

CONTRIBUȚII PRIVIND EVALUAREA MULTICRITERIALĂ A TEHNOLOGIILOR SAP

Teză destinată obținerii
titlului științific de doctor inginer
la
Universitatea "Politehnica" din Timișoara
în domeniul Automatică
de către

Ing. Adela Diana BERDIE

Conducător științific: prof.univ.dr.ing. Octavian PROȘTEAN
Referenți științifici: acad.prof.univ.dr.ing. Florin FILIP
prof.univ.dr.ing. Șerban AGACHI
prof.univ.dr.ing. Ștefan HOLBAN

Ziua susținerii tezei: 17.01.2014

Seriile Teze de doctorat ale UPT sunt:

- | | |
|---|--|
| 1. Automatică | 9. Inginerie Mecanică |
| 2. Chimie | 10. Știința Calculatoarelor |
| 3. Energetică | 11. Știința și Ingineria Materialelor |
| 4. Ingineria Chimică | 12. Ingineria sistemelor |
| 5. Inginerie Civilă | 13. Inginerie energetică |
| 6. Inginerie Electrică | 14. Calculatoare și tehnologia informației |
| 7. Inginerie Electronică și Telecomunicații | 15. Ingineria materialelor |
| 8. Inginerie Industrială | 16. Inginerie și Management |

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, 2013

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 baza cercetărilor efectuate în cadrul stagiului de practică ERASMUS din anul 2011, la firma SP LION SA, Heddesheim din Germania și cuprinde o serie de studii critice cu privire la stadiul actual al tehnologiilor UI integrate pe platforma NetWeaver a SAP. Aplicațiile implementate în tehnologiile Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI, precum și analiza rezultatelor obținute, au facilitat realizarea unei metodologii în vederea evaluării și ierarhizării acestora.

Prezenta lucrare de doctorat este rodul pasiunii mele pentru tehnologiile SAP, dar, în cea mai mare parte, și a faptului că am colaborat în acest domeniu cu colega mea șef lucr.dr. Mihaela OSACI, care mi-a împărtășit din bogata ei experiență didactică, mi-a acordat sprijinul de la începutul carierei didactice până în prezent și căreia îi mulțumesc din suflet.

O contribuție majoră în abordarea acestui domeniu a avut-o domnul dipl.dr.ing.Reinhold HAIPL, Senior Consultant la BASF SE, Frankfurt am Main, Germania, căruia îi prezint sincerele mele omagii.

Mulțumirile mele se adresează și fostei mele colega dr.ing.Ana Daniela CRISTEA (căș. HAMMES) de la firma Cellent AG, Stuttgart, Germania, alături de care am căpătat experiență în acest domeniu, publicând o serie de articole împreună.

Pentru sprijinul acordat de-a lungul întregii perioade de realizare a lucrării, pentru îndrumările competente, recomandările făcute cu înalt profesionalism, adresez, cu deosebit respect și considerație, cele mai sincere mulțumiri domnului **prof. dr. ing. Octavian PROȘTEAN**, în calitate de conducător științific.

Pentru studierea atentă, pentru observațiile interesante și utile efectuate asupra lucrării, sunt recunoscătoare și aduc sincere mulțumiri domnului prof.dr.ing Ștefan HOLBAN.

Tot pe această cale, mulțumesc conducerii firmei SP LION din Heddesheim, Germania, pentru sprijinul acordat pe întreaga perioadă de desfășurare a practicii. În mod special, mulțumesc domnului Ullrich GELLERT, manager-ul firmei, pentru acceptul de a desfășura activități de cercetare la cel mai înalt nivel, pentru disponibilitatea și amabilitatea de care a dat dovadă în timpul cercetărilor experimentale.

Totodată, țin să mulțumesc domnilor ing.Horia BOCHIȘ, ing.Johann Eduard KELLER și ing. Dragoș FLORESCU, consultanți SAP din cadrul companiei SAP AG Walldorf, Germania, pentru sfaturile și sugestiile acordate pe parcursul realizării cercetărilor experimentale.

De asemenea, mulțumirile mele se îndreaptă și către cadrele didactice de la Facultatea de Inginerie din Hunedoara, în special domnului decan conf.dr.ing. Caius PĂNOIU, d-lui conf.dr.Simion JITIAN și d-lui prof.dr.Ștefan MAKSAY care au sprijinit, direct sau indirect, realizarea acestei teze de doctorat.

În elaborarea tezei, esențial pentru mine a fost sprijinul moral și material al familiei mele: părinții, soțul și copiii, cărora le mulțumesc din suflet.

În mod special, doresc să dedic această lucrare de doctorat memoriei unei persoane dragi mie, d-lui prof.dr.ing.ANTON SAIMAC, care și-a dorit să mă număr printre doctoranzii săi, dar care, din păcate, a trecut în neființă înainte de înscrierea mea la doctorat.

Timișoara, decembrie 2013

Cu aleasă considerație,
Adela Diana BERDIE

Berdie, Adela Diana

Contribuții privind evaluarea multicriterială a tehnologiilor SAP

Teze de doctorat ale UPT, Seria 1, Nr. 18, Editura Politehnica, 2013, 168 pagini, 85 figuri, 43 tabele.

ISSN: 1842-5208

ISBN: 978-606-554-753-7

Cuvinte cheie: tehnologii web, SAP, sisteme integrate, programare ABAP, analiză multicriterială.

Rezumat,

Prin subiectul abordat, teza de doctorat răspunde unor probleme de maximă actualitate privind evaluarea tehnologiilor UI recomandate de SAP pentru dezvoltarea aplicațiilor web business pe platforma SAP NetWeaver.

Tema este tratată în mod interdisciplinar, îmbinând aspecte legate de platforma tehnologică SAP NetWeaver și tehnologiile ABAP: Web Dynpro, Floorplan Manager și CRM WebClient UI, cu elemente de prelucrare statistică a datelor, analiză corelațională, de regresie și multicriterială.

Rezultatele obținute în urma aplicării metodologiei de evaluare multicriterială propuse, conduc la stabilirea unei ierarhizări în vederea alegerii tehnologiei optime.

Cuprins

Abrevieri.....	7
Lista figurilor.....	9
Lista tabelelor.....	11
1. Introducere.....	13
1.1. Motivație.....	14
1.2. Obiectivele tezei.....	14
1.3. Structura și conținutul tezei.....	15
2. Tehnologii UI ale AS ABAP pe platforma SAP NetWeaver	17
2.1 Caracteristicile platformei SAP NetWeaver și ale server-ului de aplicații ABAP.....	17
2.2 Tehnologii ABAP UI de pe platforma Business Suite a SAP NetWeaver.....	23
2.2.1 Web Dynpro.....	23
2.2.1.1 Modelul architectural Web Dynpro ABAP.....	27
2.2.1.2 Modelul de programare Web Dynpro ABAP.....	29
2.2.2 Floorplan Manager.....	35
2.2.2.1 Modelul architectural Floorplan Manager.....	35
2.2.2.2 Modelul de programare Floorplan Manager.....	37
2.2.3 WebClient UI.....	40
2.2.3.1 Modelul architectural WebClient UI.....	42
2.2.3.2 Modelul de programare WebClient UI.....	45
2.3 Elemente de ansamblu specifice tehnologiilor WD ABAP, FPM și WebClient UI.....	48
2.4 Concluzii.....	50
3. Implementarea și testarea aplicațiilor web la nivelul SAP AS.....	51
3.1 Descrierea tipului de aplicație Web implementată în cele 3 tehnologii UI.....	51
3.2. Implementarea aplicației web în tehnologiile Web Dynpro ABAP, Floorplan Manager și WebClient UI.....	52
3.2.1 Implementarea aplicației în tehnologia Web Dynpro ABAP.....	53
3.2.2 Implementarea aplicației în tehnologia Floorplan Manager.....	56
3.2.3 Implementarea aplicației în tehnologia CRM WebClient UI	60
3.3 Analiza performanțelor aplicațiilor web business.....	61
3.4 Concluzii.....	65
4. Prelucrarea statistică a datelor experimentale.....	67
4.1 Analiza preliminară a datelor.....	67
4.1.1 Filtrarea erorilor.....	67
4.1.2 Determinarea estimației punctuale.....	70
4.1.3 Verificarea caracterului aleator al datelor – Testul Young.....	73
4.2 Funcții de distribuție.....	74
4.2.1 Funcția de distribuție și densitatea de probabilitate.....	74
4.2.2 Verificarea normalității datelor experimentale.....	74

4.2.2.1 Testul Kolmogorov-Smirnov (K-S).....	75
4.2.2.2 Testul Anderson - Darling. (A-D).....	76
4.2.2.3 Testul Hi pătrat (Chi-Squared).....	76
4.2.3 Identificarea distribuțiilor de probabilitate aferente datelor analizate și generarea de valori pe baza acestora.....	77
4.3 Compararea informațiilor statistice a datelor experimentale și a datelor generate.....	86
4.3.1 Informații statistice despre datele măsurate.....	86
4.3.2 Informații statistice despre datele generate.....	87
4.4 Concluzii.....	89
5. Analiza dependențelor dintre parametrii RT, CPU, DBRT și HTTPWT	90
5.1 Analiza datelor experimentale.....	90
5.1.1 Coeficienții de corelație dintre parametrii mășurați.....	90
5.1.2 Stabilirea corelațiilor analitice și grafice dintre parametrii mășurați.....	93
5.2 Analiza datelor generate.....	97
5.2.1 Coeficienții de corelație dintre parametrii generați.....	97
5.2.2 Stabilirea corelațiilor analitice și grafice dintre parametrii generați.....	99
5.3 Analiza comparativă a corelațiilor dintre parametrii.....	104
5.4 Concluzii.....	106
6. Metodologie de analiză și evaluare multicriterială a tehnologiilor ABAP UI	108
6.1 Metode de analiză multicriterială.....	108
6.1.1 Metoda utilității globale.....	112
6.1.2 Metoda AHP.....	113
6.1.3 Metoda TOPSIS.....	116
6.1.4 Metoda ELECTRE.....	118
6.2 Metodologie de evaluare și ierarhizare a tehnologiilor ABAP UI.....	119
6.2.1. Etapele metodologiei procesului de evaluare.....	119
6.2.2 Modelul MEIAHP.....	123
6.2.3 Algoritmul AITOP.....	128
6.2.4 Algoritmul AIEL.....	132
6.3 Concluzii.....	136
7. Concluzii, contribuții personale și dezvoltări viitoare	138
7.1 Concluzii finale.....	138
7.2 Contribuții personale.....	140
7.3 Dezvoltări de viitor.....	142
Bibliografie.....	143
Anexe.....	152
Lista lucrărilor științifice publicate.....	165

