

CONTRIBUȚII LA UTILIZAREA NOILOR TEHNOLOGII INFORMAȚIONALE ÎN DIAPORAMA DIGITALĂ

Teză destinată obținerii
titlului științific de doctor inginer
la
Universitatea "Politehnica" din Timișoara
în domeniul INGINERIE ELECTRONICĂ ȘI
TELECOMUNICAȚII
de către

Ing. Cristian Țecu

Conducător științific:
Referenți științifici:

Prof.univ.dr.ing. Radu Vasiu
Prof. Dr. Ing. Mircea Petrescu
Prof. Dr. Ing. Aurel Vlaicu
Prof. Dr. Ing. Nicolae Robu

Ziua susținerii tezei: 16.09.2010

Seriile Teze de doctorat ale UPT sunt:

- | | |
|------------------------|---|
| 1. Automatică | 7. Inginerie Electronică și Telecomunicații |
| 2. Chimie | 8. Inginerie Industrială |
| 3. Energetică | 9. Inginerie Mecanică |
| 4. Ingineria Chimică | 10. Știința Calculatoarelor |
| 5. Inginerie Civilă | 11. Știința și Ingineria Materialelor |
| 6. Inginerie Electrică | |

Universitatea „Politehnica” din Timișoara a inițiat seriile de mai sus în scopul diseminării expertizei, cunoștințelor și rezultatelor cercetărilor întreprinse în cadrul școlii doctorale a universității. Seriile conțin, potrivit H.B.Ex.S Nr. 14 / 14.07.2006, tezele de doctorat susținute în universitate începând cu 1 octombrie 2006.

Copyright © Editura Politehnica – Timișoara, 2010

Această publicație este supusă prevederilor legii dreptului de autor. Multiplicarea acestei publicații, în mod integral sau în parte, traducerea, tipărirea, reutilizarea ilustrațiilor, expunerea, radiodifuzarea, reproducerea pe microfilme sau în orice altă formă este permisă numai cu respectarea prevederilor Legii române a dreptului de autor în vigoare și permisiunea pentru utilizare obținută în scris din partea Universității „Politehnica” din Timișoara. Toate încălcările acestor drepturi vor fi penalizate potrivit Legii române a drepturilor de autor.

România, 300159 Timișoara, Bd. Republicii 9,
tel. 0256 403823, fax. 0256 403221
e-mail: editura@edipol.upt.ro

Cuvânt înainte

Teza de doctorat a fost elaborată pe parcursul activității mele în cadrul Departamentului de Comunicații al Universității „Politehnica” din Timișoara. Aceasta constituie rezultatul unui efort susținut de-a lungul unei perioade de cinci ani.

Lucrarea de doctorat este dedicată unui domeniu aparent de nișă, însă cu mare potențial în procesul educațional, cu o dezvoltare semnificativă în ultimii ani – prezentările audio-vizuale de tip diaporamă. Diaporama analogică a avut perioada sa de glorie, însă odată cu trecerea la tehnologia digitală, a fost înlocuită de cea numerică. Posibilitățile de exprimare cu ajutorul său au crescut simțitor, atât datorită accesului la fotografia digitală, cât și datorită unor aplicații specifice care permit un mare salt calitativ, atât tehnic cât și artistic. Cu toate acestea, trecerea de la analogic la digital, în diaporamă, este privată de interactivitatea totală între proiectant și opera sa. Practic, dispărea cea atingere umană care face ca fiecare proiectie să fie unică în felul său.

Prezenta lucrare cuprinde o serie de studii critice cu privire la stadiul actual al diaporamei, începând cu perioada analogică și terminând cu principalele aplicații de profil. De asemenea, aceasta oferă câteva soluții practice care să elimine neajunsurile sesizate la alte aplicații de diaporamă. Soluțiile reprezintă trei aplicații, fiecare din ele corectând neajunsurile celei precedente, astfel încât rezultatul final constă dintr-o aplicație și un studiu aferent care adaugă un plus de interactivitate prezentărilor de diaporamă. Aplicația finală a fost supusă testării de către două grupuri distincte de evaluatori, rezultatele obținute, care fac parte din această lucrare, confirmând utilitatea și valoarea sa.

În încheiere doresc să aduc mulțumiri deosebite conducătorului de doctorat, Prof. Dr. Ing. Radu Vasiliu, care, prin supraveghere constantă și sfaturi neprețuite a contribuit la realizarea prezentei lucrări și Dianei Andone, cea care a catalizat finalizarea ei.

Mulțumirile mele se adresează și:

Roxanei Damaschin – Jecu, pentru cele 12 luni de înțelegere.

Diaporamiștilor români Geo Neumann, Dan Buruleanu, Christian Mike Sugar și Savel Cheptea, de la care am avut atâtea de învățat. Diaporamiștilor britanici Peter Coles și Colin Balls, pentru o diaporamă care m-a inspirat de atâtea ori.

Lui Cristian Goțiu, pentru colaborarea la Digital Diaporama, Adrianei Lolea, pentru colaborarea la PhotoSlide Toolbar, dar mai ales lui Adrian Popescu, fără de care Motorway ar fi rămas, poate, doar un drum județean.

Iasminei Ermalai (mulțumiri speciale), Andrei Ternauciuc, Bogdan Drăgulescu, Marian Bucos, Mihai Onița și tuturor colegilor care m-au ajutat pe parcursul anilor de doctorat.

Vouă, tuturor celor care ați venit de atâtea ani la prezentările mele de diaporamă, mi-ați fost aproape, și m-ați încurajat să continui.

Jecu, Cristian

Contribuții la utilizarea noilor tehnologii informaționale în diaporama digitală

Teze de doctorat ale UPT, Seria 7, Nr. 26, Editura Politehnica, 2010, 160 pagini, 61 figuri, 3 tabele.

ISSN: 1842-7014

ISBN: 978-606-554-156-6

Cuvinte cheie:

diaporamă, digital, tehnologii informaționale, C#, PowerPoint, analogic, audio-vizual, educație, comunicare, proiecții, testare, prezentare, imagini

Rezumat,

Teza de doctorat este dedicată unui domeniu aparent de nișă în România, diaporama digitală. Aparent, deoarece publicul nostru tinde să asocieze proiecțiile de diaporamă cu acele spectacole analogice ilustrând de cele mai multe ori cu o muzică plăcută diapozitive realizate în concediu. Evoluția digitală a democratizat însă atât fotografia, cât și schimbul de informație, ceea ce a dus și la expansiunea prezentărilor audiovizuale. Pe de o parte, PowerPoint a devenit principala unealtă destinată prezentărilor de acest gen, dar diaporamistii consacrați au acum parte de câteva aplicații foarte specializate care îi ajută în munca lor.

Folosirea PowerPoint a devenit o condiție sine-qua-non a prezentărilor academice și științifice; în mod normal, suportul vizual al oricărei teze de doctorat trebuie să fie de acest tip. ProShow Gold, Pictures to EXE și Wings Platinum conduc piața aplicațiilor specializate. Dezavantajul acestor aplicații este că niciuna nu oferă un control interactiv al prezentărilor. Permit aranjarea imaginilor în ordinea dorită, a tranzițiilor dintre imagini la durată aleasă, sincronizarea cu o coloană sonoră dedicată și exportul în diverse formate convenabile ca dimensiuni și calitate, dar aceste aplicații nu facilitează controlul în timp real al tranzițiilor dintre imagini, așa cum se putea face în perioada diaporamei analogice. Această carență m-a inspirat și motivat să găsesc o soluție practică de simulare a sistemelor analogice de prezentare a unei diaporame, însă folosind mijloacele digitale contemporane.

Activitatea desfășurată în cadrul tezei de doctorat a presupus analiza stadiului actual al aplicațiilor specifice, precum și studiul tehnic al dispozitivelor analogice, pentru a găsi o soluție digitală a problemei. Folosind diverse platforme, am conceput și finalizat trei aplicații care permit controlul deplin, în timp real, al unei diaporame. Pe prima aplicație o consider doar o etapă premergătoare a studiului, întrucât rezultatele calitative nu m-au mulțumit. Experiența acumulată și feed-back-ul primit au constituit baza de la care am pornit a doua aplicație. Calitatea deosebită a tranzițiilor și deplina interactivitate au reprezentat punctele sale forte. Dificultățile în instalare și dorința personală de a oferi un produs finit la înălțimea așteptărilor m-au determinat să nu las studiul în acea etapă. Rezultatul final, aplicația pe care am numit-o „Motorway”, este produsul pe care îl găsesc cel mai apropiat de conceptul dorit de la care am pornit. Rezultatele primite de la evaluatori au confirmat acest lucru.

Fiabilitatea metodelor implementate a fost evaluată prin testare din mai multe perspective: a ușurinței de utilizare și a utilității, a calității tranzițiilor în diverse situații ce implică o putere mai mică sau mai mare de calcul, a eficienței mouse-ului ca și controller al prezentării. Am constatat un entuziasm crescut al evaluatorilor față de aplicație. Cel mai încurajator lucru mi s-a părut faptul că, deși aceștia au fost împărțiți pe două grupe, una cu motivație artistică și cealaltă cu educațională, răspunsul amândurora a fost pozitiv. Aceasta a arătat că diaporama rămâne un excepțional mijloc artistic de prezentare audiovizuală, dar și că utilizarea sa în cadrul procesului didactic poate oferi satisfacții mari, atât pentru prezentator, cât și pentru grupul de receptori ai prezentării.

Cuprins

1	Introducere	1-12
2	Analiza diaporamei analogice	2-16
2.1	Studiul evoluției memorării imaginii	2-16
2.2	Studiul diaporamei și imaginii statice.....	2-17
2.3	Analiza tranziției între imagini	2-19
2.4	Studiul diapozitivului. Sisteme de proiecție a imaginii și de sonorizare.....	2-20
2.5	Analiza tranziției mecanice și electronice a imaginilor	2-23
3	Implementări ale diaporamei digitale	3-26
3.1	Studiu asupra tehnologiei digitale de proiecție	3-26
3.2	Implementarea treptată a diaporamei digitale	3-28
3.2.1	Studiul aplicațiilor software de prezentare	3-29
3.2.2	Microsoft PowerPoint	3-30
3.2.3	Analiza critică a principalelor aplicații pentru diaporamă.....	3-32
3.2.3.1	Pictures to EXE.....	3-32
3.2.3.2	ProShow Gold.....	3-33
3.2.3.3	Wings Platinum.....	3-35
3.2.3.4	Neajunsuri ale aplicațiilor digitale	3-37
4	Elaborarea și implementarea primului model de aplicație. “Digital Diaporama” 4-38	
4.1	Aplicația „Digital Diaporama”.....	4-38
4.1.1	Prezentare generală	4-38

4.1.2	Structura aplicației.....	4-38
4.1.3	Interfața cu utilizatorul.....	4-39
4.1.4	Navigarea pe hard-disk	4-40
4.1.5	Previzualizarea	4-40
4.1.6	Lista de fișiere a diaporamei	4-40
4.1.7	Bara de status	4-41
4.1.8	Bara de meniu	4-41
4.1.9	Fereastra de diaporamă	4-44
4.1.10	Obiectul imagine	4-44
4.1.11	Panoul de comandă al diaporamei.....	4-44
4.1.12	Efectele de tranziție între imagini	4-45
4.1.13	Contopirea imaginilor	4-47
4.1.14	Butoanele cu durată predefinită a trecerii.....	4-48
4.1.15	Controlul automat al derulării imaginilor.....	4-49
4.1.16	Controlul manual al tranziției imaginilor	4-49
4.1.17	Redarea imaginilor cu transparență predefinită.....	4-50
4.1.18	Coloana sonoră.....	4-50
4.1.19	Bara de status a diaporamei.....	4-51
4.2	Unelte software pentru implementarea aplicației „Digital	
	Diaporama”	4-51
4.2.1	De ce am ales C# ?	4-52
4.2.2	Codul.....	4-54
4.2.2.1	Clasele utilizate	4-54
4.2.2.2	Punctul de pornire al aplicației.....	4-55
4.2.2.3	Momentele cheie „FormLoad” și „FormClose”	4-56
4.2.2.4	Regiunile mediului de dezvoltare “Visual Studio”	4-57
4.2.2.5	Fișierul de configurație al aplicației	4-57
4.2.2.6	Previzualizarea	4-59
4.2.2.7	Funcții apelate din meniu	4-62
4.2.2.8	Inserarea de fișiere în lista de diaporamă	4-62
4.2.2.9	Salvarea listei ce grupează imaginile din diaporamă	4-64
4.2.2.10	“Drag and Drop” în lista imaginilor pentru diaporamă	4-65

4.2.2.11	Structura funcțională a clasei „FullScreen”	4-67
4.2.2.12	Prima metodă executată	4-67
4.2.2.13	Butoanele de navigație	4-71
4.2.2.14	Butoane cu efect de tranziție a căror durată este predefinită	4-72
4.2.2.15	Metoda de redesenare a obiectului imagine	4-73
4.2.2.16	Controlul tranziției cu ajutorul mouse-ului	4-74
4.3	Testarea aplicației “Digital Diaporama”	4-76
4.4	Contribuții aduse prin aplicația “Digital Diaporama”	4-79
5	Elaborarea și implementarea celui de-al doilea model de aplicație.	
	“PhotoSlide Toolbar”	5-81
5.1	Unelte software folosite pentru “PhotoSlide Toolbar”	5-81
5.1.1	Microsoft Visual Studio 2008	5-81
5.1.2	De ce am ales .NET?	5-86
5.2	SpicIE – Simple plug-in creator for Internet Explorer	5-88
5.3	Aplicația “PhotoSlide Toolbar”	5-91
5.3.1	Descrierea interfeței cu utilizatorul. Fereastra browser	5-91
5.3.2	Descrierea interfeței cu utilizatorul. Fereastra FullScreen	5-97
5.4	Implementarea aplicației	5-99
5.5	Contribuții aduse prin aplicația Photoslide Toolbar	5-105
6	Elaborarea și implementarea celui de-al treilea model de aplicație.	
	“Motorway” 6-107	
6.1	Necesitate	6-107
6.2	Prezentarea aplicației “Motorway”	6-107
6.2.1	Fereastra browser-ului Internet Explorer	6-108
6.2.2	Fereastra de diaporamă	6-110
6.2.3	Comenzile spectacolului de diaporamă	6-110
6.2.3.1	Controlul cu ajutorul tastaturii numerice	6-110
6.2.3.2	Controlul cu ajutorul mouse-ului	6-111

6.3	Mediile de dezvoltare	6-112
6.4	Limitări și cerințe	6-113
6.5	Mecanisme interne ale aplicației	6-114
6.5.1	Mecanismul deschiderii aplicației	6-114
6.5.2	Mecanismul tranziției	6-115
6.5.3	Mecanismul audio	6-116
6.6	Aspecte legate de programare	6-116
6.6.1	Codul HTML pentru File Browser	6-117
6.6.2	Codul Javascript de construcție a căii	6-117
6.6.3	Codul HTML pentru acuratețea mouse-ului	6-117
6.6.4	Codul HTML pentru rezoluția de afișare	6-117
6.6.5	Codul JavaScript pentru setarea acurateței mouse-ului și rezoluției de afișare (generated file)	6-117
6.6.6	Codul Javascript pentru controlul temporizatorului	6-118
6.6.7	Codul Javascript pentru control audio și video	6-119
6.6.8	Codul JavaScript pentru acuratețea deplasării mouse-ului	6-122
6.6.9	Codul JavaScript pentru „DIV” și pentru obiectul player	6-125
6.6.10	Codul JavaScript pentru controlul coloanei sonore	6-126
7	Testarea și evaluarea aplicației Motorway	7-128
7.1	Prezentarea către respondent. Adobe Captivate.	7-129
7.2	Imagini ce au făcut obiectul testării	7-131
7.3	Instrucțiuni de utilizare	7-134
7.4	Testarea aplicației	7-135
7.4.1	Vârstă, domeniu de activitate	7-136
7.4.2	Experiența evaluatorilor	7-137
7.4.3	Aplicații folosite de respondenți	7-137
7.4.4	Puterea de calcul a computerelor evaluatorilor	7-138
	Configurarea computerelor respondenților din domeniu artistic:	7-138
	Configurarea computerelor respondenților din domeniu educațional:	7-138

7.4.5	Vizionarea aplicației	7-139
7.4.6	Ușurința de utilizare și utilitatea	7-139
7.4.7	Calitatea la 800 pixeli	7-140
7.4.8	Calitatea la 1200 pixeli	7-141
7.4.9	Calitatea tranziției pentru mouse.....	7-143
7.4.10	Erori în funcționare	7-144
7.4.11	Recomandări	7-145
7.5	Concluzii.....	7-146
8	Concluzii, contribuții și perspective	8-147
8.1	Contribuții teoretice	8-148
8.2	Contribuții aplicative	8-150
8.3	Direcții de cercetare pentru viitor.....	8-150

LISTĂ DE FIGURI

Fig. 1 - Diaproiectoare cu magazii dreptunghiulare	2-20
Fig. 2 - Diaproiectoare Kodak cu magazii circulare. Admit schimbarea obiectivelor.	2-21
Fig. 3 - Magnetofonele Kashtan și Nagra	2-22
Fig. 4 - Casetofonele pe 4 piste simultane Fostex, modelele X12 și X28	2-23
Fig. 5 - Dispozitive mecanice de topire înlănțuită. Piaptăn și diafragme.	2-23
Fig. 6 - Sisteme electronice de comandă a tranziției. Kodak, Simda și potențiomtru Leica.	2-25
Fig. 7 - Videoproiectoare digitale	3-26
Fig. 8 - Prezentare Powerpoint cu ajutorul unui videoproiector într-o sală de curs a UPT.	3-27
Fig. 9 - Aparatul foto compact Nikon Coolpix S1000pj, prevăzut cu videoproiector.	3-28
Fig. 10 - Tipuri de tranziții ale PowerPoint.	3-31
Fig. 11 - Fereastră a aplicației PicturesToExe Deluxe 6.0	3-33
Fig. 12 - Fereastră a aplicației ProShow Gold, cu un proiect personal	3-34
Fig. 13 - ProShow Gold oferă variante numeroase și diverse de tranziții.	3-34
Fig. 14 - Fereastră a aplicației Wings Platinum	3-36
Fig. 15 - „Digital diaporama” - interfața cu utilizatorul	4-39
Fig. 16 - Bara de status a aplicației "Digital Diaporama"	4-41
Fig. 17 - Modul de afișare „Details” al aplicației "Digital Diaporama"	4-43
Fig. 18 - Modul de afișare „Thumbnails” al aplicației "Digital Diaporama"	4-43
Fig. 19 - Modul de afișare „ Small Icons” al aplicației "Digital Diaporama"	4-43
Fig. 20 - Panoul de control al diaporamei	4-45
Fig. 21 - Exemplificarea tranziției imaginilor	4-47
Fig. 22 - Regiuni din interiorul codului aplicației "Digital Diaporama"	4-57
Fig. 23 - Interfața Visual Studio 2008	5-81
Fig. 24 - Printscreen modern de tip GUI	5-82
Fig. 25 - Fereastra de dialog la începerea unui nou proiect	5-84
Fig. 26 - Asistare IntelliSense la scrierea unui program	5-85
Fig. 27 - Modalități de rulare ale unui program	5-86
Fig. 28 - Emblema SpicIE	5-88
Fig. 29 - Arhitectura SpicIE	5-89
Fig. 30 - Plug-in de tip Explore Bar	5-90
Fig. 31 - Plug-in de tip Explore Menu	5-90
Fig. 32 - Interfața "PhotoSlide Toolbar" în browser	5-92
Fig. 33 - Activarea plug-in-ului	5-93
Fig. 34 - Cele trei ferestre principale ale aplicației	5-93
Fig. 35 - Butoanele de control disponibile în fereastra browser	5-94
Fig. 36 - Mesaj de eroare afișat dacă se încearcă salvarea unei liste vide	5-95
Fig. 37 - Printscreen al ferestrei de previzualizare	5-97

Fig. 38 - Interfața FullScreen a plug-in-ului	5-98
Fig. 39 - Modulul de comandă analogică KODAK S-AV PDC	6-108
Fig. 40 - Pentru deschiderea aplicației "Motorway" trebuie acceptat conținutul blocat	6-108
Fig. 41 - Fereastra browser-ului aplicației "Motorway"	6-109
Fig. 42 - Toate interacțiunile trebuie aprobate	6-109
Fig. 43 - Controlul tranziției cu ajutorul mouse-ului	6-112
Fig. 44 - Se începe întotdeauna cu selectarea "Slideshow.htm" de pe hard-disk	6-115
Fig. 45 - Se alege rezoluția de afișare a diaporamei	6-115
Fig. 46 - Mecanismul tranziției	6-116
Fig. 47 - Aspect al demonstrației de instalare în fază inițială. Print Screen.	7-130
Fig. 48 - Aspect al demonstrației de instalare în fază finală. Print Screen.	7-131
Fig. 49 - Imaginile folosite pentru testarea aplicației	7-132
Fig. 50 - Etape ale tranziției între imaginile 04 și 05	7-133
Fig. 51 - Etape ale tranziției între imaginile 11 și 12	7-133
Fig. 52 - Etape ale tranziției între imaginile 28 și 29	7-133
Fig. 53 - Etape ale tranziției între imaginile 29 și 30	7-134
Fig. 54 - Experiența repondenților domeniul artistic (stânga) respectiv educațional (dreapta)	7-137
Fig. 55 - Aplicații utilizate de cei din domeniul artistic (stânga) respectiv educațional (dreapta)	7-137
Fig. 56 - Vizualizarea aplicației de către cei din domeniul artistic (stânga) respectiv educațional (dreapta)	7-139
Fig. 57 - Aprecierea ușurinței și utilității aplicației de către cei din domeniul artistic (A) respectiv educațional (E)	7-140
Fig. 58 - Calitate vs. viteză de răspuns a tranziției la 800 pixeli pentru cei din domeniul artistic (A) respectiv educațional (E)	7-141
Fig. 59 - Calitate vs. viteză de răspuns a tranziției la 1200 pixeli pentru cei din domeniul artistic (A) respectiv educațional (E)	7-142
Fig. 60 - Comparație între calitatea tranziției la 800 și la 1200 pixeli	7-143
Fig. 61 - Calitate vs. viteză de răspuns a tranziției controlate de mouse pentru cei din domeniul artistic (A) respectiv educațional (E)	7-144

LISTĂ DE TABELE

Tabelul 1: Tablou al termenilor internaționali folosiți în sincronizarea verbală	2-24
Tabelul 2: Tranziții/Întârzieri	4-77
Tabelul 3: Întârzierile de tranziții pentru diferite biblioteci de imagini	4-79

1. Introducere

Anii 1950 au lansat un procedeu de comunicare ce avea să cunoască o răspândire nebănuită. Folosind imagini statice și o coloană sonoră dedicată, mijlocul audio-vizual cunoscut ca și "diaporamă" oferea avantajul accesibilității de producție și de diseminare [1].

Prezentarea audio-vizuală se desfășoară prin proiecția imaginilor statice în succesiune, folosind două diaproiectoare cu imagini suprapuse, tranziția dintre ele fiind de tip „topire înlănțuită”. Imaginile care se proiectează sunt sincronizate cu o sursă sonoră pe care este imprimată o coloană sonoră dedicată. Alăturarea imaginilor și a sunetului realizează o operă coerentă, având un scenariu, o regie și un aport intelectual al autorului [2].

Acest mod de prezentare, apanaj al erei analogice, a făcut trecerea firească spre mijloacele digitale de prezentare, păstrându-și însă caracteristicile de exprimare specifice.

Ideea cercetării mele doctorale se bazează pe diaporama digitală, care a preluat majoritatea caracteristicilor analogice, anumite aspecte din analogic nefiind însă acoperite de digital. Cercetarea în cadrul doctoratului cu tema „**Contribuții la utilizarea noilor tehnologii informaționale în diaporama digitală**” s-a axat pe trei direcții:

1) Studiul și analiza critică a diaporamei analogice și digitale.

2) Crearea unor modele de aplicație digitală **interactivă** pentru diaporamă.

3) Implementarea, testarea și evaluarea aplicațiilor, în contextul validării ca și rezultat al cercetării mele.

În **contextul 1)** am analizat din punct de vedere tehnic mijloacele de realizare și prezentare ale unei diaporame, evidențiind aspectele analogice și trecerea la digital, și realizând un studiu critic al principalelor aplicații de gen.

Suportul vizual al prezentării analogice consta din diapozitive (de regula 24/36mm) care au fost ulterior înlocuite de imagini digitale. Suportul audio a folosit mijloacele analogice ale vremii pentru stocarea sunetului; trecerea la digital a implicat utilizarea diverselor formate audio specifice [3].

Diaporama analogică a fost privită ca având la vremea sa marele avantaj al simplității realizării și difuzării, în condițiile în care erau la îndemâna oricui un aparat foto, două diaproiectoare și un magnetofon / pick-up, tehnologie suficient de accesibilă spre apogeul acestui gen de spectacol. Sincronizarea dintre imagini și coloana sonoră a folosit diverse metode, de la proiecția „live”, în care autorul asculta coloana sonoră și acționa tranziția și avansul imaginilor, până la înregistrarea unor comenzi electronice de schimbare a imaginilor pe una din piste benzii magnetice ce conținea coloana sonoră. Redând ulterior banda, aceasta va conține informația înregistrată în prealabil [4].

Tranziția dintre imagini se putea face fie mecanic, folosind două diafragme ce lucrau complementar obturând optica proiectoarelor, fie electronic. În această situație, tensiunea aplicată becurilor proiectoarelor se diminuea respectiv creștea

complementar, realizându-se astfel efectul de „topire înlănțuită”. Comanda variației de tensiune și a avansului imaginilor se făcea folosind taste cărora le era asociat un timp de tranziție. De asemenea, anumiți producători ofereau o telecomandă cu fir, pe care erau montate două potențiometre liniare. Deplasând cursoarele potențimetrelor, acestea comandau un circuit ce transmitea comanda de modificare a tensiunii de alimentare a becului [5].

Diaporama analogică oferea deci două opțiuni distincte de prezentare; cea interactivă care presupunea „interpretarea” lucrării, sincronizarea în direct a imaginilor și a coloanei sonore, respectiv prezentarea înregistrată, care permitea portabilitatea operei. Prezentarea interactivă a oferit avantajul spectacolului original și irepetabil, în care rolul autorului proiecționist devenea primordial. Mișcarea diaporamistică mondială s-a bazat foarte mult pe a doua trăsătură când a fost vorba de festivaluri internaționale, unde autori din toată lumea puteau trimite spre vizionare lucrări, fără ca prezența lor să mai fie necesară [6].

Beneficiind de acces la astfel de aparatură, am participat și eu în anii 1990-2000 la astfel de manifestări, fiind printre pușinii români prezenți. Pe de altă parte, am participat activ la organizarea și desfășurarea a trei festivaluri internaționale de diaporamă la Timișoara, unde au fost proiectate lucrări din toată lumea, folosind tehnica menționată.

Diaporama digitală a evoluat în paralel cu declinul firesc al celei analogice, până când a ajuns să o domine în proporție covârșitoare. Accesibilitatea fotografiei digitale și a fișierelor audio au fost un factor determinant în evoluția sa fulminantă. În paralel, existența unor aplicații specializate pentru realizarea de prezentări audio-vizuale a facilitat enorm răspândirea acestui fenomen. Acest fapt a dus la diversificarea domeniilor de prezentare, cel educațional câștigând tot mai mult teren. Aplicațiile de prezentare includ de regulă trei funcții majore: un editor ce permite inserarea și formatarea unui text, o metodă de inserare și manipulare a imaginilor grafice, și un sistem de prezentare ce afișează conținutul.

În prezent, PowerPoint (un produs Microsoft), a ajuns să domine piața prezentărilor audio-vizuale de tip diaporamă. Aplicație prezentă pe majoritatea sistemelor PC, oferă posibilitatea ordonării imaginilor, inserarea de text sau grafică, asocierea unei coloane sonore și a unor tranziții specifice între imagini. Produsul finit poate fi exportat într-un format autoexecutabil sau într-unul în care prezentatorul poate interveni parțial, comandând tranziția imaginii. Fenomenul PowerPoint se regăsește cu precădere în domeniul educațional; practic orice prezentare de profil folosește această aplicație. Prezentările la conferințele științifice internaționale trebuie realizate, de asemenea, în Powerpoint [7], [8].

Colaborarea mea pre-doctorală cu Universitatea „Politehnica” din Timișoara mi-a relevat importanța prezentărilor audio-vizuale în cadrul procesului educațional. Și în acest domeniu, diapozitivul didactic și folia retroproiectorului au fost înlocuite de mijloacele digitale de prezentare; ca peste tot, PowerPoint domină aplicațiile de prezentare, cu beneficiile și neajunsurile aferente.

O serie de aplicații de profil au apărut pentru a oferi diaporamiștilor unelte adecvate muncii lor. În urma unui studiu, am selecționat pe cele mai importante trei, folosite de marea majoritate a diaporamiștilor artiști. Lucrările prezentate la concursurile artistice de profil din lume sunt realizate folosind Pictures to EXE, ProShow Gold sau Wings Platinum [9], [10], [11].

Numitorul comun al acestora e dat de posibilitatea ordonării imaginilor, asocierea coloanei sonore, a tipului și duratei de tranziție. Produsul finit poate fi exportat atât în format de tip .avi cât și într-un executabil ce oferă o calitate bună a imaginii la volum mic de date.

14 Introducere - 1

Avantajele aplicațiilor specializate pentru diaporamă sunt clare. La un preț accesibil, oferă posibilitatea creării unei diaporame folosind imagini și sunet în format digital. Produsul finit poate fi accesibil oricărui utilizator conectat la Internet, nu necesită instalarea unor aplicații colaterale (de exemplu codec-uri sau plug-in-uri), poate fi vizionat atât în cadru intim, pe ecranul computerului, cât și într-o prezentare de amploare. Tranzițiile între imagini în diaporama digitală sunt de mult mai multe tipuri. Tranziția de tip "topire înlănțuită", specifică diaporamei analogice, a devenit acum doar una din nenumăratele posibilități de a trece de la o imagine la alta [12], [4].

Principalul neajuns al aplicațiilor prezentate anterior, așa cum l-am perceput eu, e dat de lipsa de interactivitate specifică prezentărilor analogice. Programând un spectacol de diaporamă, acesta poate fi rulat oriunde, fără a mai fi necesară prezența autorului. Diaporamei actuale îi lipsește posibilitatea de a improviza în timpul prezentării, de a modifica spontan efectele de trecere dintre două imagini, de a opri după dorință tranziția imaginilor într-un anumit moment sau capacitatea de a schimba fondul sonor din mers și de a-l adapta la o situație neprevăzută.

Prin comparație, proiecțiile de diaporamă analogică ofereau prezentatorului posibilitatea de a asculta coloana sonoră și de a stabili momentul apariției pe ecran a unei imagini, de a decide timpul în care acea imagine apare pe ecran de la 0 la 100%, de a derula prezentarea înainte sau înapoi. Diaporama analogică era în bună măsură o artă interpretativă și mulți autori au profitat din plin de acest avantaj. Faptul că tehnica a permis înregistrarea prezentărilor a fost absolut binevenit, dar chiar și acea înregistrare se baza pe o interpretare a autorului și reda cu acuratețe comenzile proiectantului asupra diaporamei.

Contextul 2) redă contribuția mea la crearea unei aplicații care să permită o interactivitate sporită a unei prezentări de diaporamă. Faptul că actualele aplicații nu permit interactivitatea mi-a dat ideea întoarcerii la metodele analogice de prezentare a unei diaporame, dar folosind tehnologia digitală disponibilă. Inspirația am avut-o de la dispozitivele analogice electronice, care având încorporate taste ce comandau durata tranziției se aseamănă cu dispozitivele digitale. Urma să concep o aplicație cu ajutorul căreia să comand tranziția și avansul unor imagini digitale și să controlez o coloană sonoră. Evident, tastatura computerului avea să înlocuiască tastele analogice.

Deoarece telecomanda analogică s-a dovedit a fi o unealtă extrem de eficace din punct de vedere al interactivității, am căutat soluții pentru a o simula cât mai realist cu putință. Inițial, am avut ideea realizării unui dispozitiv hardware separat, care să fie atașat computerului, însă aceasta ar fi redus numărul celor ce ar fi utilizat aplicația la unul egal cu cel al posesorilor de telecomandă! Ulterior mi-a venit ideea de a înlocui telecomanda cu mouse-ul, ce însoțește orice computer, aceasta simplificând mult lucrurile și făcând aplicația extrem de accesibilă.

Cercetarea s-a desfășurat în trei etape, de-a lungul mai multor ani. În acest timp **am conceput, realizat și testat trei aplicații distincte**, fiecare preluând de la precedentă elementele favorabile și eliminând aspectele indezirabile. Provocarea a constat în elaborarea unei aplicații capabile să controleze în timp real și în mod interactiv o prezentare de diaporamă. Rezultatul perceput de public nu trebuie să fie diferit de cel oferit de o aplicație consacrată, însă față de acestea, prezentarea trebuie să fie controlabilă în timp real.

În **prima etapă de studiu**, am conceput și elaborat o aplicație capabilă de control interactiv, numită **Digital Diaporama**. Folosind C#, am conceput această aplicație, capabilă să încarce o listă de imagini și una de fișiere audio, pe care să le utilizeze ulterior într-o prezentare interactivă de diaporamă. Aceasta permite

controlul tranzițiilor și al coloanei sonore, folosind tastatura, și un control separat al tranzițiilor, folosind mouse-ul. Coloana sonoră rulează în fundal de pe Winamp, și poate fi controlată cu ajutorul tastaturii. Capitolul 4 tratează pe larg acest studiu [13], [14].

A **doua etapă de studiu** a apărut ca o necesitate de a soluționa problemele apărute la aplicația precedentă. „**PhotoSlide Toolbar**”, rezultatul etapei a doua de studiu, oferă un control al tranzițiilor și al coloanei sonore ce alcătuiesc o diaporamă. Calitatea rezultatului final este net superioară față de prima aplicație. La concepție și implementare am folosit atât C#, Visual Studio, cât și SpicIE, pentru a concepe un plugin pentru Internet Explorer, care controlează în timp real prezentarea unei diaporame. Imaginile ce alcătuiesc diaporama pot fi stocate fie pe mediul personal, fie încărcate direct pe o pagină web. Coloana sonoră poate fi și ea controlată din tastatură. Capitolul 5 tratează pe larg acest studiu [14], [15], [16].

Deoarece „PhotoSlide Toolbar” a avut și neajunsuri, am conceput a **treia aplicație**, numită „**Motorway**”, care înglobează întreaga experiență dobândită pe parcursul studiului. Am folosit HTML, JavaScript, CSS și Internet Explorer, pentru a pune la punct ceea ce consider că reprezintă o soluție viabilă și aplicabilă universal pe PC, pentru a prezenta o diaporamă la modul interactiv, așa cum am sperat la începutul acestui studiu. Controlul se realizează atât folosind tastatura cât și mouse-ul, iar aplicația nu necesită instalare. Calitatea imaginii și a tranziției, interactivitatea totală, denotă împlinirea dezideratului inițial. „Motorway” a reprezentat etapa finală a studiului, gata pentru a fi oferită celor interesați. Acest studiu se găsește în capitolul 6 [17], [18].

Am supus această aplicație unor teste, parte a **contextului 3**). În acest scop am oferit-o spre evaluare către două grupuri distincte, primul format din persoane care au avut tangențe cu prezentări audio-vizuale artistice, al doilea grup folosind acest tip de aplicații în domeniul educațional. Rezultatul evaluării a relevat utilitatea și ușurința de utilizare, calitatea bună a tranziției pentru diverse dimensiuni ale imaginii, atât pentru situația când tranziția e controlată din tastatură, cât și cu mouse-ul. Evaluarea, rezultatele sale și analiza lor fac obiectul celui de-al șaptelea capitol .

Ultima parte a studiului, respectiv capitolul 8, e consacrată menționării contribuțiilor personale, atât teoretice, cât și aplicative. Concluziile și direcțiile unei viitoare cercetări încheie această teză.

2. Analiza diaporamei analogice

2.1 Studiul evoluției memorării imaginii

Una din primele dovezi ale trecerii ireversibile a omului la statutul de Homo Sapiens Sapiens a constituit-o abilitatea de a comunica vizual folosind termeni abstracti. Picturile rupestre rămase din *paleolitic*, cu precadere cele din peșterile Chauvet, Altamira, Lascaux, Trois Freres, sau Cuciulat, Gaura Chindiei, Coliboaia o demonstrează cu prisosință. Reprezentări zoologice și antropomorfe schematizate sau detaliate, ritualuri șamanice și de vânătoare, imagini în negativ ale palmelor sunt cele dintâi mesaje vizuale transmise posterității care au rămas până în zilele noastre [19].

Dezvoltarea firească a societății umane a fost legată intrinsec de diversificarea mijloacelor de comunicare, pasul major fiind dat de apariția scrierii folosind o convenție de asociere a unui sunet sau grupuri de sunete cu o reprezentare vizuală, numită alfabet. Atât cel cuneiform, apanaj al civilizației sumeriene cât și ulterior, cel hieroglific, utilizat de înfloritoarea civilizație egipteană, vor fi asociate cu primele formațiuni statale [20].

Artele vizuale, principala formă de comunicare abstractă ce avea să dăinuie până în prezent, au cunoscut în timp o puternică înflorire, atât ca subtilitate de exprimare cât și ca răspândire și impact. De cele mai multe ori pictura a fost asociată cu fenomenul religios, specific fiecărei culturi și zone geografice. Excepția constă, firește, în reprezentări ale potentăților vremii, fie că e vorba de lideri politici sau spirituali.

Specific picturii rămâne noțiunea de unicitate, de tușă personală a autorului. Nume cu rezonanță ale genului, de la Rubliov și Da Vinci, Rafael și Michelangelo, până la Picasso, Hundertwasser și Dali (și lista rămâne deschisă) au lăsat omenirii opere inestimabile prin unicitatea lor [21].

Revoluția industrială a adus cu sine o remarcabilă descoperire ce avea să revoluționeze comunicarea vizuală. Francezul Nicephore Niepce, în paralel cu elvețianul Hypolite Baillard au reușit prin procedee chimice specifice să fixeze imagini captate pe peretele unei **camera obscura**, și ulterior să le multiplice [22], [23]. Cu pași șovăielnici, dictați de factori tehnici, fotografia (din grecescul **photon** + **graphein**, scriere cu lumină) își făcea intrarea în lume, în vara lui 1826.

Realizate inițial pe plăci impregnate cu colodiu umed, fotografiile au evoluat devenind din ce în ce mai accesibile utilizatorului. Anul 1935 a marcat apariția filmului fotografic pe suport de celuloid, extrem de versatil. Materialele fotosensibile color - negativ și ulterior cele color - reversibil (diapozitiv) sau Polaroid [24] au adus necesarul plus de informație cromatică. Până în 1995, diapozitivele color erau singurul mediu fotografic calitativ folosit pentru tipărituri serioase, și au fost utilizate pe scară largă în fotografia comercială, de reportaj, sport, stoc și de peisaj [3] [25].

După apariția dispozitivelor digitale de fotografiere, importanța diapozitivului color s-a diminuat, rămânând apanajul artiștilor și a celor încă nemulțumiți de calitatea imaginii digitale.

Un experiment realizat de Eadweard J. Muybridge în 11 iunie 1878 a pus bazele apariției unei noi forme de comunicare vizuală, cu implicații nebănuite. Dispunând 12 aparate foto în linie pe un hipodrom, și declanșându-le în momentul trecerii calului în dreptul lor, imaginile obținute au fost proiectate în succesiunea inițială, cu o viteză mare, dând iluzia mișcării continue a patrupedului. Se născuse

cinematografia, iar traiectoria sa puternic ascendentă nu fusese bănuită de pionierii acestei noi arte [26].

Urmărind evoluția fotografiei și a cinematografeiei, e imposibil să nu remarcăm ascensiunea vertiginoasă a amândurora, evident în conexiune cu marea explozie tehnologică ce a urmat revoluției industriale. Trecerea de la plăcile de sticlă impregnate cu colodiu la filmul pe suport de celuloid, la început monocrom și ulterior color negativ și respectiv reversibil, a "democratizat" fotografia [27], [23].

În acest timp, de pellicula pe suport de celuloid a profitat puternic și industria cinematografică. Acest fapt a dus la transformarea sa, de la un simplu *entertainment* la un puternic și accesibil mijloc de comunicare. Trebuie menționat un punct de cotitură în cinematografie, respectiv apariția așa-numitului limbaj cinematic, care folosind mijloace artistice specifice genului, impunea o serie de standarde de interpretare a unei creații de gen. Aceasta a dus la impunerea filmului cinematic ca un mijloc artistic de sine stătător [28].

Mari regizori au lăsat omenirii atât succese de casă, capodopere unanim acceptate, de la începuturile filmului mut, până la cele specifice showbiz-ului contemporan, cât și lucrări de nișă, destinate unui public elevat.

Ca trăsătură dominantă, creația cinematografică se bazează pe un scenariu, în care fiecare întâmplare este defalcată și abordarea sa este tratată individual. Coloana sonoră este dedicată imaginii, conținând muzică, vocea personajelor, sunet ambiental.

Ținând cont de contextul prezentat, devine evidentă apariția diaporamei în anii '50 în cadrul modalităților de prezentare audio-vizuale [2].

2.2 Studiul diaporamei și imaginii statice

O serie de definiții s-au lansat în timp, pentru noțiunea de diaporamă. Conform Federației Internaționale de Artă Fotografică (FIAP), diaporama este "un mijloc de comunicare audio-vizual, realizat plecând de la proiecția de diapozitive sonorizate". Conform dicționarului de neologisme, DIAPORĂMĂ: s. f. spectacol în cadrul căruia sunt proiectate diapozitive. (< fr. diaporama) [29] DIAPORĂMĂ s.f. Montaj fotografic sonorizat, în cadrul căruia sunt proiectate cu acompaniament muzical o suită de diapozitive [30].

Madier [1], cel care a formulat denumirea acestui tip de prezentare, afirma că "diaporama este un ansamblu de imagini statice proiectate, în general însoțite de un acompaniament de muzică și text, înregistrat pe bandă magnetică".

În principiu, diaporama are la bază următoarele caracteristici [4]:

- Este un mijloc de expresie și de comunicare făcând parte din sfera audio-vizualului.

- Baza vizuală este alcătuită din imagini statice.

- Imaginile sunt grupate în secvențe compuse după principii ale limbajului cinematic, de logică și măiestrie, iar secvențele la rândul lor se articulează armonios între ele, formând un tot unitar, care evidențiază un eveniment, un fenomen, o trăire sau o idee. Efectul vizual e completat și emfizat de coloana sonoră: comentariu vorbit, muzică, sunet de fond, etc.

- Modul de comunicare este proiecția seriei de imagini, sincronizată cu coloana sonoră dedicată.

- Diaporama este un gen de spectacol ce se desfășoară într-un spațiu public, conformându-se cerințelor impuse de ideea de spectacol.

●Proiecția se desfășoară prin tranziția tip "cross-dissolve" după denumirea în engleză, sau francezul „fendu enchainee”, topire înlăntuită.

Proiecția se desfășoară prin "topirea" imaginilor, efect al cărei durată este variabilă, determinată de dorința autorului. Această trecere poate fi rapidă ("Cut"), până la câteva secunde, fiind permise nenumărate efecte, de la suprapuneri simple, apariții, supraimpresiuni, crearea celei de „a treia imagine”.

Imaginile care se proiectează sunt sincronizate cu o sursă sonoră pe care sunt imprimate voci, muzică, zgomote de fond, etc. Totuși, îmbinarea imagine-sunet nu este suficientă pentru a crea o adevărată diaporamă. În afară de acestea, trebuie realizată o operă coerentă, având un fir conducător variabil și un anumit aport intelectual al autorului.

Prima caracteristică a diaporamei e dată de asocierea sa cu imaginile statice. Spre deosebire de imaginile în mișcare, obținute cu aparatul de filmat, imaginea statică are certe avantaje:

- calitate tehnică superioară
- compoziție cu mari valențe artistice
- posibilitatea spectatorului de a cerceta și analiza un timp fiecare element al imaginii

●calitatea de a-l face pe privitor un spectator activ; parcurgând imaginea. își caută singur înțelesurile și își trage propriile concluzii, fapt ce nu se aplică la imaginea în mișcare unde privitorul devine spectator pasiv, ghidat de intențiile operatorului și regizorului

- concentrare asupra esențialului, prim posibilitatea eliminării unor așa numite planuri de legătură necesare numai la imaginile filmate

- posibilitatea de a crea un ritm prin variația timpului de proiecție al fiecărei imagini. Fiecare imagine ce intră în componența unui spectacol de diaporamă trebuie bine gândită, iar includerea ei în serie trebuie să fie temeinic motivată. Criteriile alegerii sunt într-un fel deosebite de cele ale alegerii unei imagini izolate pentru un concurs sau o expoziție. Acestea sunt subordonate integrării imaginii într-o secvență, potrivirii ei cu imaginea care o precede și cu cea care o urmează precum și cu ideea, atmosfera și nuanța întregii secvențe.

Dacă unei imagini izolate i se cere cât mai mult din punctul de vedere al explicitării subiectului și conținutului, al formei bine încheiate, celei alese pentru diaporamă i se cere calitatea de a se încadra într-un tot, fără a-i rupe omogenitatea. Ea nu trebuie să arate "totul", ci să contribuie la dezvoltarea gradată a unei idei, la o îmbogățire treptată a mesajului care se transmite.

De aceea, la imaginea pentru diaporamă există două categorii de cerințe: cele privitoare la calitatea tehnică și artistică a fiecărei imagini precum și cele privitoare la compatibilitatea ei cu suita de imagini [6].

Grupajele de fotografii pe o anumită idee, temă sau alt element de legătură se numesc "serii" și evidențiază o nevoie pe care o resimt artiștii fotografi de a evada din cadrul strâmt al unei singure fotografii, pentru a se putea exprima complet și nuanțat [5].

Prin "secvență" de fotografii se înțelege dezvoltarea unei teme în mod cronologic (ieri, azi, mâine) sau pe baza unei unități de loc (Timișoara zilelor noastre) sau pe baza timpului (ora 7 în 7 orașe). Secvența are multe trăsături comune cu fotoreportajul, care relatează ceva într-o anumită ordine, având început și sfârșit, introducere și concluzie [31].

De exemplu, o diaporamă sau o serie, bazată pe ideea "Sub 0°" ar prezenta și peisaje de iarnă cu formații de gheață, patinaj artistic, schiori, dar și portrete de

oameni cu "priviri înghețate", copii mâncând înghețată, interiorul unui frigider (plin sau gol), îndrăgostiți ale căror sentimente "au înghețat" și alte imagini ținute împreună de ideea "Sub 0°". Revenind la diaporamă, se poate spune că ea poate uneori să fie o serie în totalitatea ei, în sensul arătat mai sus. De obicei, astfel de diaporame sunt mai scurte (cca. 3-4 minute). În cele mai multe cazuri, diporama este alcătuită dintr-o "secvență de secvențe".

2.3 Analiza tranziției între imagini

Unul dintre efectele caracteristice ale proiecției de diaporame este tranziția imaginii. Se pot proiecta diapozitive și cu un singur proiector, în acest caz, trecerea de la o imagine la cealaltă se face fie observându-se din sală translația (verticală sau orizontală) de la o imagine la alta, fie, cum este cazul cu proiectoarele mai evolute, prin obturarea luminii în timpul schimbării diapozitivelor, ceea ce a căpătat denumirea de "pauză neagră". La ședințele prelungite de proiecție, atât repetarea monotonă a translației cât și a pauzei negre pot deveni obositoare, enervante, neplăcute. Pentru înlăturarea acestor neajunsuri, s-a trecut la proiecția cu două aparate.

Condiția primordială de reușită era ca cele două aparate să fie de aceeași marcă, același tip, cu parametri tehnici identici, cu obiective de aceeași distanță focală și luminozitate și cu lămpi de proiecție cu randament egal. Aceste cerințe sunt necesare pentru a se putea obține pe ecran o suprapunere perfectă a imaginilor proiectate de cele două aparate.

Efectul se produce prin întunecarea treptată și parțială a imaginii, în timp ce cealaltă imagine se luminează și apare treptat, dând iluzia la un moment dat că ar fi văzută prin cealaltă. Este ca și cum o imagine s-ar "topi" și ar dispărea, lăsând locul celeilalte. Acesta este efectul de tranziție a imaginii. Diaporamiștii români au preluat termenul francez „fondue enchainee”, traducându-l „topire înlănțuită” sau „contopire” [32].

Urmărind această "contopire" treptată, se observă că imaginea a doua devine vizibilă, ea apare mai întâi în porțiunile întunecate ale primei imaginii. Pe baza acestei constatări, se pot crea în mod intenționat, încă de la fotografiere, efecte controlate. De exemplu, printr-o arcadă întunecoasă apare un personaj, un vehicul, un peisaj.

Prin "topirea înlănțuită a imaginii" apare, în afară de cele două imagini de la care se pleacă, o a treia imagine rezultată din suprapunerea celor două [33].

Diferitele tipuri de tranziție prin topire înlănțuită a imaginii se deosebesc, indiferent de tehnica utilizată, prin timpul pe durata căruia se desfășoară. Cel mai des se recurge la topirea scurtă (cut) datorită implicațiilor tehnice. Urmează, după criteriile dictate de aspectul imaginilor care se succed, după necesitatea creării unui ritm de proiecție, potrivirea cu cadența acompaniamentului muzical, tranziții care variază de la 2-3 secunde până la 9-10 secunde sau chiar mai mult [2].