Abrevieri

ABAP	- Advanced Business Application Programming
AC	- Application Controller
ACC	- Application Configuration Controller
AET	- Application Enhancement Tool
AHP	- Analitical Hierarhy Process
AIEL	- Algoritm Ierarhizare - ELECTRE
AITOP	- Algoritm Ierarhizare - TOPSIS
ALV	- ABAP List Viewer
ARIA	- Accesibile Rich Internet Application
AS	- Application Server
BAPI	- Business Application Programming Interface
BOL	- Business Object Layer
BSP	- Business Server Page
CDF	- Cumulative Distribution Function
CHIP	- Collaborative Human Interface Part
CPU	- Central Processing Unit
CPUT	- CPU Time
CRM	- Customer Relationship Management
CTS	- Change and Transport Sistem
DB	- Database
DBRT	- DB Request Time
DDIC	- Data Dictionary
DE	- Efort de dezvoltare
ECC	- ERP Central Component
ECDF	- Funcția de distribuție cumulativă empirică
EE	- Efort de extindere
EM	- Efort de mentenanță
ERP	- Enterprise Resource Planning
FLUID	- Flexible User Interface Designer
FPM	- Floorplan Manager
GAF	- Guided Activity Floorplan
GenIL	- Generic Interaction Layer
GEV	- Gen. Extreme Value (distribuție statistică)
GUIBB	- Generic User Interface Building Block
HTML	- Hypertext Markup Language
HTTP	- Hypertext Transfer Protocol
HTTPS	- Hypertext Transfer Protocol Secure
HTTPWT	- HTTP-Watch Time
ICM	- Internet Communication Manager
IDE	- Integrated Development Environment
ITS	- Internet Transaction Server
JS	- JavaScript
JSP	- Java Server Script
MVC	- Model View Controller
OIF	- Object Instance Floorplan

OLTP	- Online Transaction Processing
OO	- Object Oriented
OTR	- Online Text Repository
OVP	- Overview Page Floorplan
PDF	- Probability Density Function
POC	- Point-Of-Care
QAF	- Quick Activity Floorplan
RFC	- Remote Function Call
RIA	- Rich Internet Application
RT	- Response Time
SAP	- Systems Applications and Products in Data Processing
SCM	- Supply Chain Management
SMTP	- Simple Mail Transfer Protocol
SO	- Select Option Component
SOA	- Service Oriented Architecture
SOAP	- Simple Object Access Protocol
SQL	- Structured Query Language
ST	- Simple Transformation
STAD	- Business Transaction Analysis
UDDI	- Universal Description, Discovery and Integration
UI	- User Interface
UIBB	- User Interface Building Block
UIF	- User Interface Framework
VCI	- Valoare Critică Inferioară (testul Young)
VCS	- Valoare Critică Superioară (testul Young)
WCUI	- WebClient UI
WD	- Web Dynpro
WDSL	- Web Service Description Language
WP	- Work Processes
XHTML	- EXtensible HyperText Markup Language
XML	- Extensible Markup Language
XSLT	- Extensible Stylesheet Language Transformation

Lista figurilor

- Fig. 2.1 Platforma SAP NetWeaver[16]
- Fig. 2.2 SAP NetWeaver Application Server [27]
- Fig. 2.3 Arhitectura SAP Web Application Server ABAP [28]
- Fig. 2.4 Implementarea MVC în cadrul framework-ului Web Dynpro [5]
- Fig. 2.5 Metamodelul aplicațiilor Web Dynpro [47]
- Fig. 2.6 Arhitectura componentei Web Dynpro [56]
- Fig. 2.7 Structura internă a unui View într-o componentă Web Dynpro [38]
- Fig. 2.8 Componenta Web Dynpro [38]
- Fig. 2.9 Floorplan Manager în SAP Business Suite [69]
- Fig. 2.10 Modelul pe etape (Phase Model) Floorplan Manager [69]
- Fig. 2.11 Realizarea unei aplicații FPM [72]
- Fig. 2.12 MVC în arhitectura BSP [78]
- Fig. 2.13 Captură din pagina principală a aplicației CRM WebClient UI
- Fig. 2.14 Arhitectura CRM WebClient UI [79]
- Fig. 2.15 Nivelul de logică business (Business Layer) [79]
- Fig. 2.16 Conceptul MVC în CRM WebClient UI [79]
- Fig. 2.17 Componenta WebClient UI [82]
- Fig. 2.18 MVC în aplicații WebClient UI [82]
- Fig. 2.19 Conceptul de extindere la nivel de componentă [82]
- Fig. 3.1 Interfața principală a proiectului
- Fig. 3.2 Captură cu tabelele bazei de date SAP CRM corespunzătoare aplicației web
- Fig. 3.3 Interfața aplicației implementată în tehnologia Web Dynpro ABAP
- Fig. 3.4 Structura componentei Y_WD_BUCOMP
- Fig. 3.5 Structura ROOTUIELEMENTCONTAINER corespunzătoare Layout-ului V_SEARCH_TEXT
- Fig. 3.6 Metodele ComponentController
- Fig. 3.7 Integrarea Views în Windows
- Fig. 3.8 Maparea Context-ului din ComponentController în View
- Fig. 3.9 Aplicația WD FPM ZADBUPA
- Fig. 3.10 Configurarea componentei ZADBUPA_SEARCH
- Fig. 3.11 Implementarea interfețelor în clasa Feeder
- Fig. 3.12 Diagrama de clase UML
- Fig. 3.13 Interfața aplicației implementată în tehnologia FPM
- Fig. 3.14 Aplicația BSP YBP_SEARCH
- Fig. 3.15 Secvență din codul metodei EH_ONSERACH
- Fig. 3.16 Analiza timpului în procesul de lucru, afișat prin tranzacția STAD
- Fig. 3.17 Distribuția timpilor la nivelul server-ului SAP AS [27].
- Fig. 3.18 Componentele Timpului de răspuns – Response Time [95]
- Fig. 3.19 Captură HTTP Watch pentru testarea aplicației implementate în CRM WebClient UI
- Fig. 4.1 Distribuții ale eșantionului de date RT - Web Dynpro ABAP
- Fig. 4.2 Distribuții ale eșantionului de date DBRT - Web Dynpro ABAP
- Fig. 4.3 Distribuții ale eșantionului de date CPUT - Web Dynpro ABAP
- Fig. 4.4 Distribuții ale eșantionului de date HTTPWT - Web Dynpro ABAP
- Fig. 4.5 Distribuții ale eșantionului de date RT - Floorplan Manager
- Fig. 4.6 Distribuții ale eșantionului de date DBRT - Floorplan Manager
- Fig. 4.7 Distribuții ale eșantionului de date CPUT - Floorplan Manager
- Fig. 4.8 Distribuții ale eșantionului de date HTTPWT - Floorplan Manager
- Fig. 4.9 Distribuții ale eșantionului de date RT - CRM WebClient UI

- Fig. 4.10 Distribuții ale eșantionului de date DBRT - CRM WebClient UI
- Fig. 4.11 Distribuții ale eșantionului de date CPUT - CRM WebClient UI
- Fig. 4.12 Distribuții ale eșantionului de date HTTPWT - CRM WebClient UI
- Fig. 4.13 Graficele PDF pentru RT și DBRT în Web Dynpro ABAP
- Fig. 4.14 Graficele PDF pentru CPUT și HTTPWT în Web Dynpro ABAP
- Fig. 4.15 Graficele PDF pentru RT și DBRT în Floorplan Manager
- Fig. 4.16 Graficele PDF pentru CPUT și HTTPWT în Floorplan Manager
- Fig. 4.17 Graficele PDF pentru RT și DBRT în CRM WebClient UI
- Fig. 4.18 Graficele PDF pentru CPUT și HTTPWT în CRM WebClient UI
- Fig. 5.1 Funcția de regresie liniară $RT=f(DBRT)$ pentru tehnologia Web Dynpro ABAP (WDA).
- Fig. 5.2 Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia Web Dynpro ABAP
- Fig. 5.3 Funcția de regresie $RT=f(CPUT)$ pentru tehnologia Floorplan Manager (FPM).
- Fig. 5.4 Funcția de regresie $RT=f(CPUT,DBRT)$ pentru tehnologia Floorplan Manager (FPM).
- Fig. 5.5 Funcția de regresie $RT=f(DBRT)$ pentru tehnologia CRM WebClient UI
- Fig. 5.6 Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia CRM WebClient UI
- Fig. 5.7 Funcția de regresie liniară $RT=f(DBRT)$ pentru tehnologia Web Dynpro ABAP (WDA 96 valori)
- Fig. 5.8 Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia Web Dynpro ABAP (WDA 96 valori)
- Fig. 5.9 Funcția de regresie $RT=f(CPUT)$ pentru tehnologia Floorplan Manager (FPM 99 valori)
- Fig. 5.10 Funcția de regresie $RT=f(CPUT,DBRT)$ pentru tehnologia Floorplan Manager (FPM 99 valori)
- Fig. 5.11 Funcția de regresie $RT=f(CPUT,HTTPWT)$ pentru tehnologia Floorplan Manager (FPM 99 valori)
- Fig. 5.12 Funcția de regresie $RT=f(DBRT)$ pentru tehnologia CRM WebClient UI (96 valori)
- Fig. 5.13 Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia CRM WebClient UI (96 valori)
- Fig. 5.14 Dependența parametrului dependent RT de ceilalți parametri (pentru datele măsurate)
- Fig. 5.15 Dependența parametrului dependent RT de ceilalți parametri (pentru datele generate)
- Fig. 6.1 Ierarhie AHP
- Fig. 6.2 Algoritmul metodei TOPSIS
- Fig. 6.3 Modelul ierarhic MEIAHP
- Fig. 6.4 Chestionarul comparării perechilor de criterii
- Fig. 6.5 Matricea decizională a criteriilor
- Fig. 6.6 Raportul de inconsistență al comparației criteriilor
- Fig. 6.7 Matricea normalizată a ponderilor (Super Matrice)
- Fig. 6.8 Matricea limită
- Fig. 6.9 Prioritățile elementelor modelului ierarhic
- Fig. 6.10 Ierarhizarea alternativelor
- Fig. 6.11 Analiza senzitivă în raport cu: a) DBRT și b) HTTPWT
- Fig. 6.12 Analiza senzitivă în raport cu: a) RT și HTTPWT; b) HTTPWT și DBRT
- Fig. 6.13 Algoritmul AITOP
- Fig. 6.14 Algoritmul AIEL

Lista tabelelor

Tabelul 2.1	Principalele deosebiri între tehnologiile Web Dynpro ABAP și Web Dynpro Java
Tabelul 2.2	Diferențele dintre tehnica componentizării faceless și tehnica componentizării cu clasă de asistență.
Tabelul 4.1	Filtrarea și eliminarea valorilor afectate de erori grosiere, din setul de date măsurate în tehnologia WDA
Tabelul 4.2	Filtrarea și eliminarea valorilor afectate de erori grosiere, din setul de date măsurate în tehnologia FPM
Tabelul 4.3	Filtrarea și eliminarea valorilor afectate de erori grosiere, din setul de date măsurate în tehnologia CRM WC UI
Tabelul 4.4	Parametrii statistici obținuți prin prelucrarea datelor experimentale din tehnologia Web Dynpro ABAP
Tabelul 4.5	Parametrii statistici obținuți prin prelucrarea datelor experimentale din tehnologia Floorplan Manager
Tabelul 4.6	Parametrii statistici obținuți prin prelucrarea datelor experimentale din tehnologia CRM WebClient UI
Tabelul 4.7	Valori critice inferioare și superioare
Tabelul 4.8	Rezultatele aplicării testului Young asupra eșantionului de date experimentale
Tabelul 4.9	Valorile D și p pentru cele mai bune distribuții identificate a valorilor măsurate
Tabelul 4.10	Ecuțiile funcțiilor de distribuție
Tabelul 4.11	Valorile parametrilor statistici D și p pentru seturile de date generate
Tabelul 4.12	Date statistice obținute pentru setul de măsurători aparținând aplicației realizate prin tehnologia Web Dynpro ABAP (11 valori)
Tabelul 4.13	Date statistice obținute pentru setul de măsurători aparținând aplicației realizate prin tehnologia Floorplan Manager
Tabelul 4.14	Date statistice obținute pentru setul de măsurători aparținând aplicației realizate prin tehnologia CRM WebClient UI
Tabelul 4.15	Date statistice obținute pentru setul de măsurători generate, utilizând tehnologia Web Dynpro ABAP (cu 96 valori pentru fiecare variabilă)
Tabelul 4.16	Date statistice obținute cu programul DataFit pentru setul de măsurători generate utilizând tehnologia Floorplan Manager (cu 99 valori pentru fiecare variabilă)
Tabelul 4.17	Date statistice obținute cu programul DataFit pentru setul de măsurători generate utilizând tehnologia Customer Relationship Management WebClient UI (cu 96 valori pentru fiecare variabilă)
Tabelul 4.18	Abateri standard ale parametrilor măsurați și generați
Tabelul 5.1	Matricea coeficienților de corelație pentru setul de date aparținând aplicației realizate prin tehnologia Web Dynpro ABAP.
Tabelul 5.2	Matricea coeficienților de corelație pentru setul de date aparținând

Tabelul 5.3	aplicației realizate prin tehnologia Floorplan Manager. Matricea coeficienților de corelație pentru setul de date aparținând aplicației realizate prin tehnologia Customer Relationship Management WebClient User Interface.
Tabelul 5.4	Matricea coeficienților de corelație pentru setul de date generate utilizând tehnologia Web Dynpro ABAP (cu 96 valori pentru fiecare variabilă)
Tabelul 5.5	Matricea coeficienților de corelație pentru setul de date generate utilizând tehnologia Floorplan Manager (cu 99 valori pentru fiecare variabilă)
Tabelul 5.6	Matricea coeficienților de corelație pentru setul de date generate utilizând tehnologia Customer Relationship Management WebClient User Interface (cu 96 valori pentru fiecare variabilă)
Tabelul 5.7	Funcțiile de regresie liniară și coeficienții de corelație corespunzători, pentru cele trei tehnologii utilizate la interogarea bazei de date SAP
Tabelul 5.8	Funcțiile de regresie multiplă și coeficienții de determinare corespunzători parametrilor mășurați, pentru cele trei tehnologii utilizate la interogarea bazei de date SAP
Tabelul 5.9	Dependența parametrului dependent RT de ceilalți parametri (în cazul datelor măsurate)
Tabelul 6.1	Modele ale procesului decizional [129]
Tabelul 6.2	Scala preferințelor relative Saaty 1980 [137]
Tabelul 6.3	Matricea decizională cu variantele decizionale și criteriile de decizie
Tabelul 6.4	Procedee de normalizare
Tabelul 6.5	Matricea normalizată și a ponderilor de importanță
Tabelul 6.6	Matricea normalizată ponderată
Tabelul 6.7	Valorile cele mai avantajoase și cele mai dajavantajoase ale criteriilor analizate
Tabelul 6.8	Valorile depărtării fiecărei alternative față de valoarea cea mai avantajoasă, respectiv cea mai dezavantajoasă și ierarhizarea variantelor decizionale față de soluția ideală
Tabelul 6.9	Matricea decizională cu variantele decizionale și criteriile de decizie
Tabelul 6.10	Matricea utilităților și a ponderilor de importanță.
Tabelul 6.11	Valorile utilității globale și ordinea de preferință pentru cele 3 tehnologii analizate
Tabelul 6.12	Matricea indicilor de concordanță și de discordanță
Tabelul 6.13	Matricea diferențelor dintre indicii de concordanță și cei de discordanță
Tabelul 6.14	Matricea surclasărilor

1. Introducere

În opinia autorului, teza de față prezintă cercetările referitoare la tehnologiile UI specifice sistemelor informatice integrate de tip SAP ERP, cu reliefarea rolului acestora în dezvoltarea de aplicații web business.

În teza de doctorat de față s-au integrat rezultatele practice obținute în timpul stagiului de practică ERASMUS, efectuat în anul 2011, la compania SP Lion din Heddeshheim Germania. Astfel, s-au implementat aplicații web business în trei tehnologii ABAP UI, pe platforma NetWeaver a SAP. Pe lângă analiza teoretică și măsurătorile efectuate, prezenta teză constituie și suport pentru evaluarea și ierarhizarea tehnologiilor SAP UI analizate.

SAP (Systems Applications and Products in Data Processing) sau Sistemul Aplicațiilor și Produselor [1], a fost printre primele companii dezvoltatoare de software pentru sistemele informaționale de întreprindere. Aplicațiile de pe platforma SAP permit nu numai îmbunătățirea relațiilor cu clienții și partenerii, ci și eficientizarea întregii activități. În decursul anilor, SAP s-a dezvoltat și restructurat continuu, portofoliul său de produse fiind de mai multe ori restructurat și redenumit [2]. În acest context, sistemul ERP a evoluat de la sistemele R/3 la SAP ERP și apoi la SAP ERP Central Component (ECC), iar în domeniul tehnologiilor de Internet, de la SAP ITS la SAP AS și mySAP, până la mySAP Business Suite, SAP NetWeaver și Xapplications [3].

Beneficiind de tehnologii web avansate, SAP a lansat inițial platforma SAP Netweaver 2004, care a însemnat introducerea de tehnologii noi și suport pentru Java și diferite scripturi. Versiunea următoare, SAP NetWeaver 2004s, a introdus tehnologia Web Dynpro pentru ABAP (limbajul nativ al sistemului SAP) [4], care împreună cu tehnologia Web Dynpro pentru Java [5] au făcut posibilă dezvoltarea de noi aplicații web business pe această platformă. Tehnologia Web Dynpro pentru ABAP a fost în continuare îmbunătățită, astfel încât, în prezent ea este utilizată în dezvoltarea aplicațiilor web pentru platforma Business Suite, platformă care are ca suport platforma SAP NetWeaver.

La baza fiecărui produs de pe platforma SAP NetWeaver (SAP ERP, SAP CRM, SAP SCM) stă serverul SAP de aplicații web AS [6], [7], a cărui arhitectură este orientată spre servicii (SOA). Diversitatea acestor produse și multitudinea de tehnologii prin care pot fi realizate aplicații pe platforma SAP NetWeaver, fac destul de dificilă luarea unei decizii în ceea ce privește alegerea uneia dintre ele.

În această teză sunt analizate și evaluate trei tehnologii UI pentru aplicații specifice căutărilor în baza de date, și anume: Web Dynpro, Floorplan Manager și WebClient UI. Tehnologiile analizate sunt recomandate de SAP pentru dezvoltatorii ABAP. Pe baza acestei analize s-au formulat criteriile, care au permis dezvoltarea unor algoritmi pentru ierarhizarea tehnologiilor amintite. Criteriile sunt definite atât pe considerente arhitecturale (efort de dezvoltare, de mentenanță și de extindere), cât și din punct de vedere al performanțelor aplicațiilor implementate în cele trei tehnologii (timp de răspuns, timp consumat de procesor, timp necesar modificărilor în baza de date, timp obținut în front-end).

Contribuțiile aduse de autor în această lucrare constituie o sinteză a rezultatelor lucrărilor publicate în acest domeniu.

1.1. Motivație

Prin experiența acumulată alături de colaboratorii SAP în cadrul stagiului de practică ERASMUS efectuat la compania SP Lion AG din Heddeshheim Germania, a cărei clienți sunt în general întreprinderile mici și mijlocii, în diferite situații de proiecte bazate pe tehnologii SAP, a devenit clar că pentru dezvoltatori, o mare incertitudine este aceea de a găsi soluția la întrebarea: *care tehnologie ar trebui folosită la implementare pentru ca proiectul să satisfacă cerințele clientului, să aibă rezultate cât mai optime din punct de vedere al timpului de execuție și să poată fi în viitor extins cu ușurință?*

Decizia însă, este de multe ori limitată datorită faptului că baza de clienți SAP este segmentată, fiecare având achiziționate anumite produse SAP.

Întrebările care apar în acest context ar fi:

- Care sunt tehnologiile și mediile de dezvoltare cele mai utilizate pentru aplicațiile web pe platforma NetWeaver a SAP?
- Care sunt elementele de ansamblu pe baza cărora se poate face o evaluare multicriterială a tehnologiilor SAP UI ?
- Ce caracteristici comune, specifice tehnologiilor SAP pot fi considerate criterii într-un algoritm pentru ierarhizarea acestora?
- Cum poate să fie cât mai aproape de adevăr rezultatul obținut în urma aplicării unei metodologii de evaluare multicriterială?

În acest context, am considerat necesar efectuarea unei cercetări în această direcție, teza fiind structurată pentru a răspunde acestor întrebări.

1.2. Obiectivele tezei

Obiectivul principal al acestei teze este de a dezvolta o metodologie pentru evaluarea și ierarhizarea tehnologiilor Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI. Metodologia de evaluare multicriterială propusă, se bazează pe definirea unor criterii și ponderi, necesare implementării algoritmilor de ierarhizare. Scopul acestei evaluări este nu doar de facilitare a procesului de decizie, ci și de consiliere eficientă a celor interesați.

Metodologia propusă are la bază:

- analiza comparativă a tehnologiilor ABAP UI dezvoltate pe platforma SAP NetWeaver;
- seturile de măsurători obținute la testarea aplicațiilor implementate în tehnologiile WD ABAP, FPM și CRM WebClient UI;
- analiza statistică și determinarea dependențelor dintre parametrii obținuți prin măsurători.

Etapele de evaluare multicriterială presupun:

- stabilirea alternativelor, seturilor de criterii și a ponderilor corespunzătoare acestora;
- stabilirea metodelor de analiză multicriterială;
- construirea algoritmilor pentru ierarhizarea tehnologiilor.

Seturile de criterii alese pentru realizarea modelelor ierarhice se bazează nu doar pe caracteristici software tehnice și arhitecturale, ci și pe indicatori de performanță ca: timp de răspuns, timp consumat de procesor, timp necesar modificărilor în baza de date și timp obținut în front-end. Rezultatele obținute în urma aplicării metodologiei de evaluare multicriterială și ierarhizare propuse, conduc la stabilirea tehnologiei optime.

Problematika tezei se adresează în special consultantilor și dezvoltatorilor de tehnologii SAP UI, însă și categoriilor de persoane care se ocupă în contextul proiectelor IT cu diverse tehnologii și doresc să își facă o imagine de ansamblu asupra terminologiei și posibilităților specifice SAP.

Lucrarea, în ansamblul său, poate fi de un real folos și pentru profesioniștii din domeniul IT care nu s-au ocupat de sisteme integrate și produse SAP și care, astfel, au posibilitatea obținerii unei perspective asupra platformei tehnologice de integrare și aplicații SAP NetWeaver.

Lucrarea de față nu pretinde să arate, sau să pună în discuție, toate formele posibile de implementare, sau toate soluțiile de scenarii pentru aplicațiile web, ci se limitează doar la capacitățile stivei tehnologice SAP ABAP, direcționându-se spre aplicațiile web de pe platforma NetWeaver.

1.3. Structura și conținutul tezei

Având în vedere obiectivele anterior prezentate, lucrarea are următoarea structură: introducere, cinci capitole specifice domeniului cercetat, concluzii, contribuții personale și referințe bibliografice.