Oarecum asemănător cu cea de-a treia imagine, însă imobil, este efectul de suprapunere. Aceasta trebuie realizată astfel încât să rezulte o îmbinare, o completare, un "colaj" interesant, armonios, între un portret și un peisaj sau zid, între o fotografie și un desen sau grafică. Efectul de suprapunere poate fi utilizat cu rezultate foarte bune în diaporame cu caracter documentar sau pedagogic. Dacă, de

exemplu, este necesar să se arate pe rând diferite componente dintr-un montaj electronic, se poate lua imaginea montajului și apoi crea imagini care să conțină doar săgeți indicatoare, în diferite poziții menite să arate componentele în discuție.

2.4 Studiul diapozitivului. Sisteme de proiecție a imaginii și de sonorizare

Diaporama analogică a fost privită ca având la vremea sa marele avantaj al simplității difuzării, în condițiile în care erau la îndemâna oricui un aparat foto, un diaproiector și un magnetofon / pick-up, tehnologie suficient de răspândită la începutul și spre apogeul acestui gen de spectacol.

Diapozitivul este elementul fundamental care stă la baza diaporamei analogice. Acesta este, conform dicționarului: **DIAPOZITIV** *s.n.* Copie fotografică pozitivă făcută pe sticlă sau pe alt material transparent și servind la proiectarea imaginilor cu ajutorul diascopului. [*< fr. diapositive*] [30].

Cel mai uzual tip de diapozitiv a fost cel realizat pe film de tip 135, (24/36mm), însă au existat și diapozitive de format 120 (60/60mm), destinate profesioniștilor. Odată expus, diapozitivul era prelucrat chimic prin diverse metode specifice fiecărui producător, apoi uscat, tăiat la marginile zonei expuse în aparatul fotografic și pus în ramă. Astfel înrămat, se introducea în diaproiector, cu emulsia înspre obiectiv, și cu susul în jos. Proiecția putea începe [33].

Calitatea vizionării diapozitivelor depindea atât de performanțele aparatului de proiecție cât și de calitatea ecranului. Aparatele de proiecție ce se foloseau pot fi diferențiate după dimensiunile și tipurile de diapozitive pentru care erau destinate, forma constructivă, performanțele sistemului lumino-optic (sursă de lumină, calitățile condensatorului și ale obiectivului), capacitatea magaziei de diapozitive, gradul de automatizare, etc.

Caracteristicile de bază ale unui diaproiector destinat amatorilor erau: magazie pentru diapozitive de formă dreptunghiulară, închisă sau decupată, obiectiv fix de 80 mm, un bec halogen de 24V/150W. Acestea puteau fi cuplate cu diverse dispozitive pentru automatizarea proiecției, precum schimbarea automată a diapozitivelor, sau sincronizarea diapozitivelor cu coloana sonoră.



Fig. 1 - Diaproiectoare cu magazine dreptunghiulare

Un aparat destinat spectacolelor profesionale avea următoarele caracteristici: dotat cu magazie circulară orizontală de 80 de diapozitive, un set de 9 obiective cuprinzând o gamă a distanțelor focale de la 28mm până la 250 mm și numeroase

2.4 Studiul diapozitivului. Sisteme de proiecție a imaginii și de sonorizare 21

accesorii (dispozitive de retur la zero, dispozitiv de topire înlănțuită rapid, dispozitiv de fondu variabil, programator pentru proiecție în sistem multivision, etc). Ca sursa de lumină se utiliza o lampă cu halogen alimentată la 24V, capabilă de o putere de 250W [34].



Fig. 2 - Diaproiectoare Kodak cu magazinele circulare. Admit schimbarea obiectivelor.

Un rol hotărâtor în performanțele aparatului de proiecție îl avea sistemul lumino-optic. Acesta era constituit dintr-o sursă de lumină, un reflector, un condensator și un obiectiv de proiecție. Ca surse de lumină pentru diaproiectoare de format mic se foloseau lămpi, de 100 și 500W la tensiune de rețea sau lămpi de 100 și 150 W la tensiune joasă (24V). Unele lămpi erau construite cu un reflector încorporat în becul halogen, acesta având rolul de a da un randament crescut fluxului luminos. Este renumit faptul că diaproiectoarele clasice s-au dovedit a fi radiatoare demne de luat în seamă pentru energia calorică. Aceasta putea determina ridicarea temperatură în temperatura imaginii, fapt ce la rândul lui conducea la schimbarea stării termice a diapozitivului, având repercursiuni asupra calității proiecției (fenomenul de bombare, apariția petelor sau a inelelor Newton, degradarea culorilor ori dezlipirea straturilor). Pentru a evita aceste efecte nedorite, aparatele de proiecție erau dotate cu filtre anti-calorice, din sticlă special tratată, de 2,5mm grosime la becul de 150W sau 5mm la 250W, și chiar cu suflante pentru răcire [35].

Ecranul are un rol esențial în obținerea unei proiecții de bună calitate și nu poate fi considerat ca un simplu accesoriu. Ecranele folosite în proiecții pot fi reflectante sau translucide. Cele reflectante sunt destinate proiecției frontale și sunt cu reflexie difuză, când unghiul de difuzie este de 100° - 140° , cu reflexie difuz-direcționată când unghiul de difuzie este de aproximativ 50° și cu reflexie asemănătoare oglinzilor.

Ecranele cu cele mai bune caracteristici de reflexie a diapozitivelor sunt așa-numitele ecrane perlate, ce se obțin prin acoperirea fondului ecranului cu o pulbere formată din sfere minuscule de sticlă având diametrul de 0,1 - 0,2 mm care au capacitatea de a reflecta lumina primită într-un unghi limitat, reflexia luminii făcându-se în acest caz pe direcția incidentă, asigurându-se un spor de lumină de 4-5 ori mai mare decât în cazul unui ecran obișnuit.

Ecranele pot fi și metalizate, acestea fiind acoperite cu un strat metalic reflectorizant capabil să ofere o luminanță de circa 10 ori mai mare decât în cazul ecranului convențional la o proiecție în unghi de până la 50° către 110° .

Ecranul translucid este destinat retroprotecției și se execută din sticlă mată sau opacă. Este destul de rar folosit de către amatori.

Magnetofonul a fost instrumentul de bază pentru o bună sonorizare a diaporamei analogice. Se folosea aparatul cu bandă de 6,35 mm pe vitezele de 9,5

22 Analiza diaporamei analogice - 2

sau 19 cm/s, ca fiind cel mai indicat. Racordarea diferitelor surse de sunet (microfon, magnetofon, casetofon) se făcea prin intermediul unui mixer audio.



Fig. 3 - Magnetofonele Kashtan și Nagra

Problema majoră apărea în momentul în care se dorea sincronizarea imaginii cu sunetul. De cele mai multe ori, arta proiecției consta în cunoașterea de către proiecționist (de cele mai multe ori chiar autorul lucrării) în amănunt a coloanei sonore, astfel încât să reușească să sincronizeze perfect sunetul cu imaginea, după indicațiile regizorale. Atunci când autorul nu își putea proiecta singur lucrarea, de exemplu în cazul în care lucrarea era trimisă la un festival/ prezentare internațională, era necesar un sistem de sincronizare a imaginilor cu coloana sonoră. În vederea soluționării acesteia s-a recurs la un artificiu ce consta în utilizarea unui magnetofon stereo ce funcționa cu o bandă înregistrată pe două piste: canalul stâng, respectiv canalul drept. Astfel se înregistra pe una dintre piste „coloana sonoră” propriu-zisă, mono, iar pe cea de-a doua pistă se înregistra programul de sincronizare.

În timp, sistemele au evoluat puternic, ajungându-se la magnetofone ce citeau 4 piste simultan, lăsând loc stereofoniei pe canalele 1 și 3, respectiv semnalelor analogice de sincronizare, pe pista 4 [36]. Anii '90 au marcat trecerea de la magnetofonul pe 4 piste la casetofonul similar, scăderea în dimensiune și greutate a sistemului audio fiind demnă de luat în seamă.



Fig. 4 - Casetofoanele pe 4 piste simultane Fostex, modelele X12 și X28

2.5 Analiza tranziției mecanice și electronice a imaginilor

Aparatura specifică diaporamei analogice este dotată din fabrică cu un dispozitiv de reglare a intensității luminoase, de la zero la maxim, a becului aparatului de proiecție. Sunt și alte modalități de a realiza acest efect specific cu grad mai mare sau mai mic de complexitate. O metodă simplă, mecanică, este plasarea unei măști culisante (tip piaptăn) în fața obiectivului. Dacă proiecția se realizează cu două aparate, masca va acoperi alternativ cele două conuri de lumină. O altă soluție, ceva mai complicată, este adaptarea unor diafragme pe montura obiectivelor diaproiectoarelor. Articulația comenzilor celor două diafragme se face la o manetă rotitoare ce comandă deschiderea respectiv închiderea spotului de lumină [2].

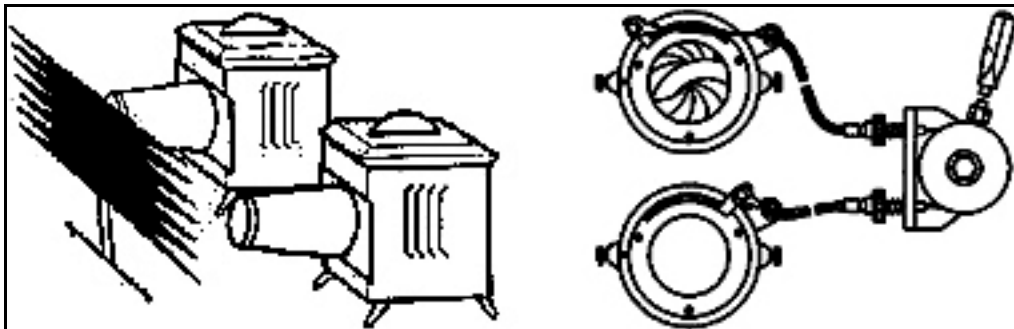


Fig. 5 - Dispozitive mecanice de topire înlănțuită. Piaptăn și diafragme.

Sincronizarea este procedeul prin care se face legătura între componentele unei diaporame în general, respectiv între sunet și imagine. Ritmul, legătura imagine-muzică sau imagine-voce se realizează prin stabilirea unei bune sincronizări.

Cea mai simplă formă de sincronizare constă în comenzi verbale înregistrate, care indicau momentul tranziției între imagini și durata acestora.

24 Analiza diaporamei analogice - 2

Termeni curenți	Germană	Engleză	Franceză	Obs.
Topirea imaginii	fondus	Fade	fondus	F
Topire lentă (+6 secunde)	lang	Iong	lent	FL sau L
Topire medie (3-5 secunde)	medium	medium	medium	FM sau M
Topire rapidă (2 secunde)	rapid	Rapid	rapid	FR sau R
Trecere fără topire	cut	Cut	cut	
Proiectarea imaginii următoare și întorcerea la prima fără schimbare	aller-retour	A-return	aller-retour	AR
Efecte speciale				
Oprire pe imagine în timpul topirii	stop	Stop	stop	
Trecere sacadată	staccato	staccato	staccato	
Apariție ca de fulger și reîntoarcere imediată	blitz	Flash	eclair	

Tabelul 1: Tablou al termenilor internaționali folosiți în sincronizarea verbală

O primă inovație a fost adusă prin lipirea de material reflectorizant, de regulă staniol, pe partea suport a benzii de magnetofon. În timp ce stratul magnetic conținea informația muzicală, staniolul de pe suportul părții magnetice reflecta către o fotodiodă lumina emisă de un beculeț. Fotodioda făcea parte dintr-un circuit ce comanda avansul diapozitivului. Neajunsul consta în faptul că sistemul nu controla durata tranziției, ci comanda doar schimbarea imaginilor [3].

Impunerea diaporamei, trecerea sa dincolo de domeniul amatorilor, a determinat diferite firme să conceapă sisteme de control al tranziției imaginilor, al avansului, cu posibilitatea înregistrării și reproducerii prezentării. Nume de prestigiu pe piața audio-vizualului, *Stumpfl*, *Basgen*, *Simda*, *Imatronic*, *Kodak* și-au perfecționat propriile sisteme.

În principiu, cu excepția sistemului Kodak PDC, sistemul de control constă dintr-o telecomandă cu fir, pe care este montat un potențiomtru liniar. Deplasând cursorul potențiometrului, acesta comandă un circuit ce transmite unui triac/tiristor atașat proiecteurului de diapozitive, comanda de modificare a tensiunii de alimentare a becului. În mod normal, telecomanda dispune de două potențiometre așezate în sens opus, astfel încât fiecare din ele comandă câte unul din cele două diaproiectoare, în mod complementar. Diminuarea tensiunii unuia implică creșterea

2.5 Analiza tranziției mecanice și electronice a imaginilor 25

celuilalt. La capătul cursei potențimetrului, câte un microcomutator comandă avansul diapozitivului corespunzător diaprojectorului cu tensiunea zero pe bec.



Fig. 6 - Sisteme electronice de comandă a tranziției. Kodak, Simda și potențimetru Leica.

În paralel cu comanda triacelor, sistemele trimit semnale codate audio către un dispozitiv de înregistrare (casetofon, magnetofon, de regulă către pista 4 pentru a beneficia de coloană sonoră stereo). În timp ce acesta redă coloana sonoră a diapoamei, în timp real autorul proiectează lucrarea înregistrând semnalele audio de sincronizare pe pista 4. Redând ulterior banda, aceasta va conține informația înregistrată în prealabil.

Kodak propunea ca în locul telecomenzii, durata tranziției să fie controlată de o tastatură căreia îi corespunde o anumită durată: cut, 1,2,4,8,12 secunde.

Succesul acestor sisteme de control a fost imens, permițând producătorilor de diaporame să își trimită lucrările (respectiv diapozitivele, numerotate corect, și caseta/banda de magnetofon conținând coloana sonoră și semnalele de sincronizare) în lumea întreagă, având garanția unei reproduceri perfecte a spectacolului.

Pe tot parcursul activității mele de diaporamist, am folosit aparatură analogică începând cu Diascol, Aspectomat J24B, Rollei, culminând cu Kodak Carousel AV2000 și Royale. Ca dispozitive de înlănțuire a imaginii, am folosit frecvent mai multe sisteme manufacturate de tip „piaptăn”, urmat de diafragmă, pentru ca ulterior, după 1990, să accesez tehnologia vestică: Simda, Imatronic, Kodak PDC. Participând activ la organizarea unor Festivaluri Internaționale de Diaporamă în Timișoara, sub Înaltul Patronaj F.I.A.P., am avut șansa unui bogat schimb de experiență atât artistic, cât și tehnic, cu diaporamiști din întreaga Europă.

3. Implementări ale diapramei digitale

3.1 Studiu asupra tehnologiei digitale de proiecție

Diaproiectoarele sunt acum înlocuite de către proiectoarele video. Ideea din spatele celor două dispozitive este esențialmente aceeași, doar că sursa imaginii este alta. Acesta preia semnalul video de la o ieșire a unui alt dispozitiv electronic (calculator personal, tuner TV, etc) și realizează o proiecție a acestuia pe un ecran folosindu-se de un sistem de lentile de reglaj.



Fig. 7 - Videoproiectoare digitale

Toate proiectoarele video folosesc o lumină foarte puternică pentru a proiecta imaginea, cele mai noi fiind capabile să realizeze reglaje fine asupra imaginii proiectate pentru a elimina orice imperfecțiune și pentru a se asigura o redare cât mai fidelă a imaginilor pe ecran.

Costul unui astfel de echipament nu a fost atât de redus ca și în cazul diaproiectoarelor clasice, determinând inițial o răspândire mai mică în rândul utilizatorilor amatori. Această situație, întâlnită până la mijlocul acestui deceniu, avea însă să se schimbe. Prețul unui astfel de dispozitiv nu a fost determinat numai de rezoluția capabilă să o redea, ci și de către capacitatea de iluminare, zgomotul introdus ori contrast. Proiectoarele video comune oferă posibilitatea de a ilumina un ecran de dimensiuni normale, într-o încăpere fără prea multă lumină sau într-o încăpere închisă. Pentru a realiza o prezentare într-o cameră de dimensiuni mai mari, eventual și cu lumină ambientală, este nevoie de un proiector video capabil să redea mai mulți lumeni. Astfel, având la dispoziție un proiector cu capacitatea de 1000 până la 1500 lumeni, nu va fi posibilă o prezentare decât într-o încăpere cu lumină de intensitate scăzută, pe un ecran de dimensiuni reduse.

3.1 Studiu asupra tehnologiei digitale de proiecție 27



Fig. 8 - Prezentare Powerpoint cu ajutorul unui videoproector într-o sală de curs a UPT.

Un proiector video cu capacitate de la 1500 la 3000 de lumeni se pretează unor încăperi de dimensiuni medii cu lumină ambientală prezentă. Peste 3000 de lumeni, proiectoarele video se folosesc în încăperi mari, unde nu există posibilitatea de a controla lumina ambientală, pentru a face prezentări pe ecrane de dimensiuni mari, de exemplu într-o sală de conferințe.

Dimensiunile imaginii proiectate sunt de asemenea importante, deoarece lumina totală nu se modifică în intensitate peste o anumită limită superioară, dar dimensiunile ecranului pot varia. Odată cu creșterea dimensiunilor ecranului, contrastul imaginii redată scade semnificativ. Astfel, dacă se mărește diagonala ecranului pe care se face prezentarea și cu 25 de procente, intensitatea luminoasă va scădea cu 35 de procente, pe când o creștere a diagonalei cu 41 de procente va determina scăderea intensității luminoase la jumătate [37], [38].

Cu toate acestea, tehnologia a permis în timp o scădere vizibilă a costurilor, precum și o creștere a calității videoproiectoarelor, astfel încât la data încheierii acestu studiu, prezența unui astfel de aparat la îndemâna oricărui diaporamist nu mai ridică probleme.

O noutate în domeniu, ce merită a fi menționată, este un produs al firmei Nikon, Coolpix S1000pj, care include un "pico proiector" într-un aparat foto de tip compact. Imaginile achiziționate cu ajutorul său pot fi proiectate pe un ecran cu ajutorul proiectorului încorporat. Acesta permite proiectarea fotografiilor sau filmelor pe orice suprafață plată, la atingerea unui buton. Oferă până la 10 lumeni, imagini la calitate VGA de la 5 la 40 de inch (13 până la 100 cm). Aparatul foto este furnizat cu telecomandă și suport proiector [39].



Fig. 9 - Aparatul foto compact Nikon Coolpix S1000pj, prevăzut cu videoproiector.

3.2 Implementarea treptată a diaporamei digitale

De îndată ce tehnologia a permis, pasul de la diaporama analogică către prezentările bazate pe fotografia numerică a fost unul așteptat și de neevitat.

La noi în țară, începând cu noul mileniu, spectacolele artistice de diaporamă au devenit din ce în ce mai rare și au tendința de a fi lăsate în uitare, acest gen de prezentări fiind practicate doar de către entuziaști și într-un spațiu din ce în ce mai restrâns. Cea mai plauzibilă explicație e dată de portabilitatea prezentărilor digitale; acum diaporamele pot fi vizionate pe internet, fără a mai fi necesară prezența într-o sală de spectacol. Fenomenul se regăsește în tot ce ține de prezentări media. Din fericire, în Europa, numărul prezentărilor artistice de diaporamă a crescut semnificativ, iar calitatea tehnică și artistică a lucrărilor a ținut pasul cu progresul IT [40]. În ciuda celor prezentate mai sus, diaporama digitală folosită în scop educativ are o ascensiune puternică, datorită aplicațiilor specifice, aflate acum la îndemâna oricui. O prezentare detaliată a celor mai folosite aplicații va fi prezentată în continuare.

La fel ca și în cazul diaporamei analogice, diaporama digitală este caracterizată de mai multe direcții de dezvoltare dintre care putem enumera: prezentări de tip profesional (legate de locul de muncă al privitorilor), prezentări comerciale, educative, sau artistice.

Diaporama digitală vine să completeze sau să înlocuiască tehnologia învechită a "fluturașilor" a tablei și a cretei, a cartoanelor explicative, a posterelor ori a foliilor transparente pentru proiectoare. Textul, grafica, filmele și alte obiecte de acest tip sunt poziționate pe pagini individuale numite "folii" sau dacă e să folosim cuvântul preluat din limba engleză, "slide-uri". Slide-urile pot fi tipărite pe suport de hârtie sau cel mai adesea redactate pe ecranul calculatoarelor ori cu ajutorul proiectoarelor pe ecrane asemănătoare cu cele folosite în diaporamele analogice. Tranzițiile dintre slide-uri sunt de cele mai multe ori comandate de către autor ori cel ce prezintă diaporama sau pot rula "automat" preprogramate să se schimbe la un anumit interval de timp și cu tranziții de un anumit tip.

Tranzițiile între slide-uri în diaporama digitală sunt de mult mai multe tipuri datorită noilor posibilități ivite odată cu creșterea puterii de calcul a calculatoarelor personale. Tranziția de tip "topire înlănțuită", specifică diaporamei analogice, a

devenit acum, doar una din nenumăratele posibilități de a trece de la o imagine la alta.

3.2.1 Studiul aplicațiilor software de prezentare

Aplicația de prezentare este un software folosit pentru afișarea de informații, de regulă sub forma unei diaporame. De obicei, include trei funcții majore: un editor ce permite inserarea și formatarea unui text, o metodă de inserare și manipulare a imaginilor grafice, și un sistem de prezentare ce afișează conținutul.

Primele aplicații de prezentări grafice rula de pe stații de lucru, ca și cele produse de Genigraphics, Autographix sau Dicomed. A devenit facilă intervenția de ultim moment, comparativ cu procedeul analogic. De asemenea, a devenit mult mai ușor să se producă un număr mare de slide-uri. Totuși, aceste stații necesitau operatori calificați, iar o singură stație de lucru necesita o investiție între 50.000 și 200.000 USD, la valoarea lor din 1979.

Dezvoltarea computerelor la jumătatea anilor '80 a schimbat modul de concepere a prezentărilor. Aplicații ieftine și specializate au adus la îndemâna oricărui posesor de PC sau Macintosh posibilitatea de a realiza prezentări cu aspect profesional.

Inițial, aceste programe au fost folosite pentru a genera informație ce era imprimată optic pe film reversibil (diapozitiv), pentru a fi prezentate ulterior cu ajutorul unui diaproiector. Pe măsură ce aceste aplicații deveneau comune la sfârșitul anilor '80, mai multe firme au pus la punct servicii care acceptau dischete cu imagini ce erau transpuse pe diapozitiv. În 1990, ecrane cu LCD ce erau adaptate la un retroproiector începeau să înlocuiască diapozitivele, pentru ca la rândul lor, acestea să fie înlocuite de videoproiectoare.

Prima aplicație comercială concepută special pentru prezentări de tip WYSIWYG (ceea ce vezi este ceea ce primești) a fost dezvoltată de Hewlett Packard în 1979 și numită BRUNO, iar mai apoi HP-Draw. Prima aplicație ce afișa o prezentare pe ecranul unui computer a fost VCN ExecuVision, dezvoltată în 1982. Permitea utilizatorilor să aleagă fișiere dintr-o bibliotecă de imagini, pentru a acompania textul unei prezentări.

Prezentul oferă multe tipuri de prezentări, incluzându-le pe cele profesionale, educative, de entertainment, pentru comunicare generală. Programele de prezentare pot completa sau înlocui mai vechea tehnologie vizuală. Textul, grafica, filmul sau alte obiecte pot fi poziționate pe pagini individuale, numite și slide-uri. Acestea pot fi tipărite sau, mult mai frecvent, afișate pe ecran și navigabile la comanda prezentatorului. Tranzițiile între imagini pot fi animate în diverse feluri. De obicei, o prezentare are multe constrângeri, iar una din cele mai importante e dată de timpul limitat pentru a afișa informații consistente. O aplicație de prezentare trebuie să ajute atât prezentatorul, cu un acces mai ușor la propriile idei, cât și participanții, cu informații vizuale ce completează discursul.

Multe aplicații oferă imagini predefinite (clip-art) și/sau au posibilitatea de a importa imagini grafice. Grafică personalizată poate fi creată și în alte softuri, gen Adobe Photoshop sau Adobe Illustrator, iar apoi importate. Conceptul de "clip-art" provine din biblioteca de imagini complementară aplicației VCN ExecuVision, din 1983.

Odată cu explozia fotografiei digitale, multe aplicații ce lucrează cu astfel de fișiere includ funcții de prezentare, pentru a le afișa într-un format similar diaporamei. Muzica poate însoți diaporama.

Similar extensiilor de programare pentru un sistem de operare sau pentru un browser de internet, „add-ons” sau plugin-uri ale aplicațiilor de prezentare pot fi folosite pentru a le mări capacitățile. De exemplu, ar fi utilă exportarea unei prezentări PowerPoint sub forma unei animații Flash sau a unui document PDF. Transmiterea lor cu ajutorul memoriilor portabile sau pe internet ar fi mai ușoară. Deoarece fișierele de tip PDF sunt concepute pentru a fi împărțite indiferent de platformă, și pentru că majoritatea browserelor au deja plugin-uri pentru vizionarea fișierelor de tip Flash, aceste formate ar permite diversificarea considerabilă a prezentărilor.

Anumite aplicații de prezentare oferă de asemenea un element hardware integrat, conceput pentru a capacita audiența sau a facilita prezentarea în diverse locații îndepărtate. Alte dispozitive ușurează activitatea prezentatorului (indicatoare laser sau table interactive).

Cel mai cunoscut și utilizat rămâne Microsoft PowerPoint, deși există alternative, gen OpenOffice sau Keynote al firmei Apple. Aplicațiile de prezentare online, dintre care se disting Zoho Show, Empressr, 280Slides și SlideRocket tind să devină viitoarea generație.

Diaporamiștii consacrați folosesc cu precădere PicturesToExe, ProShowGold sau mai complexul WingsPlatinum [40]. Asupra acestora voi insista cu prezentări detaliate, deoarece reprezintă un punct de plecare foarte important pentru studiul următor. Alte aplicații ce merită menționate, sunt: ACDSee, Adusoft Photo DVD Slideshow, AquaSoft Living Photo, Ashampoo Burning Studio, Diashow pro, DVD Slideshow Builder, Facebook, Flash, iPhone Slideshow, Magix PhotoStory, One-click Slideshow, Photo DVD Maker, Photo Manager, Photo Slideshow Builder, Pictures Slideshow Maker, Quick Slide Show, QR Photo, Silverlight, Slide Show Pilot, Picasa, Slideshow pro, Slideshow Suite, SWF 'n Slide, ULead PictureShow, Xatshow. Lista rămâne deschisă.

3.2.2 Microsoft PowerPoint

Microsoft PowerPoint [7] este o aplicație de prezentare dezvoltată de Microsoft. Parte a sistemului Microsoft Office, operează sub Windows și/sau Mac OS. Versiunea Windows poate rula sub sistemul de operare Linux, sub stratul de compatibilitate Wine.

Utilizat cu precădere de oameni de afaceri, educatori, studenți și traineri, PowerPoint e una din cele mai predominante forme de tehnologie persuasivă. Versiunile actuale sunt Microsoft Office PowerPoint 2007 pentru Windows și 2008 pentru Mac [41].

Prezentările PowerPoint constau dintr-un număr de pagini individuale, sau „slide-uri”, termen intrat ca neologism și în vocabularul nostru. În engleză, slide = diapozitiv, iar analogia se referă evident, la diapozitivul analogic. Slide-urile pot conține imagini, text, elemente grafice, filme, muzică, și alte elemente ce pot fi aranjate după dorință pe pagină. PowerPoint oferă diferite șabloane de prezentare, ce vin în sprijinul utilizatorului.

Rezultatul poate fi tipărit la imprimantă sau prezentat pe un computer, utilizatorul putând naviga prin prezentare la comandă. Publicul larg se poate bucura

3.2 Implementarea treptată a diaporamei digitale 31

de prezentare prin intermediul unui videoproiector. Slide-urile pot sta la baza unui webcast [42].

PowerPoint oferă trei tipuri de acțiuni :

1.Elementele de intrare, emfază și de ieșire ale unui slide sunt controlate de funcția numită animație personalizată (Custom Animation).

2.Tranzițiile reprezintă treceri între slide-uri. Ele pot fi animate în diverse feluri.

3. Animațiile personalizate pot fi folosite pentru a crea mici scenari prin animarea intrării, ieșirii și mișcării imaginilor.

În principiu, se atribuie expunerii slide-urilor o anumită durată (în secunde) și una din multele tipuri de tranziții. Acestea din urmă pot avea trei durate, rapid, mediu sau lent. In timpul vizionării, dacă dorește, prezentatorul poate comanda avansul imaginilor prin intermediul tastaturii sau a butoanelor mouse-ului.



Fig. 10 - Tipuri de tranziții ale PowerPoint.

Prezentările pot fi salvate în diferite formate, cele mai uzuale fiind tip prezentare (cu extensia .ppt) sau spectacol – PowerPointShow (cu extensia .pps). În primul caz, prezentatorul comandă desfășurarea prezentării, în al doilea aceasta are loc automat în funcție de programarea sa inițială. O coloană sonoră poate fi atașată prezentării [8].

Atât partizanii cât și criticii agreeau la începuturile prezentărilor digitale că facilitatea acestora poate economisi mult din timpul de lucru al celor ce ar fi folosit mijloace mecanice și/sau analogice în acest scop. Lejeritatea utilizării i-a încurajat până și pe cei mai reținuți să creeze prezentări audio-vizuale. Pe măsură ce stilul PowerPoint, animațiile, abilitățile multimedia, au devenit mai sofisticate, și pe măsură ce aplicația a devenit mai facilă, diferența dintre așteptări și realitate s-a aplanat [43] [44].

3.2.3 Analiza critică a principalelor aplicații pentru diaporamă

A devenit tot mai obișnuit să vedem diaporame digitale, personalizate pentru cei ce își folosesc fotografiile, muzica, orice document ce poate fi scanat. Unii le numesc diaporame, alții, prezentări, sau slideshow-uri. Ele pot fi create atât pe dvd video, dar și în format HD și fișiere executabile pentru PC, folosind aplicații specializate.

Diaporamele digitale oferă mult mai mult decât o prezentare a unor imagini; există posibilitatea de a adăuga tranziții, efecte de panoramare și zoom, videoclipuri, coloană sonoră, narațiune, explicații scrise, etc. Anumite aplicații specializate măresc puterea de editare a fotografiilor, într-un fel de neconceput până nu demult.

O aplicație de diaporamă include câteva funcții majore: editarea imaginilor (contrast, luminozitate, crop, rotire, etc), tranziții, coloana sonoră. Fișierele se exportă de la formatele DVD (MPEG2) sau CD-ROM, până la formate suportate de telefoane mobile (3GP), sau MP4, Microsoft Windows sau Apple Mackintosh.

3.2.3.1 Pictures to EXE

Este o aplicație profesională destinată utilizării sub Microsoft Windows, dezvoltată de WnSoft [45], [46]. Prețul variază între 49 USD pentru versiunea standard, și 75 USD, pentru cea deluxe. Este folosit pentru a crea diaporame digitale în fișiere executabile, HD video, pe DVD și export pe YouTube. Aplicația are o interfață simplă și intuitivă, ce permite utilizatorului obișnuit, cât și fotografului profesionist, să creeze diverse tipuri de diaporame. Pe suprafața utilă a slide-ului se pot insera atât imagini, cât și text). Oferă diverse tipuri de tranziții (cerc, romb, efect de răsfoire a paginii, 3D, etc) ce pot fi utilizate la latitudinea utilizatorului, precum și opțiunea asupra duratei fiecărei imagini, sau dacă fundalul, comentariile sau coloana sonoră sunt necesare unei anume imagini. Comentariile scrise pot fi inserate oriunde pe suprafața imaginii, existând opțiunea de alegere a fonturilor. Avantaj extraordinar, permițând autorilor să adauge o traducere a textului vorbit, în format de subtitrare [10], [47].

Fiecare imagine poate să fie personalizată cu o animație complexă, unde efecte de panoramare, transfocare, rotație, mascare pot fi adăugate. Imaginile pot fi sincronizate cu coloana sonoră. Calitatea imaginii finale este ireproșabilă.

Versiunea standard permite crearea unei diaporame în format executabil pentru PC, sau în format AVI. Versiunea Deluxe oferă și opțiunea HD video (MP4 H264), DVD precum și export pentru YouTube, Vimeo, iPhone/iPod. VideoBuilder-ul din ediția Deluxe permite chiar copierea pe dvd, fără a mai fi necesare aplicații dedicate, gen Nero. DVD-urile astfel realizate pot rula pe orice cititor DVD, fie că e vorba de o unitate separată, PC sau MAC. O diaporamă în format EXE poate fi rulată pe orice computer, fie că folosește Windows XP, Vista sau 2000. Niciun program adițional (nici măcar PicturesToExe) nu este necesar să fie instalat pentru a rula diaporama. Fișierul executabil poate fi protejat de o parolă, ce poate fi atribuită pornirii aplicației și/sau unei date de expirare. Butonul Print Screen poate fi blocat în timpul vizionării, din aceleași considerente ca și utilizarea unei parole [48].



Fig. 11 - Fereastră a aplicației PicturesToExe Deluxe 6.0

3.2.3.2 ProShow Gold

ProShow Gold, un produs al corporației Photodex [9], permite exportul diaporamei în formate EXE, sau DVD și CD - pentru redarea pe playerele DVD. Are de asemenea posibilitatea de a exporta diaporama în fișiere video MPEG pentru vizionare pe PC, precum și la crearea de screen-savere, și CD-uri auto-run. Prețul e asemănător cu cel al aplicației precedente, 70 USD pentru versiunea descărcată de pe pagina web a producătorilor, respectiv 80 USD pentru varianta fizică, în cutie. Există deasemenea și opțiunea de upgrade, la prețul de 45 respectiv 55 USD [49].

Când vine vorba de partajarea diaporamelor online, Photodex ProShow Gold excelează. Folosind un serviciu online gratuit Photodex, utilizatorul poate posta diaporama pe website-ul Photodex, și să invite la vizionare pe cei interesați. În testele utilizând o conexiune de bandă largă la Internet, redarea pe Internet a fost perfectă.

ProShow Gold se bazează pe o abordare tip storyboard a diaporamei de autor. Inițial, imaginile se selectează într-o fereastră ce le afișează în ordinea dorită. Coloana sonoră se include ulterior în spectacol. ProShow Gold permite alegerea din mai mult de 200 de tranziții între fiecare set de imagini. De fiecare dată când cursorul se rulează peste un efect de tranziție, se poate previzualiza desfășurarea tranziției [50].

ProShow Gold oferă și efectul de panoramare. De asemenea, se pot identifica puncte specifice ale unei fotografii pe care se poate mai apoi transfoca digital. Alte caracteristici includ posibilitatea de a adăuga fișiere video și/sau text. ProShow Gold este caracterizat de o interfață intuitivă care ajută mult la realizarea unei diaporame [51].

34 Implementări ale diaporamei digitale - 3



Fig. 12 - Fereastră a aplicației ProShow Gold, cu un proiect personal

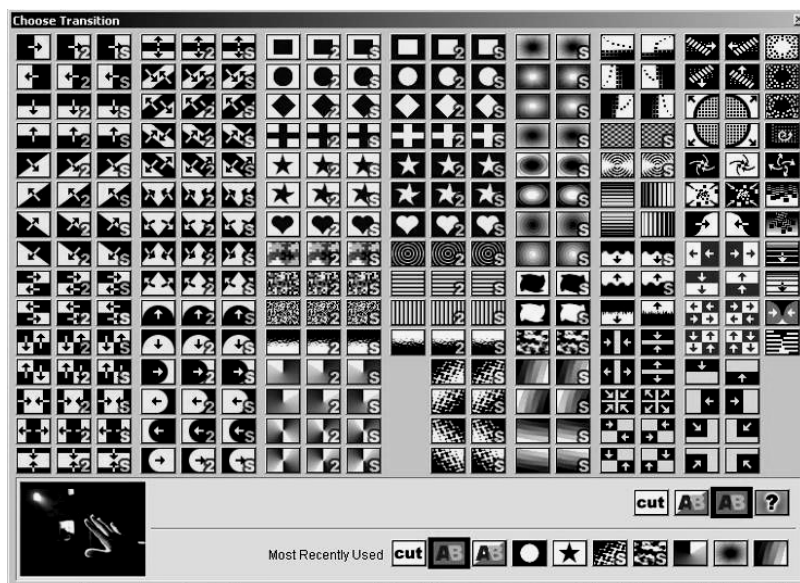


Fig. 13 - ProShow Gold oferă variante numeroase și diverse de tranziții.

Personal, folosesc această aplicație din 2005 și îi găsesc foarte multe calități. Cu ajutorul său reușesc să mă exprim din punct de vedere artistic [52], [53], iar calitatea tehnică a produsului finit este excelentă. Mă nemulțumește faptul că nu permite adăugarea de subtitrări, lucru necesar atunci când trimit la festivaluri internaționale de diaporamă lucrări cu text vorbit în limba română. Un alt neajuns,

întâlnit la toate aplicațiile de acest gen, este detaliat în următorul subcapitol, și face obiectul acestei teze.

3.2.3.3 Wings Platinum

Wings Platinum, un produs al firmei austriece Stumpf [11], oferă utilizatorului o interfață întunecată și elegantă. Designul permite utilizatorului să se concentreze asupra conținutului vizual, care devine imediat vizibil în momentul adăugării de imagini sau clipuri video la timeline. Platinum Wings permite organizarea și editarea oricărui tip de fișier media ca obiect în timeline. În acest fel, imagini de înaltă rezoluție, clipuri video HD și multitrack-uri audio se pot folosi pentru prezentări AV de o calitate deosebită.

Wings Platinum ofera posibilitatea creării unei prezentări prin plasarea de fișiere media, în timeline. Acestea pot fi imagini, sunete, videoclipuri, texte, comenzi de control de la distanță. Editarea acestora se bazează pe un concept unitar și pașii de editare sunt identici ori de câte ori este posibil. Pe timeline, diferitele fișiere media sunt aranjate ca obiecte în ordinea cronologică necesară. Orice editare este non-distructivă, adică toate fișierele media de pe hard disk rămân neschimbate și pașii de editare pot fi anulați sau modificați. Există anumite track-uri pentru diferite fișiere media, de exemplu, imagine, sunet și video, cu ieșiri specifice.

Ecranul este de tip proiecție virtuală, adică întreaga zonă este utilizată pentru proiectarea prezentării. Dimensiunea și formatul pot fi definite în mod liber. Ecranul poate fi afișat ca o fereastră sau în modul full-screen. Controlul key-frame permite mișcarea aparatului de fotografiat, efectul "imagine-în-imagine", mișcări în direcțiile X și Y, distorsiuni, transfocare, rotație a spațiului în jurul celor trei axe și modificări de culoare. Chiar și o combinație a tuturor acestor funcții este posibilă, care poate fi, de asemenea, aplicată la toate obiectele vizuale, și chiar și pentru video, text sau clipuri video în direct.

Fiecare prezentare are minim un timeline, care conține track-uri pentru aranjarea fișierelor media. Mai multe timeline-uri sunt adecvate pentru spectacole interactive sau în mai multe limbi. Track-urile permit tranziții atât pentru imagini statice, cât și pentru audio și video.



Fig. 14 - Fereastră a aplicației Wings Platinum

Imagini de aproape orice rezoluție pot fi integrate într-o prezentare Platinum Wings; acesta produce imagini cu o rezoluție de editare corespunzătoare, pentru a permite editarea rapidă în orice moment. După scalarea imaginilor, Platinum Wings va intensifica - opțional - claritatea (sharpening), pentru a asigura o calitate și claritate optimă a imaginii. Fiecare tranziție poate proiectată în mod individual, existând opțiunea de adăugare de puncte de inflexiune ce vor modifica durata/intensitatea acesteia. O gamă variată de preseturi pot fi folosite. Efecte vizuale dinamice, cum ar fi transfocarea, translația, rotația și controlul de culori, oferă efecte "imagine-în-imagine", sau de rotație spațială și înclinare a imaginii.

Fișiere audio de formate diferite pot fi folosite împreună într-un singur proiect. Conversia ratei de eșantionare a fișierelor MP3 este efectuată online în timpul redării. Creativitatea utilizatorului poate interveni, prin conceperea unor tranziții audio personalizate. Muzica de pe CD-uri audio se copiază fără pierdere. La cerere, Platinum Wings poate accesa o bază de date pe Internet pentru a afla date despre autor și disc. Fișierele audio pot fi exportate foarte flexibil într-o varietate de formate diferite.

Editarea de filme video se face în timp real. Output-ul (EXE, AVI / MPEG video, DVD-video, etc) se bazează pe date de înaltă rezoluție pentru a obține calitatea optimă. Pentru a asigura redarea optimă de material întrețesut, Platinum Wings conține o caracteristică de deîntrețesere de calitate înaltă. Exportul se poate face în formate MPEG 2, AVI sau WMV.

Obiecte de tip text pot fi utilizate pentru titluri într-o prezentare. Textul este afișat pe un fond transparent și poate fi combinat cu diferite fundaluri. Textul poate fi animat atât pe plan orizontal cât și vertical. Practic, toate efectele controlate key-

frame, cum ar fi de mișcare, de rotație și de control de culori sunt disponibile pentru obiecte de tip text.

Wings are, de asemenea, posibilitatea de a controla direct proiectoarele analogice de diapozitive printr-o interfață serială. Mediul de prezentare poate fi o bandă magnetică, un CD audio cu semnale de control sau un card CF.

Toate versiunile Platinum Wings, de exemplu, permit editarea de clipuri video DV-AVI la rezoluție standard și export MPEG 2, pentru DVD-uri video. Formatul video HD, pe de altă parte, este rezervat doar pentru anumite versiuni.

Prețurile încep cu 275 Euro pentru versiunea Starter, cresc la 550 Euro pentru Advanced, 885 Euro pentru Pro, ajungând la 1530 Euro pentru versiunea Multidisplay. Cu toate acestea, faptul că Stumfl oferă gratuit versiunea Basic este pe de o parte meritoriu, dar și strategic din punct de vedere al plasării pe piață.

3.2.3.4 Neajunsuri ale aplicațiilor digitale

Principalul neajuns al aplicațiilor prezentate anterior e dat de lipsa de interactivitate specifică prezentărilor analogice. Programând un spectacol de diaporamă, acesta poate fi rulat oriunde, fără a mai fi necesară prezența autorului. Pe de altă parte, după primul val al euforiei digitale, mulți autori resimt lipsa dispozitivelor analogice, care permiteau o interactivitate sporită. Diaporamei actuale îi lipsește atingerea artistului, posibilitatea de a improviza în timpul prezentării, de a modifica efectele de trecere dintre două imagini, de a opri după dorință tranziția imaginilor într-un anumit moment sau capacitatea de a schimba fondul sonor din mers și de a-l adapta la o situație neprevăzută. Deși o lucrare simfonică va suna fără desincronizări pe sintetizatoare MIDI, îi vor lipsi acele mici imperfecțiuni ce o fac umană și unică.

Prin comparație, proiecțiile de diaporamă analogică ofereau prezentatorului posibilitatea de a asculta coloana sonoră și de a stabili momentul apariției pe ecran a unei imagini, de a decide timpul în care acea imagine apare de la 0 la 100% pe ecran, de a derula prezentarea înainte sau înapoi. Diaporama analogică era în bună măsură o artă interpretativă și mulți autori au profitat din plin de acest avantaj. Faptul că tehnica a permis înregistrarea prezentărilor a fost absolut binevenit, dar chiar și acea înregistrare se baza pe o interpretare a autorului, și reda cu acuratețe comenzile proiectantului asupra diaporamei.

Acest dezavantaj, poate subiectiv, mi-a dat ideea întoarcerii la metodele analogice de prezentare a unei diaporame, dar folosind tehnologia digitală disponibilă. Continuarea acestei teze reprezintă un studiu prin care am încercat să ofer soluții de întoarcere la metodele analogice de proiecție a unei diaporame, folosind tehnologia și cunoștințele mileniului III.

4. Elaborarea și implementarea primului model de aplicație. "Digital Diaporama"

4.1 Aplicația „Digital Diaporama”.

4.1.1 Prezentare generală

Am conceput „Digital diaporama” în ideea unui studiu asupra posibilității de a transfera spectacolul analogic de diaporamă (așa cum era realizat până nu demult cu ajutorul proiectoarelor, diapozitivelor, magnetofonelor ori casetofonelor) în realitatea zilelor noastre. Odată cu avansul luat de tehnologie, aducerea pe piață a unor procesoare din ce în ce mai rapide, precum și datorită evoluției fabuloase pe care au avut-o camerele foto digitale, s-a făcut pasul către prezentările multimedia actuale. Diapozitivele au fost înlocuite de imagini digitale, diaproiectorul este înlocuit cu videoproiectorul, sunetul redat de magnetofon este înlocuit de o aplicație care citește fișierele audio în format digital. Aplicația de diaporamă digitală încearcă să înglobeze cât mai multe din caracteristicile spectacolului clasic de diaporamă, să îi confere acestuia un grad mai înalt de personalizare, încercând în același timp să fructifice toate avantajele date de noua tehnologie ce ne stă la dispoziție [54].

4.1.2 Structura aplicației

Orice aplicație software ce lucrează cu fișiere, fie că sunt fișiere grafice, audio, video, este gândită să aibe cel puțin două părți distincte, fiecare cu funcționalitatea sa bine definită și pe deplin justificată. Una dintre ele are menirea de a se ocupa de interfața utilizatorului cu fișierele propriu-zise, cealaltă realizează exclusiv prelucrarea datelor de intrare, în speță fișierele selectate prin intermediul funcționalității anterioare.

În cazul aplicației „Digital diaporama”, am proiectat structura aplicației în așa fel încât să delimitez bine cele două funcționalități. Astfel, o funcționalitate mare realizează interfața dintre utilizator și fișierele grafice pe care acesta dorește să le folosească în diaporama sa, respectiv cea de-a doua realizează prelucrarea datelor de intrare luând naștere astfel prezentarea de diaporamă dorită.

Fiecare dintre acestea au facilitățile lor specifice și dedicate unor anumite acțiuni ce pot fi cerute de către utilizator în demersul său. În paragrafele ce urmează, voi descrie funcționalitățile în detaliu, de la structura acestora la metodele pre-definite ce au fost folosite pentru a implementa o anumită cerință, de la problemele întâmpinate sau cele probabile într-o eventuală dezvoltare a aplicației până la ideea ce stă în spatele fiecărei clase sau metode nou create.

4.1.3 Interfața cu utilizatorul

Orice funcționalitate de acest tip trebuie proiectată și gândită de la început cu scopul de a fi cât mai facilă pentru un potențial utilizator. Interfața cu acesta trebuie fie capabilă să realizeze toate cerințele proiectului și să vină în ajutorul celui ce o folosește.

În cazul particular al „Digital diaporama”, am conceput interfața cu utilizatorul pentru a căuta și selecta fișierele pentru spectacolul de diaporamă, într-o manieră cât mai intuitivă.

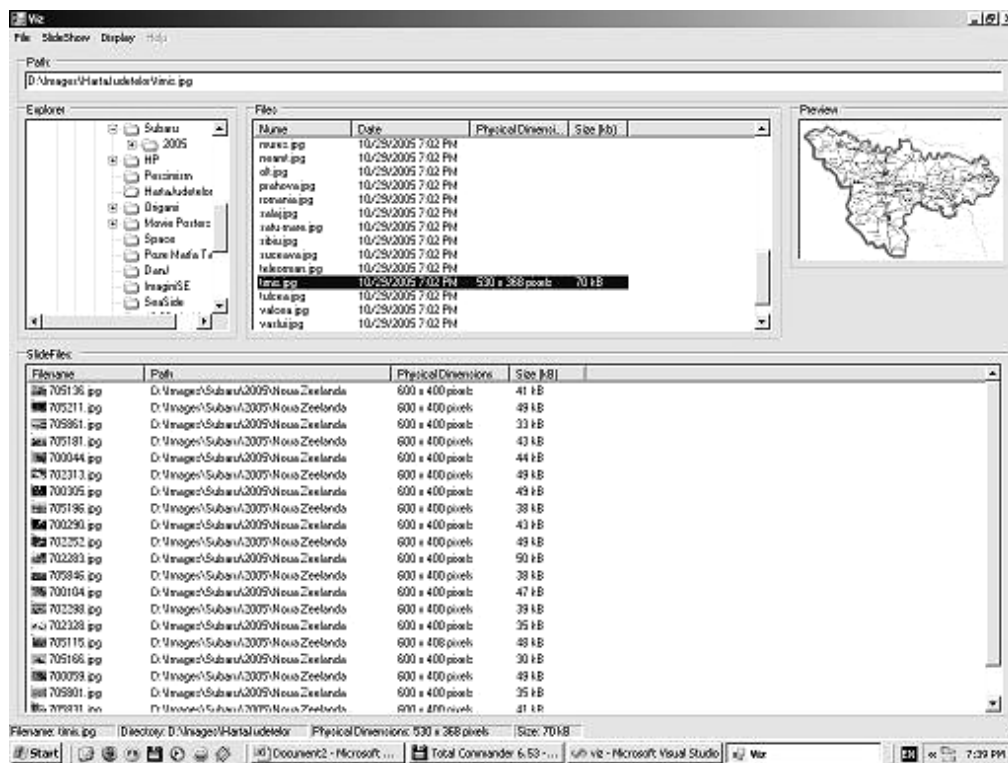


Fig. 15 - „Digital diaporama” - interfața cu utilizatorul

Analizând imaginea de mai sus, se pot distinge mai multe zone în care a fost împărțită fereastra principală a aplicației. Fiecare dintre acestea are rolul său bine definit în funcționalitatea interfeței.

Am dorit ca de la bun început să ofer utilizatorului posibilitatea de a alege pentru spectacolul său de diaporamă fișiere grafice din mai multe directoare ale hard-disk-ului, eliminând astfel restricțiile date de folosirea unui singur director pentru un anumit spectacol. În partea de sus a ferestrei am poziționat un „file browser” clasic destinat în totalitate fișierelor grafice, iar în partea de jos un obiect listă ce conține la un moment dat toate fișierele ce se dorește a face parte din spectacolul de diaporamă.