În primul capitol sunt prezentate motivația, obiectivele principale ale lucrării și modul de structurare al acesteia, funcție de obiectivele și problematica abordată.

Capitolul 2 tratează aspecte legate de platforma tehnologică SAP NetWeaver versiunea 7.02 și server-ul de aplicații SAP AS, scoțând în evidență facilitățile oferite de acestea referitor la dezvoltarea și implementarea de aplicații web. De asemenea, este prezentată o analiză critică asupra tehnologiilor recomandate de SAP pentru dezvoltatorii de aplicații web de pe platforma Business Suite a NetWeaver: Web Dynpro ABAP, Floorplan Manager, WebClient UI și Web Dynpro Java. Tehnologiile sunt abordate atât din punct de vedere arhitectural, cât și din punct de vedere al modelului de programare.

În capitolul 3 sunt tratate aspecte legate de implementarea aplicațiilor web, la nivelul server-ului ABAP, în tehnologiile UI recomandate de SAP pentru platforma Business Suite a NetWeaver. În acest context, sunt evidențiate caracteristicile specifice tehnologiilor Web Dynpro ABAP, Floorplan Manager și WebClient UI, la implementarea aceluiași model de aplicație web. Aplicațiile implementate în cele trei tehnologii sunt testate cu ajutorul instrumentelor specializate: SAP Business Transaction Analysis și HTTP Watch pentru Internet Explorer, în vederea analizării performanțelor acestora. Testele presupun interogarea bazei de date cu afișarea rezultatelor pentru trei cazuri diferite.

În capitolul 4 este realizată prelucrarea și interpretarea statistică a datelor experimentale obținute la testarea aplicațiilor în cele trei tehnologii SAP UI (Web Dynpro ABAP, Floorplan Manager și WebClient UI) în capitolul 3. Prelucrarea statistică a datelor urmărește în primul rând eliminarea valorilor afectate de erori grosiere și verificarea caracterului aleator al acestora. Pentru determinarea corectă a legăturilor dintre parametrii obținuți prin măsurători, sunt generate noi seturi de date pe baza funcțiilor de distribuție identificate pentru fiecare parametru.

În capitolul 5 este efectuată o analiză corelațională și de regresie pentru seturile de date experimentale și generate. În baza acestei analize se obțin informații referitoare la corelațiile dintre parametrii datelor experimentale și dintre parametrii datelor generate. Se stabilesc funcțiile analitice și graficele de dependență dintre parametri, în vederea stabilirii parametrilor dependenți și independenți.

În capitolul 6, cele trei tehnologii SAP, WD ABAP, FPM și WebClient UI sunt evaluate, utilizând o serie de metode și instrumente de cercetare. Prima parte a acestui capitol prezintă o descriere sintetică a metodelor de analiză multicriterială utilizate pentru ierarhizarea tehnologiilor. Cea de-a doua parte este dedicată etapelor de realizare a metodologiei de evaluare, în care sunt formulate soluțiile alternative, sunt definite seturile de criterii și implementați algoritmi metodelor multicriteriale. Ponderile de importanță aferente criteriilor sunt obținute cu ajutorul metodei AHP, în cadrul unui model ierarhic implementat, utilizând instrumentul Super Decision. Algoritmii de ierarhizare propuși, au la bază metodele de analiză multicriterială: TOPSIS, ELECTRE și MUG.

În finalul lucrării sunt prezentate concluziile, contribuțiile personale și posibile direcții de cercetare, în opinia autorului.

Teza se întinde pe 168 pagini și conține 85 figuri, 43 tabele, 9 anexe și 167 titluri bibliografice. O mare parte din contribuții a fost validată prin publicarea a 27 lucrări științifice în domeniul tezei.

2. Tehnologii UI ale AS ABAP pe platforma SAP NetWeaver

Capitolul de față este structurat în două părți. Prima parte tratează aspecte legate de platforma tehnologică SAP NetWeaver versiunea 7.0 și server-ul de aplicații SAP AS, scoțând în evidență facilitățile oferite de acestea referitor la dezvoltarea și implementarea de aplicații web.

Cea de-a doua parte are la bază o analiză critică asupra tehnologiilor Web Dynpro ABAP, Floorplan Manager și WebClient UI, recomandate de SAP pentru dezvoltatorii de aplicații ABAP de pe platforma Business Suite a NetWeaver. Tehnologiile sunt abordate atât din punct de vedere arhitectural, cât și din punct de vedere al modelului de programare.

Scopul acestui capitol este de a evidenția caracteristicile fiecărei tehnologii în parte, astfel încât acestea să poată fi evaluate și ierarhizate.

2.1. Caracteristicile platformei SAP NetWeaver și ale server-ului de aplicații ABAP

SAP NetWeaver [8], [9], [10] reprezintă platforma completă de integrare și aplicații bazată pe servicii web, care realizează „integrarea globală” atât a tehnologiilor, cât și a proceselor de afaceri în toate formele de organizare. În prezent, platforma reprezintă fundamentul tehnologic pentru aproape toate aplicațiile SAP din SAP Business Suite [11], soluțiile partenerilor SAP și aplicațiile dezvoltate de clienți, asigurând totodată și suportul pentru Enterprise Services Architecture [12], [13], conceptul SAP pentru soluții orientate către procese de afaceri și bazate pe standarde Web service.

SAP NetWeaver este prima platformă încrucișată de aplicații (cross-application platform), cu o arhitectură de sistem deschis, orientată nu numai spre dezvoltarea aplicațiilor SAP, dar și spre integrarea informațiilor și proceselor care utilizează tehnologii diverse precum Java, IBM WebSphere și Microsoft NET [14], [15].

Componentele care intră în alcătuirea platformei SAP NetWeaver, figura 2.1 [16], pot fi împărțite în două categorii [17]: componente de integrare ce aparțin celor patru nivele ale platformei și instrumente de dezvoltare și management.

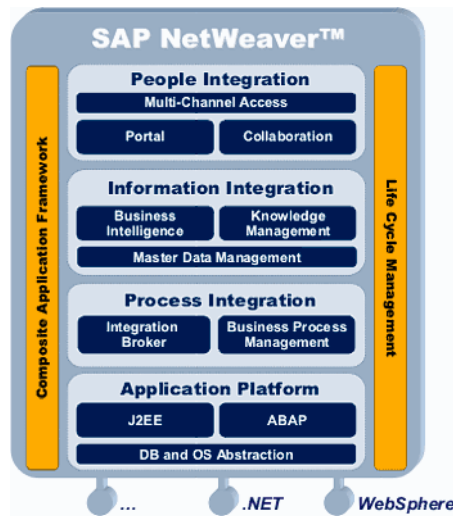


Fig. 2.1. Platforma SAP NetWeaver [16]

Cele trei componente de integrare (People Integration, Information Integration și Process Integration) ce au ca fundament platforma de aplicații (Application Platform) sunt independente, însă colaborează între ele și se dezvoltă împreună [18], [19].

Platforma de aplicații **Application Platform** are la bază SAP Application Server (SAP AS) și este fundamentul pentru toate componentele amintite anterior. Platforma oferă o infrastructură complexă, care face posibilă dezvoltarea și implementarea aplicațiilor ABAP și/sau Java pentru stocarea și procesarea datelor. Suportă J2EE (Java 2 Enterprise Edition Platform) standard complet și, începând cu versiunea 6.10, server-ul de aplicații este dezvoltat și pentru aplicații Web. Limbajul nativ al sistemului, ABAP, este utilizat pentru realizarea de aplicații care au ca mediu de dezvoltare ABAP Workbench, iar pentru realizarea de aplicații în limbaj Java, se folosește mediul SAP NetWeaver Developer Studio [20], [21].

Pentru implementarea și testarea aplicațiilor web, această platformă oferă o serie de facilități printre care:

- tehnologie completă pentru gestionarea tuturor etapelor de realizare a aplicațiilor web, de la proiectare, dezvoltare, implementare, până la testare;
- suport pentru platforma SAP Business Suite, de unde se vor alege cele trei tehnologii UI: WD ABAP, Floorplan Manager și WebClient UI; cadrele de lucru pentru tehnologiile WD ABAP cu Floorplan Manager inclus și CRM WebClient UI au conținut integrat și preconfigurat pentru implementare rapidă;
- asigură integritatea datelor în cadrul unor infrastructuri IT eterogene, oferind totodată servicii pentru consolidarea și gestionarea centrală a datelor. MaxDB oferă o bază de date formată din tabele standard relaționate, o parte dintre tabele furnizând datele pentru aplicațiile propuse;
- o varietate de unelte și tehnici care ajută în procesul de programare: suport MVC necesar realizării componentelor web ce stau la baza aplicațiilor, suport pentru programarea orientată pe obiect pentru realizarea claselor,

instrumente pentru crearea și modelarea logică a bazelor de date – Data Dictionary și Data Modeler [22], [23];

- instrumente pentru testarea performanțelor aplicațiilor (STAD, SE30, Runtime Analysis) [24].

Server-ul de aplicații SAP AS reprezintă componenta de bază a platformei SAP NetWeaver, cu ajutorul căruia pot fi dezvoltate aplicații atât în limbajul de programare ABAP, cât și în Java. Cele două sisteme ABAP și Java pot exista simultan, însă pot să ruleze și independent unul de celălalt. Existența simultană a acestora presupune însă o conexiune între cele două. Această abordare oferă dezvoltatorilor și utilizatorilor o singură infrastructură cu avantaje din ambele părți [25],[26].

SAP AS are la bază tehnologia clasică client-server, iar nucleului său SAP kernel, este extins cu un proces ICM (Internet Communication Manager), figura 2.2, care face posibilă comunicarea directă cu Internet-ul sau Intranet-ul prin intermediul unui browser.

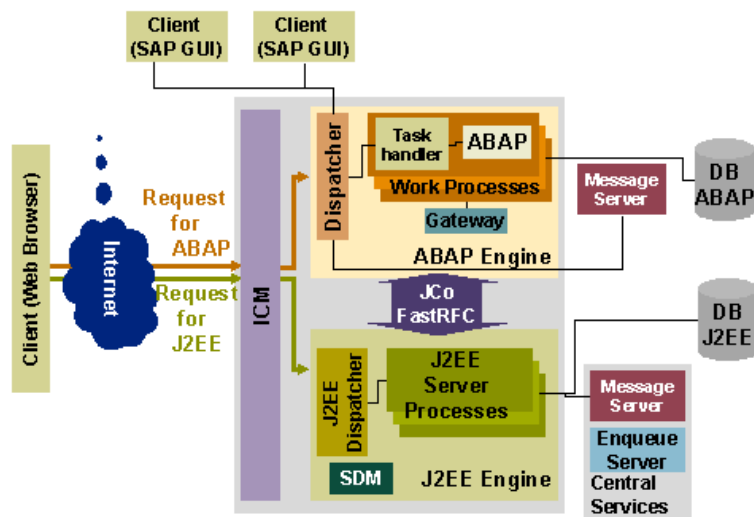


Fig. 2.2. SAP NetWeaver Application Server [27]

Întrucât implementarea aplicațiilor pentru test va avea loc la nivelul server-ului de aplicații ABAP, se vor descrie în continuare arhitectura și funcționalitățile acestuia.

Ca și **arhitectură**, SAP AS ABAP este format din trei nivele, figura 2.3 [28]:

- nivelul de persistență sau nivelul în care sunt păstrate datele în baze de date;
- nivelul de aplicații sau nivelul logic în care sunt executate programele;
- nivelul de prezentare care răspunde de tot ceea ce înseamnă prezentarea datelor și primirea intrărilor utilizatorilor.

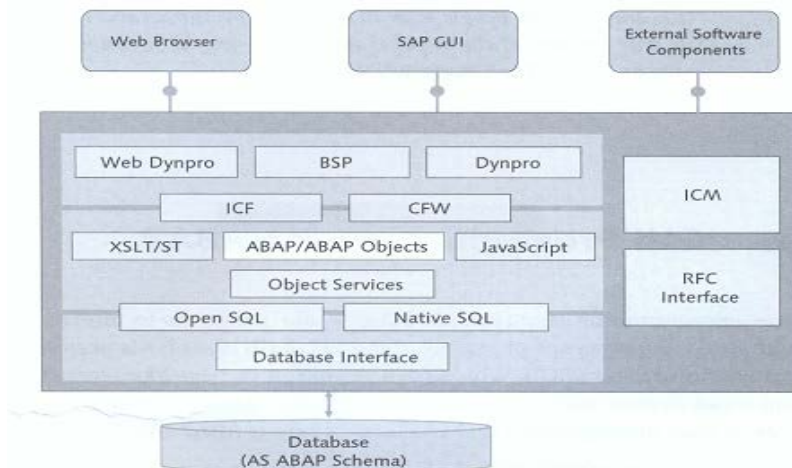


Fig. 2.3 Arhitectura SAP Application Server ABAP [28]

a) La *nivel de persistență* datele sunt păstrate într-o bază de date centrală care poate fi accesată prin interfața ei (Database Interface) cu ajutorul limbajelor Open SQL și Native SQL. Această interfață asigură dezvoltare independentă a procesului logic față de baza de date și sistemul de operare. Open SQL [29], [30] este alcătuit din declarații ABAP, fiind un subset al limbajului SQL (Structured Query Language), iar Native SQL constă din instrucțiuni care sunt transmise direct sistemului de baze de date. Întrucât există mai multe structuri de tabele la nivelul server-ului, pentru a putea fi realizate aplicațiile de test în toate cele trei tehnologii, se alege o structură de tabele comună acestora. Vor fi alese trei dintre cele mai importante tabele ale platformei Business Suite și anume tabelele standard din CRM: BUT000, BUT020 și ADRC.

b) *Nivelul de aplicații AS ABAP* are la bază un nucleu scris în C/C++ care servește ca platformă pentru programele de aplicații și care furnizează procesorii (mașinile virtuale) pentru următoarele limbaje de programare: ABAP cu ABAP Objects, XSLT (Extensible Stylesheet Language Transformation) și ST (Simple Transformation), limbaj specific pentru transformarea datelor în format XML și JavaScript (JS). Pe lângă acestea, SAP AS suportă standarde ca SOAP, WDSL și UDDI pentru dezvoltare și acces la servicii web care să facă legătura cu diferite aplicații SAP și non-SAP [31].

La acest nivel, vor fi implementate în limbajul nativ al sistemului, aplicațiile de test atât în Web Dynpro ABAP, cât și în Floorplan Manager și CRM Web Client UI. Cu ajutorul mediului ABAP Workbench și a framework-urilor WD ABAP și WebClient UI puse la dispoziție de către sistem, vor fi construite componentele web de bază, view-urile corespunzătoare interfețelor, iar la nivel de logică se va implementa codul necesar transferului rezultatelor către utilizator.

ABAP este un limbaj de programare interpretat, prelucrarea codului fiind asigurată de un cod intermediar care se execută într-o mașină virtuală. Câteva dintre caracteristicile sale ar fi [32]:

- conține elemente de limbaj stocate în biblioteci;

- acces integrat la baza de date prin Open SQL;
- tabele interne pentru stocarea dinamică și procesarea datelor în memorie;
- conceptul OLTP (online transaction processing) integrat în mediul de execuție ABAP, care permite accesarea simultană a bazei de date;
- interfețe integrate pentru alte limbaje de programare prin RFC;
- interfață integrată pentru XML;
- administrarea și accesul la obiectele partajate din memoria sistemului;
- administrarea și accesul la obiectele persistente din baza de date;
- accesul la Internet;
- accesul la interfețele utilizator;
- limbaj multilingual (elementele specifice de limbaj fiind separate de codurile sursă sunt încărcate în acord cu limba curentă în momentul în care programul este executat);
- suportă ambele sisteme Unicode sau non-Unicode.

c) La *nivel de prezentare*, AS ABAP poate fi folosit de utilizatori sau de aplicații software. Prin intermediul browser-ului, interfețele utilizator pot fi accesate direct de pe desktop-ul calculatorului. Componentele software pot avea acces doar prin protocolul Remote Function Call (RFC), conectarea la Internet realizându-se prin Internet Communication Manager (ICM) utilizând protocoale ca HTTP/HTTPS/SMTP. Interfața RFC este o interfață clasică, ce face legătura între diferite AS ABAP sau între AS ABAP și un sistem extern. SAP NetWeaver AS [33] poate funcționa ca și client web, dar și ca server web.

Ca și server web, el are sarcina de a procesa solicitările individuale HTTP de la diferiți clienți web (cum ar fi Microsoft Internet Explorer, Mozilla Firefox, etc) și a trimite înapoi răspunsul la client. SAP Web Dispatcher este cel care gestionează aceste cereri și în cazul unui sistem format din mai multe servere de aplicații se face redistribuirea lor. În rolul de client web, SAP NetWeaver AS, crează cereri HTTP (înglobate în programe ABAP) pe care le trimite la un server web. Prezentarea în web browser [34] se bazează pe tehnologia Web Dynpro ABAP și Business Server Page (BSP), similară cu Java Server Script (JSP). Pentru host-ul implicit, SAP furnizează un arbore HTTP service în care este inclus și service-ul BSP, utilizat pentru apelarea aplicațiilor BSP. La acest nivel de prezentare sunt procesate solicitările aplicațiilor web implementate în sistem, care utilizează ca și protocol HTTP.

Funcționalitatea aplicațiilor SAP, precum și extinderea lor cu soluții complementare, presupune înțelegerea a trei concepte cheie care stau la baza viziunii arhitecturale SAP și a abordării dezvoltării SAP Business Suite. Aceste concepte sunt [35]:

- păstrarea modelului fundamental al proiectării unei soluții;
- software componentizat pentru procesele de afaceri end-to-end;
- abordarea arhitecturală de soluții pentru rețele de afaceri.

Indiferent de platforma tehnologică utilizată pentru implementarea aplicațiilor web, există patru principii de bază pe care SAP le promovează și le respectă [35]. Acestea sunt:

- separarea părții logice de partea de aplicație;
- proiectarea aplicației astfel încât să satisfacă nevoile utilizatorului;

- armonizarea interfeței cu utilizatorul (coerență și ușurință la utilizare);
- realizarea unei interfețe accesibile pentru cât mai mulți utilizatori posibili.

Tehnologiile recomandate de SAP pentru dezvoltarea de interfețe UI actuale sunt [35]:

- Web Dynpro ABAP;
- Floorplan Manager;
- Web Dynpro Java;
- WebClient UI (dezvoltat pentru SAP CRM doar pentru limbajul ABAP).

Tehnologiile neagreate de SAP sunt [35]:

- Business Server Pages (BSP);
- XHTML sau HTML;
- ITS flow logic;
- ABAP Dynpro.

Toate tehnologiile recomandate de SAP stau la baza realizării soluțiilor de pe platforma Business Suite și se adresează mai multor domenii specifice cum ar fi: bănci, construcții, asistență medicală, asigurări, sector public. Dezvoltarea și implementarea aplicațiilor business pe această platformă, se poate realiza doar cu ajutorul server-ului de aplicații SAP AS de pe platforma SAP NetWeaver. Acest server, ce are la bază două stive tehnologice ABAP și JAVA, poate dezvolta aplicații în ambele limbaje de programare, alegerea uneia dintre ele depinzând de situația, condițiile și cerințele specifice proiectului, dar și ale companiei. Modelul de aplicații web ABAP presupune alegerea unei scheme de bază de date ce se află pe server-ul de aplicații ABAP, pe când modelul aplicațiilor Java necesită implicarea server-ului de aplicații Java, care dispune de o schemă de bază de date diferită.

2.2. Tehnologii ABAP UI de pe platforma Business Suite a SAP NetWeaver

2.2.1. Web Dynpro

Web Dynpro [36], [37] este tehnologia UI bazată pe web care utilizează server-ul de aplicații SAP ca backend. Există o versiune atât pentru server-ul de aplicații ABAP, cât și o versiune pentru server-ul de aplicații Java. Deși ambele versiuni oferă interfețe în mare parte identice, diferențele ce există se datorează implementărilor, care sunt complet diferite: aplicațiile WD ABAP sunt dezvoltate în mediul ABAP Workbench, iar aplicațiile WD Java sunt dezvoltate în mediul SAP NetWeaver Developer Studio, bazat pe Eclipse.

Interfețele dezvoltate în cele două medii pot fi realizate utilizând două tehnici de programare: declarativă (când structura interfeței este cunoscută înainte de execuție) și dinamică (când structura interfeței este parțial cunoscută în timpul execuției).

Tehnica declarativă [38], presupune legarea setului de controale stabilite și programarea funcționalității acestora, cu elementele care conțin datele aplicației, codul interfeței fiind automat generat de framework-ul respectiv.

Programarea dinamică [39], [40] este necesară în cazul în care interfața unei componente este bogată în elemente, manipularea lor vizuală devenind dificilă. Avantajul acestei tehnici este acela că asigură o manipulare mult mai ușoară a aplicației.

Atât în ABAP cât și în Java [41], procesul de proiectare și implementare al unei aplicații Web Dynpro are la bază paradigma Model View Controller [42], [43] care facilitează decuplarea părții logice de interfața aplicației, (figura 2.4). Funcționarea pe principiul MVC a framework-ului Web Dynpro [44], face ca întreaga structură să fie concepută din componente, separarea datelor de design concretizându-se prin intermediul obiectelor care generează date și a obiectelor care consumă date.

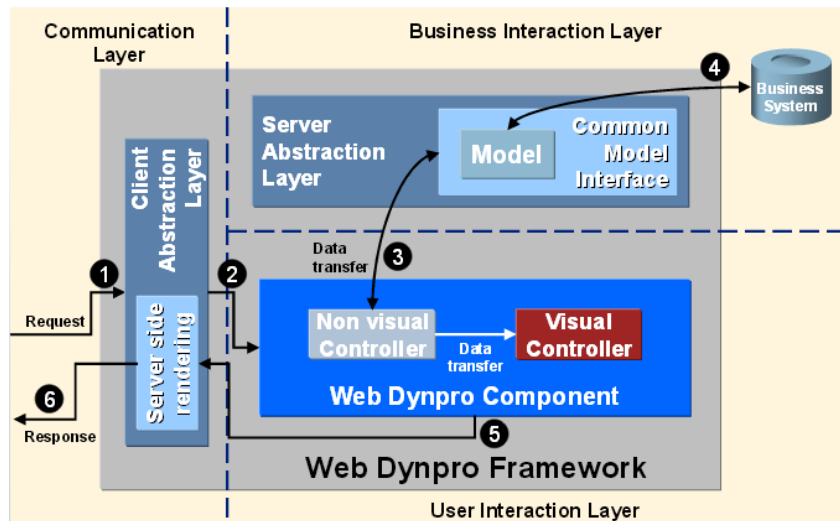


Fig. 2.4 Implementarea MVC în cadrul framework-ului Web Dynpro [5]

Având la bază arhitectura MVC, o aplicație WD ce rulează în front-end are acces la back-end-ul sistemului prin intermediul unui serviciu. Conectarea cu back-end-ul sistemului se poate realiza fie prin interfațe generate, utilizând RFC, fie prin interfețe ce apelează Web Service, fie prin interfețe proprii [45].