4.1.4 Navigarea pe hard-disk

File-browser-ul l-am alcatuit din câteva obiecte plasate într-un mod intuitiv, asemenea oricărei aplicații de acest tip. Pentru a se naviga spre imaginile de pe hard-disk se folosesc două obiecte tip "Windows Explorer", unul ce conține lista de directoare până la cel curent, respectiv altul care conține fișierele grafice, filtrate după extensie, din directorul în care ne aflăm.

Obiectul dedicat directoarelor de pe hard-disk este asemenea celui din "Windows Explorer" având aceleași imagini simbol pentru diferitele tipuri de directoare, o structură arborescentă prin care se poate naviga cu ușurință respectiv aceleași modalități de comandă. De exemplu, se folosesc aceleși dosare de culoare galbenă pentru a simboliza un director obișnuit, aceleași simboluri de "+", respectiv "-" pentru a indica un director închis, respectiv deschis sau se folosește un simplu click pentru a schimba directorul curent.

Am proiectat lista cu fișierele grafice din directorul curent pentru a afișa toate detaliile necesare despre o imagine selectată la un moment dat. Astfel, la încărcarea listei cu imaginile din directorul curent, se citește și se afișează în aceasta, pe lângă numele fișierului, și informații precum data la care acesta a fost creat ori ultima dată când a fost modificat, dimensiunile imaginii originale în pixeli sub forma <dimensiunea imaginii pe orizontală x dimensiunea imaginii pe verticală pixels>, precum și dimensiunea în kilo-octeți a acestuia. Pentru a solicita cât mai puțin resursele sistemului de operare și implicit a încetini execuția aplicației, datele despre fișiere se citesc doar la previzualizarea pentru prima dată a imaginii.

Pentru a se da și mai multă informație legată de calea de pe hard-disk unde se află utilizatorul, conținutul unui obiect de tip text afișează în orice moment calea completă și numele fișierului selectat, dacă este cazul.

4.1.5 Previzualizarea

Tot în această secțiune, am prevăzut aplicația cu o fereastră de previzualizare a imaginii, plasată în partea dreaptă a listei de fișiere grafice din directorul curent. Funcția de previzualizare oferă utilizatorului informație suplimentară asupra fișierului grafic ce urmează a fi introdus în lista spectacolului de diaporamă. Imaginea în discuție este redată scalat la dimensiunea predefinită a obiectului de previzualizare.

4.1.6 Lista de fișiere a diaporamei

Lista din partea de jos a ferestrei principale, denumită și listă "SlideFiles", conține doar informații despre imaginile ce pot face parte la un moment dat din spectacolul propriu-zis de diaporamă. Adăugarea unui fișier se poate face doar după ce în prealabil a fost deschis directorul în care acesta este poziționat. Asemenea listei ce afișează fișierele grafice din directorul curent, lista fișierelor pentru diaporamă pune la dispoziția utilizatorului informații suplimentare despre imaginile selectate. Fiecare imagine adăugată atrage după sine înscrierea în listă a unor date suplimentare referitoare la caracteristicile acesteia în contextul spectacolului de diaporamă. Dacă se vizualizează lista în modul în care sunt afișate toate detaliile despre imagini, acestea vor conține o previzualizare scalată a imaginii la

dimensiunile unui pătrat de 16 x 16 pixeli pentru a conferi o informație succintă asupra acesteia, numele acesteia însoțit de extensia caracteristică fișierelor grafice, calea completă de la care a fost încărcată imaginea, dimensiunile imaginii originale în pixeli sub forma <dimensiunea imaginii pe orizontală x dimensiunea imaginii pe verticală pixels>, dimensiunea în kilo-octeți a acesteia, precum și informații legate de timpul ce determină modul în care este folosită imaginea în ansamblul spectacolului de diaporamă. După ce mai multe imagini au fost adăugate în lista de slide-show, utilizatorul poate modifica după voia sa ordinea în care acestea vor apărea în spectacol, prin utilizarea facilității de "drag & drop", pe care am implementat-o.

4.1.7 Bara de status

Tot în încercarea de a furniza celui ce se află la comanda aplicației cât mai multă informație disponibilă într-un mod cât mai facil, am inclus în "Digital diaporama" o "bară de status" ce vine să repete informațiile despre fișierul curent selectat. Fie că se realizează selecția unei imagini prin intermediul listei din "file-browser", fie că se selectează o imagine din lista de diaporamă, bara de status își va actualiza conținutul, afișând caracteristicile acesteia și anume: numele și extensia fișierului selectat, calea completă de la care acesta a fost încărcat, dimensiunile pe verticală respectiv orizontală în pixeli, precum și spațiul în kilo-octeți, ocupat în memorie de către aceasta. În situația în care nici o imagine nu este selectată, implicit previzualizată, "bara de status" nu va afișa niciun fel de informație.

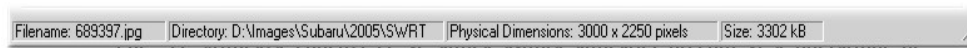


Fig. 16 - Bara de status a aplicației "Digital Diaporama"

4.1.8 Bara de meniu

Pentru a crește manevrabilitatea aplicației, am prevăzut-o cu o bară de meniu ce se află asemenea tuturor obiectelor de acest tip în partea de sus a ferestrei principale. Aceasta conține majoritatea comenzilor ce pot pleca din direcția utilizatorului către aplicație, grupate intuitiv după destinația asupra unde se dorește a fi folosite. În momentul de față, aceasta bară de meniu are în componență obiecte ca și "File", ce grupează comenzi ce se aplică asupra aplicației precum și a operațiilor de intrare-ieșire de care aceasta este capabilă, "SlideShow" ce conține toate comenzile ce pot fi introduse în contextul static al diaporamei, "Display" cu ajutorul căruia se poate selecta modul în care este organizată fereastra principală a aplicației sau meniul de "Help" ce conține scurtături către facilități auxiliare.

La rândul lor, fiecare din aceste meniuri conțin obiecte care ușurează comandarea aplicației.

Meniul "File" conține scurtăturile către comenzi precum "Load SlideShow", "Save SlideShow" sau "Exit". Consider redundantă explicarea comenzii "Exit", aceasta fiind arhicunoscută și folosită în majoritatea aplicațiilor proiectate pentru sistemele de operare bazate pe conceptul "ferestrelor". La aceasta se adaugă

comenzi în strânsă legătură cu lista de imagini destinate spectacolului de diaporamă. Această listă, odată creată și setată, poate fi, dacă se dorește, salvată într-un fișier exterior cu extensia ".ssf" (slide show file), acesta putând fi la rândul lui încărcat de aplicație. În acest fel se restaurează spectacolul de diaporamă odată creat. Cele descrise mai sus se realizează prin acționarea item-urilor "Save SlideShow", respectiv "Load SlideShow" ce vor deschide niște dialoguri de tip "Save As *.ssf" respectiv, dacă vorbim despre deschiderea unui fișier, a unui dialog "Open *.ssf".

Meniul "SlideShow" conține scurtături către comenzi ce se aplică asupra listei de fișiere ce urmează să formeze spectacolul de diaporamă. Astfel, putem întâlni comenzi de adăugare de fișiere, de ștergere ale acestora din listă, precum și comanda ce îi transmite aplicației să intre în modul prezentare. Toate aceste comenzi au fost predefinite cu "scurtături" de la tastatură pentru a face introducerea acestora mai rapid odată cu aprofundarea modului de operare al aplicației. Operațiile de adăugare, respectiv ștergere, se pot efectua asupra unei singure imagini sau asupra tuturor imaginilor din liste. Astfel, spre exemplu, pentru adăugarea unei singure imagini din directorul curent, este suficient să acționăm asupra item-ului SlideShow\Add sau să apăsăm tasta Insert. Rezultatul acestei acțiuni va fi modificarea listei diaporamei prin adăugarea la sfârșitul acesteia a imaginii selectate. Ștergerea din listă a unui fișier funcționează analog operațiunii de adăugare. Odată selectat în lista diaporamei, utilizatorul are posibilitatea de a șterge această intrare prin acționarea item-ului SlideShow\Remove File sau prin combinația de taste Ctrl+D. Posibilitatea de a șterge unul sau toate fișierele din lista de diaporamă nu va fi activă decât în momentul în care aceasta va conține cel puțin un element.

Meniul „Display” colectează comenzile ce țin de aspectul imaginii principale a aplicației. Astfel, sunt disponibile reglaje asupra modalității în care sunt afișate fișierele în liste, asupra modalității de aranjare a obiectelor în fereastra principală sau asupra vizibilității unor obiecte sau proprietăți de către utilizator. Lista ce găzduiește fișierele destinate viitorului spectacol de diaporamă are mai multe variante de vizualizare, fiecare cu avantajele și dezavantajele sale. Implicit, imaginile selectate sunt afișate cu toate informațiile secundare la vedere, în mod „Details”. Există și alte posibilități de afișare, de exemplu în mod „Thumbnails”, caracterizat de faptul că imaginile sunt afișate fără nici un fel de informație secundară, fiind vizibile doar un pătrat având latura de 96 de pixeli (o previzualizare scalată) și numele fișierelor. Modul de afișare „Small Icons” este foarte asemănător cu cel descris anterior cu diferența că previzualizarea imaginii se face scalată la un pătrat cu latura mai mică, de doar 16 pixeli. Fiecare din aceste variante de afișare are avantajele și dezavantajele sale. „Details” este cel mai complet, clasificând modurile după cantitatea de informație oferită. Acesta afișează lista completă a informațiilor suplimentare precum calea de la care a fost încărcată imaginea, data la care a fost ultima dată modificată ori spațiul în kilo-octeți ocupat pe disc. „Details” însă nu oferă o previzualizare de o calitate ridicată, din pricina dimensiunii reduse a acesteia (pătrat cu latura de 16 pixeli). Modul de vizualizare „Thumbnails” este exact opusul față de modul „Details”, acesta oferind o previzualizare a imaginilor mai bună, însă informațiile suplimentare despre fișier lipsesc cu desăvârșire. Cea de-a treia variantă „Small Icons” nu are nici unul din avantajele modurilor prezentate mai sus, el afișând doar numele fișierului însoțit de previzualizarea imaginii grafice scalată asemenea modului „Details” la un pătrat cu latura de 16 pixeli. Modul de vizualizare „Small Icons” se pretează mai ales diaporamelor cu foarte multe imagini.

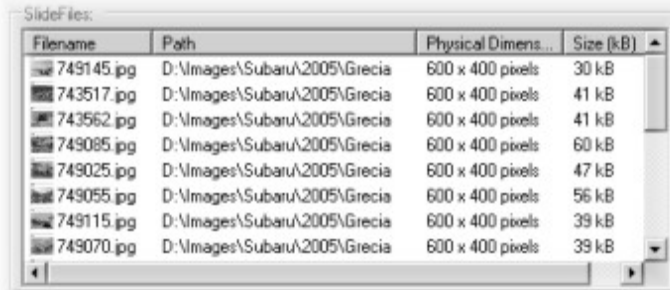


Fig. 17 - Modul de afișare „Details” al aplicației "Digital Diaporama"



Fig. 18 - Modul de afișare „Thumbnails” al aplicației "Digital Diaporama"

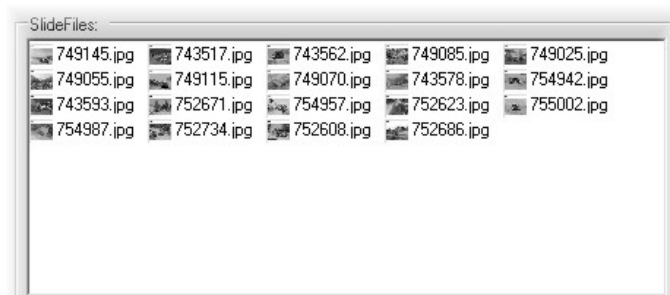


Fig. 19 - Modul de afișare „ Small Icons” al aplicației "Digital Diaporama"

Meniul „Help” adună toate comenzile secundare ale aplicației și care nu au efect concret asupra funcționării acestora. De aici se poate apela, de exemplu, comanda de afișare a informațiilor despre autor ori despre mediul în care aplicația este rulată.

4.1.9 Fereastra de diaporamă

În clipa în care toate imaginile ce vor face parte din spectacolul de diaporamă sunt introduse în lista corespunzătoare din fereastra principală de interfață cu utilizatorul, sunt îndeplinite toate condițiile pentru a da startul la succesiunea de imagini, prin accesarea submeniului SlideShow->FullScreen, sau prin folosirea combinației de taste Ctrl+F.

Ecranul de pânză sau vinilin pe care se proiectau imaginile diaporamei analogice este cel mai frecvent înlocuit cu ecranul monitorului calculatorului personal sau cu videoproiectorul. Acest nou ecran pe care se vor succeda imaginile digitale este de fapt cea de-a doua fereastră a aplicației "Digital Diaporama", o fereastră cu fundal de culoare neagră, întotdeauna ocupând tot ecranul, fiind în permanență maximizată.

Complexitatea acestei ferestre a aplicației nu poate concura, raportat la numărul obiectelor ce sunt găzduite, fereastra interfață cu utilizatorul, din simplul motiv că nu este necesar.

4.1.10 Obiectul imagine

Cel mai important obiect prezent pe fereastra de diaporamă este cel ce va primi imaginile diaporamei și în același timp cel ce va primi rezultatul algoritmilor de tranziție ale celor două imagini ce se succed. Acesta este un obiect de tip "PictureBox" [55] și prin codul implementat l-am gândit a fi centrat relativ la mijlocul ecranului.

4.1.11 Panoul de comandă al diaporamei

Cel de-al doilea obiect ca importanță al ferestrei este cel care grupează toate elementele de control ale spectacolului de diaporamă. Asemeni poziționării obiectului descris mai sus, am implementat și pentru acesta codul astfel încât în permanență să fie centrat în partea de jos a ecranului în funcție de înălțimea pe care o are.

Punctul central al acestui panou de comandă este o listă care conține toate imaginile ce au fost selectate în prealabil pentru a face parte din spectacolul de diaporamă. Pentru fiecare imagine a fost creat un element de previzualizare asemeni celui ce era prezent în cadrul listei imaginilor diaporamei, listă vizualizată în modul "Thumbnails". Astfel utilizatorul primește informații despre imaginile selectate precum numele și extensia fișierului grafic, respectiv o imagine previzualizată, destul de sugestivă. De asemenea, această listă poate avea rol în navigația prin structura ce va forma spectacolul de diaporamă.



Fig. 20 -Panoul de control al diaporamei

Elementele care trebuie să primească atenție în primul rând sunt butoanele, care au rol în navigația în lista de imagini ce formează diaporama. În număr de două, acestea conferă utilizatorului funcționalitatea de a naviga "înainte" și "înapoi" în funcție de indexul imaginii prin lista de fișiere grafice. După cum se vede și în imaginea de mai sus, acestea sunt protejate împotriva depășirii granițelor în care variază indexul. În cazul în care se vizualizează prima imagine din listă, butonul de navigare "înapoi" este inactiv, respectiv dacă se vizualizează ultima imagine din listă, butonul de navigare "înainte" este de asemenea inactiv.

Pentru a rămâne în domeniul modului de afișare al imaginilor în fereastră, am conceput aplicația de diaporamă digitală pentru a reda imagini de dimensiuni diferite, ce pot fi mai mari sau mai mici decât rezoluția ecranului pe care se efectuează afișarea. Astfel, panoul de control al diaporamei oferă posibilitatea utilizatorului de a modifica modul în care este redată o anumită imagine, micșorându-i dimensiunile ori amplificându-le pentru a fi adaptate la capacitatea de redare a ecranului. Aceste operații de micșorare, ori amplificare se fac proporțional, astfel că aspectul imaginii de redat nu se schimbă ca rezultat al acestor modificări. Odată selectată una din opțiunile "Shrink" sau "Enlarge" prin bifarea căsuțelor prevăzute de panoul de control, algoritmul ce stă în satele acestor funcționalități calculează și adaptează imaginile la noile dimensiuni și coordonate.

4.1.12 Efectele de tranziție între imagini

Aplicația de diaporamă digitală este capabilă de a realiza tranziții de tip „cross-dissolve” între imagini, întocmai cum se obișnuia în cadrul spectacolelor de diaporamă analogică. Efectul „cut” este în sine o trecere instantanee de la imaginea curentă la cea următoare, fără „topirea” acestora.

Cel de-al doilea efect disponibil este cel de „contopire” a imaginilor, ce are ca și corespondent efectul „cross-dissolve” din diaporama analogică. Contopirea simulează proiectarea imaginilor pe ecran cu lumină de intensitate diferită, însă complementară.

Selectarea tipului de trecere între imagini se face prin intermediul butoanelor tip „Radio” din partea dreaptă-sus a a panoului de control al diaporamei.

În cazul în care utilizatorul selectează opțiunea de trecere „No effect”, la apăsarea butonului de trecere la imaginea următoare, imaginea precedentă va dispărea de pe ecran, în locul ei fiind redată imaginea următoare, asemenea reducerii la zero a intensității luminii ce stă în spatele unui diapozitiv și ridicarea la maxim a nivelului intensității luminii cu care se proiectează celălalt diapozitiv, raportat la diaporama analogică.

Pentru situația în care este selectată opțiunea de trecere „Fade effect”, la apăsarea butonului de trecere la următoarea imagine, cele două imagini se vor contopi, pe măsură ce timpul predefinit pentru durata efectului se epuizează. Această contopire a celor două imagini poate fi comparată cu trecerea „cross-dissolve”, sau „topire înlănțuită”, pe care s-au bazat cele mai multe spectacole de diaporamă analogică. Simularea modificării progresive a intensității luminoase ce stătea în spatele diapozitivelor folosite în trecut se realizează prin modificarea atributului de transparență a celor două imagini ce urmează a se afișa simultan. Dacă la începutul tranziției avem intensitate luminoasă maximă în spatele primului diapozitiv, respectiv intensitate luminoasă zero în spatele celui de-al doilea, aplicația de diaporamă digitală afișează cele două imagini cu transparență diferită, minimă pentru imaginea de la care se pleacă și respectiv maximă, pentru imaginea la care se dorește a se ajunge. Transparența 100% este echivalentă cu invizibilitatea. Pe durata efectului, fie că el se desfășoară într-un timp determinat (cazul trecerilor predefinite), ori că durata acestuia nu este dinainte cunoscută, în cazul tranzițiilor dintre imagini controlate manual, transparența celor două imagini succesive se modifică pentru a simula redarea cu intensitate luminoasă variabilă a imaginilor. Dacă intensitatea luminoasă ce ar sta în spatele primului diapozitiv ar trebui să scadă și în același timp intensitatea luminoasă ce ar sta în spatele celui de-al doilea diapozitiv ar trebui să crească, raportând totul în digital, trebuie să reprezentăm prima imagine cu transparență din ce în ce mai mare, respectiv cea de-a doua imagine trebuie redată cu transparență cât mai redusă, adică opacitate din ce în ce mai mare.



Fig. 21 - Exemplificarea tranziției imaginilor

4.1.13 Contopirea imaginilor

Efectul de contopire a imaginilor ce se succed în spectacolul de diaporamă este elementul definitoriu al acestuia. În cazul diaporamelor digitale ce sunt disponibile în acest moment publicului larg, efectele incluse sunt toate predefinite a se desfășura pe o anumită durată de timp. Acesta este și cazul aplicației de diaporamă digitală pentru care sunt implementate două tipuri de tranziție, unul de tip „cut”, ce nu presupune existența unui efect anume, iar celălalt de tip „cross-dissolve”, ce realizează trecerea de la o imagine la alta prin schimbarea succesivă a proprietății de transparență pentru cele două imagini în cauză. Pentru efectul din urmă sunt disponibile mai multe variante de control al modului prin care acesta se desfășoară.

Am pus la dispoziție utilizatorului atât durata în care are loc efectul cât și numărul de pași prin care acesta tranziționează. Spre exemplu, dacă se dorește o trecere normală în cadrul unui spectacol de diaporamă, se poate selecta o durată de 3 secunde realizată din 45 de pași (15 pași/secundă). Acest lucru se realizează prin modificarea indicatorilor din cadrul grupului Effects (Effect Time, Effect Steps) la valori de 3000 ms, respectiv 15.

Algoritmul ce realizează un anumit număr de treceri într-o secundă a reprezentat o parte mai problematică a studiului, deoarece există limite superioare din punctul de vedere al numărului de pași ce pot fi realizați. Cele mai importante variabile ce intră în ecuația ce stă la baza capacității de procesare a efectului sunt dimensiunea imaginilor ce trebuie să fie parcurse secvențial, și puterea de procesare a sistemului pe care se rulează aplicația. Dacă se dorește redarea unor imagini de dimensiuni mari, de ordinul megapixelilor, este necesară în primul rând scalarea în

interiorul aplicației ale acestora la dimensiunile și rezoluția ecranului, iar ulterior procesarea imaginilor scalate. Aceasta necesită rularea unui bloc mai mare de instrucțiuni, mare consumator de timp, rezultând o redare corectă a unui număr mai redus de pași decât cel ce a fost în prealabil precizat.

Concluzionând referitor la dependența dintre dimensiunea imaginilor ce compun succesiunile și numărul de pași ce pot fi redați, se observă o variație invers proporțională între aceste date. Cu cât imaginile au mai mulți pixeli, cu atât e mai dificil de redat mai puțini pași într-o secundă de efect de contopire.

Din experiența dobândită în urma testelor efectuate, precum și din alte studii [51], s-a dovedit că o o trecere de 15 pași într-o secundă asigură un confort vizual satisfăcător privitorilor, neexistând o diferență prea mare de transparență între imaginile intermediare. Dacă e să introducem în problemă și dimensiunea imaginilor, spre exemplu pentru un format de 1600 x 1200 pixeli, nu se va putea obține un număr suficient de pași pentru a asigura confortul vizual despre care am vorbit, redându-se doar un număr aproximativ de 10 pași de transparență într-o secundă. Dacă scădem dimensiunea imaginilor până la 800 x 600 pixeli, se obține cel mai bun compromis între calitatea imaginilor redate și numărul de treceri reprezentate, aceasta dacă se rulează aplicația pe un sistem cu performanțe medii din punctul de vedere al puterii de procesare.

Această problemă este rezolvată de alte soluții software de acest tip, prin pre-procesarea imaginilor ce vor fi incluse în efectul ce urmează a fi redat. Rezolvarea menționată nu se poate aplica însă în cazul de față deoarece nu se poate stabili o formulă ce guvernează toate tranzițiile dintre oricare două imagini. Am dorit ca aplicația „Digital diaporama” să se poată baza pe o mare interactivitate din partea utilizatorului în procesul ce stă la baza efectelor de tranziție. Intervenția utilizatorului în procesul de contopire elimină orice urmă de predictibilitate în algoritmul de trecere dintre imagini și a forțat orientarea codului către eficiență.

4.1.14 Butoanele cu durată predefinită a trecerii

Panoul de control al diaporamei digitale permite rularea efectului de contopire a imaginilor în modul manual, pe durată de timp variabilă, accesibilă utilizatorului prin apăsarea unor taste, sau a unor butoane dacă panoul de control este vizibil. Tranzițiile dintre imaginile succesive se pot realiza pe durată predefinită, de la una până la nouă secunde, prin apăsarea tastelor de la 1 până la 9 al blocului numeric al tastaturii. Aceste treceri predefinite se vor efectua cu atâția pași cât este specificat de către blocul „Effect Steps” descris într-unul din paragrafele anterioare.

De exemplu, dacă se dorește o trecere de două secunde printr-un anumit număr de pași, se selectează în prealabil numărul de pași dorit, iar mai apoi se apasă tasta „2” sau butonul ce indică „2s” din grupul „Predefined”. Această acțiune va determina creșterea indexului imaginii din listă, lucru similar cu navigarea în direcția „înainte” prin aceasta. Este posibilă și navigarea cu timp predefinit în sens invers prin lista de imagini, o așa numită navigare „înapoi”, funcționalitate disponibilă utilizatorului în mod asemănător, prin utilizarea blocului numeric al tastaturii concomitent cu apăsarea tastei „Ctrl”. De exemplu, dacă este realizată combinația de taste „Ctrl+3” ori dacă se apasă tasta „Ctrl” și se acționează cu mouse-ul asupra butonului ce va indica „-3s”, iar imaginea actuală nu se află pe

prima poziție din listă, se va efectua o trecere către imaginea anterioară pe durata a trei secunde.

4.1.15 Controlul automat al derulării imaginilor

Asemenea oricărei aplicații de diaporamă digitală, cea pe care am conceput-o oferă posibilitatea utilizatorului de a programa automat derularea imaginilor pe ecranul calculatorului/videoproietorului. Fiind în prealabil selectată durata pe care se desfășoară trecerile, și selectând modul de derulare automat al acestora, spectacolul de diaporamă va rula singur, până la intervenția celui ce are acces la tastatură, dacă acesta consideră că este necesar să intervină.

Durata redării pe ecran a două imagini succesive se selectează prin modificarea valorii din casuța corespunzătoare plasată în cadrul grupului „SlideShow Interval”. Spre exemplu, dacă se dorește rularea unui spectacol cu o iterație de 2 secunde între imaginile succesive, se va modifica valoarea căsuței menționate mai sus la valoarea de 2000 ms.

Pornirea diaporamei automate se realizează la bifarea căsuței cu ce are indicativul „Enable SlideShow”. În acest moment, un sistem de temporizatoare interne ale aplicației se activează și determină evenimente ce sunt folosite în algoritmul de tranziție dintre imagini. Aceste aspecte vor fi descrise pe larg în partea ce va avea ca și punct de orientare descrierea codului utilizat.

Sunt disponibile două sensuri de a rula spectacolul de diaporamă; cu derulare „înainte”, în acest fel imaginile se succed în ordinea în care apar în lista din centrul panoului de control și respectiv cu derulare „înapoi”, caz în care imaginile se succed pe ecran în ordinea inversă celei în care acestea au fost adăugate în listă. Aceste moduri de navigare se pot selecta odată cu activarea controlului automat al derulării diaporamei, prin bifarea uneia din casuțele „Forward” ori „Backward” (Înainte sau Înapoi) după caz. Implicită este derularea imaginilor în sensul în care acestea au fost adăugate de utilizator în lista de imagini, radio-butonul „Forward” fiind bifat. Este obligatoriu ca întotdeauna când derularea automată a imaginilor este activă să fie bifată una din opțiunile ce specifică sensul de derulare al acestora.

4.1.16 Controlul manual al tranziției imaginilor

Funcționalitatea ce apropie cel mai mult aplicația "Digital diaporama" de spectacolele de diaporamă analogică și o deosebește ca și concept de realizare de alte aplicații de prezentare este funcționalitatea de a controla manual timpul de tranziție dintre imagini. Celui ce face prezentarea digitală, asemenea celor ce au făcut-o analogic, îi este acordată interactivitatea, posibilitatea de a decide momentul când se începe contopirea imaginilor succesive, de a decide cât timp durează, ori de a întrerupe procesul de contopire.

În varianta analogică electronică, operatorul avea la dispoziție un pupitru de comandă (telecomandă) bazat pe niște divizoare rezistive prin care comanda reducerea sau creșterea intensității luminoase din spatele diapozitivelor.

Inițial am căutat soluții hardware care să comunice cu aplicația prin intermediul unui port USB, și care să simuleze telecomanda analogică. Dezavantajul major ar fi fost dat de faptul că aplicația nu ar mai fi fost accesibilă oricui. Era necesar un sistem universal de control, mult mai facil și mai practic. Am avut ideea

de a înlocui telecomanda rezistivă cu mouse-ul, și de a continua studiul în această direcție.

Odată apăsat butonul mouse-ului în perimetrul ferestrei ce servește ca și bază pentru imaginile prelucrate, operatorul semnalează intenția de a începe un proces de tranziție controlat manual către imaginea următoare. Pe măsură ce acesta, ținând butonul mouse-ului apăsat, deplasează cursorul -devenit între timp invizibil-pe o direcție verticală descendentă, procesul de contopire a celor două imagini progresaază proporțional cu această mișcare.

Este, de asemenea, posibilă „întoarcerea în timp” controlată, aceasta fiind o modalitate de a exprima derularea efectului în sens invers față de cel normal, de la imaginea cu index mai mare înspre imaginea cu index mai mic în succesiunea diaporamei. Aceasta se poate realiza odată ce butonul mouse-ului a fost apăsat, cursorul trecut de mijlocul traiectoriei posibile (mijlocul ecranului), dar cursa acestuia nu își urmează traiectoria firească, pe verticală din partea de sus a a ecranului înspre partea de jos, ci invers, din partea de jos a ecranului înspre partea de sus.

Cursa mouse-ului se poate întinde pe întreaga dimensiune verticală a monitorului, dar în cazul în care operatorul eliberează butonul mouse-ului înainte de finalul cursei, pe ecran va fi afișată imaginea de la care s-a plecat, dacă mouse-ul nu a fost trecut de mijlocul traiectoriei posibile sau imaginea la care se dorește să se ajungă în cazul în care mouse-ul a fost trecut de mijlocul ecranului.

4.1.17 Redarea imaginilor cu transparență predefinită

Unul din efectele prezente în diaporama analogică electronică era redarea simultană a ambelor diapozitive ce se succed, cu o intensitate luminoasă redusă la jumătate față de intensitatea maximă posibilă. Acest efect este de asemenea implementat în cadrul aplicației de diaporamă digitală și este disponibil utilizatorului prin apăsarea butonului denumit „Half Transparency” plasat pe panoul de control al aplicației.

Accesarea acestei funcționalități determină redarea pe ecranul calculatorului a unei a treia imagini formată din combinația prin suprapunere a celor două imagini ce ar trebui să se succedă însă fiecare dintre aceste este reprezentată cu un nivel de transparență redus la jumătate.

4.1.18 Coloana sonoră

Asemenea diaporamei analogice, diaporama digitală trebuie să aibă implementată o funcționalitate ce permite redarea unei coloane sonore ce vine să susțină imaginile ce se succed.

Am optat pentru utilizarea unei alte aplicații specializate pentru redarea de fișiere audio, Winamp [56], controlată însă din interiorul aplicației de față de către utilizator. Motivele care stau la baza acestei alegeri sunt know-how-ul acumulat de către un utilizator neavizat în utilizarea unui calculator personal pentru o aplicație binecunoscută, evitarea unor probleme ce pot apărea când se încearcă dezvoltarea unor soluții software de acest tip [57]. Alt motiv ar fi posibilitatea folosirii unei aplicații ce a fost testată în prealabil, oferindu-se astfel garanția unei funcționări în parametrii optimi. Astfel, este posibilă navigarea înainte și înapoi în lista de fișiere

audio, pornirea, oprirea totală ori temporară a coloanei sonore, reglarea volumului de redare al coloanei sonore, toate aceste operații fiind disponibile în timp real și fiind accesibile utilizatorului din interiorul aplicației de diaporamă digitală [58].

4.1.19 Bara de status a diaporamei

Asemenea cazului interfeței aplicației, panoul de control dispune de o bară de status ce dă informații utilizatorului referitor la modul de rulare al spectacolului de diaporamă.

Am plasat bara de status a diaporamei în partea de jos a panoului de control și am împărțit-o în mai multe celule, fiecare cu rolul său în a reda un anumit tip de informație.

Poate cel mai important element al barei de status este cel care semnalează poziția în cadrul spectacolului, numărul de ordine al imaginii curente. Acesta este semnalat sub forma indexului imaginii curente raportat la numărul total de imagini prezente în listă și ajută utilizatorul să aibă o orientare superioară în lista de imagini.

Cel de-al doilea element conține numele fișierului grafic curent însoțit de extensia acestuia. În cazul când imaginile au o denumire sugestivă, acesta servește asemeni raportului anterior la îmbunătățirea orientării în lista de imagini.

Sunt de asemenea prezente două elemente strâns legate de dimensiunea fizică a fișierului curent redat. Unul dintre acestea conține dimensiunile reale ale fișierului grafic, iar celălalt conține dimensiunile actuale la care a fost scalată imaginea, dacă a fost necesar sau impus acest lucru, ambele sub forma: dimensiunea pe verticală x dimensiunea pe orizontală. Dacă, de exemplu, este redată o imagine de 800 x 600 pixeli și se solicită redarea pe întreaga suprafață a ecranului, presupunând că sistemului de operare îi este specificat să redea o rezoluție de 1024 x 768, cele două elemente ale barei de status vor avea următoarea formă: 800 x 600, primul dintre ele, respectiv 1024 x 768 cel de-al doilea.

Bara de progres este alt element al barei de status, acesta fiind plasat în extrema dreaptă. Aceasta se modifică proporțional cu progresul efectului de contopire al imaginilor.

4.2 Unelte software pentru implementarea aplicației „Digital Diaporama”

În Iulie 2000, viziunea inițială a Microsoft asupra conceptului de Internet era departe de conținutul rețelelor web din zilele noastre. Microsoft considera Internetul viitor ca o lume a serviciilor interdependente dezvoltate de indivizi diferiți, scrise în limbaje diferite, desfășurate pe tipuri diferite de suportul fizic, și găzduite de diferite sisteme de operare Internet.

Oferta companiei pentru aceasta nouă lume este platforma .NET. De fapt, .NET nu este o platformă în sensul tradițional al noțiunii; nu este un sistem software sau hardware comun. În schimb, .NET este o colecție de protocoale care permit diverselor aplicații internet să se folosească de servicii disparate care rulează pe mașini diferite.

Un rezultat direct al strategiei firmei Microsoft este faptul că versiunea de Microsoft Visual Studio include limbaje remodelate, modificate explicit să suporte platforma .NET [59].

4.2.1 De ce am ales C#?

C# este cel un limbaj important de programare, dezvoltat de Anders Hejlsberg la Microsoft [60], fiind primul proiectat de la bun început pentru Internet.

Este un limbaj modern care combină cele mai bune caracteristici ale celor mai folosite limbaje de programare. Odată cu C#, Microsoft a lansat și platforma .NET, care permite compilarea și interfațarea de programe scrise în limbaje diferite [14].

C# este foarte asemănător, în ceea ce privește sintaxa, cu Java. Păstrează însă o apropiere mai mare de C++. Atât Java cât și C# compilează mai întâi într-un limbaj intermediar: Java byte-code, respectiv Microsoft Intermediate Language (MSIL). În C#, compilarea codului intermediar în cod nativ este însă mai eficientă.

C# permite utilizarea OOP (Programare orientate pe obiecte) respectând toate principiile acesteia, principii pe care le vom enunța în cadrul acestui capitol.

Evoluția tehnicilor de programare începe cu programarea nestructurată (program simplu ce utilizează doar variabile globale) și se încheie cu programarea orientată pe obiecte, marcând programarea procedurală (program principal deservit de subprograme), respectiv programarea modulară (gruparea subprogramelor cu funcționalități similare în module), fiecare cu avantaje proprii în evoluția către OOP.

Programarea orientată pe obiecte (programe cu noi tipuri ce integrează atât datele cât și metodele asociate creării, prelucrării cât și distrugerii acestor date) își datorează avantajele abstracției programării (avantaje precum flexibilitate sau ușoara reutilizare a codului).

Principiile OOP (Programării Orientate pe Obiecte):

- Încapsulare
- Moștenirea
- Polimorfismul

Programarea orientată pe obiecte constituie o bună metodă de organizare a programelor de calcul (software). Proprietățile OOP conduc la un cod principal compact și elegant. Obiectele pot descrie mai bine conceptele pe care le reprezintă, fiind mai logice și intuitive decât modul tradițional, cu simple structuri de date.

Programarea OOP este foarte potrivită în cazul metodei elementelor finite, impunându-se ca o necesitate pentru dezvoltarea programelor complexe.

Din punct de vedere educațional, aplicarea OOP conduce la formarea rapidă a unor concepte globale de funcționare a metodei elementului finit. Spre deosebire de metodele tradiționale folosite în învățământ, care pleacă de la o descriere detaliată a metodei, a conceptelor și a noțiunilor, cu construirea unui program de la scară mică la scară mare, programarea OOP permite construcția unor aplicații prin asamblarea unor module existente, la nivel global. OOP reprezintă astfel o metodă modernă, logică și eficientă nu numai în dezvoltarea de programe, dar și în utilizarea și înțelegerea acestora [61], [14], [62].

C# are de asemenea un garbage collector, despre care s-a demonstrat matematic că este foarte aproape de optimul posibil. Ca și Java, C# a renunțat la moștenirile multiple, în favoarea unui model de moștenire simplă extins de moștenirea multiplă a interfețelor [61].

Spre deosebire de Java, care a renunțat total la pointeri, C# permite folosirea acestora, dar numai în cazuri speciale, marcate „unsafe”.

Avantajele C# sunt:

- Dezvoltare rapidă de aplicații [63]
- Funcționare inter-platforme.
- Acces la resursele caracteristice platformei.
- Suport pentru platformele COM și .NET.
- Posibilitatea generării automate a documentației pe baza codului sursă.[64]

Dacă luăm în considerare toate caracteristicile expuse deja, suportul vizibil pentru RAD (Rapid Application Development) pare a fi unul din principalele obiective ale Microsoft pentru C#. Sunt și alte nevoi de dezvoltare importante la care C# trebuie să răspundă. Acestea includ dezvoltarea independentă de platformă, acces la resurse tipice platformei și suport pentru platformele COM și .NET. Vom examina facilitățile C# prin care se intenționează a satisface aceste nevoi.

- C# rezolvă problema funcționării independente de platformă în același mod ca și Java. Compilatorul de C# generează un flux de cod binar care este interpretat de .NET runtime. Componenta runtime lucrează ca și o mașină virtuală Java; o aplicație poate fi rulată pe orice mașină pe care a fost portat .NET runtime.

- Accesul aplicației la resurse specifice platformei. Un program C# poate folosi, de exemplu, Windows API pentru a rula ca o aplicație completă sub Windows XP. Același program poate rula pe un PDA folosind subsetul Windows CE API. Bineînțeles, nu toate serviciile pe care aplicația le așteaptă vor fi disponibile pe toate mașinile. De aceea, este responsabilitatea dezvoltatorului să testeze aplicația pe toate platformele țintă, și să scrie cod special, dacă este nevoie, care să permită funcționarea în absența unor servicii așteptate.

- Suport pentru COM și .NET. C# include o importantă facilități a limbajului, numită "atribute". Atributele facilitează suportul oferit de C# unor tehnologii specifice precum COM sau .NET fără a îngreuna specificațiile limbajului în sine. De exemplu, C# pune la dispoziție clase de atribute care convertesc interfețe C# în interfețe COM. Altele convertesc clase C# în clase COM. Unii comentatori ai limbajului au declarat că toate clasele C# sunt obiecte COM. Acest lucru nu corespunde tocmai adevărului, dar nu atributele corespunzătoare, orice clasă C# poate fi exprimată ca un obiect COM. Altă bibliotecă de atribute oferită împreună cu compilatorul de C# împachetează clasele și funcțiile ca servicii Web. Un serviciu Web este un modul software găzduit care poate fi chemat prin internet folosind un protocol. Protocolul împachetează apelări de metode, împreună cu parametrii și valorile returnate de acestea, în pachete de date XML. Serviciile Web pot fi scrise în multe limbaje și livrate pentru multe platforme software și hardware. În plus, mai multe servicii Web pot coopera pentru ca să formeze o întreagă aplicație Web. Folosind atributele corespunzătoare, un programator poate modifica cu ușurință orice clasă sau funcție C# într-un serviciu Web [62, 64].

Multe din facilitățile pe care C# le folosește în aplicațiile Internet sunt inspirate din Delphi și Java. Totuși Microsoft pretinde că C# descende din C și C++. De aceea sunt examinate unele caracteristici prezente la C++ și care lipsesc la C# [65].

- Scopul și dereferirea operatorilor. C# nu folosește operatorul de scop din C++ ":", în schimb se bazează pe echivalentul din Delphi și Java, punctul ".". Mai mult, deoarece se bazează pe sistemul valoare/ referință de tipul Delphi/ Java, C# adesea nu folosește operatorul de dereferire din C++ (->). În schimb, implementează din nou punctul. În timp ce operatorul punct supraîncărcat poate

aduce confuzie în rândul programatorilor de C++ care tocmai învață C#, le va fi familiar celor care au fost dezvoltatori Delphi sau Java.

- Sintaxa declarării referinței. În C#, deoarece clasele, interfețele și tipurile delegate sunt implicit referințe, lipsește declarația de referințe de la C++. În loc, noul limbaj folosește o sintaxă similară cu declararea din Delphi a parametrilor transmiși prin referință:

- un parametru nemarcat este parametru de intrare
- cuvântul cheie referință indică un parametru de intrare-ieșire (asemănător cu utilizarea din Delphi)
- cuvântul cheie "out" indică un parametru de ieșire
- cuvântul cheie "params" indică o listă de parametri de lungime variabilă (asemănător cu varianta din Delphi de parametri șiruri deschise)

- Template-uri: La fel ca Java și Delphi, C# nu are template-uri. Prin urmare, nu are un mecanism pentru a sprijini declarația puternic tipizată de colecții. În schimb, fiecare clasă este în final derivată din clasa de bază comună "object". O clasă generică de colecție trebuie să fie o colecție de objects.

Totuși, chiar și fără aceste facilități ale limbajului C++, C# este un limbaj puternic, proiectat cu facilități asemănătoare Java și Delphi pentru a permite dezvoltarea de aplicații internet. Datorită orientării spre mașină, și a lipsei de suport RAD, C++ nu ar fi fost o bază potrivită pentru un limbaj menit să fie folosit la dezvoltarea aplicațiilor Internet. Orice dezvoltator de C++ care trebuie să scrie aplicații și servicii pentru platforma .NET, va fi sfătuit să învețe C#. Iar acestea două diferă mai mult decât ar putea sugera numele.

4.2.2 Codul

4.2.2.1 Clasele utilizate

În momentul în care se ia decizia scrierii de cod pentru orice tip de aplicație, trebuie clarificate anumite aspecte înainte de a se începe propriu-zis dezvoltarea acestuia. Algoritmul ce stă în spatele oricărei funcționalități a aplicației trebuie să fie cât mai simplu, pentru a permite o execuție rapidă a acestuia, dar în același timp trebuie să acopere complexitatea cerută de aplicație, și nu în ultimul rând trebuie luată în considerare o eventuală modificare a acestuia prin adăugarea unor funcționalități noi de către alte persoane. Aceste deziderate au fost urmărite și pe parcursul dezvoltării aplicației de diaporamă digitală.

Pentru aplicația ce face obiectul acestui studiu, am creat două clase mari conținute de namespace-ul „viz”, aceasta fiind o facilitate specifică a mediului de programare C#, ce grupează funcționalitățile proiectului, ambele clase moștenind clasa primară „System.Windows.Forms.Form” și făcând vizibile pentru utilizator două ferestre cu care va lucra mai departe.

Derivarea din clasa "Form" a celor două clase utilizate este:

```
public class vizform : System.Windows.Forms.Form
public class FullScreen : System.Windows.Forms.Form
```

Prima dintre acestea este „vizform” și grupează funcționalitățile ferestrei principale a aplicației descrise mai sus, cea care pornește imediat după lansarea în execuție a aplicației, iar cea de-a doua creează fundalul de tip „full-screen” pentru

spectacolul propriu-zis de diaporamă. În continuare voi prezenta aspectele cele mai importante legate de „vizform”.

Orice obiect plasat în interiorul ferestrei principale trebuie la rândul lui derivat dintr-un obiect standard de bibliotecă specific C#. Astfel pentru a declara lista în care vor apărea toate fișierele grafice din calea curentă se va apela la o declarație de tipul:

```
private System.Windows.Forms.ListView fileview;
```

unde „fileview” reprezintă obiectul de tip „ListView” caracteristic ferestrei principale. De acum încolo, de fiecare dată când ne vom referi la lista de fișiere din directorul curent, de exemplu pentru a aplica diferite operații, de inserare ori ștergere de intrări în listă, o vom putea apela prin „vizform.fileview”, ori simplu „fileview”. Cuvântul cheie, „private” plasat înaintea declarației derivării, se numește modificator de acces și spune compilatorului faptul că acest obiect este vizibil doar în interiorul clasei ce îl definește, adică „vizform”. Detaliile explicative sunt asemănătoare pentru toate celelalte obiecte create de clasa „vizform”, în total fiind folosite aproximativ 50 de obiecte în fereastra principală. Toate aceste derivări de clase se numesc atribute ale clasei „vizform” și dacă sunt declarate cu modificatorul de acces „private” sunt accesibile doar în interiorul acestei clase. Pentru a descrie și o excepție de la regulă, s-a folosit pe parcursul dezvoltării proiectului un atribut de tip „public”, acesta putând fi accesat atât din clasa ce îl conține cât și din altă clasă, după cum va fi descris mai jos:

```
public string[] inputArgs;
```

Atributul „inputArgs” s-a folosit pentru a înregistra și prelucra argumentele de intrare ale aplicației, aceasta fiind capabilă să primească argumente la rulare din linia de comandă. Acestea trebuie să fie vizibile în orice clasă ce intră în componența aplicației, acesta fiind motivul pentru care acest obiect este însoțit de cuvântul cheie „public” [66].

4.2.2.2 Punctul de pornire al aplicației

Când se dezvoltă aplicații orientate pe obiecte, orice funcție creată în interiorul unei clase poartă denumirea de metodă.

Pentru cel ce scrie cod în C#, metoda de la care pornește este cea ce se rulează prima, și anume „Main”. În cazul aplicației de diaporamă digitală aceasta are următoarea formă:

```
static void Main(string[] args)
{
    SingleInstanceApplication App = new
    SingleInstanceApplication();
    App.Run(args);
}
```

ceea ce spune de fapt compilatorului să ruleze o aplicație de tip „Single Instance”, permițând unui singur proces pentru „Digital diaporama”, indiferent de numărul de lansări a fișierului executabil.

Metoda „Run” ce se apelează imediat după aceea, face trimiterea către constructorul clasei „vizform”, acesta fiind disponibil cu argumente, după cum este înscris în codul de mai jos. Constructorul cu argumente:

```
public vizform(string[] args)
{
    InitializeComponent();
}
```

Se remarcă apelul metodei „InitializeComponent()” odată cu lansarea în execuție a aplicației. Aceasta este o metodă generată automat și întreținută de mediul de dezvoltare, ce conține date despre toate obiectele prezente în fereastra principală a aplicației. De fiecare dată când se adaugă un obiect nou sau când se modifică atributele unuia deja existent, conținutul metodei menționate mai sus se modifică.

Odată cești pași parcurși, starea aplicației se va schimba în funcție de evenimentele ce au loc, în interiorul acestuia: apăsarea unui buton de mouse într-o anumită zonă, mișcarea acestuia în interiorul spațiului unui obiect ori apăsarea unui buton al tastaturii.

4.2.2.3 Momentele cheie „FormLoad” și „FormClose”

Prima metodă ce se va apela după execuția constructorului clasei „vizform” va fi cea determinată de evenimentul „Form Load” ce apare în momentul în care se afișează fereastra principală a aplicației, în cazul de față aceasta având următoarea structură:

```
private void Form1_Load(object sender, System.EventArgs e)
{
    // Initializeaza Structura Directoarelor
    this.folderview.InitFolderTreeView();

    // Citeste fisierul de configurare
    this.ReadConfiguration();
}
```

În interiorul acestei metode se face apelul către două alte metode, prima dintre ele, ținând cont de ordinea cronologică în care se face apelul acestora, InitFolderTreeView() inițializează obiectul cu ce conține lista de directoare. Acesta nu este o metodă specifică clasei „vizform”. Cea de-a doua metodă, ReadConfiguration(), una din metodele clasei „vizform”, realizează citirea, prelucrarea și utilizarea fișierului de configurare a aplicației de diaporamă digitală.

Fișierul de configurare este citit și salvat la fiecare lansare în execuție a aplicației, respectiv la fiecare terminare a acesteia. Astfel, de fiecare dată când aplicația de diaporamă digitală se închide se va apela următoarea metodă:

```
private void vizform_FormClosing(object sender,
FormClosingEventArgs e)
{
```



```
this.SaveConfiguration();  
}
```

Metodele ce realizează operațiile de input/output cu fișierul de configurare al aplicației sunt `ReadConfiguration()` pentru citirea acestuia, respectiv `SaveConfiguration()` pentru scrierea de date în acesta.

4.2.2.4 Regiunile mediului de dezvoltare "Visual Studio"

"Visual Studio" permite folosirea cuvintelor cheie "#region", respectiv "#endregion". Acestea pot marca o zonă din cod ce realizează funcționalități înrudite, fiind o facilitare oferită de mediul de dezvoltare, ignorată de către compilator, neavând nici o amprentă asupra fișierului final executabil. Folosirea acestora crește considerabil capacitatea de a "recita" codul, mai ales când vorbim de aplicații a căror linii de comandă sunt în număr foarte mare, atunci când se dorește de exemplu îmbunătățirea unor metode scrise în prealabil. Dacă e să urmărim conjunctura în care am folosit aceste cuvinte cheie, partea de cod de mai sus grupează operațiile de citire/scriere asupra fișierului de configurare a aplicației. Aceste cuvinte cheie, odată folosite în interiorul mediului de dezvoltare permit așa-numita facilitare de colaps/expand a codului. În cazul în care codul este "expandat" acesta va fi vizibil în întregime programatorului. Când codul este "compactat", acesta va fi "invizibil" pentru programator, dar vizibil pentru compilatorul de C#, după cum se poate vedea în imaginea de mai jos.