Deoarece ambele medii Web Dynpro dispun de instrumente speciale, aplicațiile web pot fi construite sub forma unui metamodel [46], (figura 2.5), structura principală a părții vizuale nedepinzând de limbajul de implementare.

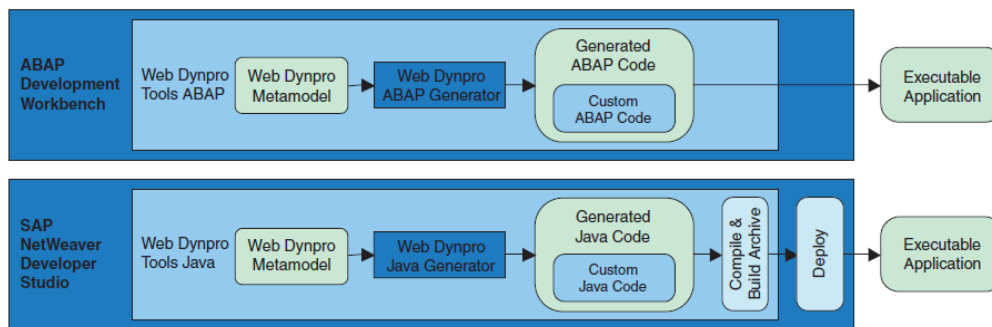


Fig. 2.5 Metamodelul aplicațiilor Web Dynpro [47]

În lucrările autorului [48], [49], [50], [51] au fost analizate principalele deosebiri și asemănări care există între tehnologiile Web Dynpro ABAP și Web Dynpro Java și care sunt sintetizate în tabelul 2.1.

Tab.2.1. Principalele deosebiri între tehnologiile Web Dynpro ABAP și Web Dynpro Java

Deosebiri	Web Dynpro ABAP	Web Dynpro Java
Construirea aplicațiilor	direct pe server-ul ABAP și stocate în baza de date	local, compilate, arhivate și apoi implementate pe mașina AS Java
Afișarea cadrelor în interfață	Este permisă afișarea unui singur view în fereastra dedicată interfeței; pentru mai multe view-uri este necesar un element UI ViewContainerUI Element	Elementul UI ViewSet face posibilă integrarea mai multor View-uri și aranjarea lor în fereastra interfeței
Instrumente suplimentare	Nu sunt disponibile instrumente grafice	Navigation Manager și Diagram View necesare mapeării între componente
Reutilizare de componente	Componentele se realizează separat și apoi se conectează prin usage	Prin containerele de tipul Development Component se poate realiza usage intern și extern de componente
Navigare	Plug-uri de intrare și ieșire; doar plug-urile de intrare pot fi parametrizate prin metode automat generate	Plug-uri de intrare și ieșire care pot fi și parametrizate prin metode automat generate
Context dinamic	Crearea dinamică de noduri și atribute nu oferă o anume distribuție	În cod se poate opta pentru diferite variante de noduri și atribute
Directorul MIME	Creat la cerere	Creat automat
Componente standard	Select Option Component (SO) ABAP List Viewer Component (ALV)	Componentele SO și ALV din ABAP nu există
Includerea structurilor în structuri	Admisă	Nu este admisă
Multilingualitate	Instrumentul OTR (Online Text Repository) cu traducerea textelor	Nu există OTR, traducerea se realizează separat pentru fișierele *.XLF. Externalizarea șirurilor de caractere face posibilă modificarea fișierelor originale
Exportul proiectului	Proiectul este stocat în baza de date pe server	Prin opțiunile meniului fișierele sunt stocate în computer
Mesajele	Clasa de mesaje cu texte salvate în tabelul T100, sau stocarea mesajelor în clasa de asistență	Se generează automat un container de mesaje (statice sau dinamice) la crearea componentei

Asemănări:

1. Ambele aplicații sunt bazate pe conceptul MVC în care view-ul este responsabil cu nivelul de prezentare sau interfața cu utilizatorul, model-ul cu nivelul logic sau procesarea datelor și controller-ul face comunicarea între ele.
2. Elementele principale ale unei componente sunt aceleași: controller, view și window.
3. Același principiu pentru stocarea datelor: noduri și atribute.
4. Același concept pentru transferul datelor: legarea elementelor UI de noduri și atribute.
5. Același concept folosit la aspectul realizării ecranelor de interfață (Matrix, Grid, Flow Layout).
6. Aceleași modalități pentru realizarea aplicațiilor: componentizare și reutilizarea componentelor.
7. Ambele medii dispun de un număr mare de elemente UI pe care le folosesc în comun.
8. Ambele medii dispun de metode Hook implicite (specifice, de ex: WDDOINIT – ca metodă constructor, WDDOBEFOACTION, WDDOEXIT- ca metodă destructor) și metode generate automat care pot fi programate. Acestea oferă programatorilor posibilitatea de a interveni în anumite situații (salvează datele, renunță, etc) pentru a concretiza o anumită acțiune.
9. Ambele medii oferă posibilitatea programării statice și dinamice.
10. Ambele aplicații pot fi integrate în portalul SAP NetWeaver.

Alegerea uneia dintre tehnologii (ABAP sau Java), depinde de situațiile specifice. Atunci când implementarea aplicațiilor web business necesită date de la server-ul de aplicații ABAP este bine să se aleagă WD ABAP și nu WD Java.

Pentru stocarea datelor (ABAP, Java) se dispune de două sisteme de baze de date diferite. Este posibil să se schimbe date între ele, dar acest lucru afectează în mod negativ durata de execuție. De aceea, această comunicare se face numai în situații speciale, iar în toate celelalte cazuri, se recomandă să se opteze pentru utilizarea tehnologiei Web Dynpro a server-ului de aplicații respectiv.

Deoarece în cercetarea de față este realizată compararea tehnologiilor la nivelul server-ului ABAP, în continuare va fi tratată tehnologia WD ABAP atât din punct de vedere arhitectural, cât și din punct de vedere al modelului de programare.

2.2.1.1. Modelul arhitectural Web Dynpro ABAP

Web Dynpro ABAP este tehnologia actuală de pe platforma tehnologică SAP NetWeaver pentru dezvoltarea de interfețe utilizator profesionale Business Suite. Așa cum s-a menționat [39], [52], [53], WD ABAP are la bază server-ul de aplicații web ABAP, propriul său mediu de execuție și proiectare, cu unelte speciale care formează un cadru de lucru - framework, ABAP Workbench.

Aplicațiile Web Dynpro pot fi văzute ca simple obiecte care conțin un set de informații și care sunt alcătuite din componente. Din aceste informații este generat un Business Server Page care servește ca și punct de plecare pentru execuția framework-ului și a aplicației [54]. Componentele aplicațiilor Web Dynpro, (figura 2.6), sunt organizate conform paradigmei MVC în seturi de Controller-e, View-uri și Window-uri [55], [56].

De partea logică a aplicației se ocupă modelele, care sunt responsabile nu numai cu acțiunile și operațiile asupra datelor, ci și cu integrarea diverselor clase ce permit procesarea informațiilor din baza de date.

Afișarea datelor aparține View-urilor, iar procesarea datelor aparține controller-elor, care reprezintă sistemul comunicativ și decizional ce face legătura între modele și view-uri, ele putând fi atât consumatoare, cât și generatoare de date.

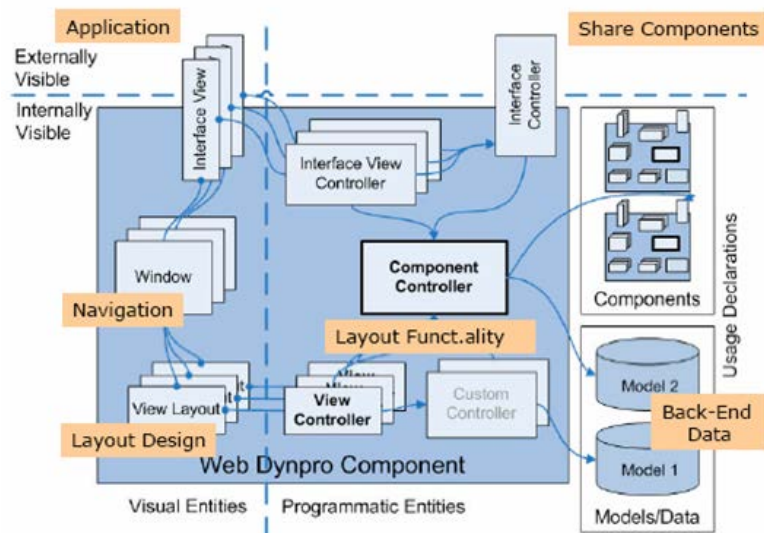


Fig. 2.6 Arhitectura componentei Web Dynpro [56]

Întrucât pot fi definite mai multe modele de date pentru o aplicație web, trebuie ca aceste modele să fie legate de componentele web care le utilizează.

Datele corespunzătoare modelelor pot proveni din surse diferite, de exemplu [27], [44]:

- date SAP provenind din back-end-ul sistemului SAP prin utilizarea de BAPI-uri;
- date proprii prin crearea de modele construite în dicționarul de date DDIC al sistemului SAP, sau folosind o structură de tabele realizată din baza de date proprie SAP ;
- date provenind din apelarea și folosirea Web Service;
- date provenind din combinarea celor trei variante anterioare;
- date externe importate.

La nivelul unui view, într-o componentă se deosebesc trei elemente distincte, care interacționează atât între ele cât și cu restul componentei: Controller-ul, Context-ul și un View(Layout), (figura 2.7).

Controller-ul este afișat doar ca un set de metode. Aceste metode pot fi de două feluri: metode implicite (Hook), care corespund framework-ului și sunt chemate doar pentru anumite momente (de exemplu la inițializare WDDOINIT) și metode Event Handler, folosite pentru procesarea unui eveniment și care pot fi formulate de către utilizator (de exemplu clic pe un buton), sau rezultate din definirea unei navigări.

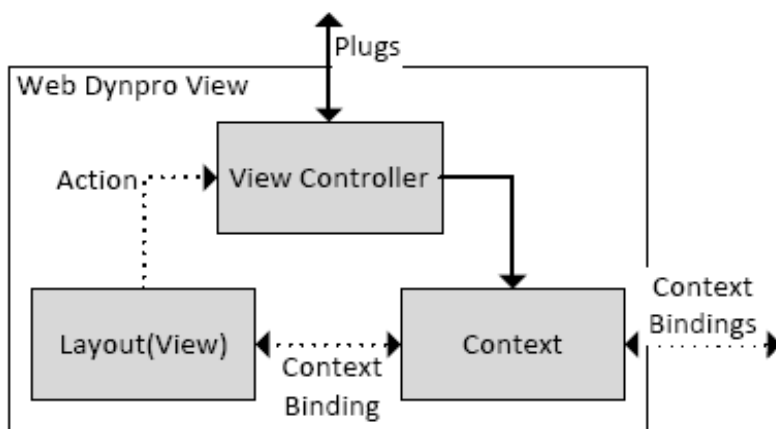


Fig. 2.7 Structura internă a unui View într-o componentă Web Dynpro [38]

Layout-ul este format din elemente UI (de exemplu: etichete, butoane, câmpuri) aranjate într-o structură arborescentă, a căror proprietăți sunt legate de un context corespunzător format din noduri și atribute.

Legătura între context și elementele UI este în ambele sensuri. Pe de o parte pot fi specificate date și variabile care să fie introduse, pe de altă parte, pot fi schimbate atribute care afectează comportamentul controalelor referitor la acestea.

Context-ul, din punct de vedere al dezvoltatorului, este o structură arborescentă de noduri și atribute, care deține date și care poate fi mapat între diferite componente.

Windows-urile au o structură asemănătoare cu a view-urilor: un context, un controller care facilitează comunicarea și plug-uri pentru navigarea între diferite ferestre.

Orice componentă Web Dynpro mai conține pe lângă View-uri și Windows-uri un Controller care, la fel ca și celelalte Controller-e conține metode, atribute și Context și care, în plus asigură interfața externă pentru componentă. Pentru realizarea acestui lucru, atât metodele cât și elementele din Context sunt marcate corespunzător.

2.2.1.2. Modelul de programare Web Dynpro ABAP

Modelul de programare Web Dynpro [57], [58] utilizează o interpretare proprie a paradigmei MVC. La nivelul unei componente WD, Model-ul este realizat din contexte, View-ul corespunde Layout-ului și Controller-ului la nivel de View, iar rolul de Controller revine ComponentController-ului și Controller-ului la nivel de Windows, (figura 2.8).

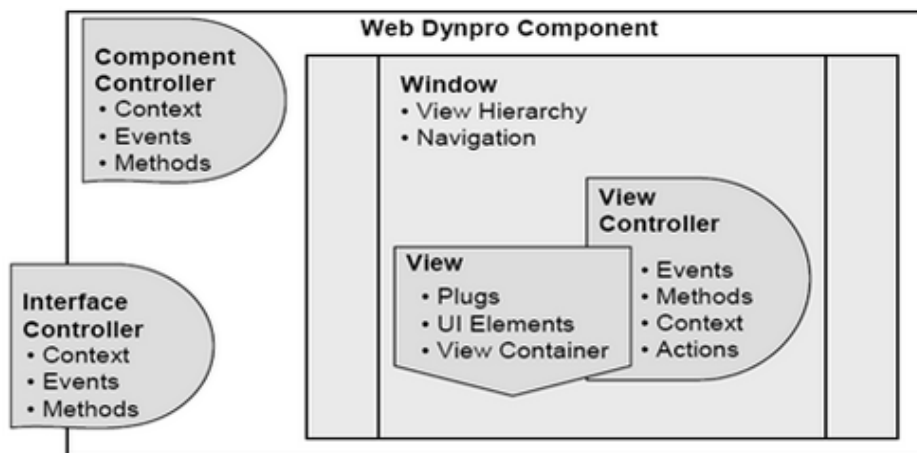


Fig. 2.8. Componenta Web Dynpro [38]

Tehnic, aceste componente au în mediul de dezvoltare sarcini și interacțiuni bine definite. Practic, o aplicație Web Dynpro este implementată ca o succesiune de View-uri legate între ele și integrate într-un Windows. Legăturile între View-uri se realizează static în faza de proiectare. Tot în Windows pot fi integrate View-uri care conțin elemente de tip ViewContainer UI, în care pot fi la rândul lor inserate alte View-uri.

Din punct de vedere tehnic, aplicația WD are ca punct de plecare spre execuție o adresă HTTP generată de către framework-ul Web Dynpro. Acesta

realizează comunicarea între interfața aplicației și aplicația propriu-zisă. Astfel, orice acțiune la interfața aplicației reprezintă o informație codată la nivelul framework-ului (de exemplu, acțiunea asupra unui buton este legată de metoda corespunzătoare atașată butonului respectiv din View-ul corespunzător).

În cazul în care este necesară inițializarea unui anumit element UI, se utilizează metodele predefinite Hook [49], [59] corespunzătoare framework-ului. Apelul acestor metode presupune:

- inițializarea Componentei, Window și View implicit (WDDOINIT);
- actualizarea View (WDDOMODIFYVIEW);
- manipularea Window Plug implicit (HANDLEDEFAULT);
- manipularea Component Controller înaintea navigării: (WDDOBEFORENAVIGATION);
- manipularea Component Controller după navigare: (WDDOPOSTPROCESSING).

Interacțiunea între componente poate fi realizată cu ajutorul Plug-urilor în două moduri:

- la navigarea între View-uri și Window (framework-ul WD este cel care instanțiază componentele țintă conform căii de navigare);
- la transmiterea datelor, Plug-urile având rolul de a controla datele cu care comportamentul componentei țintă este afectat. Schimbul de date între utilizator și aplicație se realizează și prin intermediul Context-elor, care pot fi mapate intern și extern, toată această arhitectură fiind integrată în mediul ABAP Workbench.

Așa cum s-a analizat de autor în lucrarea [60], datorită abordării declarative, este greu de încălcat conceptul arhitectural. Editorii de componente sunt cei care sprijină crearea și managementul optim al acestora, codurile putând fi înscrise doar în metodele predefinite, neexistând acces la clasele care le definesc.

Pe interfața unei componente Web Dynpro elementele pot fi programate atât dinamic cât și static în modul design, utilizând instrumentele vizuale ale framework-ului. Web Dynpro ABAP oferă o serie de clase a căror metode pot fi utilizate în codurile de programare în timpul execuției elementelor de interfață. La realizarea interfeței în modul design, elementele de interfață sunt adăugate prin inserare sub nodul ROOTUIELEMENTCONTAINER, acest nod realizând disponerea lor. Funcționalitatea acestor elemente UI trebuie ulterior programată. În cazul realizării dinamice a interfeței, toate modificările au loc la nivelul metodei Hook, WDDOMODIFYVIEW, interfața obținută fiind aceeași ca și în cazul în care se utilizează instrumentele vizuale ale framework-ului.

Această modalitate de realizare a interfeței a fost surprinsă de autor în lucrarea [40], unde, o analiză asupra timpului de execuție pentru componenta a căror elemente UI sunt realizate dinamic, demonstrează clar că acesta este cu mult mai mic decât timpul de execuție rezultat pentru aceeași componentă a căror elemente UI sunt realizate static. Acest aspect reprezintă un mare avantaj, pe lângă faptul că prin programarea dinamică se asigură o manipulare mai ușoară a componentei. Un dezavantaj al acestui tip de programare ar fi faptul că se scrie foarte mult cod.

Unul dintre conceptele utilizate de framework-ul Web Dynpro este cel de *reutilizare* sau *usage*, care oferă posibilitatea încapsulării componentelor și reutilizării funcționalităților acestora. Aceasta se poate realiza atât la nivelul unei componente (de exemplu încapsularea unui anumit View în mai multe containere pentru afișare în Window), cât și la nivelul unei aplicații formate din mai multe componente (de exemplu o componentă delegată cu autentificare poate fi parte a mai multor componente care necesită această funcționalitate), legăturile între date rămânând neschimbate în contextul respectiv.

Un studiu de caz care scoate în evidență acest concept al reutilizării este prezentat de autor în lucrarea [61], în care este analizată componenta framework-ului SALV_WD_TABLE. Reutilizarea acesteia pentru mai multe componente scoate în evidență funcționalitățile pe care le oferă referitor la prezentarea datelor din tabele.

În acest sens, programând diferite clase și interfețe specifice componentei SALV_WD_TABLE se poate :

- sorta datele din tabel în conformitate cu o anumită coloană;
- stabili numărul vizibil de rânduri;
- stabili antetul de coloană;
- afișa numai anumite coloane;
- afișa datele din coloane sub formă de arbore;
- crea dinamic un nod pentru afișarea datelor.

Diverse studii [62], [63] arată că nu există o regulă generală pentru organizarea și dezvoltarea unui proiect IT. În tehnologia WD, în prezent, există mai multe tehnici de realizare, cele mai utilizate fiind tehnicile de componentizare faceless și clasa de asistență.

Etapetele pe care le presupune realizarea unei aplicații Web Dynpro prin *tehnica de componentizare faceless* țin de pregătirea și implementarea componentelor care alcătuiesc aplicația, structura acesteia bazându-se pe trei componente principale: componenta model, componenta view (sau faceless) și componenta controller, care face legătura între ele și care conține aplicația web. Întrucât componenta Web Dynpro funcționează după principiul Model View Controller, partea vizuală a aplicației poate fi decuplată ușor de partea logică prin transferul în componenta faceless a tuturor metodelor care nu conțin secvențe de navigare și prin realizarea de noduri vizibile în tot context-ul aplicației.

Construirea logicii în componenta faceless face ca proiectul să fie întreținut și extins mult mai ușor, iar componentele să fie reutilizate și în alte aplicații. O secvență de cod utilizată în mai multe componente poate fi scrisă doar o singură dată ca metodă de interfață în componenta model, iar apoi metoda poate fi invocată ori de câte ori este necesar [53].

În esență, munca pentru proiectarea, modelarea și implementarea unui proiect IT poate fi clar împărțită pentru fiecare dintre membrii care alcătuiesc echipa respectivă, în funcție de aptitudinile fiecăruia dintre membrii.

Obiectivele implementării aplicației pot fi de asemenea împărțite pe componente, fiecare având sarcina de a rezolva vizual obiectivul respectiv. Astfel, rezultă componente independente care pot fi testate și apoi pregătite pentru legare în proiect.

Fiecare etapă în managementul proiectului poate fi cu ușurință controlată și testată. Prin decuplarea părții vizuale, controlul logic al proiectului este concentrat

pe componenta model, astfel încât orice schimbare solicitată de client în etapele intermediare poate fi ușor realizată în ceea ce privește codurile în cauză. Deși necesită un efort de programare destul de mare, prin decuplare se asigură un grad ridicat de transparență și de calitate a proiectului [64].

Un mare dezavantaj al tehnicii componentizării faceless, este procedura vastă de mapare externă și internă a nodurilor de date, care poate provoca erori, proiectele bogate în noduri de date fiind dificil de depanat. Acest dezavantaj este însă compensat de vizibilitatea completă a nodurilor de date în proiect. Tot ca dezavantaj poate fi considerată și procedura laborioasă la decuplarea modelului de date de partea vizuală. Acest aspect este compensat de o reflexie clară a conceptului Model-View-Controller, în care metodele din componenta model sunt legate de logica părții vizuale.

Marele câștig al utilizării acestei tehnici în realizarea unui proiect este acela de a putea modifica și extinde atât componentele ce alcătuiesc proiectul, cât și structura întregului proiect, prin adăugarea sau eliminarea de componente, gestionarea acestuia fiind foarte ușoară. Aceasta tehnică face ca un proiect foarte complex să devină ușor de planificat, monitorizat, dezvoltat și ulterior de modificat.