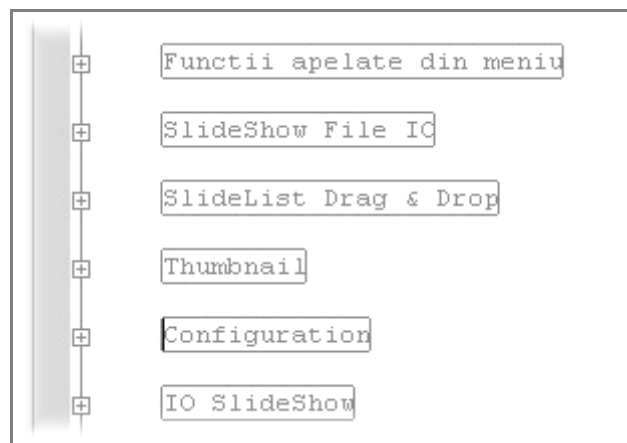


Fig. 22 - Regiuni din interiorul codului aplicației "Digital Diaporama"

4.2.2.5 Fișierul de configurație al aplicației

Fișierul de configurare are un format de tip XML, ales datorită faptului că structura acestuia de tip arborescent permite programatorului să includă cât mai multe atribute, dar în același timp este foarte ușor de utilizat. Fișierul de configurare

pentru aplicațiile scrise în mediul de dezvoltare C# are de obicei extensia ".config" și poate fi de următoarea formă, de exemplu în cazul în care acesta salvează ultimul director curent înainte de a se închide aplicația:

```
viz.exe.config
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <appSettings>
    <add key="InitialPath"
value="D:\Images\Subaru\2005\Mexic" />
  </appSettings>
</configuration>
```

Aceasta este doar o variantă intermediară a acestuia, odată cu o eventuală dezvoltarea a aplicației. Pe măsură ce se va dovedi necesar, metodele de scriere, respectiv citire a acestuia se vor modifica și cu aceasta și numărul de date conținut în fișierul de configurare. În momentul de față, metodele de citire/scriere a fișierului de configurare au următoarea formă:

```
#region Configuration

private void ReadConfiguration()
{
string initialPath =
ConfigurationManager.AppSettings["InitialPath"];

if (!this.folderview.DrillToFolder(initialPath))
MessageBox.Show("Invalid path în 1st argument");
}

private void SaveConfiguration()
{
System.Configuration.Configuration config =
ConfigurationManager.OpenExeConfiguration
(ConfigurationUserLevel.None);

// Update initial path
config.AppSettings.Settings.Remove("InitialPath");
config.AppSettings.Settings.Add("InitialPath",
this.folderview.GetSelectedNodePath());

// Save the changes în *.config file.
config.Save(ConfigurationSaveMode.Modified);
}

#endregion
```

Metoda de citire a configurației încearcă să preia datele din fișierul cu extensia ".config" prin instanța clasei "ConfigurationManager", respectiv prin metoda "AppSettings" apelată cu parmetrul "InitialPath". În cazul în care nu se poate naviga

prin structura de directoare către calea citită din fișier, se va afișa un mesaj pe ecran ce va informa utilizatorul de această eroare.

Metoda de scriere a fișierului de configurare funcționează într-un mod analog cu cea de citire, doar că operațiile făcute sunt în celălalt sens, respectiv de ștergere a vechilor intrări și apoi de scriere a noilor date, în speță directorul curent.

4.2.2.6 Previzualizarea

După cum mai menționat, am implementat în aplicația de diaporamă digitală funcționalitatea de previzualizare a imaginilor selectate la un moment dat. Aceasta înseamnă că odată ce utilizatorul selectează dintr-una din cele două liste o imagine (din cea care conține fișierele grafice din directorul curent, respectiv din cea care grupează toate imaginile ce urmează să alcătuiască spectacolul de diaporamă), în fereastra principală un obiect de tip grafic își va schimba conținutul cu imaginea selectată, însă scalată la un dreptunghi cu dimensiunile de 200 x 150 pixeli.

Metodele ce se rulează în momentul când utilizatorul schimbă imaginea, fiind necesară previzualizarea acesteia, diferă foarte puțin în cazul în care imaginea se schimbă din lista fișierelor grafice din directorul curent, ori din lista imaginilor ce urmează să formeze spectacolul de diaporamă, bazându-se pe același model: în primul rând este necesară găsirea căii de pe hard-disk la care se află noua imagine selectată, respectiv modificarea conținutului barei de status cu datele noii imagini selectate.

Cele două metode comune în cazul selectării imaginilor din liste diferite sunt prezentate în secțiunea de cod de mai jos:

```
private void DisplayImageInfo(string path, string filename,
string filesize)
{
    this.statusBar.Panels[0].Text = "Filename: " + filename;
    this.statusBar.Panels[1].Text = "Directory: " + path;
    this.statusBar.Panels[2].Text =
    "Physical Dimensions: " +
    originalWidth.ToString() + " x " +
    originalHeight.ToString() +
    " pixels";
    this.statusBar.Panels[3].Text =
    "Size: " + filesize + " kB";
    if (!this.statusBar.Visible) this.statusBar.Visible = true;
}
```

Metoda "DisplayImageInfo" ia ca argumente trei șiruri de caractere și în funcție de acestea realizează modificările asupra barei de status a aplicației. Cele trei șiruri sunt denumite intuitiv "filename", "path", respectiv "filesize" și sunt concatenate ca și variabile cu șirurile constante "Filename:", "Directory:", respectiv "Size:" pentru a forma noile componente ale barei de status.

Pe lângă acestea, bara de status va conține și dimensiunile (în pixeli) ale imaginii ce se dorește a fi previzualizată. Se observă apelul metodelor originalWidth.ToString(), respectiv a originalHeight.ToString(). Acestea realizează de fapt conversia într-un șir de caractere a dimensiunii imaginii nou selectată, ce este

conținută în atribute de tip "private", deoarece citirea acestuia poate fi făcută doar în momentul în care se realizează conexiunea între fișierul de pe hard-disk și obiectul corespunzător din aplicația de diaporamă digitală, mai exact apelul metodei LoadImage(). În realitate succesiunea celor două metode "LoadImage", respectiv "DisplayImageInfo" este inversă celei prezentate în documentul de față, înainte realizându-se legătura și mai apoi se face actualizarea barei de status.

```
private void LoadImage(string path)
{
    this.previewbox.Image =
    this.ImagetoSize(path, this.previewbox.Width,
    this.previewbox.Height, Color.Wheat);
    this.previewbox.Refresh();
}
```

Metoda LoadImage(), realizează de fapt una din cele mai importante funcționalități ale ferestrei principale ale aplicației de diaporamă digitală, previzualizarea. De fapt, aparența ascunde apelul unei alte metode ImageToSize, ce este de fapt o funcție de scalare a unei imagini la anumite dimensiuni, folosită și în alte părți ale aplicației, de exemplu la scalarea imaginilor simbol ce apar în diferitele moduri de vizualizare a fișierelor, "Details", "Thumbnails" ori "Small Icons", a listei imaginilor ce vor forma diaporama.

```
private Bitmap ImagetoSize(string fileName, int desiredWidth,
int desiredHeight, Color penColor)
{
    Bitmap bmp = new Bitmap(fileName);

    try
    {
        desiredWidth =
        bmp.Width > desiredWidth ? desiredWidth : bmp.Width;
        desiredHeight =
        bmp.Height > desiredHeight ? desiredHeight : bmp.Height;

        originalWidth = bmp.Width;
        originalHeight = bmp.Height;

        Bitmap retBmp = new Bitmap(desiredWidth, desiredHeight,
        System.Drawing.Imaging.PixelFormat.Format64bppPArgb);

        Graphics grp = Graphics.FromImage(retBmp);

        int destWidth = desiredWidth, destHeight = desiredHeight;

        if (bmp.Width > bmp.Height)
            destHeight = (int)((float)
            bmp.Height / (float)bmp.Width * destWidth);
        else if (bmp.Width < bmp.Height)
            destWidth = (int)((float)
            bmp.Width / (float)bmp.Height * destHeight);
    }
}
```

```
int destLeft = (desiredWidth / 2) - (destWidth / 2);
int destTop = (desiredHeight / 2) - (destHeight / 2);

grp.DrawImage bmp,
destLeft, destTop, destWidth, destHeight);

Pen pn = new Pen(penColor, 1); //Color.Wheat
grp.DrawRectangle(pn,
0, 0, retBmp.Width - 1, retBmp.Height - 1);

return retBmp;
}
catch
{
//If cant load the image, create a blank one with ThumbSize
bmp = new Bitmap(desiredWidth, desiredHeight);
return bmp;
}
finally
{
bmp.Dispose();
}
}
```

Metoda `ImageToSize` realizează de fapt transformarea unei imagini, în sensul de a micșora dimensiunile sale, dacă este cazul. Ca și parametrii cei importanți, sunt considerați calea la care se află imaginea de scalat și dimensiunile dorite, pe verticală și pe orizontală, ale imaginii rezultat. Imaginea rezultată se trimite ca și ieșirea metodei, odată ce aceasta a fost parcursă de șirul de execuție.

Funcționarea acesteia este axată pe un bloc de tipul "try, catch, finally" ce acoperă toate riscurile ce pot apărea în cazul executării metodei. De exemplu, orice excepție ridicată de execuția instrucțiunilor cuprinse în blocul „try” va determina executarea blocului de instrucțiuni cuprinse din blocul „catch”, evitând astfel afișarea unui mesaj de eroare în fața unui utilizator de cele mai multe ori neavizat. După ieșirea din metodă se vor apela instrucțiunile cuprinse în blocul „finally”, în acest caz eliberarea memoriei ocupată de variabila locală de tipul „bmp”. Această operațiune din final este oricum realizată de funcția de „Garbage Collector” a oricărui mediu de dezvoltare este OOP (Object-Oriented-Programming), însă specificarea ad-literam a acestui tip de instrucțiune reduce încărcarea execuției aplicației.

Dacă e să urmărim firul logic cuprins în această metodă, primul lucru ce se realizează este realizarea legăturii între variabila locală de tip „Bitmap” și imaginea de pe hard-disk-ul calculatorului personal. Odată realizată această legătură, se verifică nevoia reală de a scala imagine originală, fiind posibil ca dimensiunile acesteia să permită redarea fără nici un fel de operație de a propoționa lungimea și înălțimea. Următorul pas este crearea obiectului de tip grafic ce va fi returnat de către metodă, în funcție de factorul de scalare dar și de coordonatele colțului din stânga sus a imaginii calculate în prealabil. Pasul final este „desenarea” imaginii scalate, centrată în obiectul grafic creat și returnarea acestuia la ieșirea metodei.

4.2.2.7 Funcții apelate din meniu

Dacă apelul unor comenzi de tipul "Exit", "Details", "Thumbnails", "Small Icons", "Remove SlideShow file" ori "Clear SlideShow list" nu implică decât apelarea unor metode specifice obiectelor impactate, de exemplu metoda "Close()" a clasei "vizform" pentru închiderea aplicației, ori verificarea proprietății "Checked" a instanțelor clasei "MenuItem", și nu prezintă foarte mare interes și nici nu e foarte dificil să se înțeleagă mecanismul ce stă în spatele acestora, totuși sunt alte comenzi ce sunt mai complexe și merită prezentate în acest sub-capitol.

4.2.2.8 Inserarea de fișiere în lista de diaporamă

De exemplu, dacă e să aruncăm o privire în detaliu asupra metodelor ce realizează inserarea unui nou obiect în lista imaginilor ce se doresc a intra în componența spectacolului de diaporamă, acestea sunt de două tipuri, dar în mare aceleași, diferind doar modul prin care se face legătura între fișierul ce se dorește a fi inserat și calea pe hard-disk către acesta. Inserarea unui nou obiect în listă se face în cazul aplicației de față fie ghidându-se după indexul acestuia, fie direct după calea fișierului sursă. Pentru a realiza această funcție, am definit două metode:

```
private void AddSlideShowFilebyIndex(int index)
private void AddSlideShowFilebyPath(string path, string filename)
```

ale căror corp de instrucțiuni arată la fel de la momentul în care s-a stabilit calea către fișierul grafic:

```
{
    ListViewItem item = new ListViewItem();
    ...
    AddThumbnail(path,96,96,16,16,Color.Wheat);
    ...
    this.slideview.Items.Add(item);
}
```

unde prin „...” am mascat codul ce stă la bază determinării căii până la fișierul grafic în cazul primei apariții, respectiv am mascat codul ce realizează adăugarea la obiectul „item” a informațiilor secundare despre fișierul ce urmează a fi inserat (numele exact, data la care a fost ultima dată modificat, dimensiunea în pixeli a acestuia ori spațiul în kilo-octeți ocupat de acesta pe hard-disk). Fiecare nou „item” creat este adăugat la lista imaginilor ce vor forma diaporama prin apelul metodei de la final „Items.Add(item)”. Am evidențiat în mijlocul corpului acestei funcții, apelul metodei AddThumbnail(). Aceasta este necesară, deoarece toate imaginile inserate vor avea un așa numit element de previzualizare, indiferent de starea în care lista este afișată: „Details”, „Thumbnail” ori „Small Icons”. De fapt această metodă este în mare aceeași cu „ImageToSize” ce a fost descrisă puțin mai sus, însă este dublată, deoarece este nevoie de a se crea două elemente de previzualizare de dimensiuni diferite. „AddThumbnail()” ia ca și parametrii calea către fișierul grafic

căruia trebuie să-i creeze elemente de previzualizare, dimensiunile elementului mare, respectiv dimensiunile elementului mic, pe orizontală și pe verticală.

```
private void AddThumbNail(string fileName,
int imgLargeWidth, int imgLargeHeight,
int imgSmallWidth, int imgSmallHeight,
Color penColor)
{
    Bitmap bmp;

    try
    {
        bmp = new Bitmap(fileName);
    }
    catch
    {
        //create a blank image with ThumbSize
        bmp = new Bitmap(imgLargeWidth, imgLargeHeight);    }

    ...

    grp.DrawRectangle(pn, 0, 0, retBmp.Width - 1, retBmp.Height - 1);

    this.largeThumbnailSlideView.Images.Add(retBmp);

        float propScale = (float)imgSmallWidth/imgLargeWidth;

        iLeft =(int)(iLeft*propScale);
        iTop = (int)(iTop*propScale);
        tnWidth = (int)(tnWidth*propScale);
        tnHeight = (int)(tnHeight*propScale);

    grp.DrawImage(bmp, iLeft, iTop, tnWidth, tnHeight);

    grp.DrawRectangle(pn, 0, 0, retBmp.Width - 1, retBmp.Height - 1);

    this.smallThumbnailSlideView.Images.Add(retBmp);
}
```

Primul pas, asemenea cazului în care construim previzualizarea pentru imaginea curentă, este de a face conexiunea între fișierul grafic de pe hard-disk și obiectul de tip "Bitmap". Cu "..." am marcat codul ce coincide cu cel al metodei "ImageToSize()" descrisă anterior. Diferența evidentă apare în momentul în care am avea rezultatul final al primei scalări. În loc să-l trimitem ca și "ieșire" al funcției, așa cum am făcut în cazul "ImageToSize", aici îl adăugăm la o listă de imagini, prin apelarea metodei "largeThumbnailSlideView.Images.Add(retBmp)". De aici totul este nou în materie de cod. Având în vedere că trebuie să creăm două elemente de previzualizare ale aceleiași imagini, diferite ca dimensiuni, foarte ușor se poate face scalarea primului element la dimensiunile celui de-al doilea, proporțiile imaginilor fiind aceleași. Astfel se reține valoarea proporției în variabila de tip real "propScale" și toate dimensiunile (lungime, înălțime, coordonatele colțului din stânga sus pentru

noua imagine) vor fi determinate prin simpla înmulțire a dimensiunilor primei imagini cu factorul de scalare. În final, cea de-a doua previzualizare se adaugă într-o a doua listă de imagini, prin apelarea metodei "smallThumbnailSlideView.Images.Add(retBmp)".

Cele două liste de imagini se folosesc la stabilirea proprietăților de "ThumbNails" ale listei ce conține imaginile ce vor forma diaporama.

4.2.2.9 Salvarea listei ce grupează imaginile din diaporamă

Este absolut necesar ca odată formată o anumită listă pentru diaporamă să se ofere posibilitatea de a o salva într-un anumit format pe hard-disk, pentru a fi restaurată la un alt moment. Această funcționalitate este îndeplinită de către "Digital Diaporama" iar comenzile către metodele ce fac acest lucru sunt situate în meniul "File", sub-meniul "Open SlideShow", respectiv "Save SlideShow". Fișierele ce conțin lista de imagini ce intră în componența diaporamei și atributele necesare ale acestora sunt scrise în format XML (extensible mark-up language) atât datorită ușurintei în scriere și citire a datelor prin existența claselor XmlTextReader, respectiv XmlTextWriter, cât și pentru a ușura munca celui ce va modifica ulterior codul, pentru a implementa noi facilități.

Tot codul legat de operațiunile de intrare/ieșire au fost grupate în regiunea "IO SlideShow", unde prin simbolul "..." am mascat codul ce ține de declararea obiectelor folosite, inițializări ori setări ce țin de operațiile de intrare/ieșire ce trebuie să fie efectuate la închiderea unui fișier deschis pentru editare, ori citire:

```
private void LoadSlideShowFile(string path)
{
...
try
{
while (reader.Read())
{
if (reader.NodeType == XmlNodeType.Element)
{
if (reader.Name == XmlNodeType.Tag)
{
// loading SlideShow Files attributes
reader.MoveToAttribute(0);
slideFilename = reader.Value;
reader.MoveToAttribute(1);
slidePath = reader.Value;
...
AddSlideShowFilebyPath(slidePath,
slideFilename);
}
}
else if (reader.NodeType == XmlNodeType.None)
{
```



```

return;
}
}
}
...
}

```

Metoda de citire din fișier "LoadSlideShowFile" ia ca și argument calea la care este situat acesta. După ce se realizează conexiunea dintre fișierul de pe hard-disk și obiectul "reader" de tip "XmlTextReader", se poate efectua operația de citire secvențială a obiectelor conținute de fișierul în cauză. Citirea și prelucrarea datelor se efectuează pe atribute, unul câte unul, pentru fiecare nod explorat, până la întâlnirea unui nod nul, moment în care se execută instrucțiunea de ieșire din metodă "return".

```

private void SaveSlideShowFile(string path)
{
    ...
    int i;
    for (i = 0; i < this.slideview.Items.Count;i++ )
    {
        myXmlTextWriter.WriteStartElement(XmlNodeType);
            myXmlTextWriter.WriteAttributeString(XmlNodeTypeFilenameAtt,
            this.slideview.Items[i].SubItems[0].Text);
            myXmlTextWriter.WriteAttributeString(XmlNodeTypePathAtt,
            this.slideview.Items[i].SubItems[1].Text);
        ...
        myXmlTextWriter.WriteEndElement();
    }
    ...
}

```

Metoda de scriere a fișierului "XML" funcționează analog celei de citire, de data aceasta prin utilizarea unui obiect de tip "XmlTextWriter", realizând o explorare a listei ce conține fișierele destinate diaporamei, în funcție de index, de la cel mai mic spre cel mai mare și scrierea acestora ca și atribute în noul fișier ce urmează a fi creat.

Putem remarca faptul că se apelează două metode pentru o anumită intrare din listă, una la începutul scrierii atributelor, iar alta la terminarea acestora și anume "WriteStartElement", respectiv "WriteEndElement". Acestea marchează un nou nod, element specific limbajului XML.

4.2.2.10 "Drag and Drop" în lista imaginilor pentru diaporamă

Efectul "Drag&Drop" este foarte întâlnit în sistemele de operare ce au interfață grafică bazată pe ferestre, precum Windows, aproape toate distribuțiile de Linux, ori sistemele de operare de la Apple. Acesta constituie de fapt o modalitate foarte simplă de a "muta" cu mouse-ul "item-uri" dintr-o fereastră în alta ori dintr-un obiect într-altul al aceleiași ferestre sau în interiorul aceluiași obiect. Descrierea

din urmă se aplică și în cazul aplicației de față, fiind necesară la un moment dat o anumită reorganizare a imaginilor în funcție de index, deoarece în spectacolul de diaporamă imaginile sunt redată în aceeași ordine în care ele se află în lista din fereastra principală .

Cea mai facilă rezolvare a acestei probleme de reorganizare a fost să implementez bine-cunoscutului "drag&drop" pentru lista ce conține imaginile ce urmează să formeze diaporama. Ideea ce stă în spatele acestei funcționalități este una simplă și se bazează pe trei evenimente ce pot fi capturate de aplicație: apăsarea unui buton al mouse-ului, mișcarea cursorului mouse-ului în interiorul obiectului, respectiv eliberarea butonului mouse-ului ce a fost în prealabil apăsat.

Pentru urmărirea "item-urilor" din listă se vor folosi următoarele obiecte:

```
private ListViewItem itemDnD = null;  
private ListViewItem itemOver = null;
```

Obiectele de tip "ListViewItem" vor "memora" atât timp cât butonul mouse-ului este apăsat, item-ul asupra căruia a fost apăsat butonul mouse-ului, respectiv ultimul "item" ce a fost survolat de către acesta. Prima dintre cele trei metode captează momentul în care butonul mouse-ului este apăsat asupra unui "item" din lista diaporamei:

```
private void slideview_MouseDown(object sender, MouseEventArgs e)  
{  
try  
{  
itemDnD = this.slideview.GetItemAt(e.X, e.Y);  
insertIndex = itemDnD.Index;  
Cursor = Cursors.Default;  
}  
catch { }  
}
```

În cazul în care este selectat un "item" valabil, se va memora conținutul acestuia și indexul la care acesta se află. Pe parcursul mișcării mouse-ului se urmărește și reține ultimul "item" survolat și indexul la care acesta se află.

```
private void slideview_MouseUp(object sender, MouseEventArgs e)  
{  
...  
try  
{  
this.slideview.Items.Remove(itemDnD);  
...  
this.slideview.Items.Insert(insertIndex, itemDnD);  
...  
}  
...  
}
```

Metoda ce se apelează la detectarea eliberării butonului de mouse, în cazul în care acest eveniment are loc deasupra unui "item" valid realizează ștergerea intrării de la care s-a plecat inițial și inserarea acesteia în noua poziție. Prin folosirea acestei

funcționalități, utilizatorul poate controla cu ușurință ordinea în care imaginile selectate în prealabil vor fi poziționate în diaporamă.

4.2.2.11 Structura funcțională a clasei „FullScreen”

Clasa „FullScreen” reprezintă suportul reflectat în cod pentru realizarea diaporamei digitale. Asemeni clasei „vizform” ce grupează funcționalitățile legate de interfața cu utilizatorul, clasa „FullScreen” grupează la rândul ei toate metodele și atributele ce fac parte din diaporama digitală ca și concept software.

4.2.2.12 Prima metodă executată

Ca și funcționalitate, rularea diaporamei se face apelând o „scurtătură” sau accesând un submeniu din interfața aplicației cu utilizatorul. În consecință a fost nevoie ca să includ în clasa „vizform” o metodă ce va fi lansată în execuție la apariția evenimentului mai sus menționat. Metoda în discuție are următoarea structură și este definită în clasa „vizform”:

```
// Intra în modul fullscreen
private void menuItem7_Click(object sender, System.EventArgs e)
{
    int counter=0;
    FullScreen full = new FullScreen();

    while (counter < this.slideview.Items.Count)
    {
        ListViewItem item = new ListViewItem();
        ...
        full.slideShowListView.Items.Add(item);

        counter++;
    }
    ...
    full.Show();
}
```

În spatele codului stă de fapt crearea unui obiect de tip „FullScreen” prin intermediul metodei constructor „FullScreen()” din clasa cu același nume, a cărei listă va fi inițializată cu toate imaginile ce vor forma viitorul spectacol de diaporamă. Această listă este de fapt similară cu lista centrală a panoului de comandă a diaporamei.

În primă instanță se adaugă noi intrari la listă iar în următoarea fază se rulează metoda „Show” ce face vizibilă fereastra suport pentru diaporamă pe ecranul calculatorului personal.

Și în cazul clasei ce este supusă la discuție în aceste paragrafe, avem de-a face cu o metodă constructor ce poartă același nume cu cel al clasei din care face parte. Structura acestuia în cazul clasei „FullScreen” are mai mult rol de inițializare a componentelor și în consecință nu prezintă prea mult interes pentru înțelegerea modului de funcționare al clasei.

Imediat după parcurgerea constructorului clasei, se apelează metoda ce survine în urma apelării metodei „Show()” pentru obiectul „full”, după cum a fost descris în paragrafele de mai sus. Acest eveniment este generat de apariția pe ecran a noii ferestre ce va servi ca suport pentru derularea spectacolului de diaporamă și va avea următoarea definiție:

```
private void FullScreen_Load(object sender, System.EventArgs e)
{
    this.FSControlBox.Left = (int)(this.Width * 0.1);
    this.FSControlBox.Width = (int)(this.Width * 0.8);
    this.FSControlBox.Top = this.Height -
    this.FSControlBox.Height - (int)(this.Height * 0.05);
    this.effectTimer.Interval =
    (int)(1000 / this.effectStepsCounter.Value);

    pic_index = 0;
    transparency = 1.0f;

    path = GetPath(pic_index);

    this.nextImage = new Bitmap(path);

    this.Load_Full_Image(path);
}
```

Astfel, dacă analizăm metoda menționată, se remarcă ușor două părți distincte, dacă le clasificăm după funcțiile ce sunt îndeplinite de instrucțiunile componente. Prima dintre acestea realizează diferite inițializări necesare în vederea pornirii diaporamei digitale, ce au ca țintă poziționarea corectă a panoului de control a acesteia, poziționarea indexului imaginii curente pe prima imagine din listă, respectiv setarea transparenței la valoarea maximă în cazul primei imagini. Cea de-a doua secțiune conține apelul metodelor ce duc la apariția primei imagini pe ecran.

Două dintre atributele caracteristice clasei „FullScreen” ce au fost declarate pe parcursul dezvoltării aplicației și constituie în esență legătura cu diaporama analogică, reprezentând obiectele ce rețin imaginea de la care se pleacă, respectiv imaginea la care se dorește a se ajunge au următoarele definiții:

```
private Bitmap previousImage;
private Bitmap nextImage;
```

În prima fază, obiectul imagine curentă va fi inițializat cu prima imagine din listă, iar cel de-al doilea obiect imagine va trebui inițializat cu valoarea vidă. Inițializările au loc în primul apel al metodei „Load_Full_Image()” accesată cu argumentul cale curentă. Funcția în discuție are următorul corp de instrucțiuni:

```
private void Load_Full_Image(string path)
```

```

{
try
    {
        this.previousImage = this.nextImage;
    }
    catch { }

    path = this.GetPath(pic_index);
    this.nextImage = new Bitmap(path);
    ...
    this.ScaleImage(this.nextImage);
    this.ValidateNavigation(pic_index);
    this.DisplayInfo(pic_index);
}

```

Se sesizează folosirea în corpul acestei metode a două atribute specifice clasei "FullScreen" anume "path" și "pic_index". Cele două sunt declarate astfel:

```

private string path;
...
private int pic_index;

```

și au rolul de a reține în orice moment al execuției calea către imaginea curentă (din prim-plan), respectiv indexul (poziția din lista de imagini) ce o caracterizează pe aceasta. Deoarece constructorul clasei „Bitmap” ce este folosit pentru a încărca imagini noi în aplicație ia ca argument calea relativă a fișierului grafic, iar navigația în lista de imagini se face crescând sau micșorând indexul imaginii curente, am dezvoltat metode care returnează, la primirea unui index ca argument, calea la care este situată imaginea cu acel index:

```

private string GetPath(int index)
{
    return path=
    this.slideShowListView.Items[index].SubItems[1].Text+
    "\\'+
    this.slideShowListView.Items[index].SubItems[0].Text;
}

```

Această metodă prelucrează informațiile suplimentare ale intrării din lista de imagini și le aduce la forma ce reliefează o cale către fișierul grafic cu indexul primit ca și parametru.

Următorul pas în redarea corectă a imaginii pe ecran este necesitatea scalării acesteia la dimensiunile date de rezoluția la care este setat sistemul de operare, procedeu asemănător scalării din cadrul previzualizărilor din interfața cu utilizatorul. Metoda pe care se bazează această funcționalitate este:

```

Rectangle tempRect = new Rectangle();
private void ScaleImage(Bitmap src)
{
    ...

```

```

tempRect = resize_rectangle(desired, source,
this.EnlargeCheckBox.Checked, this.ShrinkCheckBox.Checked);

if (tempRect.Width > prevdest.Width)
this.fullscreenPicture.Width = tempRect.Width;
else this.fullscreenPicture.Width = prevdest.Width;

if (tempRect.Height > prevdest.Height)
this.fullscreenPicture.Height = tempRect.Height;
else this.fullscreenPicture.Height = prevdest.Height;

this.fullscreenPicture.Top =
(int)(desired.Height - this.fullscreenPicture.Height) / 2;

this.fullscreenPicture.Left =
(int)(desired.Width - this.fullscreenPicture.Width) / 2;

...

dest =
new Rectangle(OffsetX, OffsetY, tempRect.Width, tempRect.Height);

prevdest =
new Rectangle(prevOffsetX, prevOffsetY,
previousImage.Width, previousImage.Height);
}

```

Se poate observa cel mai bine în corpul acestei metode procedeul de lucru ce stă în spatele algoritmului de suprapunere a celor două imagini. Orice ajustare de dimensiune ori de culoare sau transparență are loc pe două căi, câte una pentru fiecare imagine încărcată. Odată calculate caracteristicile de poziționare pentru imaginea curentă, caracteristicile anterior calculate vor trece în planul secund însă nu se vor pierde în totalitate, fiind necesară folosirea lor în redarea imaginii de la care se pleacă.

În mijlocul acestei metode stă apelul către o metodă statică a clasei "FullScreen()", "resize_rectangle" ce ia ca și parametri două obiecte de tip „Rectangle” și două variabile de tip logice.

Cele două dreptunghiuri conțin de fapt informații despre dimensiunile fizice ale imaginii ce se dorește a fi redată pe ecran, respectiv despre dimensiunile fizice ale ferestrei maximizate, deci implicit a rezoluției la care se lucrează.

Variabilele logice pot lua doar două valori, „True” sau „False” („Adevărat”, respectiv „Fals”) în funcție de valoarea cu care se face scalarea imaginilor cu un factor supraunitar, în acest caz, având de a face cu așa numitul efect de întindere („Enlarge”), respectiv cu un factor subunitar, având aici de a face cu o operație de micșorare („Shrink”).

După efectuarea tuturor operațiilor menționate mai sus, singurul lucru ce mi-a rămas de făcut a fost afișarea diferitelor informații în legătură cu schimbările aduse de creșterea, ori descreșterea indexului imaginilor.

Metoda creată special pentru a îndeplini această funcție are următoarea formă:

```

private void DisplayInfo(int index)
{
    this.slideShowListView.Items[index].Focused = true;
    this.slideShowListView.Items[index].EnsureVisible();

    try
    {
        this.IndexPanel.Text = (index+1).ToString()+ '/' +
this.slideShowListView.Items.Count.ToString();
        this.NamePanel.Text =
this.slideShowListView.Items[index].Text;
        this.ResPanel.Text = this.source.Width.ToString()+ 'x'+
this.source.Height.ToString()+ " pixels";
        this.ActualResPanel.Text = this.dest.Width.ToString()+ 'x'+
this.dest.Height.ToString()+ " pixels";
    }
    catch { }
}

```

Pentru a fi accesibilă din orice metodă a clasei "FullScreen()", funcția de afișare a informațiilor despre imagini și progresul diaporamei ia ca și argument un număr întreg, ce reprezintă de fapt indexul numeric al imaginii din listă.

Structural, aceasta este foarte simplu realizată, modificându-se doar textul ce trebuie afișat în bara de status al panoului de control, schimbându-se la apelul metodei numărul de ordine al imaginii curente, numele fișierului grafic ce este reprezentat, dimensiunile imaginii originale, ori dimensiunile la care a fost reprezentată imaginea curentă.

4.2.2.13 Butoanele de navigație

Pe baza codului ce stă în sptele celor două butoane prin intermediul cărora se realizează navigația înainte sau înapoi în lista de imagini, se bazează toate celelalte operații de modificare ale indexului imaginii curente, fie că acesta survine în urma cererii de la un temporizator sau că se realizează un efect cu durată predeterminată.

Pot exemplifica cu codul ce stă în spatele navigației "înainte" în lista de imagini:

```

private void button2_Click(object sender, System.EventArgs e)
{
    pic_index++;
    transparency = 1.0f;
    this.GetPath(pic_index);
    this.Load_Full_Image(path);
    ...
}

```

Prima instrucțiune ce este rulată incrementează indexul imaginii curente cu o unitate, simbolizând tipul de navigare de care am aminitit.

La fel ca și în cazul încărcării primei imagini succesiunea următoarelor apeluri de metode este aceeași. În primul rând se setează transparența imaginii ce urmează

a fi redată la valoarea 1, ceea ce semnifică o imagine proiectată în diaporama analogică cu intensitate luminoasă maximă. În faza imediat următoare se preia calea imaginii cu indexul modificat și se realizează propriu-zis transpunerea ei pe ecran.

Algoritmul este similar în cazul în care navigarea se face în sens invers, diferind doar prima instrucțiune, ce va fi înlocuită de o decrementare a indexului imaginii curente.

Pentru a exemplifica modul în care se apelează metodele bazate pe evenimente ca și cele menționate mai sus și în cazul în care evenimentele pentru care ele au fost proiectate nu apar, consider codul de mai jos ce implementează rularea automată, la un interval predefinit de timp a diaporamei.

```
private void FullScreenTimer_Tick(object sender, EventArgs e)
{
    if (this.FWRadioButton.Checked)
        if (this.button2.Enabled) this.button2_Click(sender,e);
    if (this.BKRadioButton.Checked)
        if (this.button1.Enabled) this.button1_Click(sender, e);
}
```

Grupul de instrucțiuni de mai sus este cel ce stă în spatele funcționalității de rulare a diaporamei în mod automat. Metoda descrisă este apelată la apariția evenimentului de tact al unui temporizator. Temporizatorul în cauză este cel ce poate fi reglat de utilizator prin modificarea valorii din grupul "SlideShow Interval" prezent în cadrul panoului de control al diaporamei.

Logica este foarte simplă. Se pune problema activării unuia dintre cele două butoane "Radio". În cazul în care este activ cel care direcționează spre o navigare înainte în lista de imagini ("FWRadioButton.Checked" este bifat) se face apelul către aceeași metodă ce este apelată la apariția evenimentului de apăsare al butonului de navigare înainte.

Se procedează în mod analog în cazul în care este activ butonul "Radio" ce direcționează sensul de derulare al diaporamei înspre "înapoi".

4.2.2.14 Butoane cu efect de tranziție a căror durată este predefinită

Se pune problema realizării tranzițiilor dintre imagini într-o perioadă de timp de dinainte știută, spre exemplu la 1, 2 ori 6 secunde. Cea mai comodă metodă de realizare a fost să dezvolt o metodă ce ia ca și parametru durata de timp pe care urmează a se desfășura efectul și apelarea ei cu parametrul dorit.

```
private void PredefineEffect(int period)
{
    this.effectTimeCounter.Value = period;
    this.alphaButton.Checked = true;
    if (isCtrlDown)
    {
        if (this.button1.Enabled == true)
            this.button1_Click(this.Handle, EventArgs.Empty);
    }
}
```



```

else if (this.button2.Enabled == true)
this.button2_Click(this.Handle, EventArgs.Empty);
}

```

În funcție de valoarea parametrului primit, se setează valoarea temporizatorului corespunzător realizării efectului de contopire a imaginilor. Pentru a face activă această funcționalitate se bifează artificial butonul ce indică modul de trecere între imagini prin efect de transparență și se verifică dacă este vorba despre o navigație în sens crescător sau o navigație în sens descrescător în lista de imagini, în funcție de valoarea variabilei logice "isCtrlDown" ce marchează dacă este sau nu apăsată tasta "Ctrl" a tastaturii.

Iată un exemplu al apelării acestei metode pentru a realiza o trecere între două imagini cu un efect de contopire ce va dura trei secunde:

```

private void button3s_Click(object sender, EventArgs e)
{
PredefineEffect(3000);
}

```

După cum se observă, la apariția evenimentului ce semnalează faptul că utilizatorul a apăsat butonul ce semnifică efectul de trecere de 3 secunde, este apelată metoda creată în prealabil cu parametrul "3000".

4.2.2.15 Metoda de redesenare a obiectului imagine

Am gândit aplicația de diaporamă digitală ca să realizeze contopirea celor două imagini ce se succed, prin desenarea acestora pe aceeași suprafață a unui obiect dinainte creat cu proprietatea de transparență modificată într-un mod complementar. Dacă la un moment, una din cele două imagini are transparența la valoarea maximă cea de-a doua va avea această proprietate ajustată complementar, fiind minimă. Această complementaritate este păstrată pe întreaga durată de derulare a tranziției dintre cele două imagini.

Metoda ce realizează această redare duală, formând cea de-a treia imagine pe ecran ca și compunere a celor două imagini de plecare, are următoarea formă și se apelează de fiecare dată când este cerută redesenarea obiectului imagine al clasei "FullScreen".

```

protected void OnPaint(object sender,
System.Windows.Forms.PaintEventArgs e)
{
e.Graphics.Clear(this.BackColor);

// alpha value is specified by transparency attribute
nextCm.Matrix33 = transparency;
...
e.Graphics.DrawImage(this.nextImage,
dest, 0, 0,
this.nextImage.Width, this.nextImage.Height,
GraphicsUnit.Pixel, nextIA);
...

```

```
    prevCm.Matrix33 = 1 - transparency;  
    ...  
    e.Graphics.DrawImage(this.previousImage,  
    prevdest, 0, 0,  
    this.previousImage.Width, this.previousImage.Height,  
    GraphicsUnit.Pixel, prevIA);  
    }  
    }
```

Obiectul "e.Graphics" reprezintă suportul pe care se face redarea celor două imagini suprapuse. Se pot observa cele două apeluri ale metodei "DrawImage" pentru obiectele de tip "Bitmap" de dinainte "calculate", nextImage, respectiv previousImage. Acestea vor fi "desenate" în dreptunghiurile destinație, de asemenea dinainte calculate "dest" și "prevdest".

Această metodă este apelată de fiecare dată când este necesară redesenarea ferestrei suport prin apelarea metodei

```
    this.fullscreenPicture.Refresh();  
    sau  
    this.fullscreenPicture.Invalidate();
```

4.2.2.16 Controlul tranziției cu ajutorul mouse-ului

Elementul comun ce stă între aplicația de diaporamă digitală și spectacolele de diaporamă analogică este posibilitatea de a controla timpii de tranziție dintre imaginile consecutive, prin intermediul unui dispozitiv auxiliar, în speță mouse-ul.

Codul ce stă în spatele aceste funcționalități este grupat în regiunea "Mouse Control" din cadrul clasei "FullScreen".

Pentru a se urmări progresul cursorului pe ecran și starea de apăsare a butoanelor se declară următoarele variabile:

```
    bool isMouseDown;  
    int startXPos;  
    int startYPos;  
    float prevTransparency;
```

Prima dintre ele se setează în cazul în care utilizatorul apasă pe unul din butoanele mouse-ului, cea de-a doua și cea de-a treia servesc pentru a reține coordonatele de pe ecran la care s-a apăsă butonul mouse-ului, iar cea de-a patra reține transparența anterioară cu care s-a desenat imaginea curentă pe ecran.

Pentru a urmări traiectoria mouse-ului pe ecran, se folosesc trei metode bazate pe evenimente: una este apelată la apăsarea unuia din butoanele mouse-ului, a doua este apelată la eliberarea butonului în prealabil apăsă, iar apelul celei de-a treia survine la modificarea coordonatelor cursorului mouse-ului pe ecran.

La apăsarea unui buton al mouse-ului se execută următoarea secvență de cod:

```
private void FullScreen_MouseDown(object sender, MouseEventArgs e)  
{  
    isMouseDown = true;
```

```
startXPos = e.X;  
startYPos = e.Y;
```

```
...  
}
```

reținându-se coordonatele la care a survenit acest eveniment și marcându-se prin setarea variabilei booleane dedicate acestui scop că utilizatorul a apăsat unul din butoanele mouse-ului.

Când utilizatorul eliberează butonul apăsat, se apelează următoarea metodă:

```
private void FullScreen_MouseUp(object sender, MouseEventArgs e)  
{  
    isMouseDown = false;  
  
    if (transparency < 0.5f)  
    {  
        transparency = 1.0f;  
        pic_index--;  
        this.GetPath(pic_index);  
        this.Load_Full_Image(path);  
    }  
    else  
    {  
        transparency = 1.0f;  
        this.fullscreenPicture.Refresh();  
    }  
    ...  
}
```

Aceasta, asemenea evenimentului ce determină apelul ei, rulează codul complementar celui prezentat în aliniatul de mai sus, în primă fază resetându-se variabila ce ține starea de apăsare a butoanelor mouse-ului. Ulterior se ia decizia asupra imaginii ce trebuie să fie în prim plan, după cum urmează, imaginea de la care s-a plecat, dacă traiectoria mouse-ului nu trece de mediana pe verticală a imaginii, respectiv imaginea la care se dorește să se ajungă, dacă traiectoria mouse-ului a depășit mediana luată în discuție.

În cazul în care mouse-ul are o traiectorie variabilă pe verticală și unul din butoanele mouse-ului este ținut apăsat, trebuie efectuate calculele necesare pentru determinarea transparenței intermediare a imaginii curente:

```
private void FullScreen_MouseMove(object sender, MouseEventArgs e)  
{  
    float transparencyFactor;  
  
    if (isMouseDown)  
    {  
        transparencyFactor =  
        (float)(e.Y - startYPos) / (this.Height - startYPos);  
  
        transparency = transparencyFactor;
```

```
...
if (Math.Abs(prevTransparency - transparency) > 0.01)
{
this.fullscreenPicture.Refresh();
prevTransparency = transparency;
}

}
}
```

După cum se observă, calcularea transparenței intermediare se face doar în cazul în care cursorul indicator al mouse-ului se află în mișcare și unul din butoane este apasat. În acest caz, considerăm progresul pe verticală al traiectoriei mouse-ului relativ la coordonatele la care a fost apăsat butonul. În funcție de această progresie, se calculează un factor "transparencyFactor" ce este mai apoi atribuit transparenței imaginii curente.

Este necesar a se face un test de progresie a traiectoriei pentru a nu se suprasolicita capacitatea de procesare a sistemului și a determina redarea unui număr de cadre redus ce ar scădea calitatea optică a trecerilor dintre imagini. Acest test este de fapt limitarea inferioară a variației transparenței ce determină redesenarea imaginii contopite la o valoare de 1% în exemplul prezentat mai sus.

Doar în situația când noua transparență calculată are o variație pozitivă sau negativă mai mare cu 1% față de precedentă transparență ce a determinat desenarea imaginilor contopite se apelează din nou metoda de redesenare a obiectului imagine.

4.3 Testarea aplicației "Digital Diaporama"

În principiu, evaluarea calitativă a aplicațiilor de diaporama digitala va trebui să se raporteze în principal la calitatea vizuală a tranzițiilor și la respectarea duratei acestora, comparativ cu timpul comandat prin intermediul tastaturii.

Desigur, există criterii universale de evaluare a unei aplicații și este evident că și acestea vor fi utilizate.

Cel mai important test este cel al calității tranziției și al întârzierii față de durata comandată. Pentru aceasta s-a folosit un laptop ASUS, cu următoarea configurație: **Intel Pentium Dual CPU T2330 @ 1,60GHz 798Mhz, 1,87GB RAM** și un cronometru Swatch cu precizie de sutime de secundă.

Am realizat cinci biblioteci de imagini pornind de la originalele în format .NEF (RAW al firmei Nikon) [67], fiecare conținând 8 imagini. Imaginile au fost salvate în format .JPEG în cinci moduri : 1280/800 pixeli, -cu un spațiu mediu ocupat de 700 kB, respectiv 200 kB și 50 kB (corespunzătoare salvării best, medium și low al

4.3 Testarea aplicației "Digital Diaporama" 77

Adobe Photoshop)- și 800-600 pixeli –cu un spațiu mediu ocupat de 300 kB respectiv 60 kB (corespunzătoare salvării best și medium) [68], [69] .

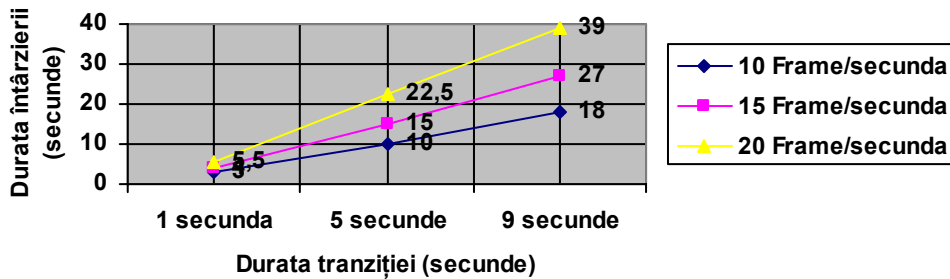
Cu ajutorul tastaturii numerice am comandat tranziții -de 1, 5 și respectiv 9 secunde- pentru fiecare bibliotecă de imagini. Aceste tranziții au fost mai apoi cronometrate, am calculat întârzierile față de timpul solicitat și am făcut media acestora. Tabelul ilustrează numeric aceste măsurători.

Dimensiunea Imaginii (pixeli)	Spațiul mediu ocupat (kB) Calitate JPEG	Numărul de frame- uri pe secunda ale tranziției	Durata reală a tranziției pt comanda de 1 secundă (s)/ întârziere(s)	Durata reală a tranziției pentru comanda de 5 secunde (s) /întârziere(s)	Durata reală a tranziției pentru comanda de 9 secunde (s) /întârziere(s)
1280-800	700 k Best	10	4/ 3	15/ 10	27/ 18
		15	5/ 4	20/ 15	36/ 27
		20	6,5/ 5,5	27,5/ 22,5	48/ 39
1280-800	200 k medium	10	2,6/ 1,6	10,5/ 5,5	18,7/ 9,7
		15	4/3	15,4/10,4	26,4/ 17,4
		20	5/4	20,1/15,1	35,7/ 26,7
1280-800	50 k Low	10	2,3/ 1,3	9,3/ 4,3	16,6/ 7,6
		15	3,4/ 2,4	14/ 9	24,2/ 15,2
		20	4,2/ 3,2	17,9/ 12,9	31/ 22
800-600	300 k Best	10	1,6/ 0,6	6,1/ 1,1	10,3/ 1,3
		15	2,2/ 1,2	8,1/ 3,1	13,5/ 4,5
		20	2,5/ 1,5	10,3/ 5,3	18,1/ 9,1
800-600	60 k medium	10	1,4/ 0,4	6/ 1	10,2/ 1,2
		15	1,5/ 0,5	6,1/ 1,1	11/2
		20	1,9/ 0,9	7,3/ 2,3	13,8/ 4,8

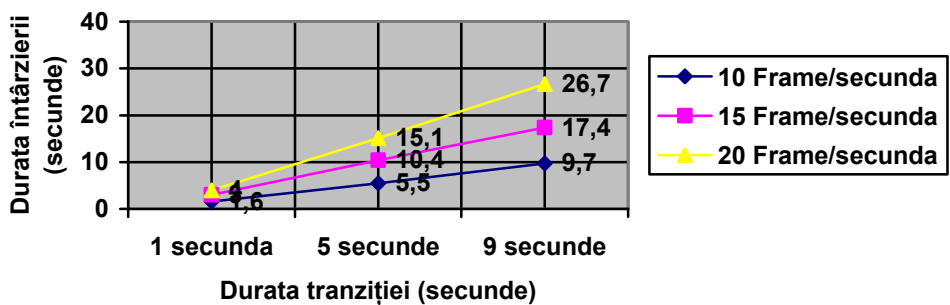
Tabelul 2: Tranziții/Întârzieri

Interpretând datele de mai sus defalcat pe biblioteci de imagini, am obținut următoarele reprezentări grafice ale întârzierilor de tranziții.

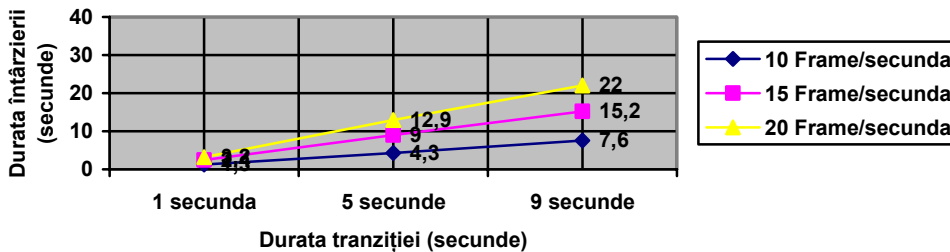
1280-800 pixeli 700k best



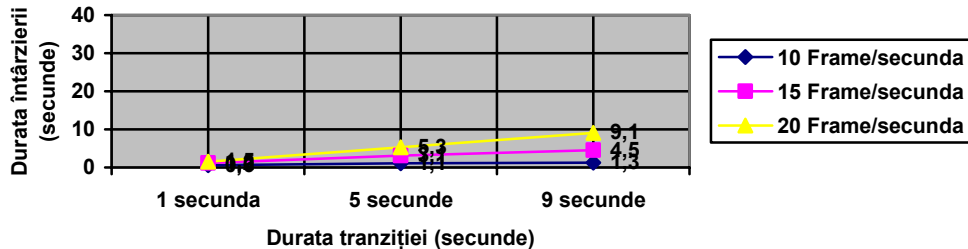
1280-800 pixeli 200k medium



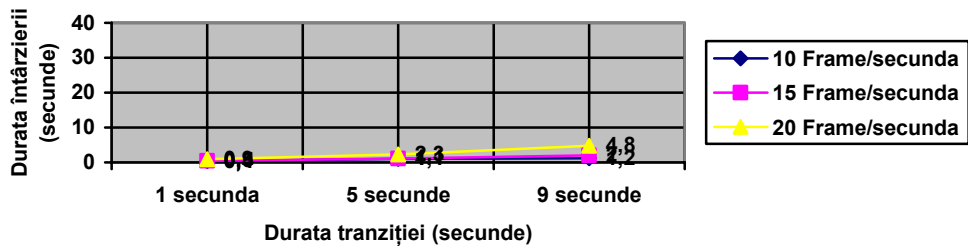
1280-800 pixeli 50k low



800-600 pixeli 300k maximum



800-600 pixeli 60k medium



Tabelul 3: Întârzierile de tranziții pentru diferite biblioteci de imagini

Devine evident faptul că întârzierile, mari, sunt proporționale atât cu dimensiunea imaginii, cât și cu spațiul ocupat respectiv calitatea imaginii. Iar aceste neajunsuri se referă doar la imagini în format .JPEG, unde au intervenit deja diverse grade de compresie care au ca efect, pe lângă scăderea spațiului ocupat, și scăderea calității imaginii.

Utilizând ulterior alte computere, rezultatele obținute de mine au fost similare, întârzierile nefiind acceptabile, din punctul meu de vedere, în situația utilizării unor imagini de o dimensiune acceptabilă calitativ.

4.4 Contribuții aduse prin aplicația "Digital Diaporama"

"Digital Diaporama" reușește să atingă obiectivul principal, acela de a controla o prezentare ce include imagini digitale, tranzițiile între ele, precum și o coloană sonoră dedicată [54].

Butoanele aplicației, aspectul său și folosirea mouse-ului pe post de telecomandă, prezintă asemănări mari cu dispozitivele similare analogice de la care s-a inspirat [35], [4].

Aplicația reprezintă o punte între diaporama analogică și cea digitală, cu avantaje și perspective clare. Neajunsurile acestora sunt date de rezultatele testării, respectiv o neconcordanță între durată solicitată a tranziției, și cea reală, precum și

80 Elaborarea și implementarea primului model de aplicație. "Digital Diaporama" - 4

o tranziție sacadată în cazul folosirii mouse-ului. Privind realist, aplicația nu ar face față unei utilizări firești.

"Digital Diaporama" rămâne doar un prim pas -important- în studiul unei aplicații ce permite controlul în timp real al tranzițiilor de imagini. Lipsa unui răspuns adecvat la expectanțele avute a dus la orientarea mea către etapa a doua de studiu, respectiv "Photo Slide Toolbar".

5. Elaborarea și implementarea celui de-al doilea model de aplicație. “PhotoSlide Toolbar”

5.1 Unelte software folosite pentru “PhotoSlide Toolbar”

5.1.1 Microsoft Visual Studio 2008

Microsoft Visual Studio este un mediu integrat de dezvoltare (Integrated Development Environment- IDE) pentru Microsoft [15].

Un mediu integrat de dezvoltare sau un mediu integrat pentru design, sau chiar mediu integrat de depanare este o aplicație software care asigură facilități cuprinzătoare pentru programatori în vederea dezvoltării de software. Un IDE în mod normal cuprinde:

- Un editor de cod sursă
- Un compilator sau interpretor de cod
- Unelte pentru construcția automată a soluției
- Un debugger

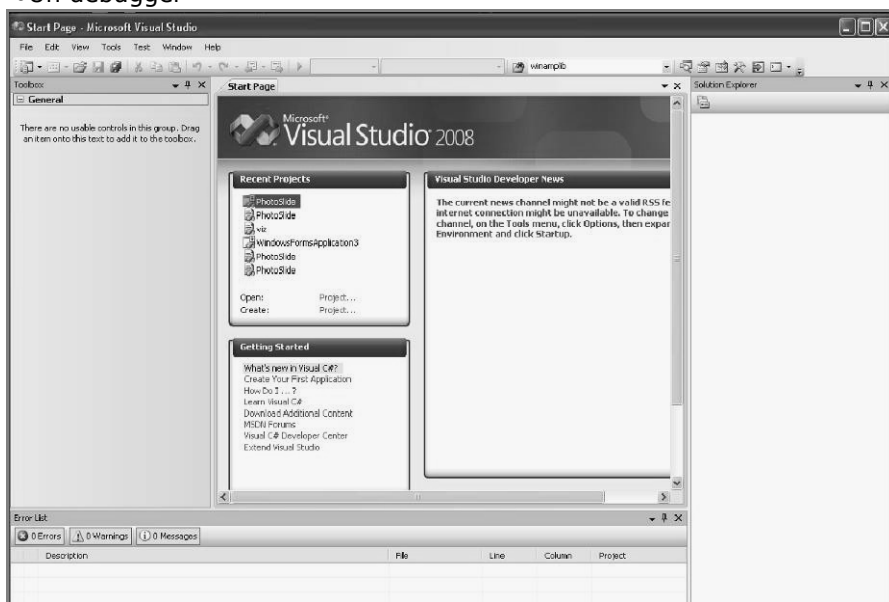


Fig. 23 - Interfața Visual Studio 2008

82 Elaborarea și implementarea celui de-al doilea model de aplicație. "PhotoSlide Toolbar" - 5

Acesta poate fi folosit pentru dezvoltarea de console și interfețe grafice cu utilizatorul, precum și aplicații de tip Windows Form (aplicații de tip fereastră), site-uri web, aplicații și servicii web, fie în cod nativ sau cu ajutorul codului de ajutor pentru toate platformele suportate de Microsoft: Windows Microsoft, Windows Mobile, WindowsCE, .NET Framework, etc.

Visual Studio include un editor de cod ce suportă IntelliSense (implementare a autocompletării codului). Debugger-ul integrat lucrează atât la nivel de sursă cât și la nivel de mașină. Alte facilități includ forme designer pentru construirea de aplicații cu Interfața Grafică cu Utilizatorul (GUI), forme designer web (pentru construirea de clase) și scheme de tip bază de date. Permite adăugarea de plug-in-uri care măresc funcționalitatea la aproape toate nivelele.

Visual Studio suportă limbaje, medii de servicii ale limbajului care permit ca orice tip de limbaj de programare să fie suportat de către editorul de cod și debugger.

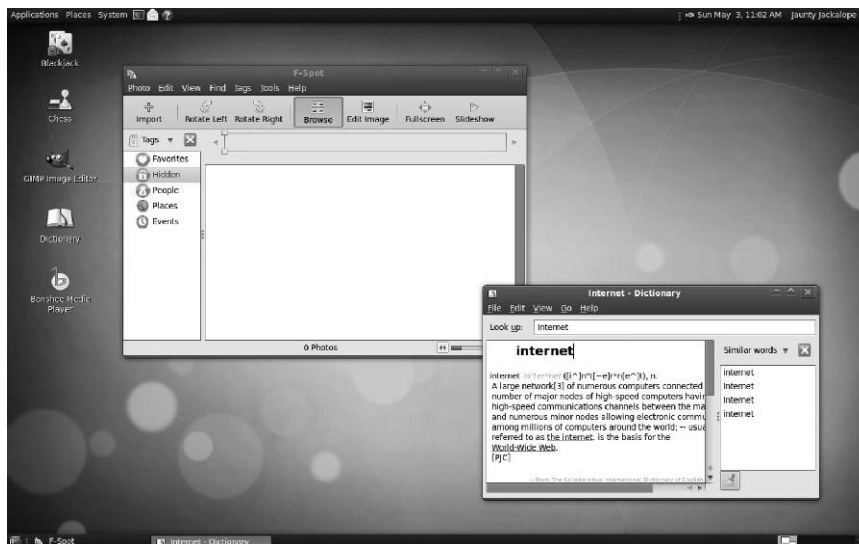


Fig. 24 - Printscreen modern de tip GUI

Limbaje incluse suportate: C/C++, VB.NET, C#, precum și alte limbaje ca F#, M, Python și Ruby, dar și altele care au devenit disponibile prin intermediul serviciilor de limbaj ce pot fi instalate separat. Suportă deasemenea XML/XSLT, HTML/XHTML, Java Script și CSS. Există variante Visual Studio care oferă o varietate limitată de limbaje, numite: Microsoft Visual Basic, Visual J#, Visual C#, Visual C++ [13].

În prezent Visual Studio 2008 precum și varianta 2005 sunt disponibile studenților în mod gratuit prin programul DreamSpark. Visual Studio 2010 era la ora realizării studiului (în 2008) în varianta beta, în testare și dezvoltare.

Variantele anterioare Visual Studio erau sincronizate, și prin urmare legate de o versiune specifică .NET Framework, care la rândul ei era legată de motorul CLR (Common Language Runtime). În 2006 Microsoft lansează .NET Framework varianta 3.0, dar spre deosebire de variantele anterioare, nu este lansată o variantă nouă de CLR sau o nouă versiune a compilatoarelor de limbaj, sau o nouă versiune Visual

Studio. Deasemenea, în 2006 sunt lansate Windows Vista și Office 2007, fiecare cu platformă de dezvoltare proprie. Ulterior, dezvoltatorii au căutat să îmbunătățească uneltele pentru a adresa corespunzător cele trei platforme. Tocmai acest gol este umplut de Visual Studio 2008, prin oferirea de support principal pentru .NET Framework 3.0, Windows Vista și Office 2007.

În același timp cu Visual Studio 2008, o nouă versiune a .NET Framework este lansată (variantea 3.5), și un pachet de servicii pentru variantele v2.0 și v3.0. Deasemenea sunt dezvoltate noi compilatoare pentru Visual Basic și C#. Printre cele mai importante avantaje ale Visual Studio 2008 este că poate lucra cu oricare dintre versiunile arhitecturilor .NET Framework, și deasemenea poate fi privit ca un superset al Visual Studio 2005: nu numai că poate co-exista pe aceeași mașină cu Visual Studio 2005, nu numai că formatul fișierelor este compatibil între cele două versiuni, dar Visual Studio 2008 vede proiecte existente ale arhitecturilor .NET Framework v2.0 și v3.0.

Mai sunt o mulțime de trăsături adiționale în ceea ce privește estetica și utilizarea Visual Studio 2008. Acesta a fost proiectat și construit ca un excelent Vista client, și evident că funcționează foarte bine și pe XP.

Visual Studio 2008 prezintă următoarele puncte de avans:

- Dezvoltarea rapidă de aplicații- de la modelare la codare și depanare, Visual Studio 2008 oferă un limbaj îmbunătățit, designer, editor și trăsături de manipulare a datelor care ajută la creșterea productivității.

- Lucrul cu datele într-un mod unificat și integrat- este îmbunătățit în mod semnificativ felul în care dezvoltatorii manevrează datele. În mod tradițional, datele trebuiau manevrate în funcție de locație, și de odul în care se poate conecta utilizatorul la ea. Odată cu LINQ (Language Integrated Query) dezvoltatorii pot folosi un singur model pentru interogarea și transformarea XML, Microsoft SQL Server și obiectele de tip data fără a utiliza sau învăța limbaj specializat, de unde rezultă o reducere a complexității și o creștere a productivității.

- Construirea de aplicații care rulează pe multiple versiuni ale .NET Framework- cu Visual Studio 2008 dezvoltatorii de aplicații au abilitatea de a folosi o singură unealtă pentru manevrarea și construcția de aplicații care au ca țintă mai multe versiuni ale .NET Framework. Visual Studio 2008 va adapta proiectele și setările disponibile pentru versiunea .NET Framework specificată. Nu mai este necesară existența mai multor versiuni Visual Studio pentru aplicații ale diferitelor arhitecturi .NET Framework.

- Colaborare între ciclurile de dezvoltare- Visual Studio încurajază dezvoltatorii, designerii, testerii, arhitecții și project-managerii să lucreze împreună cu ajutorul uneltelor comune și a integrării procesului pentru reducerea timpului de găsim a unei soluții.

- Crează experiențe remarcabile ale utilizatorului- oferă noi unelte de dezvoltare ce unesc aplicații folosind cele mai noi platforme, incluzând Web, Windows Vista, Microsoft Office 2007, Microsoft SQL Server, Windows Mobile și Windows Server 2008.

- Experiența noilor unelte în dezvoltarea de tip Web- este oferită o platformă robustă, pentru construirea, gazduirea și publicarea aplicațiilor pe Web.

- Construirea de aplicații solide și scalabile pentru Microsoft Office System- uneltele Visual Studio pentru Microsoft Office sunt acum în totalitate integrate în Visual Studio 2008 Professional Edition: se permite customizarea Microsoft Office Word, Microsoft Office Excel, Microsoft Office Power Point, Microsoft Office Outlook, Microsoft Office Visio, Microsoft Office InfoPath și Microsoft OfficeProject, pentru

îmbunătățirea usabilității și pentru a putea fi observate mai bine îmbunătățirile aduse sistemului Office 2007.

•Construirea de aplicații uluitoare pentru Windows Vista-Visual Studio 2008 include îmbunătățiri care ajută dezvoltatorii să creeze aplicații de tip Windows Vista relativ repede și ușor [60].

În cadrul utilizării acestui mediu de programare, după lansare am ales opțiunea New Project din meniul File.

În fereastra de dialog (Fig.25), am selectat tipul aplicației dorite, după care, la Name, introducem numele aplicației noastre. Fereastra în care am scris programul se numește implicit Programs.cs și se poate modifica prin salvare explicită(Save As).

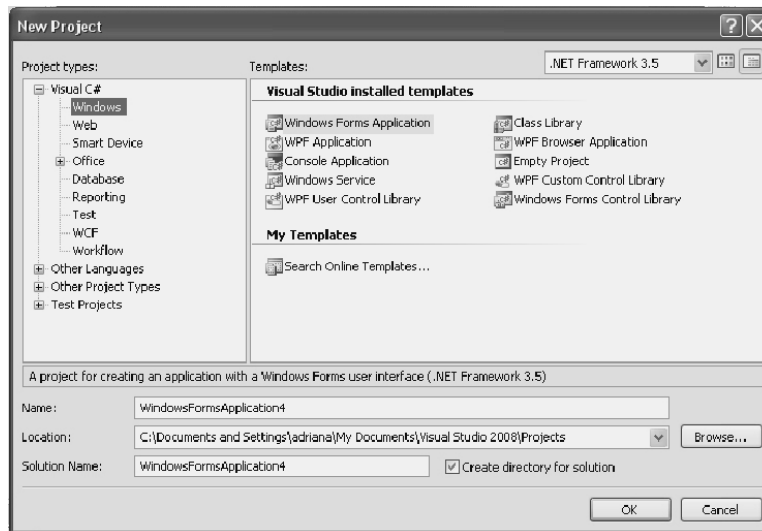


Fig. 25 - Fereastra de dialog la începerea unui nou proiect

Extensia .cs provine de la **C Sharp**. În scrierea programului am fost asistat de IntelliSense, ajutorul contextual.

Compilarea programului se realizează cu ajutorul opțiunii Build Solution (F6) din meniul Build. Posibilele erori de compilare sunt listate în fereastra Error List.

Efectuând dublu click pe fiecare eroare în parte, cursorul din program se poziționează pe linia conținând eroarea.

5.1 Unelte software folosite pentru "PhotoSlide Toolbar" 85

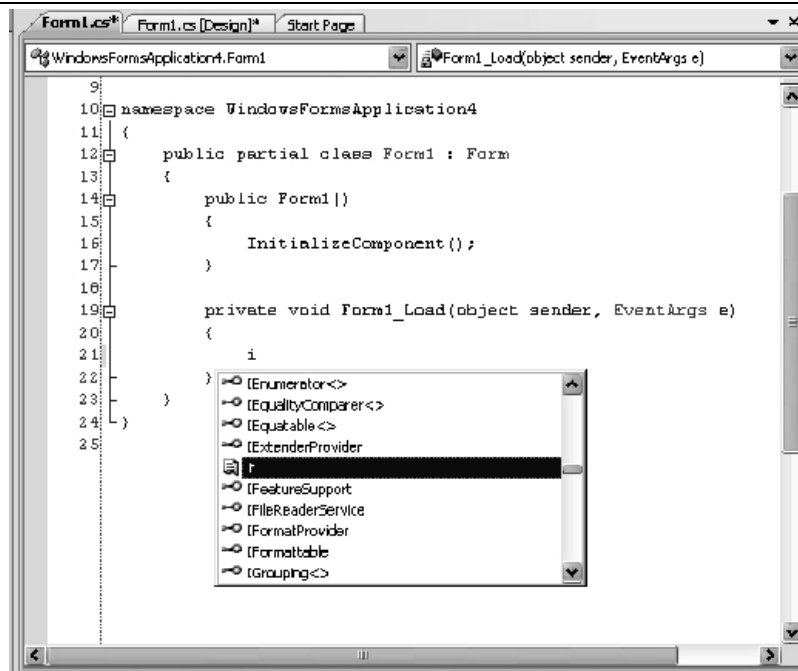


Fig. 26 - Asistare IntelliSense la scrierea unui program

Rularea programului se poate realiza în mai multe moduri: rapid fără asistență de depanare (Start Without Debugging Shift+F5), rapid cu asistență de depanare (Start Debugging F5 sau cu butonul din bara de instrumente), rulare pas cu pas (Step Into F11 și Step Over F12) sau rulare rapidă până la linia marcată ca punct de întrerupere (Toggle Breakpoint F9 pe linia respectivă și apoi Start Debugging F5).

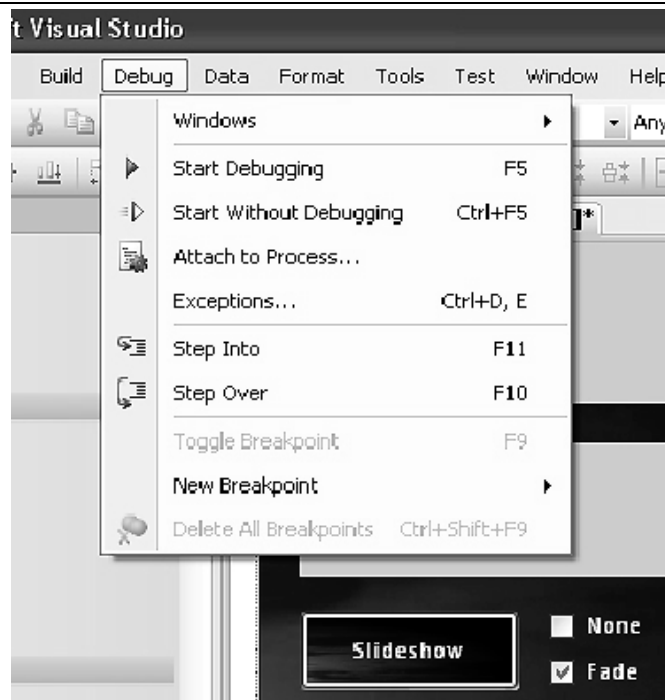


Fig. 27 - Modalități de rulare ale unui program

Încetarea urmăririi pas cu pas (Stop Debugging Shift+F5) permite ieșirea din modul depanare și revenirea la modul normal de lucru. Toate opțiunile de rulare și depanare se găsesc în meniul Debug al meniului. Fereastra de cod și ferestrele auxiliare ce ne ajută în etapa de editare pot fi vizualizate alegând opțiunea corespunzătoare din meniul View.

Ferestrele auxiliare utile în etapa de depanare se pot vizualiza alegând opțiunea corespunzătoare din meniul Debug/Windows.

5.1.2 De ce am ales .NET?

În primul rând pentru că mi-a oferit instrumente pe care le pot folosi și în alte programe, acces ușor la baze de date, permite realizarea desenelor sau a altor elemente grafice. Spațiul de nume System.Windows.Forms conține instrumente (controale) ce permit implementarea elementelor interfeței grafice cu utilizatorul. Folosind aceste controale, putem proiecta și dezvolta rapid și interactiv, elementele interfeței grafice. Tot .NET oferă clase care efectuează majoritatea sarcinilor uzuale cu care se confruntă programele și care plictisesc și fură timpul programatorilor, reducând astfel timpul necesar dezvoltării aplicațiilor [15], [70].

Microsoft .NET Framework este una dintre cele mai populare platforme de dezvoltare de aplicații software și servicii. Același model de programare poate fi utilizat pentru o varietate de aplicații – Web, Windows, console de jocuri, sisteme *embedded* și dispozitive mobile – în timp ce dezvoltatorii au la alegere o plajă foarte diversă de limbaje de programare și tehnologii pentru rezolvarea problemelor de arhitectură și dezvoltare.

Pe scurt, platforma este compusă dintr-un sistem de tipuri sigur, o bibliotecă de clase și un mediu de execuție a aplicațiilor:

•**Common Type System - CTS**

Reprezintă, în esență, un set de tipuri comun pentru toate limbajele de programare ce pot fi folosite pentru scrierea de aplicații peste .NET Framework (C#, Visual Basic etc). Mai mult, CTS este acompaniat de reguli pentru declararea și utilizarea tipurilor de date în .NET Framework;

•**Base Class Library - BCL**

Biblioteca de bază a .NET Framework este compusă din câteva mii de clase și interfețe, clasificate logic în spații de nume, ce oferă acces la o colecție consistentă de funcționalități deja implementate și care pot fi utilizate la dezvoltarea de noi componente, servicii și aplicații;

•**Common Language Runtime - CLR**

Execuția oricărei aplicații dezvoltată peste platforma .NET Framework este gestionată de CLR. Acest mediu coordonează rularea codului aplicațiilor prin implicarea unor servicii comune tuturor limbajelor în .NET Framework, cum sunt alocarea și dealocarea memoriei, modificarea permisiunilor de execuție, tratatarea excepțiilor și altele.

.NET Framework este o platformă de dezvoltare cu beneficii evidente în multe scenarii de dezvoltare – pentru Web, Windows și dispozitive mobile - și pentru o plajă diversificată de dezvoltatori. De exemplu, începătorii vor identifica în C# un limbaj elegant și ușor de învățat, în timp ce modelul de scriere de aplicații este unul riguros.

Dezvoltatorii experimentați pot alege .NET Framework pentru arhitectură, soluții pentru diverse provocări de design și dezvoltare sau pentru modul în care se realizează managementul proiectelor complexe.

Odată cu Windows Vista a fost lansată .NET Framework 3.0. Versiunea 3.0 nu include modificări în mediul de execuție al versiunii 2.0 (are același CLR) ci doar la nivelul bibliotecii de clase. .NET Framework 3.0 contribuie cu un suport superior pentru interfețe utilizator, conectivitate fără precedent și coordonarea proceselor de business.

Versiunea curentă - .NET Framework 3.5 - este disponibilă din 2007 și aduce îmbunătățiri în accesul la date și prelucrarea acestora. Alese corect, tehnologiile de management al datelor joacă un rol important în performanța aplicațiilor. Ceea ce ne oferă .NET Framework din acest punct de vedere este subsetul numit ADO.NET; iar limbajele de programare din .NET Framework oferă, mai nou, sintaxa și un limbaj unic pentru accesarea, manipularea și prelucrarea a datelor din fișiere text, fișiere XML și chiar din sistemul de fișiere, până la structuri de obiecte și baze de date. Această extensie a limbajelor de programare pentru manipularea datelor se numește LINQ- *Language Integrated Query*.

Pe scurt, componenta CLR și biblioteca FCL permit realizatorilor de aplicații să construiască următoarele tipuri de aplicații:

•**Servicii Web XML** Metode care pot fi accesate foarte ușor prin Internet, serviciile XML Web reprezintă, desigur, elementul esențial al inițiativei .NET a companiei Microsoft.

•**Web Forms** Sunt aplicații bazate pe limbajul HTML (situri Web). De regulă, aplicațiile Web Forms vor efectua interogări de baze de date și apelări ale serviciilor Web, vor combina și filtra informațiile returnate, pe care le vor prezenta apoi în cadrul unui browser, folosind o interfață cu utilizatorul bogată, bazată pe HTML. Aplicațiile Web Forms oferă un mediu de dezvoltare în stil Visual Basic 6 și Visual InterDev pentru aplicațiile Web scrise în orice limbaj din componenta CLR.

•**Windows Forms** Sunt aplicații Windows GUI bogate. În loc de a utiliza o pagină Web Forms pentru a crea interfața UI a aplicației, se poate folosi funcționalitatea mai puternică, cu performanțe mai ridicate, oferită de suprafața de lucru Windows. Aplicațiile Windows Forms pot profita de controalele, meniurile și evenimentele declanșate de mouse și tastatură și pot trata direct cu sistemul de operare aflat la bază. Ca și aplicațiile Web Forms, și acestea efectuează interogări ale bazelor de date și apelări ale serviciilor XML Web. Aplicațiile Windows Forms oferă un mediu de dezvoltare asemănător cu Visual Basic 6 pentru aplicațiile GUI scrise în orice limbaj CLR.

•**Aplicații Windows de consolă** Pentru aplicațiile cu cerințe UI foarte simple, o aplicație de consolă reprezintă o modalitate rapidă și facilă de a construi o aplicație. Compilatoarele, utilitarele și instrumentele sunt implementate, de regulă, ca aplicații de consolă.

•**Servicii Windows** Folosind platforma .NET Framework, este posibil să se construiască aplicații de serviciu, controlabile prin intermediul facilității Windows Service Control Manager (SCM).

•**Biblioteca de componente** Platforma .NET Framework permite construirea de componente (tipuri) autonome, care pot fi cu ușurință încorporate în oricare dintre tipurile de aplicații menționate mai sus.

5.2 SpicIE – Simple plug-in creator for Internet Explorer

Internet Explorer dispune de interfețe COM (Component Object Model) ce permit dezvoltarea miezului funcțional al Internet Explorer. SpicIE este o platformă de dezvoltare ce controlează extensiile browserului Internet Explorer. Aceasta permite dezvoltarea ușoară de extensii ale Internet Explorer în programe precum C#, VB.NET [71], [16].



Fig. 28 - Emblema SpicIE

Arhitectura

Interfețele COM ale Internet Explorer:

- foarte mare funcționalitate
- foarte mare complexitate
- greu de utilizat

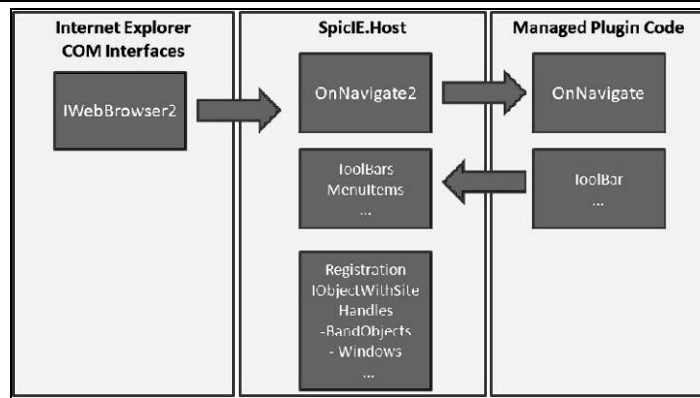


Fig. 29 - Arhitectura SpicIE

Platforma de dezvoltare SpicIE:

- încapsulează interfețele COM ale Internet Explorer
- ajută la managementul codului
- ține evidența elementelor de infrastructură și configurare
- oferă funcționalitatea de bază pentru a fi tratată ca o extensie de browser pentru Internet Explorer [72].

Tratarea evenimentelor este cea mai ușoară metodă de integrare/dezvoltare a Internet Explorer. Controlurile SpicIE, precum și elementele interfeței cu utilizatorul sunt în mod simplu desenate de SpicIE, sau în mod complex poți să-ti desenezi singur propria interfață cu utilizatorul [73].

Customizarea codului presupune:

- implementarea codului plug-in-ului ce va extinde funcționalitatea Internet Explorer
- managementul codului prin intermediul evenimentelor, elementelor de interfață cu utilizatorul, sau acces direct la interfețele COM
- limbaje de programare C#, VB.NET, C++, etc.

Cerințe de funcționalitate:

- Platforma SpicIE funcționează pe arhitecturile .NET Framework v2.0, v3.0 și v3.5.
- Pentru o cât mai ușoară dezvoltare de plug-in-uri se recomandă utilizarea oricărei variante a Visual Studio.
- Platforma SpicIE este testată pentru Internet Explorer 7.0 și Internet Explorer 8.0
- Platforma SpicIE poate fi folosită pentru limbajele C#, VB.NET, C++ sau alte limbaje de gestionare.

Iată o listă a evenimentelor ce pot fi folosite în dezvoltarea de plug-in-uri:

- OnNavigateError
- OnNavigateComplete
- OnQuit
- OnBeforeNavigate
- OnDocumentComplete
- OnDownloadBegin

- OnDownloadComplete
- OnFullScreen
- OnTheaterMode
- OnTitleChange
- OnWindowClosing
- OnFileDownload
- OnNewWindow
- OnVisible
- OnProgressChange
- OnPropertyChange
- OnSetPhishingFilterStatus
- OnUpdatePageStatus

Plug-in-urile astfel dezvoltate pot fi de mai multe feluri:
➤Explore Toolbar(de tip toolbar)

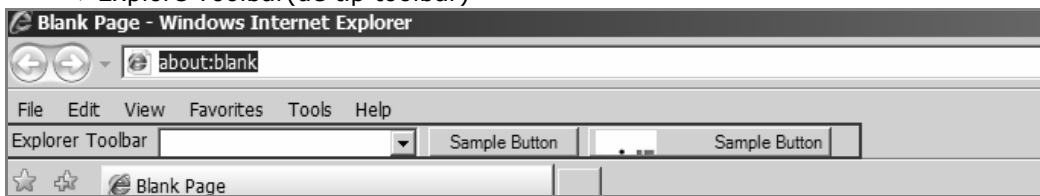


Fig. 30 - Plug-in de tip Explore Bar

➤Explore Meniu (de tip meniu orizontal/vertical)

PhotoSlide Toolbar se încadrează deasemenea la aceasta categorie, de tip meniu vertical. Dacă aceasta nu apare, ea trebuie selectată din meniu: View->Explore Bar->PhotoSlide Toolbar, sau orice alt nume s-a ales pentru meniu [74].

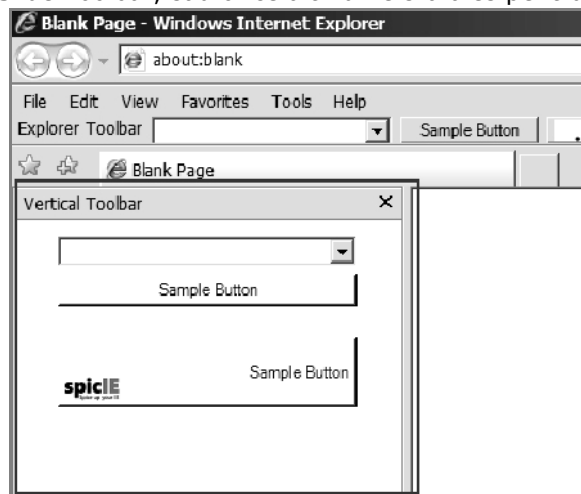


Fig. 31 - Plug-in de tip Explore Menu

5.3 Aplicația "PhotoSlide Toolbar"

5.3.1 Descrierea interfeței cu utilizatorul. Fereastra browser

Am conceput aplicația "PhotoSlide Toolbar" ca un plug-in pentru browserul furnizat de Microsoft, Internet Explorer.

Un **plugin** este un program care poate să se integreze într-un alt program (de bază) pentru a îndeplini funcții specifice. Termenul provine din verbul englez *plug-in*, care înseamnă a conecta. Exemple tipice de plugin-uri sunt cele pentru afișarea formatelor grafice (de ex. SVG dacă browserul nu include implicit acest format), pentru a executa fișiere multimedia, pentru a cripta/decripta e-mail-uri (de exemplu PGP), sau pentru a filtra imagini în programe grafice. Programul de bază (browserul sau un client de poștă electronică ș.a.) setează un standard pentru schimbul de date cu plugin-ul, permite accesul plugin-urilor la datele din program și execută cererile plugin-urilor. Mulți producători de software cu renume oferă proiectanților de programe API-uri ("*Application Programming Interface*") pentru plugin-uri, pentru a mări utilitatea produsului de bază. Pentru dezvoltarea "PhotoSlide Toolbar" am dispus de interfața API SpicIE, deasemenea sub tutela Microsoft, iar ca mediu de programare Visual Studio 2008, respectiv limbaj de programare C#. Am detaliat unele caracteristici ale acestor unelte software în capitolul destinat lor [75].

Utilitatea aplicației reunește utilizatorii de Internet Explorer care doresc să realizeze prezentări interactive de tip Diaporama, indiferent de scopul acestora: educațional, cultural, artistic, de promovare a unor produse sau instituții, asociații sau firme.

Întocmai ca la aplicația precedentă, am realizat două stări ale aplicației despre care voi vorbi în două subcapitole separate. Una dintre interfețele aplicației, prima cu care utilizatorul intră în contact este cea din fereastra browser-ului, iar cea de-a doua, FullScreen-ul efectiv al prezentării electronice de imagini selectate în cadrul primei ferestre. Voi descrie în detaliu toate controalele disponibile în ambele stări ale aplicației.

Interfața plug-in-ului este următoarea (așa arată Internet Explorer odată ce utilizatorul instalează PhotoSlideToolbar):

92 Elaborarea și implementarea celui de-al doilea model de aplicație. "PhotoSlide Toolbar" - 5

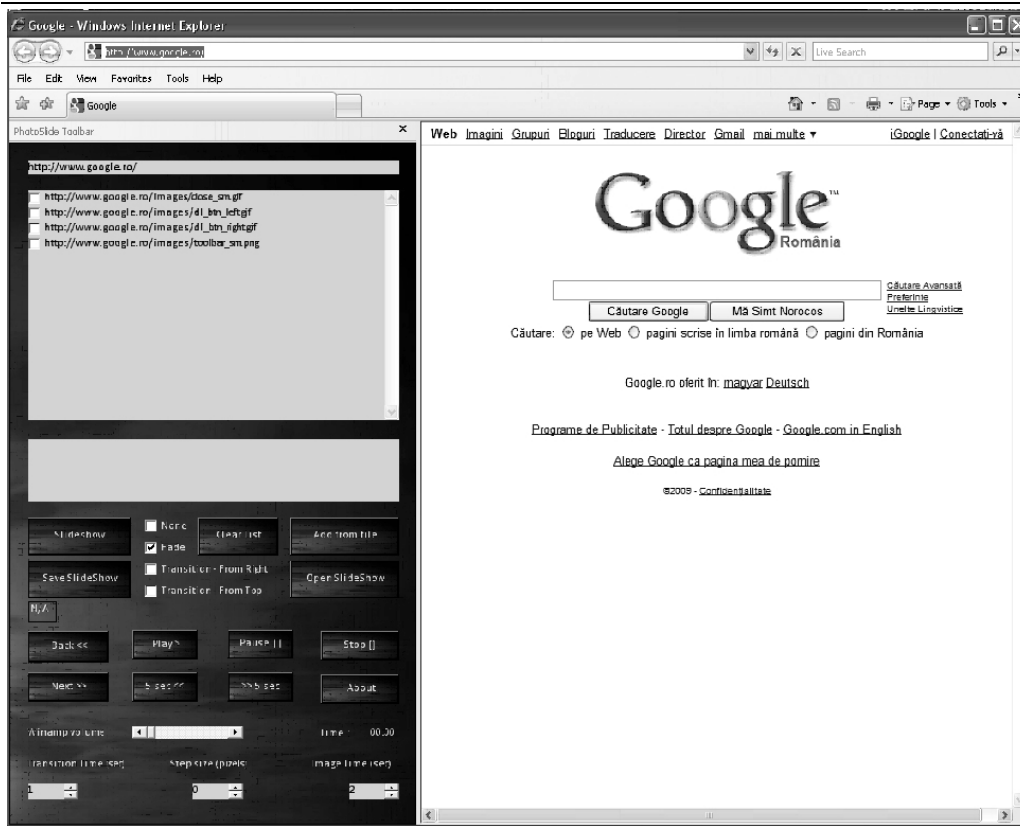


Fig. 32 - Interfața "PhotoSlide Toolbar" în browser

"PhotoSlide Toolbar" este un meniu de tip vertical ce apare în partea stângă a browserului Internet Explorer.

În partea dreaptă a ecranului va fi afișată, ca de obicei pagina web pe care se navighează în momentul respectiv.

Daca meniul vertical nu este vizibil, acesta trebuie selectat din meniul Internet Explorer:

View->Explore Bar->PhotoSlideToolbar.

Selecția se va face ulterior instalării plug-in-ului [76].

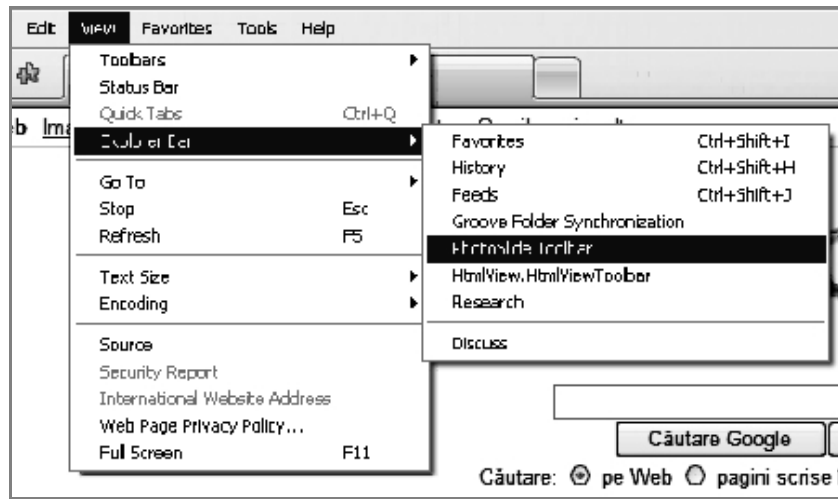


Fig. 33 - Activarea plug-in-ului

PhotoSlide Toolbar dispune de mai multe butoane de control, respectiv ferestre în care afișează informația:

- Fereastra de text pentru afișarea paginii web curente
- Fereastra de text de tip Checked List Box pentru afișarea imaginilor găsite în cadrul paginii web navigate. În cadrul acesteia se face și selecția imaginilor pentru prezentare.
- Fereastra de text pentru afișarea imaginilor pentru prezentare. Odată selectate în fereastra anterioară, imaginile sunt afișate în lista celor ce compun efectiv diaporama.

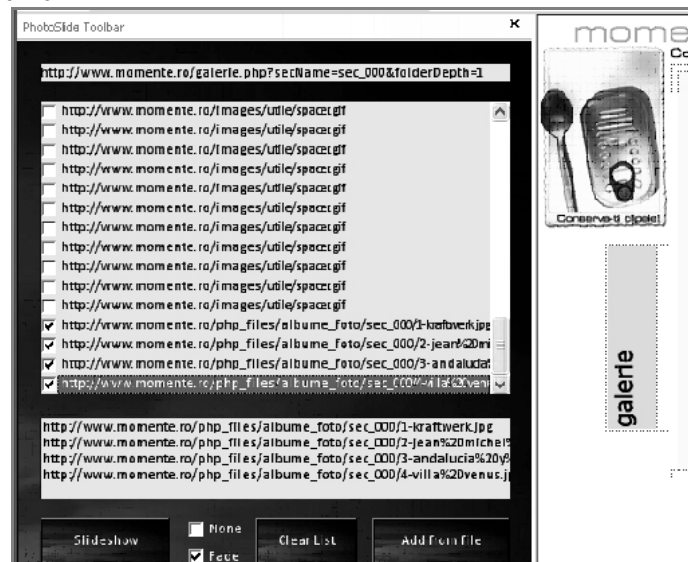


Fig. 34 - Cele trei ferestre principale ale aplicației

Putem observa în figura 32 adresa web afișată în prima fereastră, fereastra de selecție a imaginilor, în care sunt afișate toate imaginile pe care aplicația le identifică în pagina web navigată, precum și fereastra ce afișează doar cele 4 imagini selectate anterior, pentru a realiza prezentarea digitală.

Butoanele de control ale aplicației, ce pot fi identificate în figura de mai jos, se încadrează în mai multe categorii, în funcție de ce anume se poate controla prin intermediul lor:



Fig. 35 - Butoanele de control disponibile în fereastra browser

Elemente de control pentru construirea prezentării digitale:

- Buton Slideshow
- Buton SaveSlideshow
- Buton ClearList
- Buton AddFromFile
- Buton OpenSlideshow
- Check box Selecție Tip Tranzitie
- Numeric Up &Down Transition Time
- Numeric Up &Down Step size
- Numeric Up &Down Image Time

Elemente de control pentru manevrarea Winamp:

- Buton Back
- Buton Play
- Buton Pause
- Buton Stop
- Buton Next
- Buton 5 sec back
- Buton 5 sec forward
- ScrollBar Winamp Volume

Buton pentru afișare informații despre aplicație: About**TextBox pentru afișarea fișierului audio în redare****TextBox pentru afișarea timpului de redare a respectivului fișier audio**

Voi detalia acțiunea declanșată de fiecare dintre butoanele de mai sus, sau a ferestrelor de text ce afișează informații utile, deși numele acestora sunt suficient de sugestive pentru majoritatea utilizatorilor neavizați.

Butonul SlideShow este cel care ne introduce în modul Full Screen odată ce am stabilit toți ceilalți parametri ai prezentării. Acesta pornește prezentarea efectiv, cu condiția ca în fereastra de imagini selectate să existe minim o imagine. Dacă în fereastra de imagini selectate nu se află niciun element, se afișează mesaj de eroare, așa cum se poate observa în figura 34.

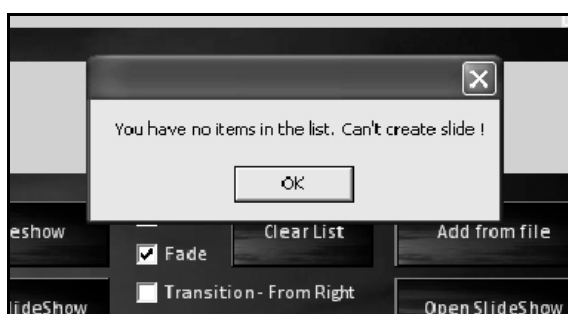


Fig. 36 - Mesaj de eroare afișat dacă se încearcă salvarea unei liste vide

Butonul SaveSlideShow ne permite să salvăm prezentarea realizată până în acel moment. La apăsarea acestui buton se scriu într-un fișier adresele tuturor imaginilor deja selectate în lista de imagini selectate. În cazul în care în fereastra pentru imagini selectate nu există nicio imagine, acesta afișează de asemenea un mesaj de eroare.

Butonul ClearList permite curățarea listei de imagini selectate, pentru a porni de la zero construcția unei noi prezentări.

Butonul AddFromFile permite adăugarea de imagini dintr-un fișier de pe disc. Acestea pot fi adăugate la o prezentare deja începută, nu există restricții în ceea ce privește sursa imaginilor. Pot fi incluse în aceeași prezentare atât imagini din cadrul paginilor web, cât și imagini din dosare de pe disc.

Butonul OpenSlideShow deschide o prezentare construită anterior, și salvată pe disc. Dat fiind faptul că la salvare se scriu într-un fișier adresele imaginilor din cadrul prezentării, cu toate că plug-in-ul poate fi folosit și în lipsa unei conexiuni la internet, dacă este vorba despre o prezentare cu imagini provenind de pe o pagină web, aceasta nu se va deschide.

Check box Selecție Tip Tranziție ne permite selectarea unui tip de tranziție dintre cele patru disponibile:

- None- fără tranziție (imaginile se schimbă pur și simplu, fără niciun efect la trecerea de la una la cealaltă- tranziție tip CUT)

- Fade este echivalentul Cross Dissolve din cadrul programelor de editare video. Imaginea anterioară este contopită în următoarea (opacitatea primei scade de la 100% la 0, în timp ce la cealaltă are loc procesul invers.) Acesta este modul tipic de tranziție utilizat în cadrul diaporamei analogice.

- Transition-FromRight –Imaginile se înlocuiesc una pe cealaltă din partea dreaptă (prima imagine se deplasează în stânga în timp ce este înlocuită de următoarea care vine din partea dreaptă a ecranului)

- Transition-FromTop- Imaginile se înlocuiesc una pe cealaltă din partea de sus a ecranului (prima imagine se deplasează în jos în timp ce este înlocuită de următoarea care vine din partea de sus a ecranului)

NumericUp&Down Transition Time permite selectarea timpului de tranziție între imagini. Acesta este condiționat de Image Time, cel care ne informează despre timpul de staționare a imaginii pe ecran. Durata tranziției nu poate fi mai mare decât timpul de staționare a imaginii pe ecran, întrucât ar fi absurd ca tranziția în sine să dureze mai mult decât imaginea, privitorul nu va avea timp să perceapă detaliile imaginii.

NumericUp&Down Step Size Step Size se referă la numărul de pixeli cu care avansează imaginea în cadrul tranzițiilor de tip: FromRight și FromTop. Acesta este controlul cu ajutorul căruia influențăm durata tranziției pentru aceste două tipuri de tranziție.

NumericUp&Down Image Time contolează durata în secunde a timpului de staționare a imaginii pe ecran. Așa cum arătam, nu se poate selecta un timp de tranziție mai mare decât timpul de afișare a imagii.

Butonul Back face parte dintre controalele pentru Winamp, și la apăsarea acestuia Winamp trece la fișierul audio anterior.

Butonul Play este cel care pornește Winamp, și abia la a doua apăsare acesta începe efectiv să redea melodiile adăugate în prealabil în playlist. Nu se poate lucra cu Winamp din punctul de vedere al realizării unui playlist, acesta trebuie conceput dinainte. Aplicația nu editează fișiere audio; din fericire însă, există suficiente aplicații care sunt deja consacrate editării audio de calitate.

Butonul Pause oprește redarea Winamp, are același efect ca și butonul de pauză al acestuia. Odată oprită redarea, aceasta poate reîncepe fie prin apăsarea acestuia încă o dată, fie prin apăsarea butonului Play. Redarea se reia din punctul în care s-a oprit.

Butonul Stop oprește redarea melodiei în curs. Winamp poate reîncepe redarea doar prin apăsarea butonului Play. Redarea se reia de la începutul melodiei în cadrul căreia s-a acționat butonul Stop.

Butonul Next trece la următoarea melodie din Winamp.

Butoanele 5sec back și 5sec forward, odată ce sunt acționate fac Winamp să decaleze coloana sonoră cu 5 secunde în față respectiv în spate. Dacă melodia este la început și se acționează 5 sec back, aceasta trece la melodia anterioară. Similar, dacă aceasta este la sfârșit și se apasă 5sec forward, se trece la melodia următoare.

ScrollBar Winamp Volume permite controlul volumului Winamp. S-a încercat adăugarea de cât mai multe controale pentru ca utilizatorul să nu fie nevoit să întrerupă aplicația pentru ajustarea prezentării.

În afara coloanei sonore care trebuie realizată a priori, eventual cu ajutorul unui editor audio (Adobe Audition sau Audacity- un software gratuit), sau prelucrarea imaginilor care se realizează deasemenea a priori cu ajutorul programelor de editare de imagini (Adobe Photoshop este unul dintre cele mai cunoscute, iar ACDSsee e unul din cele mai rentabile) utilizatorul nu trebuie să simtă nevoia utilizării unei alte aplicații pentru realizarea prezentării sale.

Butonul About este pentru afișarea informațiilor despre aplicație. **TextBox**-uri pentru afișarea melodiei în redare în cadrul Winamp, și deasemenea pentru afișarea timpului de când aceasta se afla în starea Play.

O particularitate importantă a aplicației, nespecificată anterior, este fereastra acesteia de Preview.

Atunci când se face click pe una dintre imaginile din lista de imagini detectate de plug-in în cadrul paginii web, aceasta se deschide într-o fereastră Preview așa cum se poate observa în Fig.37. Această fereastră are o deosebită importanță, întrucât în cadrul aplicației imaginile identificate în cadrul unei pagini web sunt adeseori elemente de grafică ale paginii web, nu toate dintre ele merită încadrate într-o prezentare, și de aceea este foarte important pentru utilizator să vadă imaginea pe care urmează să o adauge în lista celor selectate.

Fereastra de Preview dispăre dacă se face simplu click pe ea, sau dacă se selectează o altă imagine; aceasta se încarcă automat. Deasemenea, aceasta se poate ajusta în funcție de dimensiunea imaginii vizualizate.



Fig. 37 - Printscreen al ferestrei de previzualizare

5.3.2 Descrierea interfeței cu utilizatorul. Fereastra FullScreen

Aplicația în Full Screen arată ca în Fig 38. Trebuie precizat faptul că aplicația se află în pauză la trecerea de tip Fade dintre două imagini.

Am ales fundalul negru pentru că acesta este cel mai adesea utilizat în cadrul prezentărilor de acest tip (negrul se poate armoniza cromatic cu orice), și motivând ceea ce timpul a demonstrat, acesta este tipul de fundal ce se potrivește cu absolut orice imagine prezentată [77].

Am realizat o prezentare de imagini selectate de pe pagina personală web www.momente.ro, o galerie de fotografii on-line. Observăm cele două meniuri de control adăugate pentru această stare: Image, respectiv Winamp.



Fig. 38 - Interfața FullScreen a plug-in-ului

Cele două meniuri nu sunt vizibile decât dacă utilizatorul duce cursorul mouse-ului în zona respectivă de ecran.

Pentru lucrul cu imaginile, și implicit al prezentării în sine, am adăugat următoarele elemente de control în această stare a aplicației:

- **Back** – la acționarea acestuia se revine la imaginea anterioară.
- **Play** – dacă aplicația se află în pauză la acționarea butonului play aceasta își reia cursul. Important de precizat este faptul că atunci când se selectează SlideShow și se deschide FullScreen prezentarea este în desfășurare, nu trebuie acționat butonul Play.
- **Next** – trece la următoarea imagine din listă (practice se trece peste timpul selectat în fereastra de browser)
- **Exit** – iese din FullScreen și revine la fereastra de browser
- **Time Transition** permite influențarea timpului tranziției dintre imagini în FullScreen, dacă se dorește modificarea acestuia ulterioară afișării prezentării.

În cadrul controalelor pentru Winamp în FullScreen au fost adăugate toate cele din fereastra de browser, și evident cu aceeași denumire pentru o mai bună manevrare. Este aproape evidentă dorința utilizatorului de a controla coloana sonoră în timpul prezentării, și din acest motiv am adăugat și aceste controale de Winamp, pentru a nu fi necesară întreruperea prezentării dacă, de exemplu, este necesară creșterea volumului în Winamp.

Nu voi mai detalia acțiunea fiecărui element de control din această categorie, întrucât am precizat anterior influența acestora.

În cadrul ferestrei FullScreen am adăugat controale asociate acțiunilor pe care utilizatorul le poate întreprinde cu ajutorul tastelor sau mouse-ului:

- **La acționarea tastei Space** – se oprește prezentarea în punctul în care se află, așa cum se poate observa și în figura anterioară, iar la o nouă acționare prezentarea se reia din punctul în care se află. Dacă ne aflăm cu mouse-ul în zona

meniurilor asociate controalelor imaginilor și Winamp, apăsarea tastei Space face aplicația să iasă din mediul FullScreen și să revină la fereastra browser.

- **La acționarea tastei Escape**- aplicația iese din mediul FullScreen și revine la fereastra browser.

- **Pentru click stânga al mouse-ului** se trece la următoarea imagine, fără a se mai aștepta timpul de imagine selectat în fereastra browser.

- **Scroll înainte și înapoi al roțiței mouse-ului**- se trece la următoarea imagine, sau la imaginea anterioară, deasemenea fără a se mai aștepta timpul de imagine selectat în fereastra browser.

- **Apăsarea unei taste numerice de la 1 la 9** modifică timpul de tranziție în secunde după valoarea care s-a introdus prin tastatura.

Numărul relativ mare al controalelor introduse atât în fereastra browser cât și în cadrul desfășurării prezentării are ca scop ușurarea utilizării aplicației atât în crearea de prezentări electronice, cât și în redarea acestora. S-a urmărit oferirea pentru utilizator a posibilității de a influența desfășurarea prezentării chiar în timpul acesteia, întocmai cum era posibil în varianta analogică a spectacolelor de diaporame.

Posibilitatea intervenției umane este benefică, ea eliminând automatismul unei prezentări și oferind noi valențe artistice și creative prezentării audio-vizuale.

5.4 Implementarea aplicației

În cadrul acestui capitol voi face o mică descriere a metodelor utilizate în cadrul implementării aplicației. Așa cum am precizat anterior, am utilizat Visual Studio 2008 ca mediu de programare, C# ca limbaj de programare și SpicIE ca API ("Application Programming Interface"). Caracteristicile acestora au fost prezentate anterior în cadrul capitolului asociat descrierii uneltelor software de care am dispus în cadrul dezvoltării plug-in-ului "PhotoSlide Toolbar".

Voi descrie câteva aspecte legate de implementare. În cadrul aplicației am folosit lucrul cu controale de tip timer. Controlul Timer poate executa cod la intervale regulate cauzând apariția unui eveniment Timer. Intervalele de timp la care va apărea evenimentul Timer sunt controlate prin proprietatea *Interval* (specifică timpul în milisecunde). Implicit, valoarea luată de proprietatea *Interval* este zero, în acest caz controlul nefiind activ. Odată ce se va seta o valoare mai mare decât zero controlul va deveni activ, generând periodic (pană la dezactivarea acestuia, sau pană la închiderea ferestrei care îl conține) evenimente Timer.

Pentru rezolvarea acestei cerințe, se pot utiliza două metode:

- utilizarea proprietății *Enabled* a controlului Timer, care poate lua valorile *True* sau *False*.

- apelarea metodelor *Start* și *Stop* ale controlului Timer.

Voi începe prin a preciza metoda de implementare a tranziției de tip Fade, pentru că pe ea se bazează efectiv aplicația. Tranziția de tip Fade [78] a fost realizată lucrând cu opacitățile celor două imagini: *forma_slide* și *forma_slide1* fiind cele două imagini consecutive între care se face tranziția. În funcție de timpul selectat pentru tranziție, opacitatea primei imagini scade de la 1 (1 reprezintă 100%) la 0.1 (acesta fiind o valoare suficient de apropiată de 0), în timp ce opacitatea celeilalte imagini va crește de la 0.1 la 1.

Codul care implementează această tranziție este următorul:

```

private void timer_effect_Tick(object sender, EventArgs e)
{
    if (continua_slide == 1) // nu e apasată pauza sau exit sau..slide
show în derulare
    {
        opac += 10;
        if (opac > 100)
        {
            opac = 1;
            timer_effect.Stop();
            return;
        }
        if (ordine == 1)
        {
            forma_slide.Opacity = (double)opac / (double)100;
            forma_slide1.Opacity = (double)1 -
(double)forma_slide.Opacity;
        }
        else if (ordine == -1)
        {
            forma_slide1.Opacity = (double)opac / (double)100;
            forma_slide.Opacity = (double)0.9 -
(double)forma_slide1.Opacity;
        }
    }
}

```

Se poate observa lucrul cu cele 2 imagini mai sus amintite și deasemenea cu opacitățile lor și timer-ul asociat.

În cadrul declarațiilor de clase utilizate vom preciza clasa de bază utilizată pentru dezvoltarea acestuia, a cărei declarație este următoarea:

```

using SpicIE;
pentru lucrul cu fișiere s-a declarat:
using System.IO;

```

În timp ce pentru Winamp am folosit clasa:

```

using WinampFrontEndLib;

```

anterior dezvoltată pentru aplicația "Digital Diaporama".

Am folosit variabile pentru identificarea modurilor de lucru (mediul browser și mediul FullScreen), pentru calculul coordonatelor în raport cu ecranul pentru tranzițiile de tip Top respectiv Right, respectiv calculul pasului tranziției pentru aceleași tranziții menționate.

În cadrul controalelor asociate aplicației, dar nu numai, s-a utilizat metoda tratării evenimentelor:

```
private void PhotoSlideToolbar_Load(object sender, EventArgs e)
{
    this.forma_preview.Resize += new
System.EventHandler(mareste_fereastra);
    this.forma_preview.MouseClick += new
System.Windows.Forms.MouseEventHandler(mouse_apasat_preview);
    this.p1.MouseClick += new
System.Windows.Forms.MouseEventHandler(mouse_apasat_preview);
    this.control_pauza.Click += new
System.EventHandler(apasa_pauza);
    this.control_inapoi.Click += new
System.EventHandler(apasa_inapoi);
    this.control_urmator.Click += new
System.EventHandler(apasa_urmator);
    this.control_iesire.Click += new System.EventHandler(apasa_iesire);
    ...
}
```

Voi prezenta un exemplu de configurare a butoanelor, cum s-a setat mărimea, numele sau locația pentru fiecare dintre ele:

```
//
// control_urmator
//
control_urmator.BackColor = System.Drawing.Color.Transparent;
control_urmator.Location = new System.Drawing.Point(164, 12);
control_urmator.Name = "control_urmator";
control_urmator.Size = new System.Drawing.Size(58, 34);
control_urmator.TabIndex = 4;
control_urmator.Text = "Next >>";
control_urmator.FlatStyle = FlatStyle.Popup;
control_urmator.UseVisualStyleBackColor = false;
```

Dupa creare și configurare, butoanele au fost puse în forma plug-in-ului.

Așa cum arătam mai devreme, în dezvoltarea aplicației am utilizat `timer`. Unul dintre ele verifică o dată la fiecare jumătate de secundă starea Winamp, pentru a ști dacă este nevoie de afișarea titlului melodiei, respectiv a timpului scurs din redare.

Codul funcției care primește ca parametru adresa paginii web și preia din codul sursă al acesteia în funcție de forma standard de inserare a imaginilor în cadrul paginilor web (``) adresa sursă a imaginilor, este redat mai jos:

```
private string GetIEDocContents(string Url)
{
    SHDocVw.ShellWindows objWins = new SHDocVw.ShellWindows();
    mshtml.HTMLDocument objDoc;
```

```

string strContents="";
object refIndex = 1;

foreach(SHDocVw.InternetExplorer objIE in objWins)
{
    if(objIE.LocationURL==Url)
    {
        objDoc = (mshtml.HTMLDocument) objIE.Document;
        strContents= objDoc.body.parentElement.innerHTML;
        if(strContents.IndexOf("<FRAME">)=0)
        {
            mshtml.FramesCollection      objFrames      =
(mshtml.FramesCollection) objDoc.frames;
            mshtml.IHTMLWindow2          objFrame        =
(mshtml.IHTMLWindow2)objFrames.item(ref refIndex);
strContents=objFrame.document.body.parentElement.innerHTML;
        }

        mshtml.IHTMLControlRange  body2 = (mshtml.IHTMLControlRange)
objDoc.body;
        mshtml.IHTMLControlRange  controlRange =
(mshtml.IHTMLControlRange) body2.createControlRange();
        mshtml.IHTMLCollection objEiCol;
        objEiCol = objDoc.getElementsByTagName("IMG");
        //MessageBox.Show("pana aici !");
        checkedListBox1.Items.Clear();
        foreach(mshtml.HTMLImage objImg in objEiCol)
        {
            // MessageBox.Show(objEiCol.ToString());

            checkedListBox1.Items.Add(objImg.getAttribute("src", 1));
        }

        break;
    }
}

return strContents;
}

```

Această funcție identifică practic imaginile din cadrul paginii web, și permite afișarea acestora în fereastra de imagini astfel încât se poate lucra cu ele ulterior.

Funcția ce afișează adresa de URL la care se află browserul la momentul curent se apelează la fiecare schimbare a adresei de Internet. Aceasta este și cea care apelează funcția redată anterior, și umple apoi lista cu adresele imaginilor găsite.

Un exemplu de implementare a unuia dintre controalele din forma de control a imaginii (respectiv cel de trecere la următoarea imagine) din FullScreen este redat mai jos:

```
private void apasa_urmator(object sender, EventArgs e)
{
    if (continua_slide == 1)
    {
        timp_slide.Interval = 1;
        timp_slide.Start();
    }
}
```

Pentru ieșirea din slide-show, mai exact ca această ieșire să nu pară bruscă, este nevoie de setarea opacității pozelor la minim înainte de a le ascunde:

```
private void apasa_iesire(object sender, EventArgs e)
{
    continua_slide = 0;
    forma_controale.Hide();
    contor = 0;
    opac = 1;
    forma_slide.Opacity = 0.001;
    forma_slide1.Opacity = 0.001;
    forma_slide.Hide();
    forma_slide1.Hide();
    fundal_negru.Hide();
}
```

Funcția ce realizează slide-show-ul efectiv verifică inițial dacă sunt imagini în fereastra imaginilor selectate, dacă în fereastră nu este nicio imagine este afișat un mesaj de eroare, în caz contrar se face slide-show în funcție de tipul tranziției selectat (tip_efect este testat pentru a determina ce tip de tranziție a fost selectată în fereastra browser):

```
private void button1_Click(object sender, EventArgs e)
{
    if (listBox1.Items.Count == 0)
    {
        MessageBox.Show("You have no items in the list. Can't create slide!");
    }
    else
    {
        count++;

        continua_slide = 1;
        if (count == 1)
        {
            forma_slide.Controls.Add(poza_slide);
            forma_slide1.Controls.Add(poza_slide1);
        }
    }
}
```

```
}

forma_preview.Hide();
poza_slide.Size = forma_slide.Size;
poza_slide1.Size = forma_slide1.Size;
fundal_negru.Show();

forma_slide1.Show();

ordine = -1;
forma_slide.Show();
//forma_slide.TopMost = true;
contor = -1;
numar_elemente_lista = listBox1.Items.Count;
timp_slide.Interval = 1;

pas_dreapta = SystemInformation.PrimaryMonitorSize.Width / 10;
pas_jos = SystemInformation.PrimaryMonitorSize.Height / 10;

if (chk_Fade.Checked)
{
    tip_efect = 1;
    opac = 1;
    timp_fade.Start();
}
if (chk_Dreapta.Checked)
{
    tip_efect = 2;
    timer_dreapta.Start();
}
if (chk_Jos.Checked)
{
    tip_efect = 3;
}

if (tip_efect == 1)
{
    forma_slide.Opacity = 0.01;
    forma_slide1.Opacity = 0.01;
}

if (tip_efect == 2 || tip_efect == 3)
{
    forma_slide.Opacity = 1;
    forma_slide1.Opacity = 1;
}

timp_slide.Start();
//timer_effect.Interval = interval / 100;
```



```

        //timer_effect.Start();
    }
}

```

Funcția ce realizează afișarea imaginilor în fereastra de Preview atunci când se face click stânga pe imagine este următoarea:

```

private void checkedListBox1_SelectedIndexChanged(object sender,
EventArgs e)
{
    p1.ImageLocation = checkedListBox1.SelectedItem.ToString();

    p1.Size = forma_preview.Size;
    forma_preview.BackColor = Color.Black;
    forma_preview.TopMost = true;
    forma_preview.Text = "Preview - " +
checkedListBox1.SelectedItem.ToString();

    forma_preview.Show();
}

```

Se observă că este afișat deasemenea și titlul acesteia.

Un exemplu de funcție ce tratează lucrul cu Winamp (cea care deschide Winamp atunci când se apasă Play, sau la o a doua apăsare redă melodia în curs) :

```

private void button5_Click(object sender, EventArgs e)
{
    System.Diagnostics.Process.Start("winamp.exe");
    WinampLib.Play();
}

```

Funcția ce afișază titlul melodiei, respectiv timpul de când aceasta se află în play:

```

private void timer_winamp_Tick(object sender, EventArgs e)
{
    lbl_numelodie.Text = WinampLib.GetCurrentSongTitle()+
".ToString();
    timp_melodie.Text =
WinampTrackPosition(WinampLib.GetTrackPosition());
}

```

5.5 Contribuții aduse prin aplicația Photoslide Toolbar

În urma testelor, și a comparației cu aplicația ce a făcut obiectul primei etape de studiu, au rezultat avantaje deosebite. Tranzițiile se desfășoară impecabil, fără

intermitențe, iar diferențele între durata solicitată și cea reală a tranzițiilor au dispărut. Calitatea tranzițiilor se păstrează și în cazul imaginilor de dimensiuni mari. Coloana sonoră este redată, întocmai ca la aplicația precedentă, prin intermediul Winamp.

Faptul că aplicația poate suporta imagini de dimensiuni de până la full screen, iar tranziția lor se desfășoară perfect fără să țină cont de mărimea spațiului ocupat, reprezintă un mare avantaj, în comparație cu precedenta aplicație. Pe de altă parte, este necesară redimensionarea imaginii dacă aceasta depășește dimensiunea full screen, de regulă 1280/960 pixeli, la 72 dpi. În caz contrar, pe display va apărea doar o parte a imaginii, respectiv zona sa centrală.

Aplicația permite și tranziția dintre imagini aflate pe Internet, însă e foarte important ca utilizatorul să elimine din lista de imagini destinate prezentării de pe o anumită pagină web, pe acelea care nu fac parte efectiv din galerie ci sunt elemente vizuale auxiliare ale paginii, elementul lor comun cu fotografiile fiind doar extensia .jpg . În caz contrar, prezentarea va conține, pe lângă fotografiile aflate într-o anumită galerie on-line, și alte imagini, de exemplu stegulețul aferent limbii de comunicare, sigla de pe pagină, bannere, etc.

Principalul dezavantaj al aplicației nu e dat de neconcordanța dintre cerințe și așteptări, ci de instalare. Aceasta funcționează perfect pe orice computer doar dacă se instalează în paralel și anumite versiuni ale altor aplicații, respectiv „Internet Explorer 7” (nu versiuni mai vechi), „SpiceIE” versiunea 8.1 și „DotNetFx35”.

Situația cea mai defavorabilă e dată de necesitatea de a instala, în anumite cazuri de configurare a computerului, versiunea integrală Visual Studio 2008. Dezavantajele sunt date atât de volumul foarte mare ocupat (kit-ul ocupă 3387 MB) cât și de un neajuns supărător al acestei aplicații. Există situații în care apar conflicte între Visual Studio 2008 și alte aplicații, care duc la întreruperea funcționării celor din urmă, aceasta în situația în care aplicația Internet Explorer și implicit plug-in-ul "PhotoSlide Toolbar" nu sunt pornite.

Feedback-ul obținut de la evaluatori ai IADIS International Conference e-Society 2010 este revelator [79]:

Puncte pozitive:

- Integrarea programului pare a fi de succes. Controlul în timp real va oferi utilizatorilor posibilitatea de depăși multe probleme specifice.
- Ideea de a accentua paradigme analogice de succes, cum ar fi Diaporama,
- în era digitală este valoroasă, deoarece subliniază că nu este întotdeauna necesar să se "reinventeze roata", ci, mai degrabă, este important de a învăța din trecut pentru a proiecta un viitor mai bun.
- Descrierea tehnică a Barei de instrumente PhotoSlide este detaliată și exactă
- Soluția propusă aduce o noutate, și pare originală.

Puncte negative:

- Tehnologiile alese nu par a fi destul de adecvate. Ar trebui să ia în considerare utilizarea de software și instrumente open source.
- Tehnologiile folosite sunt vechi, și inconvenabile în termeni de desfășurare.

6. Elaborarea și implementarea celui de-al treilea model de aplicație. “Motorway”

6.1 Necesitate

Așa cum am arătat în precedentele etape ale studiului, am reușit să realizez ce mi-am propus; o aplicație care să permită controlul manual al tranzițiilor dintre imagini, în cadrul unui spectacol de diaporamă. Obiectivul propus a fost atins, în două etape de studiu. În prima etapă am reușit să pun la punct o aplicație ce permite controlul cu ajutorul tastaturii și al mouse-ului [54], iar în a doua etapă, doar cu tastatura, însă cu rezultate mai bune [79].

Neajunsurile apărute trebuie însă luate în seamă și în ideea unui studiu extins, eliminate. Tranzițiile efectuate de prima dintre aplicații sunt sacadate și mult mai lente decât ar trebui. Aplicația realizează însă un calcul în timp real, iar asta consumă mult timp de lucru. Se aplică foarte multe formule și calcule, se folosește o putere de calcul mare. Problemele vizibile apar atunci când se solicită tranziții scurte, de o secundă. Uneori, calculul durează mai mult de o secundă, ceea ce duce la întârzieri foarte mari în răspuns.

Și a doua aplicație, plug-in-ul Photoslide Toolbar, realizează un calcul în timp real. Optimizarea sa a redus vizibil, întârzierile. Principalul neajuns e dat de faptul că instalarea aplicației necesită mult prea multe resurse. Pentru C# am avut nevoie de un mediu de dezvoltare Visual Studio 2008, care știe să folosească C#. Odată aplicația finalizată, a fost compilată și a generat un pachet de fișiere. Problemele apar la portarea aplicației pe un computer, din care e posibil să lipsească anumite elemente. Aceasta se datorează faptului că la compilarea în C# nu s-a impus ca aplicația să conțină toate resursele (ceea ce i-ar fi mărit substanțial volumul).

Dincolo de aceste neajunsuri, iată câteva părți pozitive ale studiului de până în acest moment, care merită preluate în extinderea studiului: ideea de a concepe o aplicație care să suplinească unele neajunsuri ale aplicațiilor de diaporamă consacrate, concretizarea acestei idei în două aplicații ce sunt diferite din punct de vedere al construcției dar asemănătoare ca rezultat, folosirea tastaturii ca element de control în timp real al tranziției imaginilor, și respectiv a mouse-ului pentru a înlocui dispozitivul analogic de control manual al tranziției.

În acest context, și ținând cont de experiența acumulată la primele etape ale studiului, necesitatea unei noi aplicații - care să preia părțile utile ale primelor etape de studiu, și în egală măsură, să evite neajunsurile menționate anterior - devine evidentă.

Denumirea “Motorway” a fost inspirată de diaporama omonimă realizată de britanicii Peter Coles și Colin Balls. Este un tribut adresat unei lucrări pe care o consider un reper major în lumea diaporamei artistice analogice [80].

6.2 Prezentarea aplicației “Motorway”

Aplicația poate fi controlată atât prin tastatură cât și prin intermediul mouse-ului. Reprezintă echivalentul digital al dispozitivelor analogice Kodak PDC [81] –

comanda prin tastatură- respectiv Simda [82] și/sau Imatronic SX2000 [83], pentru comanda cu mouse-ul.

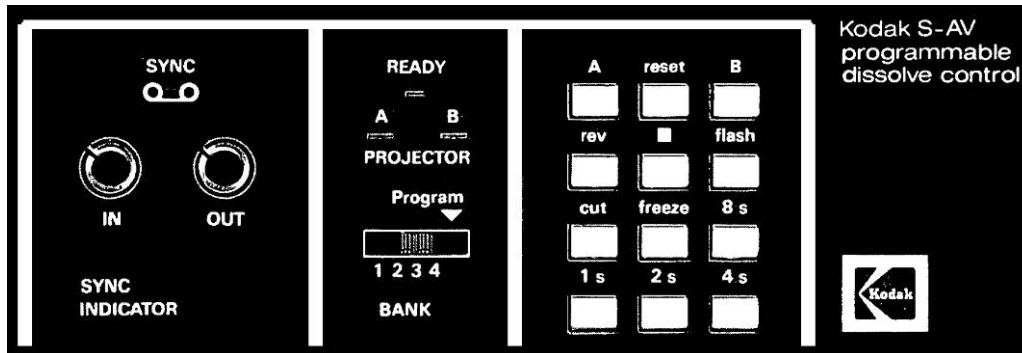


Fig. 39 - Modulul de comandă analogică KODAK S-AV PDC

Combinând experiența acumulată cu studiul precedent, și ținând cont de modul de lucru al dispozitivelor analogice Kodak PDC, Simda și Imatronic, următoarele rezultate au fost așteptate: o aplicație capabilă să controleze în timp real atât tranziția de tip „cross/dissolve” între două imagini, cât și redarea și controlul unui fișier audio.

Instalarea se realizează copiind pe hard-disk directorul aplicației. Acesta conține fișierul „index.html” și respectiv directorul „app”. La rândul său, „app” e compus din următoarele directoare: „audio”, „css_files”, „images”, „js_files” și „slides”, precum și fișierul „slideshow.htm”.

Coloana sonoră dedicată se copiază în directorul „audio”, iar imaginile ce compun diaporama, în directorul „slides”. În cazul în care coloana sonoră nu este compusă dintr-o singură piesă, așa cum se consideră a fi profesional, ci din mai multe, ele vor fi citite în ordine alfabetică. Imaginile încărcate în directorul „slides” vor fi și ele citite în ordine alfabetică, de aceea este foarte importantă denumirea și/sau numerotarea lor. Ordinea viitoareii lor apariții pe ecran este dată de numerotarea / denumirea adecvată.

Aplicația are două ferestre. Se începe prin deschiderea „ index.html”, în mod obligatoriu cu browserul Internet Explorer [84].

6.2.1 Fereastra browser-ului Internet Explorer

Deasupra paginii va apare textul următor: “To help protect your security, Internet Explorer has restricted this webpage from running scripts or ActiveX controls that could access your computer. Click here for options... ”

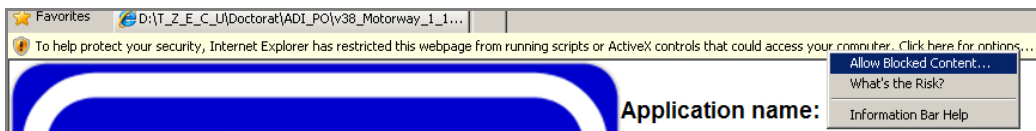


Fig. 40 - Pentru deschiderea aplicației "Motorway" trebuie acceptat conținutul blocat

Alegem „Allow Blocked Content...”, iar dacă se cere, confirmăm opțiunea [85]. Fereastra browser-ului se deschide și este funcțională.

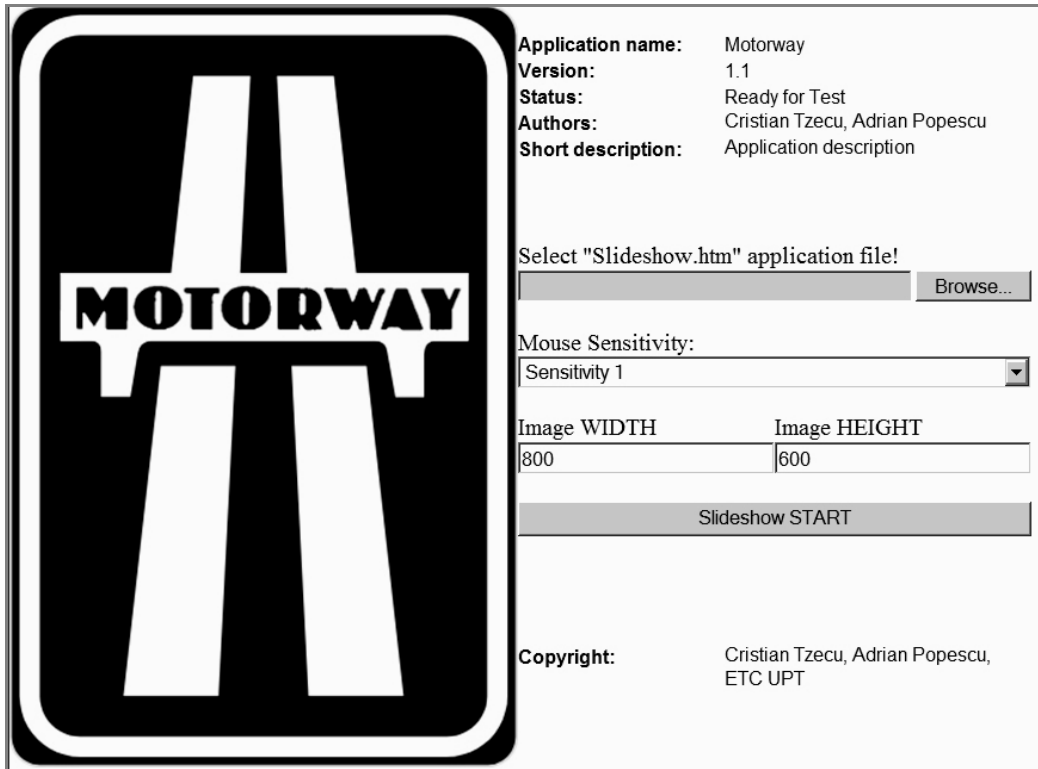


Fig. 41 - Fereastra browser-ului aplicației "Motorway"

Aceasta oferă posibilitatea unui reglaj al sensibilității mouse-ului și al dimensiunii de afișare a imaginilor, precum și selectarea fișierului "Slideshow.htm" din directorul „app”.

Odată selectat acest fișier, se apasă butonul „Slideshow START”, și intrăm în fereastra de diaporamă. Întocmai ca la intrarea în fereastra browser-ului, deasupra paginii va apare textul următor: "To help protect your security, Internet Explorer has restricted this webpage from running scripts or ActiveX controls that could access your computer. Click here for options... "

Alegem „Allow Blocked Content...”, iar dacă se cere, confirmăm opțiunea.

O ultimă restricție e dată de necesitatea de a aproba interacțiunea. Apăsăm „Yes”. Fereastra de diaporamă se deschide și este funcțională.

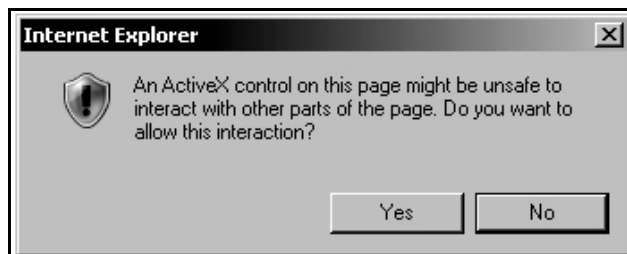


Fig. 42 - Toate interacțiunile trebuie aprobate

6.2.2 Fereastra de diaporamă

Fereastra prezintă în fază inițială prima imagine din directorul „slides”. Deoarece la prezentările analogice de diaporamă, prima imagine era obligatoriu un diapozitiv negru, recomand același lucru și în cazul digital. În acest fel următoarea imagine va apare din negru, similar efectului de apariție tip „fade in”. Vizual, e de preferat acest lucru, față de o apariție bruscă a imaginii.

Cu excepția imaginilor alese pentru prezentare, fereastra de diaporamă conține un obiect în partea dreaptă a ecranului, sub formă de dreptunghi transparent, cu margine gri, având dimensiunea de 75/22 pixeli. Aceasta este bara de control a mouse-ului.

6.2.3 Comenzile spectacolului de diaporamă

Cu ajutorul tastaturii și/sau a mouse-ului, spectacolul de diaporamă poate fi derulat și controlat în totalitate.

Comenzile din tastatură sunt următoarele:

Ctrl + Q muzica pornește

Ctrl + Q din nou muzica se întrerupe, urmând ca la o nouă comandă similară, sa pornească din punctul de întrerupere.

Ctrl + > se trece la următoarea piesă, în ordine alfabetică, dacă în folderul „audio” sunt mai multe piese. Pentru pornirea acestei noi piese, se apasă **Ctrl + Q**

Ctrl + < se trece la precedenta piesă, în ordine alfabetică, dacă în folderul „audio” sunt mai multe piese. Pentru pornirea acesteia, se apasă **Ctrl + Q**

Esc – se reia ultima piesă de la început

6.2.3.1 Controlul cu ajutorul tastaturii numerice

1 – o secundă de tranziție

2 – două secunde de tranziție

3 – trei secunde de tranziție

.....

9 – nouă secunde de tranziție, cea mai lungă tranziție controlabilă de tastatură

0 – „cut”, zero secunde de tranziție.

Space – întrerupe tranziția controlată de tastatură (pause). Apăsând din nou, reia tranziția întreruptă

~-“flash”, odată apăsând, proiectează 50% din ambele imagini. După eliberarea tastei, tranziția continuă

Săgeata stânga < - inversează ordinea de proiectie (alfabetică) a imaginilor. Permite astfel întoarcerea la o imagine ce a fost deja proiectată.

Săgeata dreapta -> revine la ordinea de proiectie (alfabetică) a imaginilor, implicită.

Ctrl + S – ascunde/arată bara de control a mouse-ului, situată în partea dreaptă a ecranului.

Ctrl+1, Ctrl+2, Ctrl+3 controlează distanța pe care mouse-ul o parcurge pentru ca o tranziție să se desfășoare integral. Ctrl+1 indică o cursă scurtă a mouse-ului, pentru tranziții ce nu necesită acuratețe, iar Ctrl+3 indicată cea mai lungă cursă a mouse-ului, pentru tranziții de mare acuratețe.

Comenzile realizate cu ajutorul tastaturii simulează comenzile analogice realizate cu ajutorul sistemului Kodak S-AV Programmable Dissolve Control [81], Fig.39.

6.2.3.2 Controlul cu ajutorul mouse-ului

Pas 1: cursorul mouse-ului se plasează stânga pe bara de control a mouse-ului, situată în partea dreaptă a ecranului. Cu mouse-ul stânga apăsat se trage de el în jos. Are loc tranziția dintre prima și a doua imagine, viteza sa depinzând de mișcarea mouse-ului.

Pas 2: se eliberează butonul de mouse stânga după apariția completă a celei de-a doua imagini.

Pas 3: cu mouse-ul stânga pe bara de control se trage de el în sus. Are loc tranziția dintre a doua și a treia imagine, viteza sa depinzând de mișcarea mouse-ului.

Pas 4: se eliberează butonul de mouse stânga după apariția completă a celei de-a treia imagini.

Dacă bara de control a mouse-ului e în poziția superioară, se pot schimba setările de acuratețe ale mouse-ului (Ctrl + 1,2,3). Tot în această poziție bara de control a mouse-ului poate fi ascunsă (Ctrl+S).

Comenzile realizate cu ajutorul mouse-ului simulează comenzile analogice realizate cu ajutorul sistemelor Simda și Imatronic.



Fig. 43 - Controlul tranziției cu ajutorul mouse-ului

6.3 Mediile de dezvoltare

Am folosit JavaScript, Html și CSS (Cascading Style Sheets).

1.HTML este prescurtarea de la *Hyper Text Mark-up Language* și este codul care sta la baza paginilor web. Paginile HTML sunt formate din etichete sau tag-uri și au extensia *.html* sau *.htm*. În marea lor majoritate aceste etichete sunt pereche, una de deschidere `<eticheta>` și alta de închidere `</eticheta>`. Browserul interpretează aceste etichete afișând rezultatul pe ecran. HTML-ul nu este un limbaj case sensitive (nu face deosebirea între majuscule și minuscule) [17].

2.Javascript este un limbaj de programare care face posibil ca paginile web să fie mai interactive. Este tot mai des recunoscut ca făcând parte din categoria "Scripting Languages". A fost lansat sub numele de "LiveScript", schimbarea numelui în JavaScript fiind legată de strategia de marketing [86].

Scripturile Javascript sunt introduse în pagina HTML și sunt interpretate și executate de browser. Javascript poate fi considerat o unealtă foarte bună dacă se dorește controlul conținutului paginilor în funcție de dată, oră, sistem de operare sau browser. De asemenea, e util pentru realizarea de site-uri interactive, care să

comunica cu vizitatorii, site-uri dinamice, pentru validarea datelor primite din formulare. Acestea sunt doar câteva din posibilitățile pe care le oferă acest limbaj.

JavaScript dă dinamicitate, interactivitate cu pagina net [18], [87].

3.CSS, prescurtarea de la Cascading Style Sheets, sunt etichete folosite pentru formatarea paginilor web (de exemplu formatare text, background sau aranjare în pagină, etc.) [88].

Beneficiile sintaxei CSS sunt:

- formatarea este introdusă într-un singur loc pentru tot documentul
- editarea rapidă a etichetelor
- datorită introducerii într-un singur loc a etichetelor se obține o micșorare a codului paginii, implicit încărcarea mai rapidă a acesteia.

Sintaxa CSS este structurată pe trei nivele:

nivelul 1 fiind proprietățile etichetelor din documentul HTML, tip **inline**
nivelul 2 este informația introdusă în blocul HEAD, tip **embedded**
nivelul 3 este reprezentat de comenzile aflate în pagini separate, tip **externe** [89]

CSS, Cascading Style Sheets, e folosit pentru partea de finețe a afișării aplicației, controlul culorii, a grosimii imaginilor, pentru definirea parametrilor, a caracteristicilor obiectelor de imagine din Slideshow. CSS dă anumite proprietăți obiectelor dintr-o pagină web.

Într-o exprimare figurată HTML reprezintă învelișul sub care lucrează aplicația, CSS o face să fie atractivă, Javascript îi comandă modul de lucru, îi conferă partea de finețe.

6.4 Limitări și cerințe

Aplicația a fost concepută ținând cont de anumite **constrângeri** și cerințe.

În primul rând, am pus accent pe **portabilitate** – aplicația să poată fi folosită pe cât mai multe configurații ale calculatoarelor, fără să fie necesară instalarea unor aplicații adiționale.

O altă constrângere a fost dată de **procesarea tranziției** de tip „cross-dissolve”. Aceasta trebuia să fie diferită de cea folosită la aplicațiile anterioare. Procesarea tranziției de tip „cross-dissolve” se făcea printr-un calcul matematic, ce dura uneori mai mult de o secundă, chiar dacă se solicita o tranziție de exact o secundă [90].

Internet Explorer a fost singura opțiune pentru a soluționa această cerință. Internet Explorer are deja implementat mecanismul ce controlează funcția tranziției de tip „cross dissolve”. Deasemenea, Internet Explorer face parte din pachetul Windows, deci ar trebui să se găsească instalat pe aproape orice PC.

Un alt browser extrem de utilizat, Firefox, ar fi putut intra în calcul. Însă Internet Explorer și Firefox sunt produse ale unor firme concurente (Microsoft respectiv Mozilla) și au anumite neconcordanțe în ce privește limbajul folosit. În cadrul Firefox, de exemplu, există standarde diferite față de Internet Explorer. În mod normal, Firefox ar trebui să fie 100% compatibil cu Internet Explorer, însă această cerință nu e încă (primăvara lui 2010) îndeplinită. Cu siguranță că o versiune avansată a aplicației mele va beneficia de un selector care să sesizeze dacă aplicația e deschisă cu Internet Explorer sau Firefox. În acest al doilea caz, va trebui să existe o porțiune de cod adaptată la Firefox [91].

Internet Explorer și Firefox sunt două aplicații din aceeași clasă, respectiv browsere de internet, *address interpretor*. Aplicațiile precedente ale studiului meu nu au abordat corect cerința, deoarece calculul tranziției dintre două imagini

durează foarte mult, fiind foarte laborios. Practic se compunea imaginea, pixel cu pixel, fără a fi vorba de o setare a transparenței. Aplicațiile care lucau în C# făceau de fapt rendering, ceea ce ducea la întâzieri mari. Calculul nu se desfășura, de fapt, în timp real. Dacă am fi avut o putere de calcul mai mare, ar fi funcționat, însă în condiții uzuale, puterea de calcul e prea mică. Internet Explorer are deja mecanismele predefinite pentru tranziția de tip „cross-dissolve”. Ele sunt testate și funcționează cu succes; ceea ce am avut de făcut a fost să le facem să lucreze pentru aplicația Motorway. Nu a mai fost necesară inventarea unor algoritmi de procesare ai imaginii. Din fericire, există deja definite metode care se pot aplica pe diferite tipuri de obiecte.

O altă restricționare a apărut pe parcurs, lucrând cu Internet Explorer. În mod normal, partea de .html și Internet Explorer este folosită la aplicații on-line. Restricționarea a apărut prin faptul că am folosit **local** o aplicație (Internet Explorer) gândită pentru lucru online. Explorer, pe de altă parte, are mecanisme interne de securitate care limitează utilizatorul atunci când vrea să lucreze local. În mod normal, pentru a avea acces total pe un site, sunt necesare un nume de utilizator și o parolă. Acestea sunt informații care se verifică online, pe server, și nu local. Similar, din motive de securitate, pentru a nu avea acces la hard-disk-ul utilizatorului, sau la informații confidențiale, am fost limitat la gestiunea fișierelor. Au trebuit abordate strategii de regândire a modului în care se accesează lista de imagini și cea de fișiere audio. În mod normal, aceste fișiere nu ar trebui accesate decât pentru upload. Pe de altă parte, utilizatorul aplicației Motorway va lua resursele direct de pe calculator, local. Nu are nevoie de server.

Gestiunea fișierelor a reprezentat o altă constrângere. A trebuit creată o listă de imagini al căror număr nu îl cunoaștem. Pe server, există funcții specifice pentru acest lucru. Local însă, nu putem ști care e folderul rădăcină, datorită limitărilor. Putem ști doar care sunt drive-urile. Alegând fișierul Slideshow, îi definim implicit partiția și rădăcina. Aceasta e logica pentru a defini punctul de start al aplicației.

În mod normal, pe un server, pagina de start este "index.html". În situația noastră, startul trebuie definit manual. Utilizatorul are libertatea de a-și copia aplicația oriunde dorește; definindu-i startul, ea poate porni.

Acestea au fost limitările și constrângerile care au apărut în implementarea aplicației; în final, la baza ei este doar o pagină de internet. Acest fapt o face simplă și accesibilă.

6.5 Mecanisme interne ale aplicației

6.5.1 Mecanismul deschiderii aplicației

În partea de setare, se selectează Slideshow, apoi se alege rezoluția de afișare, după care sensibilitatea mouse-ului. Aplicația identifică imaginile și muzica, le numără, le cataloghează și le ordonează în ordine alfabetică.

Fișierul Slideshow se selectează pentru ca aplicația să știe unde sunt de fapt fișierele de imagine și sunet. A trebuit selectat un fișier .html, care să identifice unde e situat folderul rădăcină, unde se găsește partea ce conține imaginile și respectiv partea ce conține sunetul.

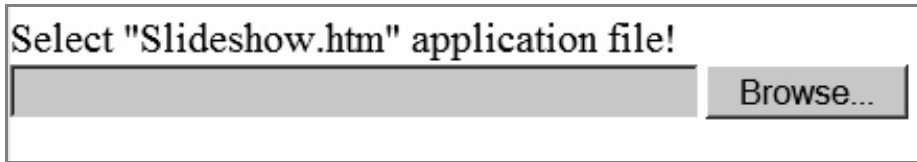


Fig. 44 - Se începe întotdeauna cu selectarea "Slideshow.htm" de pe hard-disk

În momentul instalării, imaginile și muzica se ordonează alfabetic. Trebuie să alegem rezoluția de afișare, indiferent de mărimea reală a imaginii. Utilizatorul trebuie să țină cont de faptul că aplicația nu redimensionează imaginile alese. Dacă rezoluția de afișare e mai mică decât dimensiunea originală a imaginii, doar o parte a imaginii va fi vizibilă, respectiv un decupaj centrat. Dimensiunea acestuia va fi egală cu dimensiunea rezoluției de afișare. Dacă rezoluția de afișare e mai mare sau egală cu dimensiunea originală a imaginii, aceasta va încăpea în întregime pe monitor (sau ecran).

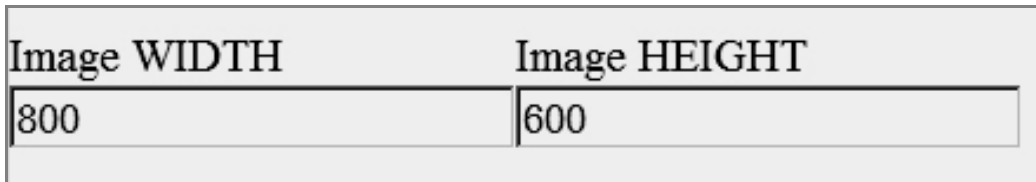


Fig. 45 - Se alege rezoluția de afișare a diaporamei

Dacă dimensiunile imaginii sunt disproporționate față de rezoluția de afișare, se va pierde proporția de afișare. Vor rezulta imagini proiectate ce vor fi prea înguste sau prea late față de cele originale. Rezoluția de afișare trebuie să nu depășească rezoluția display-ului, dar proporția trebuie păstrată. Dacă avem mai multe imagini, având aceeași proporție dar dimensiuni diferite, imaginea cea mică va fi mărită la rezoluția de afișare. Imaginea mare, însă, nu va fi micșorată, deoarece zona de afișare e cea care se întinde, nu imaginea.

6.5.2 Mecanismul tranziției

Internet Explorer este un interpretor al paginii; traduce în limbaj vizual (și uneori auditiv) o pagină de web. Internet Explorer știe să interpreteze JavaScript, precum și afișarea diverselor obiecte în pagină; poziție, layere (multistrat) cu diverse proprietăți de poziție, dimensiune, culoare, transparentă. Aplicația e concepută simplu: trebuie afișate într-o anumită manieră două imagini. În orice moment, transparența lor este și trebuie să fie complementară. Imaginile pot avea transparențele de 50%, sau una poate fi transparentă 100%, respectiv una opacă 100%. Aplicația e gândită să lucreze pe doar 2 straturi. Tranziția implică o creștere a opacității uneia dintre imagini și creșterea complementară a transparenței la cealaltă; apare efectul de "cross-dissolve". Simplitatea e dată de faptul că avem de controlat doar transparența și opacitatea. Dacă transparența imaginii e totală, 100%, și implicit opacitatea este 0%, acea imagine nu poate fi văzută. Dacă transparența e 0%, și implicit opacitatea e totală, 100%, imaginea apare în totalitate. Obiectul imagine are o proprietate de transparență *alpha* cuantificată la 100 de pași. Pentru pasul 0 imaginea e transparentă, pentru pasul 100, e opacă.

Avem două imagini, ca două obiecte. Ordinea de afișare e aceeași: imaginea 1 e în față, imaginea 2 e în spate. Când imaginea 1 e opacă, imaginea 2, fiind transparentă, nu se vede. În acel moment îi schimbăm sursa; îi spunem că imaginea 2, din spate, va fi înlocuită de altă imagine.

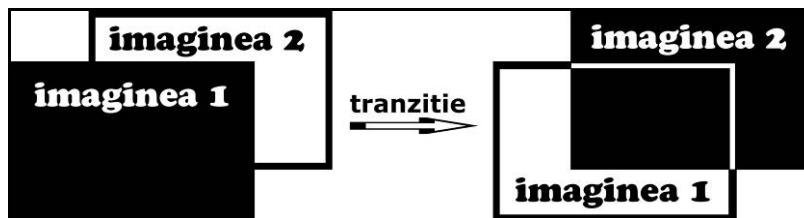


Fig. 46 -Mecanismul tranziției

6.5.3 Mecanismul audio

Internet Explorer are un mecanism intern care știe să ruleze fișiere audio, respectiv Windows Media Player. Avantajul este că nu mai e necesară pornirea în paralel a altei aplicații audio, așa cum am procedat la prima aplicație a acestui studiu. Se pot citi fișiere audio compresate în format .mp3 sau necompresate (wav). Teoretic, orice fișier audio poate fi citit de Media Player, dacă există decoderul instalat, pentru formate mai puțin obișnuite, gen .flac, .ape, .ogg [92].

6.6 Aspecte legate de programare

La realizarea codului, am ținut cont de principiul reutilizării. Astfel, nu vom mai scrie foarte multe linii de cod, când avem definite anumite funcții pe care le re folosim. Există o anumită „eleganță” ce trebuie luată în considerare atunci când se abordează scrierea codului unei aplicații. Să nu te repeți în scrierea codului ține de această calitate. Dacă programatorul descoperă un model, ce se folosește în anumite scopuri, acesta poate încerca să adapteze funcția respectivă pentru ca să o poată re folosii în cadrul aplicației sale de cât mai multe ori, pentru a reduce numărul liniilor de cod. Un alt aspect care ține de calitatea liniilor de cod e dat de comentariile pe marginea codului. Oricărui cunoscător care citește și analizează codul-sursă, ar trebui să îi fie foarte clar ce se întâmplă la un moment dat vis-a-vis de logica aplicată pentru porțiunea respectivă.

Codul în sine este vast, liniile de cod sunt numeroase. Abordate de la început la sfârșit, ele nu pot fi citite ca o carte. În mod normal, există o structură de bază, precum și niște metode aplicate pentru diverse acțiuni. Aspectul ramificat e cea mai la îndemână analogie. Logica în sine se construiește prin interconexiunea/comunicarea cu celelalte metode sau funcții. Un program în sine e compus dintr-o succesiune de comenzi, proprii limbajului ales. Ele sunt ordonate într-o succesiune logică și temporală, dar permit și apelarea „ în buclă”. În cazul nostru, apelăm de mai multe ori o funcție, și revenim la ea, vis-a-vis de ciclicitatea slideshow-ului. De exemplu, prezentarea ajunge la capătul listei de imagini, și o ia de la început.

Tranziția dintre două imagini care trebuie să dureze o secundă, folosește o anumită funcție specifică. Această funcție se folosește la toate celelalte cazuri de tranziții, însă adaptată la timpul de tranziție. Din motive menționate anterior, nu am

folosit funcții diferite pentru tranziții diferite. Dificultatea a constat în combinarea elementelor avute la dispoziție pentru realizarea prezentării de diaporamă.

Aplicația pentru diaporamă digitală Motorway pornește de la pagina index.html, apăsând „Slideshow Start”. Câmpurile sunt folosite pentru setarea inițială a variabilelor aplicației. Selectând fișierul „Slideshow.htm”, arătăm calea spre directorul rădăcină al aplicației.

6.6.1 Codul HTML pentru File Browser

```
<input type="file" class="file" id="FileBrowser" style="width: 100%">
```

6.6.2 Codul Javascript de construcție a căii

```
function slideshowStart()
{
    appPath = getPath("FileBrowser");

    if((appPath == "") || (appPath == null) || (appPath == "undefined"))
    {
        alert("Please select Slideshow.htm application file");
    }
    else
    {
        WriteToFile_APP_Path(appPath, sensitivity);
        window.open('app/slideshow.htm','Dialog',features);
    }
}
```

6.6.3 Codul HTML pentru acuratețea mouse-ului

```
<select name="Sensitivity" OnChange="setOption(this.selectedIndex)"
style="width: 100%">
    <option>Sensitivity 1
    <option>Sensitivity 2
    <option>Sensitivity 3
</select>
```

6.6.4 Codul HTML pentru rezoluția de afișare

```
<INPUT id="imgWidth" type="text" value="800" style="width: 100%">
<INPUT id="imgHeight" type="text" value="600" style="width: 100%">
```

6.6.5 Codul JavaScript pentru setarea acurateții mouse-ului și rezoluției de afișare (generated file)

Pentru setarea valorilor inițiale ale rezoluției de afișare și ale acurateții mouse-ului, fișierul „app_path.js” va fi generat în mod automat.

```

function WriteToFile_APP_Path(appPathStr, sensitivity)
{
    var file_path = "";
    var fso, ts;
    var ForWriting = 2;
    var temp_str = "";
    var imgWidthTextBox = document.getElementById("imgWidth");
    var imgHeightTextBox = document.getElementById("imgHeight");
    fso = new ActiveXObject("Scripting.FileSystemObject");
    file_path = appPath + "js_files\\app_path.js";
    ts = fso.OpenTextFile(file_path, ForWriting);
    ts.WriteLine("var appPath = \"\" + appPathStr + "\\\";");

    temp_str = "// Define mouse sensitivity - by setting multiplication step will
affect number of pixels mouse must be moved for 1 dissolve transition step";
    ts.WriteLine(temp_str);
    temp_str = "// 1 - moving mouse cursor 1 pixel results in dissolve transition
by 1 step";
    ts.WriteLine(temp_str);
    temp_str = "// 2 - moving mouse 2 pixels results in 1 dissolve transition
step";
    ts.WriteLine(temp_str);
    temp_str = "// 3 - moving mouse 3 pixels results in 1 dissolve transition
step";
    ts.WriteLine(temp_str);
    ts.WriteLine("var mouse_sensitivity = " + sensitivity + "\\;");
    ts.WriteLine("var imgWidth = " + imgWidthTextBox.value + "\\;");
    ts.WriteLine("var imgHeight = " + imgHeightTextBox.value + "\\;");
    ts.Close();
}

```

6.6.6 Codul Javascript pentru controlul temporizatorului

Durata tranziției e controlată de temporizatoare, funcții care generează acțiuni specifice și definite, la intervale de timp specificate. Tranziția dintre două imagini este realizată în 100 de pași. Pentru o tranziție de o secundă, temporizatorul este acționat la fiecare 10 milisecunde. Pentru o tranziție de nouă secunde, temporizatorul este acționat folosind formula: $\text{Trigger time} = (9 \text{ seconds} * 1000 \text{ milliseconds}) / 100 \text{ steps} = 90 \text{ milliseconds}$

```

// start timer
function startTimer(timeSec)
{
    if(timeSec == 0)
    {
        time_in_mili = 0;
        time_timeout = 0;
    }
    else

```

```
{
    time_in_mili = timeSec * 1000;
    time_timeout = time_in_mili/nr_of_transition_steps;
}

// play
pause = 1;
display();
}
```

6.6.7 Codul Javascript pentru control audio și video

Pentru schimbări dinamice ale vitezei de tranziție, direcție și control audio, evenimentele generate cu ajutorul tastaturii sunt capturate și filtrate.

```
switch(KeyID)
{
    case 48:    startTimer(0);
               break;
    case 96:    startTimer(0);
               break;
    case 49:    pause = 0;
               nr_of_transition_steps = 15;
               transition_multiplication = 7;
               startTimer(1);
               break;
    case 97:    pause = 0;
               nr_of_transition_steps = 15;
               transition_multiplication = 7;
               startTimer(1);
               break;
    case 50:    pause = 0;
               nr_of_transition_steps = 15;
               transition_multiplication = 7;
               startTimer(2);
               break;
    case 98:    pause = 0;
               nr_of_transition_steps = 15;
               transition_multiplication = 7;
               startTimer(2);
               break;
    case 51:    pause = 0;
               nr_of_transition_steps = 20;
               transition_multiplication = 5;
               startTimer(3);
               break;
    case 99:    pause = 0;
               nr_of_transition_steps = 20;
               transition_multiplication = 5;
               startTimer(3);
               break;
}
```

```
case 52:    pause = 0;
           nr_of_transition_steps = 25;
           transition_multiplication = 4;
           startTimer(4);
break;
case 100:   pause = 0;
           nr_of_transition_steps = 25;
           transition_multiplication = 4;
           startTimer(4);
break;
case 53:    pause = 0;
           nr_of_transition_steps = 30;
           transition_multiplication = 3;
           startTimer(5);
break;
case 101:   pause = 0;
           nr_of_transition_steps = 30;
           transition_multiplication = 3;
           startTimer(5);
break;
case 54:    pause = 0;
           nr_of_transition_steps = 50;
           transition_multiplication = 2;
           startTimer(6);
break;
case 102:   pause = 0;
           nr_of_transition_steps = 50;
           transition_multiplication = 2;
           startTimer(6);
break;
case 55:    pause = 0;
           nr_of_transition_steps = 70;
           transition_multiplication = 2;
           startTimer(7);
break;
case 103:   pause = 0;
           nr_of_transition_steps = 70;
           transition_multiplication = 2;
           startTimer(7);
break;
case 56:    pause = 0;
           nr_of_transition_steps = 100;
           transition_multiplication = 1;
           startTimer(8);
break;
case 104:   pause = 0;
           nr_of_transition_steps = 100;
           transition_multiplication = 1;
           startTimer(8);
break;
case 57:    pause = 0;
```

```

        nr_of_transition_steps = 100;
        transition_multiplication = 1;
        startTimer(9);
    break;
    case 105:    pause = 0;
                nr_of_transition_steps = 100;
                transition_multiplication = 1;
                startTimer(9);
    break;
    case 192:
        // "`" - Flash effect - opacity 50/50
        steps_counter = steps_counter_temp;
        arrayIndex = arrayIndex_temp;

        flash_50_50 = 0;

        //play
        if(pause == 0)
        {
            pause = 1;
            display();
        }

    break;
    case 32:
        //pause
        if(steps_counter > 0)
        {
            if(pause == 1)
                pause = 0;
            else if(pause == 0)
                pause = 1;

            display();
        }
    break;
    case 37:
        // Arrow Left
        if(direction == 0)
        {
            direction = 1;

            if(arrayIndex != 0)
            {
                var
document.getElementById("img_1");          image_1_temp          =

                image_1_temp.src          =
"slides/"+imagesNameArray[arrayIndex-1];
            }
        }
    }

```

```

        //testDisplay();
    }
    break;
    case 39:
        // Arrow Right
        if(direction == 1)
        {
            direction = 0;

            if(arrayIndex != 0)
            {
                var image_1_temp =
document.getElementById("img_1");

                if(arrayIndex == imagesNameArray.length-1)
                    image_1_temp.src = "slides/"+imagesNameArray[0];
                else
                    image_1_temp.src =
"slides/"+imagesNameArray[arrayIndex+1];
            }
            // used for debug
            //testDisplay();
        }

        break;
    }
}

```

6.6.8 Codul JavaScript pentru acuratețea deplasării mouse-ului

Pentru acuratețea mouse-ului, poziția cursorului și evenimentele generate cu ajutorul mouse-ului sunt capturate și interpretate. Coordonatele deplasării mouse-ului sunt reduse la axa verticală.

```

init : function(o, oRoot, minX, maxX, minY, maxY, bSwapHorzRef,
bSwapVertRef, fXMapper, fYMapper)

```

```

{
    o.onmousedown = Drag.start;

    o.hmode = bSwapHorzRef ? false : true ;
    o.vmode = bSwapVertRef ? false : true ;

    o.root = oRoot && oRoot != null ? oRoot : o ;

    if (o.hmode && isNaN(parseInt(o.root.style.left ))) o.root.style.left =
"0px";
    if (o.vmode && isNaN(parseInt(o.root.style.top ))) o.root.style.top =
"0px";
    if (!o.hmode && isNaN(parseInt(o.root.style.right ))) o.root.style.right =
"0px";
    if (!o.vmode && isNaN(parseInt(o.root.style.bottom))) o.root.style.bottom =
"0px";
}

```

```

o.minX = typeof minX !== 'undefined' ? minX : null;
o.minY = typeof minY !== 'undefined' ? minY : null;
o.maxX = typeof maxX !== 'undefined' ? maxX : null;
o.maxY = typeof maxY !== 'undefined' ? maxY : null;

o.xMapper = fXMapper ? fXMapper : null;
o.yMapper = fYMapper ? fYMapper : null;

o.root.onDragStart = new Function();
o.root.onDragEnd = new Function();
o.root.onDrag = new Function();
},

start : function(e)
{
var o = Drag.obj = this;
e = Drag.fixE(e);
var y = parseInt(o.vmode ? o.root.style.top : o.root.style.bottom);
var x = parseInt(o.hmode ? o.root.style.left : o.root.style.right );
o.root.onDragStart(x, y);

o.lastMouseX = e.clientX;
o.lastMouseY = e.clientY;

if (o.hmode) {
    if (o.minX !== null) o.minMouseX = e.clientX - x + o.minX;
    if (o.maxX !== null) o.maxMouseX = o.minMouseX + o.maxX -
o.minX;
} else {
    if (o.minX !== null) o.maxMouseX = -o.minX + e.clientX + x;
    if (o.maxX !== null) o.minMouseX = -o.maxX + e.clientX + x;
}

if (o.vmode) {
    if (o.minY !== null) o.minMouseY = e.clientY - y + o.minY;
    if (o.maxY !== null) o.maxMouseY = o.minMouseY + o.maxY -
o.minY;
} else {
    if (o.minY !== null) o.maxMouseY = -o.minY + e.clientY + y;
    if (o.maxY !== null) o.minMouseY = -o.maxY + e.clientY + y;
}

document.onmousemove = Drag.drag;
document.onmouseup = Drag.end;

return false;
},

drag : function(e)
{

```

```

    e = Drag.fixE(e);
    var o = Drag.obj;

    var ey = e.clientY;
    var ex = e.clientX;
    var y = parseInt(o.vmode ? o.root.style.top : o.root.style.bottom);
    var x = parseInt(o.hmode ? o.root.style.left : o.root.style.right );
    var nx, ny;

    if (o.minX != null) ex = o.hmode ? Math.max(ex, o.minMouseX) :
Math.min(ex, o.maxMouseX);
    if (o.maxX != null) ex = o.hmode ? Math.min(ex, o.maxMouseX) :
Math.max(ex, o.minMouseX);
    if (o.minY != null) ey = o.vmode ? Math.max(ey, o.minMouseY) :
Math.min(ey, o.maxMouseY);
    if (o.maxY != null) ey = o.vmode ? Math.min(ey, o.maxMouseY) :
Math.max(ey, o.minMouseY);

    nx = x + ((ex - o.lastMouseX) * (o.hmode ? 1 : -1));
    ny = y + ((ey - o.lastMouseY) * (o.vmode ? 1 : -1));

    if (o.xMapper)      nx = o.xMapper(y)
    else if (o.yMapper) ny = o.yMapper(x)

    Drag.obj.root.style[o.hmode ? "left" : "right"] = nx + "px";
    Drag.obj.root.style[o.vmode ? "top" : "bottom"] = ny + "px";
    Drag.obj.lastMouseX = ex;
    Drag.obj.lastMouseY = ey;

    Drag.obj.root.onDrag(nx, ny);
    return false;
  },

  end : function(e)
  {
    document.onmousemove = null;
    document.onmouseup = null;

    Drag.obj.root.onDragEnd(  parseInt(Drag.obj.root.style[Drag.obj.hmode ?
"left" : "right"]),
                                parseInt(Drag.obj.root.style[Drag.obj.vmode ?
"top" : "bottom"]));
    Drag.obj = null;

    MouseUpCheck(e);
  },

  fixE : function(e)
  {
    if (typeof e == 'undefined') e = window.event;
    if (typeof e.layerX == 'undefined') e.layerX = e.offsetX;
  }
}

```

```

    if (typeof e.layerY == 'undefined') e.layerY = e.offsetY;
    return e;
}

```

6.6.9 Codul JavaScript pentru „DIV” și pentru obiectul player

Pentru controlul audio, creăm un obiect „DIV”, de stocare a a player-ului audio.

```

document.write('<DIV ID="player"'
+ 'style="position:absolute;left:-1000px;top:-1000px"'
+ '></DIV>');

if (detect_browser() == "MSIE" ||
    detect_browser() == "Netscape")
{
    player.innerHTML = '<object id="sound"'
+ 'classid="clsid:6BF52A52-394A-11d3-B153-00C04F79FAA6"'
+
'codebase="http://activex.microsoft.com/activex/controls/mplayer/en/nsmp2inf.cab
#Version=5,1,52,701"'
+ 'standby="Loading Microsoft® Windows® Media Player components..."'
+ 'type="application/x-oleobject" width="160" height="144">'
+ '<param name="url" value="'+media+'"'>'
+ '<param name="volume" value="100">'
+ '<embed id="sound" type="application/x-mplayer2" src="'+media+'"'
+ 'classid="clsid:6BF52A52-394A-11d3-B153-00C04F79FAA6"'
+ 'pluginspage="http://www.microsoft.com/Windows/MediaPlayer/'
+ 'type="application/x-mplayer2"'
+ 'url="'+media+'"'
+ 'volume="100"'
+ 'width="160" height="144">'
+ '<\embed>'
+ '<\object>';
}
else // if Safari or Firefox, then load Quicktime controls
{
    player.innerHTML = '<object '
+ 'classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" '
+ 'width="160" height="144" id="sound"'
+ 'style="position:absolute;left:-1000px;top:-1000px"'
+ 'codebase="http://www.apple.com/qtactivex/qtplugin.cab">'
+ '<param name="SRC" value="'+media+'"'>'
+ '<param name="AUTOPLAY" value="true">'
+ '<param name="CONTROLLER" value="false">'
+ '<param name="VOLUME" value="100">'
+ '<param name="ENABLEJAVASCRIPT" value="true">'
+ '<param name="TYPE" value="audio/wav">'
+ '<embed classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B"'
+ 'name="sound"'
+ 'id="sound"'

```

```

+ 'src="'+media+'"'
+ 'pluginspage="http://www.apple.com/quicktime/download/"
+ 'volume="100"'
+ 'enablejavascript="true" '
+ 'type="audio/wav" '
+ 'height="16" '
+ 'width="200"'
+ 'style="position:absolute;left:-1000px;top:-1000px"'
+ 'autostart="true"'
+ '> </embed>'
+ '</object>';
}

```

6.6.10 Codul JavaScript pentru controlul coloanei sonore

Evenimentele generate cu ajutorul tastaturii sunt capturate și filtrate.

```

// Play/Pause music "Ctrl + Q"
if ((KeyID == 81) && (isCtrl == true))
{
    isCtrl = false;
    //alert('CTRL + Q');
    if(pause_sound == 0)
    {
        if(song_started == 0)
        {
            song_started = 1;
            play_song();
        }
        else
        {
            pause_sound = 1;
            pause_song();
        }
    }
    else if(pause_sound == 1)
    {
        pause_sound = 0;
        pause_song();
    }
}
// Previous track "Ctrl + ,<"
if ((KeyID == 188) && (isCtrl == true))
{
    isCtrl = false;
    next_song(1);

    song_started = 0;
    pause_sound = 0;
}
// Next track "Ctrl + .>"

```

```
if ((KeyID == 190) && (isCtrl == true))
{
    isCtrl = false;
    next_song(2);

    song_started = 0;
    pause_sound = 0; }
```

7. Testarea și evaluarea aplicației Motorway

A evalua presupune, în cel mai general înțeles, a stabili valoarea unui obiect. În cazul de față, termenul este utilizat într-un sens mai restrictiv, acela de evaluare a unei aplicații. definită ca fiind o activitate științifică îndreptată spre colectarea, analizarea, interpretarea și comunicarea informațiilor despre eficacitatea unui program [93]. Evaluările au mai multe motive practice: pentru a decide dacă un anumit program merită să fie continuat, îmbunătățit, extins sau restrâns; pentru a determina utilitatea noilor programe și inițiative; pentru a crește eficacitatea managementului de proiect.

Conform cu Peter Rossi [93], pentru a realiza o evaluare eficace și justă a unui program, trebuie ținut cont de următoarele întrebări:

Care este natura și scopul problemei?

Unde este localizată aceasta, pe cine, cum și cât de mulți sunt afectați?

Cum se justifică crearea, modificarea sau extinderea de programe sociale?

Care sunt intervențiile fezabile care pot ameliora semnificativ problema?

Care este grupul țintă?

Programul dezvoltat atinge tocmai populația țintă?

Este implementat bine?

Sunt asigurate serviciile plănuite?

S-au obținut scopurile și beneficiile propuse?

Este costul programului rezonabil în comparație cu beneficiile și eficacitatea aduse de el? și altele.

Evaluarea poate fi realizată având în vedere aspecte calitative și/sau cantitative. Michael Patton susține faptul că o evaluare calitativă rezultă din trei tipuri diferite de colectări de date: interviuri, observații directe și documente scrise [94]. Interviuurile sunt menite să colecteze informații de la oameni cu privire la experiențele, opiniile, sentimentele și cunoștințele proprii. Datele din observații constau în descrieri detaliate ale activităților, comportamentelor, acțiunilor și interacțiunilor dintre oameni. Analiza documentelor include extrase din studii, citate sau întregi pasaje din diferite tipuri de înregistrări, raporturi sau publicații oficiale, jurnale personale și răspunsuri scrise la chestionare. Rezultatele calitative ale cercetărilor pot fi utilizate individual sau în combinație cu cele cantitative. În general activitatea de cercetare presupune metode multiple, inclusiv combinații ale metodelor cantitative și calitative. Chestionarul este un bun exemplu de astfel de combinație, în el utilizându-se atât întrebări cu răspuns deschis, care vizează aspectul calitativ, cât și întrebări cu răspuns închis, pentru partea cantitativă a studiului[95].

În cazul acestui studiu, trebuie evaluată atât aplicația dezvoltată, cât și utilitatea acesteia în plan educativ și artistic.

Cea mai cunoscută definiție a evaluării procesului educațional este cea dată de Tyler în 1950, revizuită și adăugită în 1989 [96], potrivit căreia evaluarea este procesul de determinare a gradului de realizare a obiectivelor educaționale. Un comitet comun pentru standarde în evaluare, format din șaptesprezece membri reprezentând doisprezece organizații asociate cu evaluarea educațională, au publicat în anul 1981 definiția evaluării ca fiind „investigarea sistematică a valorii sau meritului unui anumit obiect” [97].

Testarea și evaluarea noilor tehnologii propuse și implementate în cadrul activității doctorale trebuie realizate ținând cont de considerentele enunțate mai sus. Cele trei perspective diferite care au fost luate în considerare pentru o evaluare corectă și completă sunt următoarele:

- Stabilitatea și funcționalitatea uneltelor implementate;
- Utilitatea/ aplicabilitatea;
- Impactul asupra grupurilor țintă.

Dirjecțiile de evaluare au fost realizate prin implementarea de chestionare utilizând platforma Zef. Aceasta este o unealtă dezvoltată de partenerii UPT din Finlanda, care permite colectarea datelor și opiniilor de la orice persoană sau grup, realizând procentajul cel mai mare de răspunsuri din domeniu. Permite analizarea, organizarea și editarea datelor într-un format vizual ușor de înțeles, oferind rapoarte utile și elegante. Exemple ale aplicațiilor existente includ studii ale gradului de mulțumire al clienților, studii ale nivelului de satisfacție la locul de muncă, precum și aprecierea nivelului de inovație, risc și calitate al unei anumite acțiuni sau proiect. Întrebările pot fi alese din șabloane existente sau pot fi create de la zero, iar invitațiile pentru completarea unui chestionar sunt transmise prin intermediul e-mail sau prin publicarea pe o pagină web a unui link spre chestionar [98].

Așa cum voi detalia ulterior, axa verticală e folosită pentru a evalua ușurința de utilizare sau calitatea tranziției, și variază de la dificil sau sacadat (în origine), la foarte ușor, respectiv foarte fin (stânga sus) Cea orizontală e folosită pentru evaluarea utilității și a vitezei de răspuns, variind de la inutil respectiv cu întârzieri mari (în origine), la foarte util respectiv precis (dreapta jos). Acest grafic bi-dimensional oferă patru zone de evaluare, corespunzătoare celor patru cadrane. Rezultatele optime se situează în primul cadran (din punct de vedere trigonometric), respectiv în colțul din dreapta sus. Al treilea cadran (din punct de vedere trigonometric) este cel mai defavorabil, în partea stângă jos. Ajutorul oferit de această unealtă de evaluare este considerabil, pentru că ajută respondentul să transpună propria părere într-un grafic care cuantizează cu precizie nivelul de apreciere. Precizia răspunsului este extrem de bună.

În cele ce urmează voi prezenta detaliat metodele de evaluare a aplicației Motorway. Prin interconectarea rezultatelor vor fi obținute o serie de concluzii necesare pentru a lua o decizie cu privire la îmbunătățirea și extinderea aplicației.

7.1 Prezentarea către respondent. Adobe Captivate.

Așa cum am arătat, folosirea aplicației e supusă unor constrângeri. Internet Explorer este singurul web-browser compatibil, iar la deschidere, utilizatorul nevizat va fi întrebat de multe ori dacă e de acord să accepte restricțiile impuse de acesta. De asemenea, popularea folderului „slides” cu imagini proprii și al celui „audio” cu o coloana sonoră poate părea neprietenoasă, în contextul în care cele mai multe aplicații posedă un browser propriu cu ajutorul căruia se pot încărca imagini și muzică.

Din acest motiv, am folosit o unealtă extrem de eficientă, care prezintă chiar și celui mai nevizat utilizator, toți pașii prin care trebuie să treacă pentru a folosi

corect aplicația. Folosind Adobe Captivate pentru a înregistra tot ce se întâmplă pe ecran, reușesc să fiu mai elocvent decât dacă aș scrie toate explicațiile.

Adobe Captivate oferă posibilitatea de a crea simulări puternice și pătrunzătoare, training pe bază de scenariu și teste robuste, fără cunoștințe de programare sau experiență multimedia. Capturează și înregistrează activitatea ecranului și sincronizează materialele video și audio. Dezvoltă și pune în aplicare simulări atrăgătoare, cu rezolvarea sarcinilor. Dezvoltă rapid materiale didactice pentru training complex pe bază de scenariu.

Adobe Captivate (fostul *RoboDemo*) este o unealtă electronică pentru învățare, atât pentru Windows cât și pentru Mac, ce poate fi folosită pentru a prezenta aplicații, a simula folosirea lor, exportând produsul finit în format .swf. De asemenea, poate converti fișiere .swf în .avi, ce pot fi încărcate pe site-uri web, ca material didactic. Pentru simularea utilizării unor aplicații, Captivate poate identifica apăsarea tastelor, și a mouse-ului, mișcarea acestuia precum și defilarea imaginilor pe ecran.

Extensia .swf provine fie de la "Small Web Format" (format mic pentru web), fie de la "Shockwave Flash". Concepute pentru a fi suficient de mici astfel încât să fie publicabile pe Internet, fișierele swf pot conține animații sau applet-uri cu diverse grade de interactivitate.

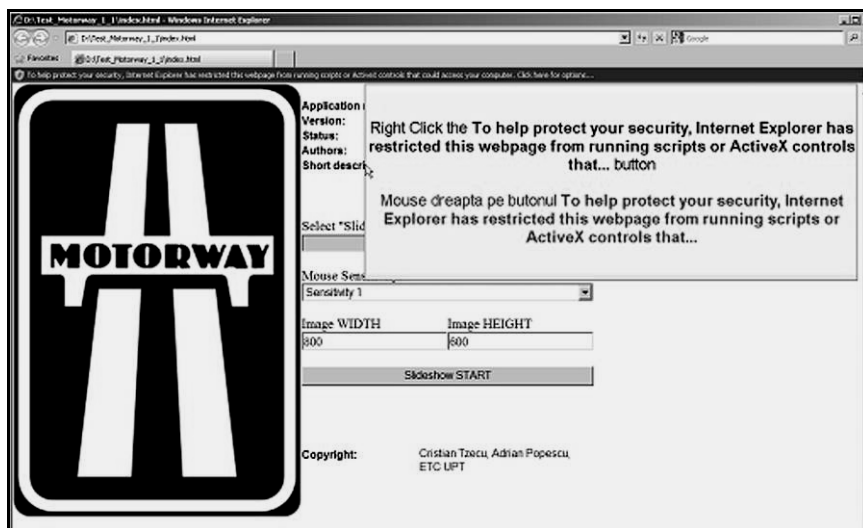


Fig. 47 - Aspect al demonstrației de instalare în fază inițială. Print Screen.

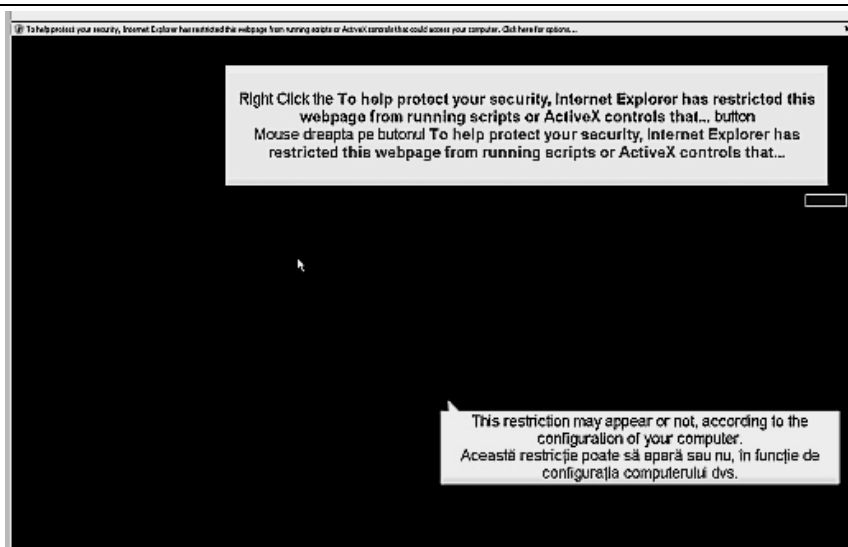


Fig. 48 - Aspect al demonstrației de instalare în fază finală. Print Screen.

Concret, utilizatorul nefamiliarizat a putut urmări un tutorial în format Flash prin care au fost explicate modalitățile de instalare a aplicației, transferul imaginilor dorite pentru diaporamă în directorul "slides", respectiv a muzicii în directorul "audio". De asemenea, metodele de acceptare a restricțiilor impuse de Internet Explorer, astfel încât de la dezarhivare până la momentul începerii diapramei, utilizatorul să nu mai întâmpine dificultăți.

7.2 Imagini ce au făcut obiectul testării

Pentru testarea aplicației am oferit un set de 35 de imagini din biblioteca personală [99] selectate astfel încât tranziția dintre ele să fie atractivă. Am căutat ca imaginile aflate în succesiune să posede elemente grafice comune, fie dispuse pe liniile de forță ale cadrului, fie într-unul din cele patru puncte de forță. În acest fel, așteptarea privitorului de a primi informații noi de la următoarea imagine este împlinită, și implicit gradul de satisfacție vizuală. Aidoma prezentărilor de diaporamă, fie ele analogice sau digitale, prima imagine, numerotată "00", este neagră.

Imaginile provin de la un aparat foto digital Fuji E900, de 9 megapixeli. Ele au fost prelucrate în Adobe Photoshop astfel încât să aibă caracteristicile de cromatică, contrast și luminozitate cât mai apropiate de realitate. Apoi, au fost redimensionate, și salvate în două directoare, fiecare corespunzător unei dimensiuni. 800/600 pixeli este prima dimensionare, și respectiv 1200/900, păstrând proporțiile. Imaginile mici au un volum mediu de 75 kB, cele mari respectiv 130 kB. Am căutat ca imaginile să aibă volume de date asemănătoare, și în acest scop am folosit opțiunea "Save for web and devices" din Adobe Photoshop, și am specificat la opțiunea "Optimize to File

132 Testarea și evaluarea aplicației Motorway - 7

Size : Desired File Size" 75, respective 130 k. Evident, rezoluția este de 72 pixeli pe inch pătrat, fiind comună monitoarelor și videoproiectoarelor.

Am ales să ofer aceleași imagini, dar la dimensiuni și volume diferite, pentru ca testarea să fie mai relevantă. Fiind vorba de un calcul în timp real, m-am așteptat ca la imaginile de dimensiune mare, unele computere cu putere redusă de calcul să nu ofere o calitate atât de bună a tranzițiilor ca și la imaginile mici.

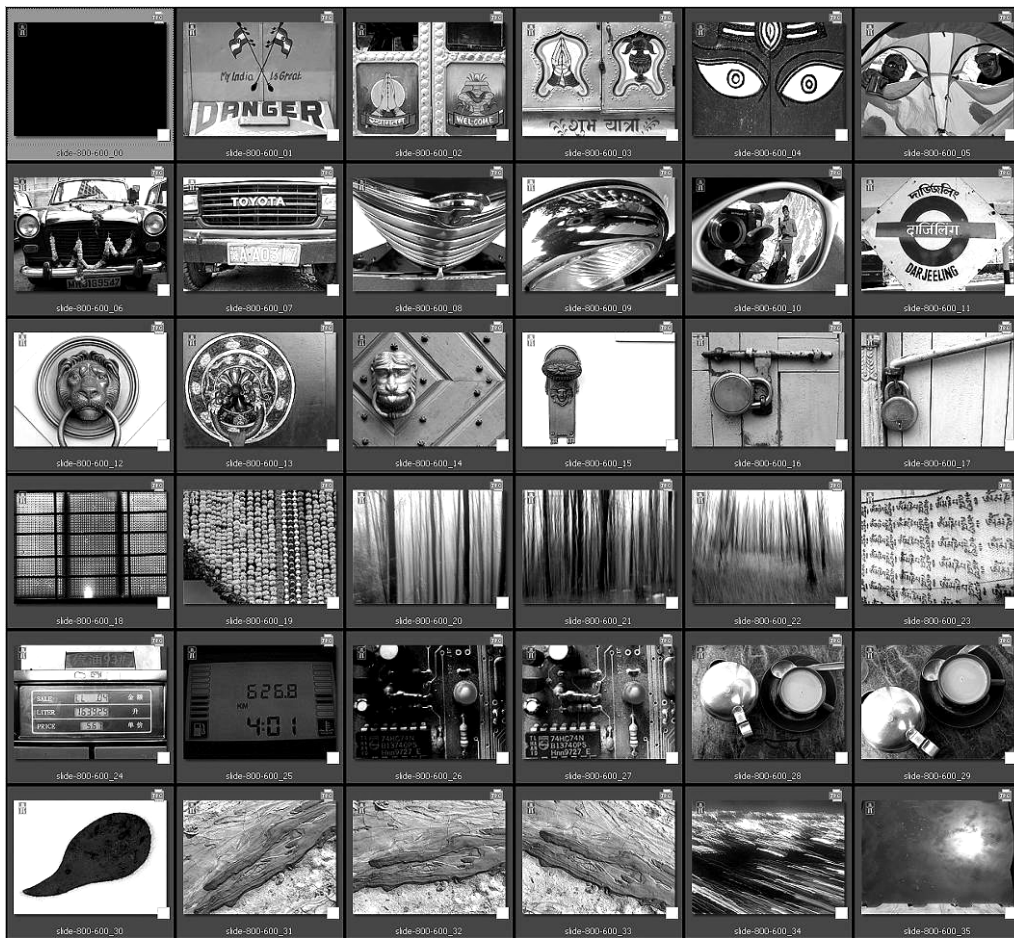


Fig. 49 - Imaginile folosite pentru testarea aplicației

Așa cum am arătat în primul capitol al tezei, am căutat să dau o importanță mare celei de "a treia imagini", obținută prin suprapunerea dintre două imagini succesive, cu elemente asemănătoare în cadru, sau dimpotrivă, complementare.



Fig. 50 - Etape ale tranziției între imaginile 04 și 05

În figura 50 se pot remarca imaginile 04 și 05 având ca elemente grafice comune ochii pictați pe poarta roșie, respectiv deschiderile cortului care se aseamănă cu doi ochi. Suprapunând imaginile, la o opacitate de 50%, remarcăm păstrarea și continuarea similarității cu un chip uman.



Fig. 51 - Etape ale tranziției între imaginile 11 și 12

În figura 51 se pot remarca imaginile 11 și 12 având ca elemente grafice comune inelul roșu de pe indicatorul feroviar indian, respectiv forma circulară a ciocanului pentru poartă. Suprapunând imaginile, la o opacitate de 50%, se crează senzația că din interiorul inelului roșu emerge capul de leu, aflat la rândul său într-un cerc concentric cu cel roșu. La aceasta ajută formele rombice foarte asemănătoare, prezente în ambele imagini.



Fig. 52 - Etape ale tranziției între imaginile 28 și 29

Imaginile 28 și 29 ilustrează același subiect, însă încadrat diferit. Trecând de la o imagine la alta, în funcție de viteză, se poate crea iluzia mișcării camerei, aidoma experimentului Muybridge [26]. Fundalul de culoare închisă trece neobservat, și ajută la detașarea celor două elemente principale ale cadrului, astfel încât privitorul remarcă cu precădere cana verde și vasul metalic. Pe parcursul tranziției, privirea observatorului va urmări aceste două elemente.



Fig. 53 - Etape ale tranziției între imaginile 29 și 30

Imaginile 29 și 30 sunt complementare. Fundalul primei imagii este întunecat, în timp ce a doua constă dintr-o zonă închisă, înconjurată de una alb-deschis. Trecând de la o imagine la alta, ceașca verde persistă pe ecran mai mult întrucât se suprapune peste o zonă închisă. În schimb, zăpada din a doua imagine "acoperă" mai repede elementele mai întunecate din prima imagine.

Aceste patru exemple demonstrează câteva trucuri vizuale care fac ca o prezentare de acest fel să fie atractivă. După ce grupul de respondenți a testat aplicația cu ajutorul acestor imagini, au fost invitați să folosească propriile bănci de imagini, pentru că în acest fel satisfacția unei prezentări proprii ar trebui să fie implicit mai mare. Acesta a fost și scopul întregului studiu, de a oferi tuturor o aplicație pe care să o utilizeze cu imagini proprii.

În directorul "audio" am inclus trei piese instrumentale în format .mp3, ale formației Tangerine Dream, care au un ritm ce permite asocierea imaginilor cu o cadență de 5-6 secunde/imagine. Fiind mai multe piese muzicale separate, utilizatorul poate folosi tastatura pentru a schimba muzica în timpul prezentării, conform instrucțiunilor de utilizare anexate. Desigur, atunci când respondenții s-au obișnuit cu aplicația, au fost încurajați să folosească muzica aleasă de ei pentru prezentări.

7.3 Instrucțiuni de utilizare

Pentru că unor utilizatori nu le sunt familiare vechile dispozitive analogice folosite pentru comanda unui spectacol de diaporamă, am atașat și un set de instrucțiuni de utilizare. Ele au fost concepute spre a fi tipărite și folosite chiar în momentul utilizării, în ideea că experiența și dexteritatea se câștigă în timp, după mai multe utilizări.

Ctrl + Q -muzica pornește

Ctrl + Q din nou – muzica se întrerupe, urmând ca la o nouă comandă similară, să pornească din acel punct.

Ctrl + > - trece la următoarea piesă, în ordine alfabetică, dacă în directorul "audio" sunt mai multe piese. Pentru

pornirea acesteia, **Ctrl + Q**

Ctrl + < - trece la precedenta piesă, în ordine alfabetică, dacă în directorul "audio" sunt mai multe piese. Pentru pornirea acesteia, **Ctrl + Q**

Esc -reia ultima piesă de la început

Control folosind tastatura numerică

1 – o secundă de tranziție
2 – două secunde de tranziție
3 – trei secunde de tranziție

9 – nouă secunde de tranziție, cea mai lungă tranziție controlabilă de tastatură
0 - cut
Săgeată Stânga _ inversează ordinea de proiecție (alfabetică) a imaginilor. Permite astfel întoarcerea la o imagine ce a fost deja proiectată (schimbă sensul tranziției).
Săgeată Dreapta _ revine la ordinea de proiecție (alfabetică) a imaginilor, implicită (schimbă sensul tranziției).
Space – pauzează tranziția. Apăsând din nou, reia tranziția întreruptă.
Ctrl + S – ascunde/arată bara de control a mouse-ului, situată în partea dreaptă a ecranului.
Ctrl+1, Ctrl+2, Ctrl+3 controlează cursa mouse-ului. Ctrl+1 indică o cursă scurtă a mouse-ului, pentru tranziții ce nu necesită acuratețe, iar Ctrl+3 indicată cea mai lungă cursă a mouse-ului, pentru tranziții de mare finețe.
Mouse control:
Pasul 1: mouse stânga pe bara de control a mouse-ului, situată în partea dreaptă a ecranului. Cu mouse-ul apăsat trageți de el în jos. Are loc tranziția dintre prima și a doua imagine, viteza sa depinzând de mișcarea mouse-ului.
Pasul 2: eliberați butonul de mouse stânga după apariția completă a celei de-a doua imagini.
Pasul 3: mouse stânga pe bara de control a mouse-ului, cu mouse-ul apăsat trageți de el în sus. Are loc tranziția dintre a doua și a treilea imagine, viteza sa depinzând de mișcarea mouse-ului.
Pasul 4: eliberați butonul de mouse stânga după apariția completă a celei de-a treia imagini.
 Dacă bara de control a mouse-ului e în poziția superioară, se pot schimba setările de acuratețe ale mouse-ului (Ctrl + 1,2,3). Tot în această poziție bara de control a mouse-ului poate fi ascunsă (Ctrl+S).
 ~ sau **MOUSE DREAPTA** – efect flash (50% transparență pentru ambele imagini, chiar și în timpul unei tranziții în derulare)
Alt + F4 –închide fereastra de diaporamă

7.4 Testarea aplicației

Partea de testare a funcționalității aplicației Motorway a cuprins următoarele zece întrebări. Respondenții au fost rugați să completeze după ce au testat aplicația de mai multe ori, la un interval de 4 zile, pentru a nu avea o parere alterată de dificultățile inițiale. Pentru că am considerat că aplicația ar putea avea două domenii majore de utilizare, am grupat respondenții pe două categorii, corespunzătoare domeniilor artistic și educațional.

Prin repondent din domeniul artistic numesc acea persoană care a vizionat conștient mai multe diaporame artistice, fie ele digitale sau analogice, și care a realizat cel puțin un montaj audio-vizual artistic și/sau documentar, fie acesta

diaporamă, film, prezentare digitală de altă natură. Am beneficiat de sprijinul a nouă repondenți din domeniul artistic.

Prin repondent din domeniul educativ numesc acea persoană care este implicată în educarea altor persoane, indiferent că este vorba de domeniul universitar, preuniversitar, training oferit unor beneficiari. Am impus ca aceste persoane să fie utilizatori de mijloace audio-vizuale moderne în sprijinul activității lor educaționale. Am beneficiat de sprijinul a șapte repondenți din domeniul educațional.

Întrebările adresate au fost următoarele:

- 1) Vârstă, domeniu de activitate.
- 2) Până acum ce tip de prezentări audiovizuale ați realizat?
- 3) Ce aplicații audiovizuale folosiți?
- 4) Configurarea computerului Dvs -> My Computer -> Properties -> General
- 5) Pentru vizionarea aplicației Motorway ați folosit laptop, monitor separat sau videoproiector (vă rog să specificați rezoluția).
- 6) Apreciați ușurința de utilizare și utilitatea aplicației Motorway.
- 7) Calitatea tranziției pentru imagini de lățime 800 pixeli.
- 8) Calitatea tranziției pentru imagini de lățime 1200 pixeli.
- 9) Calitatea tranziției pentru mouse.
- 10) Ați descoperit erori în funcționare? (precizați)
- 11) Sugestii, comentarii.

Iată răspunsurile celor 9 evaluatori din domeniul artistic și respectiv celor 7 din domeniul educațional. Fiecare subcapitol cuprinde atât o interpretare vizuală a răspunsului, oferită de Zef Solutions, cât și o interpretare personală, comparând datele existente.

7.4.1 Vârstă, domeniu de activitate

Domeniu artistic:

- ✓ 26, sănătate
- ✓ 34, dezvoltare pagini web, fotografie
- ✓ 27, educație
- ✓ 35, sănătate
- ✓ 40, artistic și educațional
- ✓ 28, artistic, muzical și tehnic
- ✓ 30, asigurari
- ✓ 39, IT, fotografie, video, diaporamă
- ✓ 31, ONG

Domeniu educațional:

- ✓ 30, scenarist
- ✓ 23, programare
- ✓ 45, audio
- ✓ 24, servicii sociale
- ✓ 25, servicii sociale
- ✓ 27, educație și cercetare
- ✓ 29, educație

Analizând aceste prime informații, constatăm că avem de a face cu persoane cu studii medii și superioare, cu vârste cuprinse între 23 și 45 de ani. Media de vârstă este puțin sub 31 de ani, ceea ce reflectă un grup de repondenți cu experiență în domeniile lor de activitate.

7.4.2 Experiența evaluatorilor

Până acum ce tip de prezentări audiovizuale ați realizat?

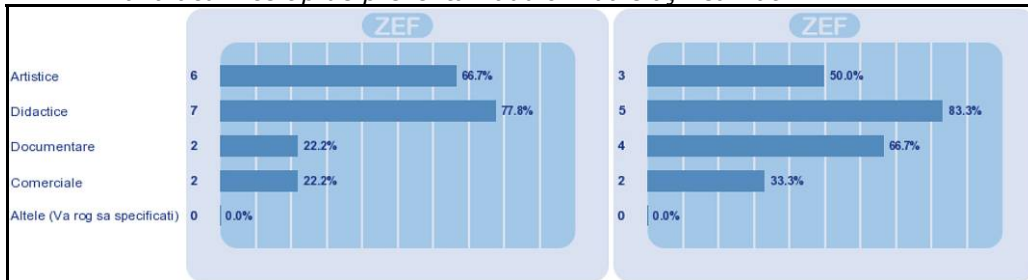


Fig. 54 - Experiența repondenților domeniul artistic (stânga) respectiv educațional (dreapta)

Diferențierea începe să se facă simțită. Domeniul artistic devansează pe cel educațional la prezentări artistice (cu 16,7%). În schimb, este devansat de educațional la prezentări didactice (cu 5,5%), la documentare (cu cifra record de 44,5%) și la comerciale (cu 11,1%). Rezultă că domeniul de realizare a unor prezentări audio-vizuale este vast, și are caracteristici specifice fiecărei grupe de repondenți. De remarcat că la repondenții din domeniul artistic prezentările didactice le devansează –paradoxal- pe cele artistice, în timp ce la repondenții din domeniul didactic, procentul mare de lucrări artistice realizate le conferă un orizont mai larg. De asemenea, este interesant de menționat că niciunul din repondenți nu a ieșit din tiparele primelor patru categorii (realizând de exemplu prezentări vizuale ce acompaniază un concert, sau prezentări audio-vizuale motivaționale).

7.4.3 Aplicații folosite de respondenți

Ce aplicații audiovizuale folosiți?

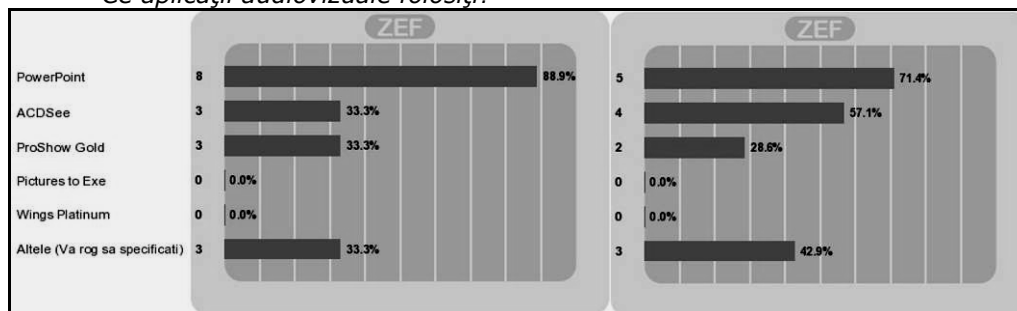


Fig. 55 - Aplicații utilizate de cei din domeniul artistic (stânga) respectiv educațional (dreapta)

Alte aplicații folosite de trei repondenți din domeniul artistic:
 1) Adobe Premiere 2) Sony Vegas, Adobe Premiere, Adobe After Effects 3) VirtualDub, Adobe Premiere

Alte aplicații folosite de trei repondenți din domeniul educațional:

1) Xtreme PhotoStory 2) Adobe Premiere, Encore 3) Adobe Presenter

O primă concluzie este aceea că PowerPoint a detronat toate celelalte metode de prezentare. Domeniul artistic folosește mai puțin ACDSee pentru prezentări decât cel educațional, aceasta fiind o aplicație plurivalentă de vizualizat și editat imagini statice da cu opțiunea de diaporamă. Îmbucurător este că din cele trei aplicații dedicate diaporamei digitale, cărora le-am consacrat câte o prezentare în această teză, una din ele este utilizată de ambele categorii, respectiv ProShow Gold, deși într-un procentaj mic. Este evident că prețul mare al Wings Platinum este cel care nu îl plasează între preferințe. De ce ProShow Gold și de ce nu Pictures to Exe pentru utilizatorii noștri, poate constitui subiectul unui studiu colateral. Demn de remarcat este faptul că trei repondenți din domeniul artistic folosesc Adobe Premiere, cea mai la îndemână aplicație profesională pentru editare video, dar care permite și facilitează deasemenea tranziția dintre imagini statice, adăugarea muzicii multitrack. Rezultatul este exportat în format .avi, singurul dezavantaj fiind volumul foarte mare al fișierelor, comparativ cu aplicațiile specializate de diaporamă digitală.

7.4.4 Puterea de calcul a computerelor evaluatorilor

Configurarea computerelor respondenților din domeniu artistic:

1. (M) AMD Athlon X2 "Brisbane", 2,0 GHz, 2GB RAM, Windows XP SP2 32-bit
2. (B) Windows 7 Ultimate Intel Core2Duo 2 GHz 4 GB RAM
3. (B) Windows 7, procesor Intel Core 2 Quad Q6600, 2,40 GHz, RAM 4 GB
4. (S) Windows XP, procesor Intel Pentium 4 CPU, 3GHz, 798Mhz, 3 GB of RAM
5. (M) Windows XP, Intel Pentium Dual CPU, 1,6GHz, 798Mhz, 1,87 GB RAM
6. (S) Laptop Lenovo: Win Vista, CPU Intel Centrino Duo 1.5 GHz, 2 GB RAM
7. (M) Windows Vista, procesor Intel(R) Core(TM)2 T5500 @1.66 GHz, 2GB RAM
8. (M) Celeron, 2.4GHz, 768 DDRII XP sp3
9. (B) Windows XP, Intel Core Duo CPU, 2,09 GHz, 2 GM of RAM

Configurarea computerelor respondenților din domeniu educațional:

1. (B) Intel Pentium Quad CPU, 2,33GHz, 4 GB of RAM
2. (S) Windows Xp Professional - SP3, procesor AMD Turion 64x2 Mobile Technology TL-56, MMX, ~1,8 GHz, 896 MB of RAM
3. (S) AMD Athlon XP 2000+, 1,67 GHz. 512 MB RAM.
4. (M) Windows XP Professional Genuine Intel CPU T2400 @1.83GHz 987 MHz 1GB RAM
5. (M) Windows XP Genuine Intel(R) CPU T2400 1.83GHz 987 MHz 1GB of RAM
6. (B) Intel Core2 Quad Q6660 2.40 GHz, 4 GB RAM
7. (B) Windows XP Professional, Intel Core 2 CPU, 6600@2.4GHz, 2.39Ghz, 1.96 GB RAM

Comparând cele două grupe de repondenți, puterea de calcul este similară. Am notat cu (B) o putere bună de calcul, cu (M) una medie și respectiv cu (S) una slabă. Acordând 3 puncte pentru (B), două pentru (M) și unul pentru (S), media celor două grupe este similară, respectiv 2,1 puncte din 3 posibile. Anumite concluzii se pot trage însă cunoscând aceste date, deoarece mă aștept ca cei ce posedă aparatură performantă (B) să fie mai mulțumiți de aplicație.

Faptul că ambele grupe au aceeași putere de calcul arată că studiul nu este alterat de avantajul clar al unei grupe în detrimentul celeilalte. Consider că această echivalență ajută mult la obiectivitate.

7.4.5 Vizionarea aplicației

Pentru vizionarea aplicației Motorway ați folosit laptop, monitor separat sau videoproiector (vă rog să specificați rezoluția).

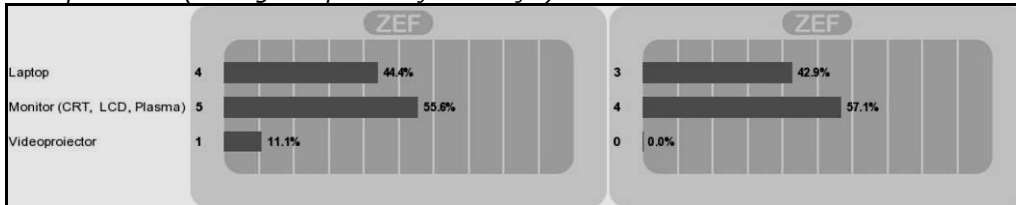


Fig. 56 - Vizualizarea aplicației de către cei din domeniul artistic (stânga) respectiv educațional (dreapta)

Respondenții din domeniul artistic folosesc: **Monitor** LCD1920/1080, LCD1280/1024, CRT1280/1024 CRT1280/960, **Laptop** 1280/800, 1280/800, 1280/800 și **Videoproiector** 1280/900. Cei din domeniul educațional folosesc: **Monitor** 1680/1050, 1280/1024 și **Laptop** 1280/800, 1280/800, 1280/800

Concluzia acestor răspunsuri este că ambele categorii de repondenți folosesc de regulă o rezoluție a monitorului de 1280/800 pixeli. Acest răspuns este extrem de favorabil studiului, întrucât setul de imagini oferite pentru testare la dimensiune 1200/900 se încadrează în full-screen-ul tuturor.

Unul singur folosește un sistem LCD care permite vizualizarea HD, prețul foarte ridicat fiind o explicație foarte plauzibilă pentru singularitate. Din păcate, doar unul singur (din domeniul artistic) folosește videoproiectorul. Rezultă de aici că prezentările realizate de repondenți nu sunt atât destinate unei adresabilități largi (așa cum facilitează un videoproiector) ci mai degrabă în cadru intim. Pe de altă parte, nu exclud posibilitatea –pentru că și eu procedez similar- ca repondenții să nu aibă videoproiector propriu, dar să nu ezite să folosească aparatură împrumutată atunci când nevoia de largă adresabilitate este evidentă.

7.4.6 Ușurința de utilizare și utilitatea

Apreciați ușurința de utilizare și utilitatea aplicației Motorway.

Această întrebare așteaptă răspunsuri bidimensionale. Aceasta înseamnă că un singur răspuns realizează evaluarea și a ușurinței de utilizare, și a utilității aplicației. Toate răspunsurile la cele două întrebări sunt centralizate în graficul ce urmează:

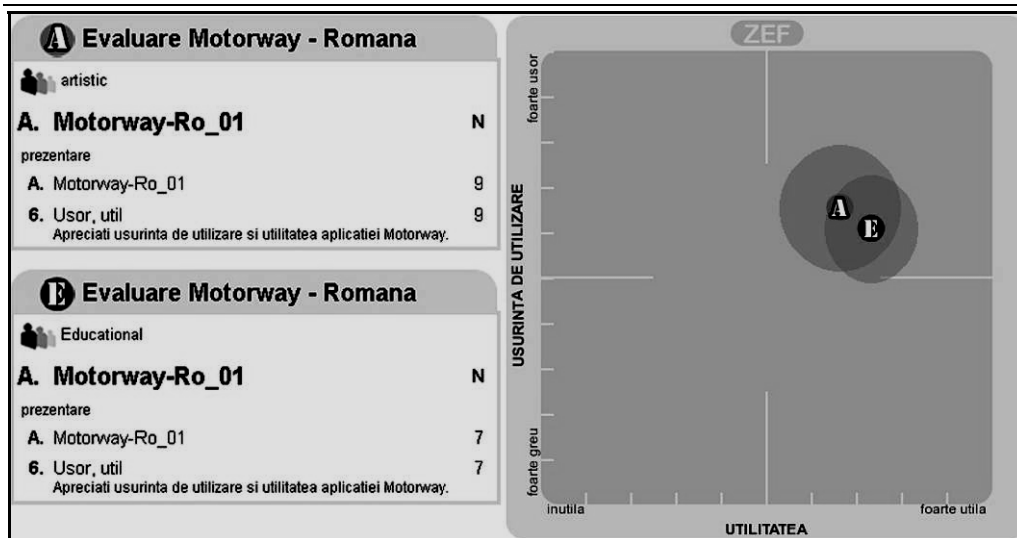


Fig. 57 - Aprecierea ușurinței și utilității aplicației de către cei din domeniul artistic (A) respectiv educațional (E)

Artistic: X=66,3 Y=65

Educațional: X=73,4 Y=60,7

Având în vedere atât restricțiile impuse de Internet Explorer cât și faptul că încărcarea imaginilor și a muzicii se face folosind Windows Explorer, Total Commander sau alte aplicații similare –și nu din vreun browser intern-, rezultatele sunt îmbucurătoare. 65% pentru domeniul artistic respectiv 60% pentru cei din educațional în ce privește ușurința de utilizare arată că evaluatorii nu au întâmpinat dificultăți majore în utilizare. De altfel, sugestiile tuturor indică dorința de instalare mai facilă; nimeni nu a avut nemulțumiri legate de ușurința de utilizare a aplicației, ci mai degrabă de faza premergătoare utilizării, respectiv instalarea. Utilitatea este cotateă cu 66,3% de artiști, în vreme ce evaluatorii din mediul educațional o cotează la 73,4%. Cele două grupuri nu au răspunsuri prea diferite. Faptul că ambele sunt situate în primul cadran (d.p.d.v trigonometric) denotă un aspect pozitiv. Sugestiile evaluatorilor sunt binevenite. Asupra lor voi reveni.

7.4.7 Calitatea la 800 pixeli

Apreciați calitatea tranziției pentru imagini de lățime 800 pixeli.

Această întrebare așteaptă răspunsuri bidimensionale. Aceasta înseamnă că un singur răspuns realizează evaluarea atât a calității tranziției la o lățime a imaginii de 800 pixeli, cât și a vitezei de răspuns a aplicației pentru aceleași condiții. Toate răspunsurile la cele două întrebări sunt centralizate în graficul care urmează:

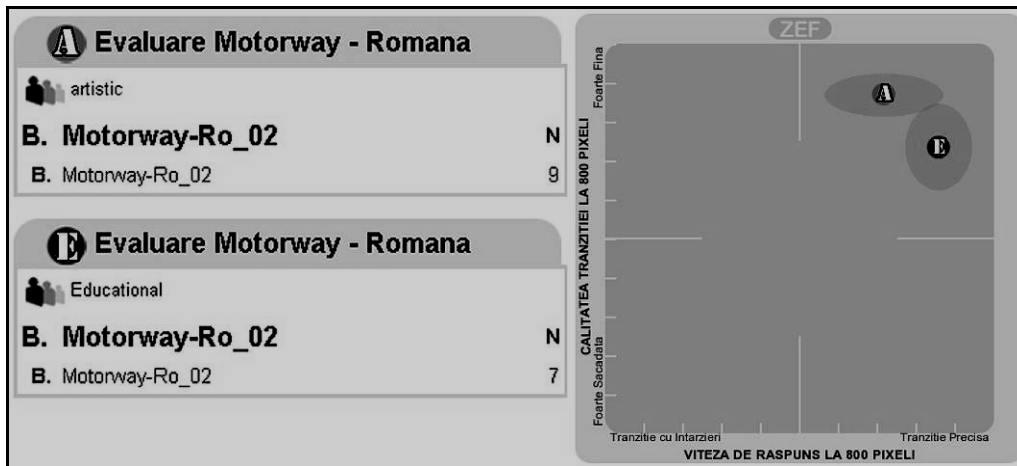


Fig. 58 - Calitate vs. viteză de răspuns a tranziției la 800 pixeli pentru cei din domeniul artistic (A) respectiv educațional (E)

Artistic : X=71,5 Y=86,6
Educațional: X=85,8 Y=73,3

Ambele grupuri de evaluare sunt satisfăcute atât de calitatea tranziției între imagini cu lățimea de 800 pixeli, cât și de faptul că acestea nu au întârzieri notabile. Există o diferență vizibilă între cele două grupe, însă aceasta nu afectează percepția globală; ambele grupe și-au orientat preferințele spre primul cadran.

Evaluatorii artiști sunt cu 13,3% mai mulțumiți decât cei din educație de calitatea tranziției. Pe de altă parte, tot artiștii sunt cu 14,3% mai puțin mulțumiți de viteza de răspuns a aplicației. O explicație ar putea fi căutată analizând echipamentul cu ajutorul căruia s-a făcut evaluarea, prezentat detaliat în subcapitolul 7.4.4.

7.4.8 Calitatea la 1200 pixeli

Apreciați calitatea tranziției pentru imagini de lățime 1200 pixeli.

Această întrebare așteaptă răspunsuri bidimensionale. Aceasta înseamnă că un singur răspuns realizează evaluarea atât a calității tranziției la o lățime a imaginii de 1200 pixeli, cât și a vitezei de răspuns a aplicației pentru aceleași condiții. Toate răspunsurile la cele două întrebări sunt centralizate în graficul care urmează:

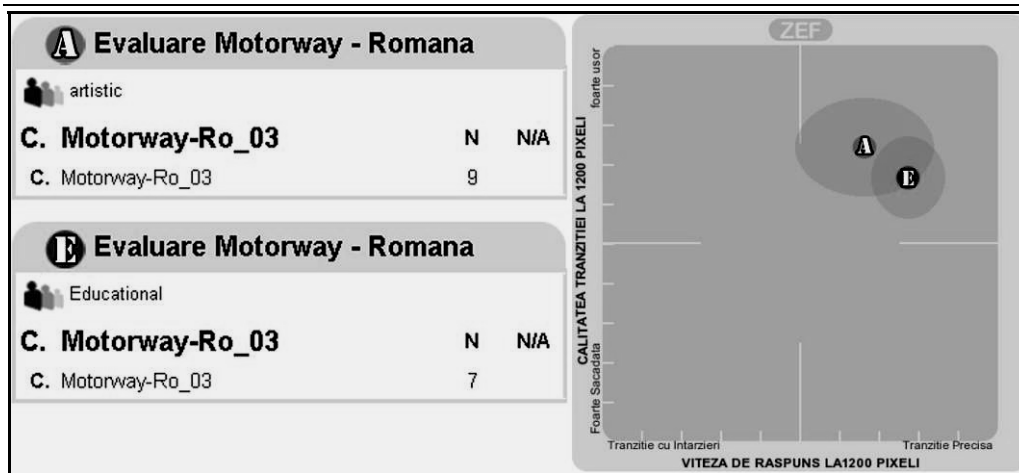


Fig. 59 - Calitate vs. viteză de răspuns a tranziției la 1200 pixeli pentru cei din domeniul artistic (A) respectiv educațional (E)

Artistic: X=66,5 Y=74,1
 Educațional: X=77,4 Y=66,3

Ambele grupuri de evaluare sunt de asemenea satisfăcute atât de calitatea tranziției între imagini cu lățimea de 1200 pixeli, cât și de faptul că acestea nu au întârzieri notabile. Există și aici o diferență vizibilă între cele două grupe, însă aceasta nu afectează percepția globală; ambele grupe și-au orientat preferințele spre primul cadran.

Evaluatorii artiști sunt cu 7,8% mai mulțumiți decât cei din educație de calitatea tranziției. Pe de altă parte, tot artiștii sunt cu 10,9% mai puțin mulțumiți de viteza de răspuns a aplicației. Explicația ar trebui să fie similară cu cea din subcapitolul precedent.

Ușoara scădere în apreciere între situația 1200 pixeli față de 800 pixeli este vizibilă, dar foarte ușor de explicat. Așa cum am arătat în capitolul 6, puterea de calcul influențează rezultatul final. Tranziția implică înlocuirea fiecărui pixel al imaginii inițiale cu pixelul aferent celei finale, trecând printr-un număr mare de pași intermediari. Cerând computerului să realizeze o tranziție ce implică înlocuirea a 480.000 pixeli, aferenți dimensiunii 800/600, acesta va utiliza în timpul solicitat o anumită putere de calcul. O configurație mai slabă a computerului poate duce la întârzieri mai mari la imaginile 1200/900, unde trebuie înlocuiți în aceeași unitate de timp 1.080.000 pixeli, mai mult decât dublul situației precedente.

Făcând abstracție de separarea evaluatorilor pe cele două domenii, am încercat să aduc la un loc toate răspunsurile celor 16 persoane. Am putut astfel compara calitatea tranziției și viteza de răspuns pentru imagini de lățime 800 pixeli, respectiv 1200 pixeli. Reprezentarea grafică a răspunsului se poate vedea în figura 60. Atât scăderea calității cât și a vitezei de răspuns pentru tranziții între imagini de 1200 pixeli sunt proporționale. Calitatea tranziției scade cu 10,3%, viteza de răspuns cu 6,2%. Nefiind vorba de scăderi dramatice doar pe una din axe, rezultă că

puterea de calcul scăzută a computerelor unor evaluatori este cea care determină acest rezultat.

O confirmare vine de la răspunsul unor din evaluatori, posesori de aparatură performantă(B) (Windows 7, procesor Intel Core 2 Quad Q6600, 2,40 GHz, RAM 4 GB) care au afirmat că nu au sesizat diferențe de calitate între tranzițiile celor două tipuri de imagini.

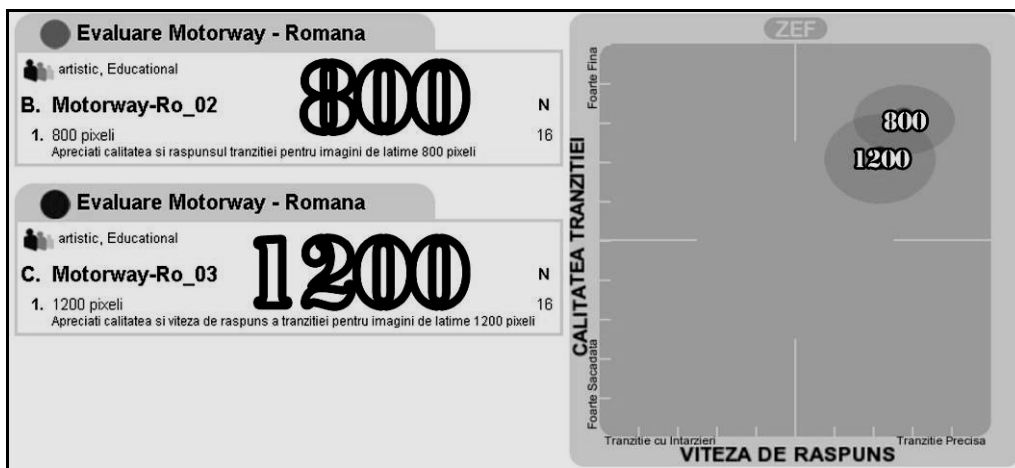


Fig. 60 - Comparație între calitatea tranziției la 800 și la 1200 pixeli

800 pixeli: X=77,8 Y=80,8
1200 pixeli: X=71,6 Y=70,5

7.4.9 Calitatea tranziției pentru mouse

Apreciați calitatea tranziției pentru mouse.

Această întrebare așteaptă răspunsuri bidimensionale. Aceasta înseamnă că un singur răspuns realizează evaluarea atât a calității tranziției atunci când se folosește mouse-ul, cât și a vitezei de răspuns a aplicației pentru aceleași condiții. Toate răspunsurile la cele două întrebări sunt centralizate în graficul care urmează:

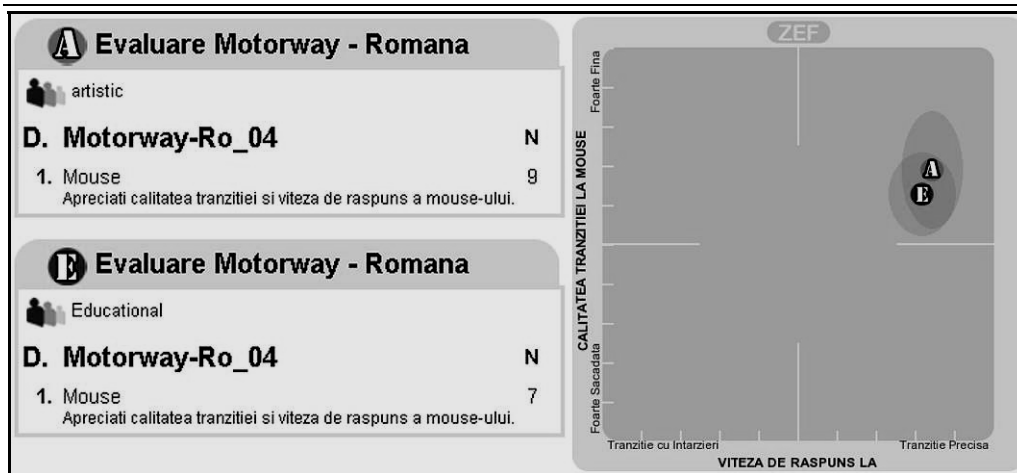


Fig. 61 - Calitate vs. viteză de răspuns a tranziției controlate de mouse pentru cei din domeniul artistic (A) respectiv educațional (E)

Artistic : X=84,3 Y=68,6

Educațional: X=81,5 Y=62,4

Ambele grupuri de evaluare sunt de asemenea satisfăcute atât de calitatea tranziției între, cât și de faptul că viteza de răspuns în cazul comenzii cu mouse-ul. Diferența între cele două grupe este mult mai mică decât la precedentele situații; ca și până acum ambele grupe și-au orientat preferințele spre primul cadran.

Există o proporționalitate între cele două grupe; cei din domeniul artistic sunt cu 2,8% mai mulțumiți de calitatea tranziției, și cu 6,2% de viteza de răspuns. Am primit și un feed-back încurajator „Preferință pentru mouse, mai mult decât pentru taste”.

7.4.10 Erori în funcționare

Ați descoperit erori în funcționare? (precizați)

Unele răspunsuri denotă că evaluatorii nu au tratat cu atenție invitația, în care menționăm clar că aplicația lucrează doar cu Internet Explorer (“Nu merge în Firefox...”, “Nu ruleaza decat în Internet Explorer, fără sunet. Eroare legată de slideshow.htm în Firefox”). Alteori, evaluatorii nu au fost atenți la instrucțiunile din capitolul 7.3 (“Probleme la ieșirea din aplicație, a fost necesar kill din Task Manager”) care oferă combinația Alt +F4 pentru a ieși din aplicație.

Altele însă merită multă atenție pentru că ridică probleme interesante. De exemplu, unii evaluatori care foloseau Windows 7 au observat apariția unor artefacte (pixeli albi) care par a se datora sistemului de operare. Aceleași artefacte au fost însă observate sub Windows 7 și la alte aplicații.

Un evaluator observă că “În cazul opririi în mijlocul tranziției și eliberarea mouse-ului, la reluare se face un salt la o imagine după care se reia tranziția normală”, ceea ce este perfect adevărat. Situația indică o exploatare greșită a

aplicației (eliberarea mouse-ului în timpul tranziției), însă problema ridicată va trebui remediată, astfel că și în cazul unor erori de exploatare, aplicația să ajute utilizatorul.

7.4.11 Recomandări

Sugestii, comentarii

În acest capitol, mă voi rezuma la a cita câteva recomandări ale celor ce au lucrat cu aplicația:

„...aplicația ar putea fi mai preferată de utilizatori în cazul în care comenzile prin tastatură ar fi înlocuite cu butoane evidente la mouse-over”,

„...pe lângă ce face acum, o opțiune în care trecerea să se facă pe culorile spectrului adică din imaginea următoare apare mai întâi tot ce e verde (de ex.) apoi galben, etc.”

„...necesită o interfață mai user-friendly și o compatibilitate mai mare (firefox, etc.)”

„Posibilitate de embedding la aplicații consacrate, gen Proshow Gold, sau PicturesToEXE. Înregistrare a prezentării.”

„1. Accelerare OpenGL sau Direct3D 2. Suport multimonitor 3. Simularea artefactelor inerente tehnicilor analogice (eventual preprocesare a fotografiilor, suport pentru filtre DX sau VST pentru coloana sonoră)”

Cea mai motivantă sugestie afirmă că “Este o aplicație ușor de folosit, cu potențial semnificativ în cazul în care este dezvoltată.”

7.5 Concluzii

Testarea și evaluare produselor/uneltelor nou dezvoltate este un proces cu o importanță vitală în stabilirea fiabilității soluțiilor propuse pentru grupul țintă căruiia îi sunt destinate, un proces necesar și obligatoriu înainte de darea în folosință a acestora. În cazul concret al evaluării aplicației dezvoltate de mine, trebuie ținut cont de faptul că este vorba despre o inițiativă interdisciplinară – IT, educație, multimedia. Evaluarea aplicației am realizat-o utilizând platforma Zef.

Testarea aplicației Motorway a fost realizată de către șaisprezece recenzori cu domenii de activitate diferite, de la IT la educație, de la comunicații la fotografie și sănătate, de la ONG la social, de la asigurări la multimedia și pagini Web. Pentru o eventuală informație suplimentară am împărșit grupul în două, fiecare subgrupă având caracteristici specifice, relativ la raportarea la prezentări audio-vizuale. Concluziile la care au ajuns evaluatorii în urma testărilor au fost colectate prin intermediul unui chestionar implementat utilizând platforma Zef. Pe baza analizării răspunsurilor date de aceștia, unele dintre ele reprezentate prin intermediul unor grafice generate de Zef, am putut trage următoarele concluzii: atât graficele bidimensionale utilizate pentru a testa ușurința de utilizare și utilitatea, cât și cele ce arată calitatea și viteza de răspuns pentru diverse situații, au colectat toate răspunsurile în cadranul 1 (peste valorile medii), indicând faptul că aplicația este ușor de utilizat și utilă, că oferă o soluție calitativă în ce privește o prezentare audio-vizuală interactivă. Desigur, tot graficele arată că ar mai putea fi îmbunătățite unele aspecte pe viitor

Ca o concluzie generală, pot afirma că studiile de evaluare și testare a aplicației propuse și implementate de mine relevă potențialul acesteia, atât într-un mediu academic cât și într-unul artistic, educațional, comercial, publicitar, etc. Aplicația poate primi îmbunătățiri și dezvoltări, ceea ce ar putea duce la o creștere considerabilă a atractivității și importanței sale, în multiple planuri.

8. Concluzii, contribuții și perspective

Teza de doctorat este dedicată unui domeniu aparent de nișă în România, respectiv diaporama digitală. Aparent, deoarece publicul nostru tinde să asocieze proiecțiile de diaporamă cu acele spectacole analogice ilustrând de cele mai multe ori cu o muzică plăcută diapozitive realizate în concediu. Evoluția digitală a democratizat însă atât fotografia cât și schimbul de informație, ceea ce a dus și la expansiunea prezentărilor audiovizuale. Pe de o parte, PowerPoint a devenit principala unealtă destinată prezentărilor de acest gen, dar diaporamistii consacrați au acum parte de câteva aplicații foarte specializate care îi ajută în munca lor.

Folosirea PowerPoint a devenit o condiție sine-qua-non a prezentărilor academice și științifice; în mod normal suportul vizual al oricărei teze de doctorat trebuie să fie de acest tip. ProShow Gold, Pictures to EXE și Wings Platinum conduc piața aplicațiilor specializate, ceea ce a dus la renașterea mișcării diaporamiste europene, cu consecințe benefice și în România.

Dezavantajul acestor aplicații este că niciuna nu oferă un control interactiv al prezentărilor. În ciuda faptului că permit aranjarea imaginilor în ordinea dorită, a tranzițiilor dintre imagini la durată aleasă, sincronizarea cu o coloană sonoră dedicată și exportul în diverse formate convenabile ca dimensiuni și calitate, aceste aplicații nu facilitează controlul în timp real al tranzițiilor dintre imagini, așa cum se putea face cu ușurință în perioada diaporamei analogice.

Această carență neacoperită de aplicațiile digitale m-a inspirat și motivat să găsesc o soluție practică de simulare a sistemelor analogice de prezentare a unei diaporame, însă folosind mijloacele digitale contemporane.

Activitatea desfășurată în cadrul tezei de doctorat a presupus analiza stadiului actual al aplicațiilor specifice, precum și studiul tehnic al dispozitivelor analogice consacrate, pentru a găsi o soluție digitală a unei probleme ce își are originile în analogic. Folosind diverse platforme, am conceput și finalizat trei aplicații care permit controlul deplin, în timp real, al unei diaporame. Pe prima aplicație o consider, acum, doar o etapă premergătoare a studiului, întrucât rezultatele calitative nu s-au ridicat la înălțimea așteptărilor. Experiența acumulată și feedback-ul primit au constituit baza de la care am pornit la elaborarea celei de a doua aplicații. Calitatea deosebită a tranzițiilor și deplina interactivitate au reprezentat punctele sale forte. Dificultățile în instalare și dorința personală de a oferi un produs finit la înălțimea așteptărilor m-au determinat să nu las studiul în acea etapă, ci să folosesc o nouă abordare. Rezultatul final, aplicația pe care am numit-o „Motorway”, este produsul pe care îl găsesc cel mai apropiat de conceptul dorit de la care am pornit. Rezultatele primite de la evaluatori au confirmat acest lucru.

Fiabilitatea metodelor implementate a fost evaluată prin testarea acestora din mai multe perspective: a ușurinței de utilizare și a utilității, a calității tranzițiilor în diverse situații ce implică o putere mai mică sau mai mare de calcul, a eficienței mouse-ului, ca și controller al prezentării. Am constatat un entuziasm crescut al evaluatorilor față de aplicație. Cel mai încurajator lucru mi s-a părut faptul că deși aceștia au fost împărțiți pe două grupe, una cu motivație artistică și cealaltă, cu educațională, răspunsul amândurora a fost pozitiv. Aceasta a arătat că diaporama rămâne un excepțional mijloc artistic de prezentare audiovizuală, dar și că utilizarea sa în cadrul procesului didactic poate oferi satisfacții mari atât pentru prezentator cât și pentru grupul de receptori ai prezentării.

Pe durata desfășurării cercetării de doctorat am adus mai multe contribuții proprii la dezvoltarea domeniului. Cele pe care le consider mai importante sunt descrise în paragraful următor.

8.1 Contribuții teoretice

(1) Sinteza și analiza critică a diaporamei analogice și digitale

Am realizat un studiu prin care am urmărit o istorie a diaporamei. Am conturat atât aspectele tehnice specifice, cât și anumite considerente artistice, care stau la baza oricărei prezentări de gen chiar și în zilele noastre. Am utilizat pentru aceasta atât literatura de specialitate și informațiile existente pe site-urile diaporamistilor consacrați, cât și vasta experiență pe care o am în domeniu, începând din 1986. Studiul posedă o puternică tușă personală, bazată pe apartenența la școala timișoreană de diaporamă. Acesta începe de la dispozitivele mecanice manufacturate care permiteau un control al contopirii imaginilor, continuând cu diaproiectoarele și dispozitivele de control cu care am lucrat personal. Am reliefat ulterior trecerea de la perioada analogică la cea digitală, cu avantajele și neajunsurile sale. Pe unul din ele am încercat să îl înlătur prin studiul meu.

Prezentarea pe larg a contribuției se găsește în capitolele 1, 2 și 3.

(2) Identificarea și studiul principalelor aplicații destinate prezentărilor audiovizuale

În studiul menționat mai sus am prezentat atât aspectele analogice cât și pe cele digitale. Cele din urmă beneficiază de o analiză a celor mai importante aplicații. Selecția lor s-a bazat pe datele obținute de la organizatorii principalelor festivaluri internaționale din Europa, organizate cu precădere în Anglia, Franța, Belgia, Polonia. Am folosit de asemenea, informații de la organizatorii Festivalurilor Internaționale de Diaporamă de la Suceava, respectiv Gura Humorului. Din datele primite, a reieșit clar predilecția diaporamistilor pentru aplicațiile ProShow Gold, Pictures to EXE și Wings Platinum. Am realizat un studiu al acestora, evidențiind calitățile, lipsurile, asemănările și diferențierile dintre acestea. M-am asigurat că nicio aplicație nu a folosit ideea mea, prezentată mai jos.

Prezentarea pe larg a contribuției se găsește în capitolul 3.

(3) Ideea controlului unei diaporame digitale în timp real pornind de la metodele analogice

Mixarea unor piese muzicale în timp real folosind dispozitive digitale este azi o normalitate. Există dispozitive de editare liniară video care permit mixarea imaginilor video; câteva aplicații pentru tranziții nelinare au apărut deja pe piață [100] [101]. Cu toate acestea, consider că ideea unui dispozitiv digital care să simuleze controlul total al unei prezentări de tip diaporamă, emulând dispozitivele analogice similare, îmi aparține.

Prezentarea pe larg a contribuției se regăsește în întreaga teză, și constituie pilonul central al acesteia.

(4) Ideea utilizării tastaturii pe post de controller al prezentării

Ideea de a simula digital dispozitivele specifice diaporamei analogice mi-a venit utilizând PowerPoint, o aplicație foarte utilă, extrem de răspândită, care permite o interactivitate acceptabilă între diaporamist și prezentarea sa. Lipsa posibilității de control atât a duratei tranziției cât și a coloanei sonore mi-a dat ideea-pilon a acestei teze. Studiind aplicațiile consacrate pentru diaporamă am constatat că nici ele nu suplinesc această lipsă. Emularea dispozitivelor de comandă analogice ar fi putut fi realizată prin construirea unui modul hardware extern, însă aceasta ar fi implicat pe lângă costurile mari și lipsa de accesibilitate a aplicației. Folosirea tastaturii deja existente în locul unui dispozitiv de comandă extern simplifică foarte mult lucrurile. Desigur, toate aplicațiile care rulează pe un computer se bazează pe utilizarea tastaturii. Cu toate acestea, consider că ideea folosirii tastaturii numerice ca echivalent temporal al controllerului digital al tranziției, îmi aparține.

Prezentarea pe larg a contribuției se găsește în subcapitolele 4.1.14, 5.3.2 și 6.2.3.

(5) Ideea utilizării mouse-ului pe post de controller al tranziției

Inițial, am avut doar ideea folosirii tastaturii numerice pe post de controller al tranzițiilor; fiecărei taste numerice trebuia să îi corespundă echivalentul în secunde al duratei tranziției. Însă deoarece telecomanda analogică pe fir a fost instrumentul care a permis adevăratul control de finețe al tranziției, m-am gândit la construirea unui dispozitiv similar, digital. Acesta urma să se conecteze la computer prin intermediul unui port USB, și ar fi avut nevoie de un kit de instalare care să îi permită să comunice cu aplicația propriu-zisă. Necesitatea unui astfel de dispozitiv specializat ar fi complicat mult lucrurile și ar fi făcut aplicația inaccesibilă altor utilizatori. Ideea folosirii mouse-ului pe post de telecomandă mi-a venit în data de 14.04.2007.

Teza mea de doctorat cuprinde trei aplicații care urmăresc interactivitatea unui spectacol de diaporamă; prima și a treia dintre ele includ și folosirea mouse-ului.

Prezentarea pe larg a contribuției se găsește în subcapitolele 4.1.16 și 6.2.3.2.

(6) Identificarea și analiza unor unelte software capabile să susțină conceperea și implementarea aplicațiilor personale

Am studiat literatura de specialitate și o serie de articole publicate începând cu anul 2000 și am identificat principalele unelte software cu ajutorul cărora să pot construi o aplicație capabilă să comande în timp real o prezentare interactivă de diaporamă. Am avut constant un feedback al muncii mele, astfel încât atunci când nu am avut parte de rezultatul dorit, am folosit noi unelte. Nu m-am rezumat la un studiu singular care să genereze o unică aplicație ci ținând cont de neajunsurile apărute, am reluat construirea aplicației folosind –parțial sau total– unelte noi.

Rezultatul final se regăsește în cea de a treia aplicație, despre care consider că soluționează întru totul cerințele aceste teze.

Prezentarea pe larg a contribuției se găsește în capitolele 4,5 și 6.

8.2 Contribuții aplicative

(1) Propunerea, conceperea și implementarea aplicației „Digital Diaporama”

Aplicația a apărut ca o primă etapă a studiului. Folosind C#, am conceput o aplicație capabilă să încarce o listă de imagini, una de fișiere audio, pe care să le utilizeze ulterior într-o prezentare interactivă de diaporamă. Aceasta permite controlul tranzițiilor și al coloanei sonore, folosind tastatura, și un control separat al tranzițiilor, folosind mouse-ul. A fost prima aplicație capabilă de control interactiv [54]. Dezavantajele observate m-au condus către elaborarea celei de-a doua aplicații.

Prezentarea pe larg a contribuției se găsește în capitolul 4.

(2) Propunerea, conceperea și implementarea aplicației „PhotoSlide Toolbar”

Aplicația a apărut ca o necesitate de a soluționa problemele apărute la aplicația precedentă. Am folosit atât C#, Visual Studio, cât și SpicIE, pentru a concepe un plugin pentru Internet Explorer, capabil să ofere un control al tranzițiilor și al coloanei sonore ce alcătuiesc o diaporamă. Imaginile ce alcătuiesc diaporama pot fi stocate fie pe mediul personal, fie încărcate direct pe o pagină web. Comparativ cu „Digital Diaporama”, această aplicație a înlăturat neajunsurile, și a oferit o soluție reală și calitativă [102] [103].

Prezentarea pe larg a contribuției se găsește în capitolul 5.

(3) Propunerea, conceperea și implementarea aplicației „Motorway”

Deoarece „PhotoSlide Toolbar” a avut și neajunsuri, legate de instalare, am conceput a treia aplicație, care înlobează experiența dobândită pe parcursul studiului [104]. Am folosit HTML, JavaScript, CSS și Internet Explorer, pentru a pune la punct ceea ce consider că reprezintă o soluție viabilă și aplicabilă universal pe PC, pentru a prezenta o diaporamă la modul interactiv. Controlul se realizează atât folosind tastatura cât și mouse-ul, iar aplicația nu necesită instalare. Singura necesitate impusă o reprezintă existența browser-ului Internet Explorer instalat pe PC. „Motorway” a reprezentat etapa finală a studiului, gata pentru a fi oferită celor interesați.

Prezentarea pe larg a contribuției se găsește în capitolul 6.

8.3 Direcții de cercetare pentru viitor

Consider că nu există perfecțiune absolută; ar fi împotriva ideii benefice de progres și evoluție. Conștientizarea posibilităților de îmbunătățire este primul pas în împlinirea lor. Desigur că nici acest studiu nu se abate de la crezul meu; chiar acum,

pe ultimele pagini ale tezei, idei de îmbunătățire încep să apară. Ele sunt legate de comparația cu aplicațiile deja consacrate; teoretic, cercetarea mea ar trebui să se orienteze în direcția ajungerii din urmă a acestora.

Concret, am identificat câteva direcții prin care aplicația „Motorway” ar putea evolua. Prima ar fi, așa cum am fost sfătuit deja, conceperea unei interfețe mai prietenoase, care să permită încărcarea imaginilor și a coloanei sonore printr-un browser intern. Am în vedere chiar alcătuirea unui playlist, care să conțină mai multe prezentări; s-ar putea trece de la un playlist la altul, fără a mai reporni aplicația.

A doua direcție de cercetare s-ar putea orienta pe înregistrarea prezentării, așa cum Simda, Imatronic, Kodak PDC, Basgen-Stumfl ofereau odinioară. Avantajul pentru diaporamist ar fi enorm, pentru că în loc să își aranjeze imaginile și tranzițiile în timeline, sincronizându-le cu coloana sonoră, nu ar avea altceva de făcut decât să „interpreteze” diaporama, cu simț artistic și cu trăire interioară. Produsul final, deși lipsit de interactivitate, ar purta în el tușa personală a proiecționistului. La capătul procesului, acest produs final ar putea rula ca un executabil, sau ca un fișier .avi, în format HD, ceea ce l-ar face accesibil noilor medii de prezentare.

Cu siguranță că în cadrul Universității „Politehnica” din Timișoara există resursele necesare continuării acestui studiu. Impunerea diaporamei în mediul didactic este pentru mine un vis cu potențial ridicat de realizare.

BIBLIOGRAFIE

- [1] Madier Claude, *Projections sonorisées et diaporamas*. Paris: Publications Photo-Cinema Paul Montel, 1971.
- [2] Iarovici Eugen, Agarici Paul, Dimitriu Mihai, Oradan Adrian, Achitei Gheorghe, *Audiovizual si diaporama* Editura Tehnica Bucuresti 1993.
- [3] Morozan Dumitru, *Realizarea si exploatarea diapozitivelor*. Bucuresti: Editura Tehnica, 1980.
- [4] Perea Joaquín, "Audiovisuales basados en la diapositiva: El diaporama y la multivisión," *UNIVERSO FOTOGRÁFICO*, vol. Nr 4, pp. 129-157, 2001.
- [5] Iarovici Eugen, "Fotografia si lumea de azi ". Bucuresti Editura Tehnica 1989.
- [6] Palazón Alfonso, "Diaporama: percepción audiovisual," *UNIVERSO FOTOGRÁFICO* vol. N°4, pp. 21-43, 2001.
- [7] microsoft.com, Microsoft Office PowerPoint (2010), [Online], Adresa: <http://office.microsoft.com/en-us/powerpoint/default.aspx>.
- [8] internet4classrooms.com, "PowerPoint Tutorials," 2009.
- [9] Photodex Corporation, ProShow 4.1 (2010), [Online], Adresa: <http://photodex.com/>.
- [10] ReponsesPhoto, "WnSoft PictureToExe," France, 2008, p. 129.
- [11] Stumpfl Reinhold, Wings Platinum 4 HDAV (2010), [Online], Adresa: http://www.avstumpfl.com/en/products/hdav/wings_platinum_4_hdav/overview, martie 2010.
- [12] Aigrain Philippe, Joly Philippe, "The automatic real-time analysis of film editing and transition effects and its applications," *Computers & Graphics* vol. 18, pp. 93-103, 1994.
- [13] Randy-Davis Stephen, Sphar Chuck, *C# 2005 for Dummies*, For Dummies; Bk&CD-Rom edition ed., 2005.
- [14] Troelsen Andrew, *C# and the .NET Platform*, Second Edition ed. Berkeley, CA: Apress, 2003.
- [15] Microsoft.com, Visual Studio (2008), [Online], Adresa: <http://msdn.microsoft.com/en-us/vstudio/default.aspx>, april 2008.
- [16] microsoft.com, SpicIE - Simple Plug-In Creator for Internet Explorer (2010), [Online], Adresa: <http://code.msdn.microsoft.com/SpicIE/Wiki/View.aspx>, april 2008.
- [17] Castro Elizabeth, *HTML, XHTML, and CSS*, Sixth Edition ed.: Peachpit Press, 2006.
- [18] Zakas Nicholas, *Professional JavaScript for Web Developers*, Second Edition ed.: Wrox, 2009.
- [19] Bleahu Marcian, *Omul și Peștera*. București: Editura Sport - Turism, 1978.
- [20] Ovidiu Drimba, *Istoria Culturii și Civilizației*, vol. I-II-III. București: Saeculum.
- [21] Venturini Lionello, *Cum sa intelegem pictura*. București: Meridiane, 1978.
- [22] Lefevbre Bertrand, *La plus ancienne photographie du monde*. Fécamp, FRANCE: L. Durand et fils, 1984.

- [23] Stapf Helmut, *Practica fotografică*. Bucuresti: Editura Tehnica, 1958.
- [24] *The Polaroid Book*: Taschen GmbH, 2005.
- [25] Newhall Beaumont, *The history of photography: from 1839 to the present day*. London, England Secker & Warburg, 1972.
- [26] Shimaura Arthur, "Muybridge in Motion: Travels in Art, Psychology and Neurology," *History of Photography*, vol. 26, 2002.
- [27] Gernsheim Helmuth, Gernsheim Alison, *The history of photography: from the camera obscura to the beginning of the modern era*. New York, USA McGraw-Hill 1969.
- [28] Fielding Raymond, *A Technological History of Motion Pictures and Television*. Berkeley, Los Angeles, CA: University of California Press, 1979.
- [29] Marcu Florin, *Marele dicționar de neologisme*. București: Saeculum, 2008.
- [30] Marcu Florin, Măneacă Constant, *Dicționar de neologisme*. București: Academiei, 1986.
- [31] Iarovici Eugen, "Fotografia, limbaj specific ". Bucuresti: Editura Meridiane, 1971.
- [32] Ward Peter, *Digital video camerawork*, 2000.
- [33] Simpson Robert, *Planificación y producción de programas con diapositivas*: Ed. Kodak, 1978.
- [34] Răduleț Toma, *Optica fotocinematografica(vol.1+2)*. Bucuresti: Editura Tehnica, 1977.
- [35] Moreno Pons Huelva Margarita, "Las diapositivas como recurso didactico: Diaporama en el aula," *Revista Cientifica de Comunicacion y Education*, pp. 95-100, 2003.
- [36] Jones Steve, "The cassette underground " *Popular Music and Society*, vol. 14, pp. 75-84, 1990.
- [37] Chiu Patrick, Kapuskar Ashutosh, "NoteLook: taking notes in meetings with digital video and ink," in *International Multimedia Conference Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, Orlando, Florida, United States 1999 pp. 149 - 158
- [38] Harman Doug, *Ghid de fotografie digitală*. Iași: Polirom, 2009.
- [39] Nikon, COOLPIX S1000pj (2010), [Online], Adresa: http://nikon.ro/ro_RO/products/product_details.page?ParamValue=Digital%20Cameras&Subnav1Param=COOLPIX&Subnav2Param=Style&Subnav3Param=0&RunQuery=13&ID=1850.
- [40] Guidicelli Maurice Gigou Laure, Le Site du Diaporama Créatif (2010), [Online], Adresa: <http://sitedudccn.com/>, aprilie 2010.
- [41] "Universitatea_DUNAREA_DE_JOS", GHID DE UTILIZARE POWER POINT (10/03/2008), [Online], Adresa: <http://www.idd.ugal.ro/Tutoriate/msppt.htm>.
- [42] Bartscha Robert, Cobernb Kristi, "Effectiveness of PowerPoint presentations in lectures," *Computers & Education*, vol. 41, pp. 77–86, 2003.
- [43] Craig Russell, Joel Amernic, "PowerPoint Presentation Technology and the Dynamics of Teaching " *Innovative Higher Education*, pp. 147-160, 2006.

- [44] West Philip, Foster Greg, Clayton Peter, "Content Exposure of Slide Show Presentations for Selective Download and Annotation via Mobile Devices," Department of Computer Science Rhodes University Grahamstown, South Africa, 2004, p. 8.
- [45] WNSoft, PicturesToExe Deluxe 6.0 (2010), [Online], Adresa: <http://www.wnsoft.com/>.
- [46] Pizolatto Jean-Charles, Réaliser des diaporamas numériques avec PicturesToExe. Version 6.5 (2009), [Online], Adresa: http://www.wnsoft.com/apr/help/6.5/PicturesToExe_User_Guide_v65_French.pdf, Ianuarie 2010.
- [47] ThePhotographer, "The picture show program," U.S.A., 2008, pp. 25-26.
- [48] Ondina Patricia, *Diaporama numerique*: Editions Générales First, 2008.
- [49] Grotta Daniel, Grotta-Wiener Sally, Make Gorgeous Slideshows with ProShow Gold (2008), [Online], Adresa: http://www.pcworld.com/article/169131/make_gorgeous_slideshows_with_pro_show_gold.html, noiembrie 2008.
- [50] Gordon Mah Ung, "ProShow Gold 3.0 The gold standard just got better," Future US, Inc., 2007. Available: <http://www.maximumpc.com>.
- [51] Jefferson Graham, Right software changes shutterbug to moviemaker (2006), [Online], Adresa: http://www.usatoday.com/tech/products/2006-08-23-personal-tech_x.htm, ianuarie 2010.
- [52] Fotoklub Zamek, Szczecin, Entered work (2009), [Online], Adresa: <http://www.fotoklub.zamek.szczecin.pl/index.php/diaporama/xxx-pomorskie-spotkania-z-diaporam-szczecin-2009/entered-work--zgoszone-prace.html>.
- [53] TrophéeDeParis, Concours International de diaporamas numériques. Oeuvres preseleccionnees (2010), [Online], Adresa: <http://site.voila.fr/tropheedeparis/pageoeuvresselect.html>, 2010.
- [54] Cristian Țecu, Radu Vasii, Cristian Goțiu, "Contributions to the use of the new computer technologies in the digital slideshows," in *IADIS International Conference Computer Graphics and Visualiyation 2008*, Amsterdam, 2008, pp. 311-314.
- [55] Gary B. Shelly, Corinne Hoisington, *Visual Basic 2008 for Windows and Mobile Applications: Introductory*: Cengage Learning, 2009.
- [56] Frankel Justin, *Mp3 Power! With Winamp* ebrandedbooks.com, 1999.
- [57] Williams Ryan, *Windows XP Digital Music For Dummies: For Dummies*, 2004.
- [58] Rathbone Andy, *MP3 For Dummies: For Dummies*, 2001.
- [59] Schulte Wolfram Barnett Mike, *Contracts, Components, and their Runtime Verification on the .NET Platform* Redmond WA, 98052-6399, USA, 2002.
- [60] Hejlsberg Anders , Wiltamuth Scott , Golde Peter, *C# Language Specification*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2003.
- [61] Archer Tom, *Inside C#*. Redmond, WA, USA: Microsoft Press, 2001.

- [62] Liberty Jesse, *Programming C#*, 3 edition ed. Sebastopol, CA, USA: O'Reilly & Associates, Inc, 2003.
- [63] Martin James, *Rapid Application Development*: Macmillan Coll Div, 1991.
- [64] Dumitriu-Lupan Nuşa, Tătăran Mihai, Programarea Orientată pe Obiecte și Programarea Vizuală cu C# .Net (2008), [Online], Adresa: <http://www.scribd.com/doc/517796/Manual-C>.
- [65] Soulié Juan, C++ Language Tutorial (2009), [Online], Adresa: <http://www.cplusplus.com/doc/tutorial/>, ianuarie 2010.
- [66] Petzold Charles, *Programming Windows with C# (Core Reference)*. Redmond, WA, USA: Microsoft Press, 2001.
- [67] Long Ben, *Real World Nikon Capture NX*: Peachpit Press, 2007.
- [68] Long Ben, *Complete Digital Photography*, 5 edition ed.: Charles River Media, 2009.
- [69] Kelby Scott, *The Adobe Photoshop CS3 Book for Digital Photographers*: New Riders Press, 2007.
- [70] microsoft.com, Microsoft .NET (2010), [Online], Adresa: <http://www.microsoft.com/net/overview.aspx>, march 2010.
- [71] Yaron Nir, COM Interface Basics (2003), [Online], Adresa: <http://www.codeproject.com/KB/COM/COMBasics.aspx>, aprilie 2008.
- [72] Gary B. Shelly, Thomas J. Cashman, Steven M. Freund, *Windows Internet Explorer 7: Introductory Concepts and Techniques*: Course Technology, 2007.
- [73] LeBlanc David Howard Michael, *Writing Secure Code*, Second Edition ed., 2002.
- [74] Jackson Paul, spicIE: Writing IE7 and IE8 Plugins in Managed Code (2009), [Online], Adresa: <http://www.lovethe.net/2009/03/spicie-writing-ie-7-and-ie-8-plugins-in.html>, aprilie 2009.
- [75] Michael Rees, "Evolving the browser towards a standard user interface architecture," in *The Third Australasian conference on User interfaces*, Melbourne, Victoria, Australia 2002, vol. 7, pp. 1-7.
- [76] Walker Mark, Making Microsoft Internet Explorer 7 Your Own (2006), [Online], march 2008.
- [77] Ion Mihaella, Teoria cromatică a culorilor (2010), [Online], Adresa: <http://www.revista-atelierul.ro/2010/04/26/teoria-cromatica-a-culorilor/>, aprilie 2010.
- [78] Philippe Aigrain, Philippe Joly, "The automatic real-time analysis of film editing and transition effects and its applications," *Computers & Graphics*, vol. 18, pp. 93-103, 1994.
- [79] Țecu Cristian , Popescu Adrian , VasIU Radu, "PhotoSlide Toolbar: Using the Internet browser for managing real-time digital slideshow," in *IADIS INTERNATIONAL CONFERENCE e-Society 2010*, Porto, 2010, pp. 503 - 506.
- [80] Coles Peter, Balls Colin, "Motorway" by Peter Coles and Colin Balls (1988), [Online], Adresa: <http://www.source.ie/video/motorway.html>, octombrie 2009.

- [81] Kodak, Kodak S-AV Programmable Dissolve Control (1990), [Online], Adresa: http://slideprojector.kodak.com/plugins/acrobat/S-AV%20PDC_english.pdf, april 2010.
- [82] SIMDA-5V, Simda Slide Projectors (2007), [Online], Adresa: http://www.simda.com/simda08/slide_projectors.php, september 2009.
- [83] Diaporama-AV-Group, Essential Notes for Audio-Visual Workers (2008), [Online], Adresa: <http://www.muirshiel.com/diaporama/Essential%20Notes.pdf>, ianuarie 2010.
- [84] Pinard Katherine, *Internet Explorer 8, Illustrated Essentials* Course Technology, 2009.
- [85] Gary B. Shelly, Steven M. Freund, *Windows Internet Explorer 8: Introductory Concepts and Techniques*: Course Technology, 2009.
- [86] Flanagan David, *JavaScript: The Definitive Guide: Activate Your Web Pages* O'Reilly Media, 2010.
- [87] Thomas Powell, Fritz Schneider, Nancy Maragioglio, *JavaScript: The Complete Reference*, Second Edition ed. New York, NY, USA: McGraw-Hill, Inc, 2004.
- [88] Håkon Wium Lie, Bert Bos, *Cascading Style Sheets*: Pearson Addison Wesley, 2004.
- [89] J. Badros Greg, Borning Alan, Marriott Kim, Stuckey Peter, "Constraint cascading style sheets for the Web," in *Proceedings of the 12th annual ACM symposium on User interface software and technology*, Asheville, North Carolina, United States, 1999.
- [90] Hart David, "Toward greater artistic control for interactive evolution of images and animation," in *ACM SIGGRAPH 2006 Sketches*, Boston, Massachusetts, 2006.
- [91] Mayer Matthias, "Web History Tools and Revisitation Support: A Survey of Existing Approaches and Directions," *Found. Trends Hum.-Comput. Interact.*, vol. 2, pp. 173-278, 2009.
- [92] McEvoy Seth, *Microsoft; Windows Media Player for Windows; XP Handbook*: Microsoft Press, 2001.
- [93] Peter H. Rossi, Mark W. Lipsey, Howard E. Freeman, *Evaluation: A Systematic Approach*, 7 ed.: Sage, 2004.
- [94] Michael Quinn Patton, *Qualitative research and evaluation methods*, 3 ed.: SAGE, 2002.
- [95] Ermalai Iasmina-Leila, "Contributii la utilizarea noilor tehnologii informationale in procesul de eLearning," PhD thesis, Universitatea Politehnica Timișoara, 2009.
- [96] Ralph W. Tyler, George F. Madaus, Daniel L. Stufflebeam, *Educational evaluation: classic works of Ralph W. Tyler*: Springer, 1989.
- [97] Ernest R. House, *New Directions in Educational Evaluation*: Taylor & Francis, 1986.
- [98] Zef, Solutions (2009), [Online], Adresa: <http://www.zefsolutions.com/en/solutions.html>.

- [99] Tzecu Cristian, momente.ro (2010), [Online], Adresa: <http://www.momente.ro/galerie.php>, martie 2010.
- [100] AvTake, CutFour DV 3 (2010), [Online], Adresa: <http://www.avtake.com/cutfour.asp>, april 2010.
- [101] Matrox, Matrox RT.X2 - Realtime transitions (2010), [Online], Adresa: <http://www.matrox.com/video/en/products/pc/rtx2/cpu/#transitions>.
- [102] Cristian Țecu, Radu VasIU, Cristian Goțiu, "PhotoSlide Toolbar – A Real Time Controller for Live Digital Slideshow Transitions " *The 4th International Scientific Conference "eLearning and Software for Education"*, pp. 371-375, 2008.
- [103] Cristian Țecu, Adrian Popescu, VasIU Radu, "PhotoSlide Toolbar: Using the Internet browser for managing real-time digital slideshow," in *IADIS INTERNATIONAL CONFERENCE e-Society 2010*, Porto, 2010, pp. 503 - 506.
- [104] Cristian Țecu, "Digital Slideshow Performed Live Using the MOTORWAY Application," in *IADIS International Conference "Applied Computing 2010"*, Timisoara, 2010.