O tehnică alternativă utilizată în realizarea proiectelor web ABAP este *componentizarea cu clasă de asistență*. Când se utilizează acest gen de componentizare, componenta model și componenta controller dispar, iar o parte logică a proiectului este preluată la nivel global de către o clasă de asistență [52]. Această clasă, moștenește clasa SAP CL_WD_COMPONENT_ASSISTANCE și este în mod automat instanțiată în componentele proiectului, care sunt utilizate. Metodele implementate de această clasă fac parte din logica proiectului.

Unul dintre dezavantajele componentizării cu clasă de asistență se referă la reflexia mai slabă a conceptului Model-View-Controller, deoarece metodele din clasa de asistență nu susțin logica părții vizuale.

Ca avantaje ale acestei tehnici deosebim:

- mai puține componente;
- procedură ușoară de a decupla modelul de date de partea vizuală;
- efort de întreținere și depanare mic al erorilor proiectului.

Diferențele între cele două tehnici la nivel de concept, așa cum au fost sintetizate de autor în lucrarea [65], sunt prezentate în tabelul 2.2.

Tab. 2.2 Diferențele dintre tehnica componentizării faceless și tehnica componentizării cu clasă de asistență.

Concept	Tehnica componentizării faceless	Tehnica componentizării cu clasă de asistență	Observații
Număr de componente în proiect	Cele 3 componente de bază: model, view, controller	O componentă pentru un view global	Avantaj pentru componentizarea clasă de asistență
Logica proiectului	Controller-ul componentei faceless utilizat de componenta controller	Clasa de asistență instanțiată în toate componentele proiectului	Egalitate

Metodele din partea logică a proiectului	Pot fi implementate cu logica părții vizuale	Partea vizuală nu suportă logică, dar lucrează cu modelul de date	Avantaj pentru componentizarea faceless
Decuplarea modelului de date de partea vizuală	Procedură complexă	Procedură simplă	Avantaj pentru componentizarea clasă de asistență
Conceptul MVC	Reflectat foarte bine	Componenta view trebuie să păstreze interfața logică	Avantaj pentru componentizarea faceless
Nodurile din contexte	Noduri vizibile din ambele direcții (componente folosite pentru componentele utilizate și invers)	Noduri vizibile în context, unidirecțional (componente folosite pentru componentele utilizate)	Avantaj pentru componentizarea faceless
Maparea nodurilor	Intern și extern - procedură complexă	Intern procedură ușoară	Avantaj pentru componentizarea clasă de asistență
Componenta executabilă a proiectului	Componenta Controller	Componenta view-ul global	Egalitate
Erori Debugging	Dificil pentru proiecte cu multe noduri	Mai puțin dificil	Avantaj pentru componentizarea clasă de asistență
Mentenanța proiect	Dificil pentru proiecte cu multe noduri	Mai puțin dificil	Avantaj pentru componentizarea clasă de asistență

În concluzie, ambele tehnici au avantajele și dezavantajele lor, însă atunci când dimensiunea unui proiect este foarte mare este de preferat realizarea sa prin metoda de componentizare cu clasă de asistență.

Pe lângă conceptul de reutilizare, framework-ul WD adoptă și *conceptul de extindere* [66]. Acest concept poate fi utilizat de metodele din Controller-ul componentei WD. Un element important însă, este legat de conținutul acestor metode care nu poate fi suprascris, în cod fiind necesară utilizarea atributului WD_THIS în locul referinței ME. Pe lângă extinderea codului este necesară și extinderea context-ului celorlate părți ale componentei, View-uri și Window, care din punct de vedere tehnic nu este dificil a fi realizat. Astfel, în View pot fi adăugate elemente de interfață noi, fără a fi modificate cele existente, pot fi ascunse diferite elemente UI, așa încât să nu mai fie părți componente ale paginii respective, sau pot fi adăugate noi elemente de tip Plug [27].

Prin extindere este de asemenea posibil să se modifice structura de navigare într-o fereastră (Window). Link-urile de navigare pot fi eliminate sau ascunse și create noi link-uri, însă această operație poate cauza probleme. Totodată pot fi adăugate ca extindere noi View-uri, Window-uri, Controllere și alte componente suplimentare fără să fie necesară modificarea codului existent. Este necesară însă

extinderea contexte-lor pentru toate componentele adăugate, iar pentru acele noduri și atribute care se bazează pe structuri din dicționarul de date este necesară și extinderea dicționarului cu noi obiecte [27].

Datorită caracterului multilingual al platformei este posibilă internaționalizarea aplicațiilor, framework-ul fiind capabil cu ajutorul instrumentului OTR specializat în traducerea textelor, să ofere interfețe în diferite limbi. Un studiu de caz în această direcție a fost prezentat de autor în lucrarea [50].

Câteva dintre avantajele pe care le oferă Web Dynpro în dezvoltarea aplicațiilor web [27]:

- prin conceptul MVC, partea logică este separată de partea de prezentare;
- codul este minimizat, iar proiectarea maximizată;
- transportul datelor se face automat prin „data binding” și web service;
- se asigură acces la datele din contextul aplicației care rămân aceleași chiar dacă interfața se modifică;
- verificare automată a intrărilor;
- internaționalizarea aplicațiilor datorită caracterului multilingual al platformei;
- implementarea aplicațiilor se reduce mult, datorită utilizării instrumentelor grafice (Wizard);
- reutilizarea componentelor asigură o mentenanță ușoară;
- aplicațiile pot rula pe mai multe platforme.

Concluzionând, Web Dynpro ca tehnologie reprezintă nu numai un model de programare pentru interfețele utilizator, ci și un set de instrumente necesare design-ului de interfețe, un mediu de execuție pentru aplicații și totodată o tehnologie pentru soft modularizat. Aplicațiile Web Dynpro pot fi încapsulate în Portal pentru ca apoi să poată fi afișate prin browser sau prin SAP NetWeaver Business Client.

2.2.2. Floorplan Manager

Noile aplicații UI ale soluției SAP Business Suite [67] integrate pe platforma SAP NetWeaver sunt realizate cu ajutorul framework-ului Floorplan Manager. Aceste aplicații oferă utilizatorilor nu numai posibilitatea optimizării strategiilor IT de afaceri, dar și de a îmbunătății și integra procesele de business astfel încât să fie cât mai competitive pe piață. Proiectarea proceselor are la bază soluții modulare care pot să lucreze atât cu aplicații SAP cât și non-SAP. [68]

Framework-ul Floorplan Manager (FPM) are la bază tehnologia Web Dynpro ABAP, beneficiind astfel de toate avantajele oferite de aceasta. Cu ajutorul său este posibil să se construiască aplicații web rapid și ușor, putând fi adaptate de client necesităților, fără modificarea codului. FPM oferă următoarele patru modele de floorplan (șabloane) pentru construirea aplicațiilor, prin care este specificat aspectul general și indirect scopul acestora [69]:

- Object Instance Floorplan (OIF) permite crearea, vizualizarea, modificarea și ștergerea unui obiect de afaceri;

- Overview Page Floorplan (OVP) permite doar vizualizarea cu navigare la diverse activități, concentrându-se numai pe obiecte;
- Guided Activity Floorplan (GAF) asistă executarea unei activități împărțite în mai multe etape;
- Quick Activity Floorplan (QAF) versiunea simplificată a OIF, permite divizarea unei activități într-o secvență logică de pași.

Framework-ul FPM, are împărțită suprafața UI în zone bine definite și cu o anumită semantică. Pe lângă conținutul de bază, dezvoltatorul poate completa conținutul acestor suprafețe cu blocuri predefinite UI (UIBuildings Block) special concepute pentru componentele Web Dynpro ale framework-ului FPM, care pot fi puse împreună cu ajutorul unui editor grafic. Zonele specifice modelelor de floorplan-uri conțin elemente stabile, care servesc unui aspect uniform, putând fi configurate de dezvoltator, doar în situațiile cerute de aplicație.

Deoarece multe caracteristici ale tehnologiei Web Dynpro se regăsesc și în tehnologia FPM [69], framework-ul FPM fiind strâns legat de framework-ul WD prin utilizarea componentelor sale, descrierea acestei tehnologii atât din punct de vedere arhitectural cât și din punct de vedere al modelului de programare al aplicațiilor, se va face numai unde există diferențe.

2.2.2.1. Modelul arhitectural Floorplan Manager

Tehnologia FPM, ca parte a tehnologiei Web Dynpro, urmărește aceeași linie arhitecturală care are la bază conceptul MVC de separare a părții logice de cea vizuală. Tehnologia FPM, (figura 2.9), pune la dispoziție un model de framework bazat pe roluri, prin care dezvoltatorul, administratorul și utilizatorul final primesc responsabilități diferite [70]. Astfel:

- dezvoltatorul – definește catalogul de câmpuri, parametri de intrare și implementează clasa feeder pentru GUIBB;
- administratorul – configurează GUIBB bazat pe catalogul de câmpuri și parametri de intrare;
- utilizatorul final – declanșază aplicația WD și alege valorile preferate pentru criteriile de căutare în blocul generic pentru căutare (Search GUIBB).

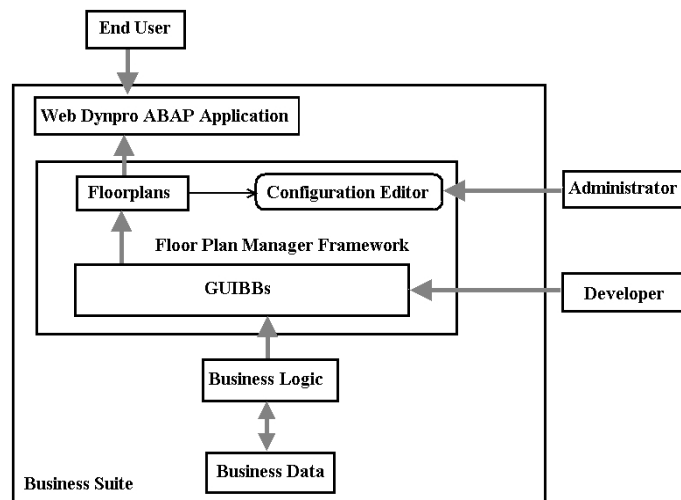


Fig. 2.9 Floorplan Manager în SAP Business Suite [69]

Pe lângă modelul bazat pe roluri, framework-ul FPM are integrat și un model pe etape (Phase Model) [69], prezentat în figura 2.10. Acest model pe etape descrie faptul că framework-ul FPM bazat pe WD ABAP este cuprins în bucla de evenimente a floorplan-ului. În timpul execuției, clasa feeder implementează interfața UIBB declanșând metodele predefinite pentru a furniza valori cataloagelor de câmpuri și parametrilor de intrare în bucla de evenimente. Ieșirea din buclă conduce la întoarcerea spre aplicația WD [71].

Întrucât tehnologia Web Dynpro permite elaborarea structurată a aplicațiilor, componentele sale, ca unități soft modularizate, ajută framework-ul FPM să reducă programarea redundantă. Combinarea și configurarea componentelor se realizează cu ușurință, rezultând astfel aplicații web complexe. Adaptarea aplicațiilor WD cu FPM în conformitate cu standardele actuale, poate fi foarte ușor realizată prin simpla configurare a acestora cu ajutorul editorului Floorplan Manager Configuration[29]. Setările de configurare pot fi schimbate și îmbunătățite prin modificarea codului, lucru care atrage după sine modificarea comportamentului dinamic al aplicațiilor.

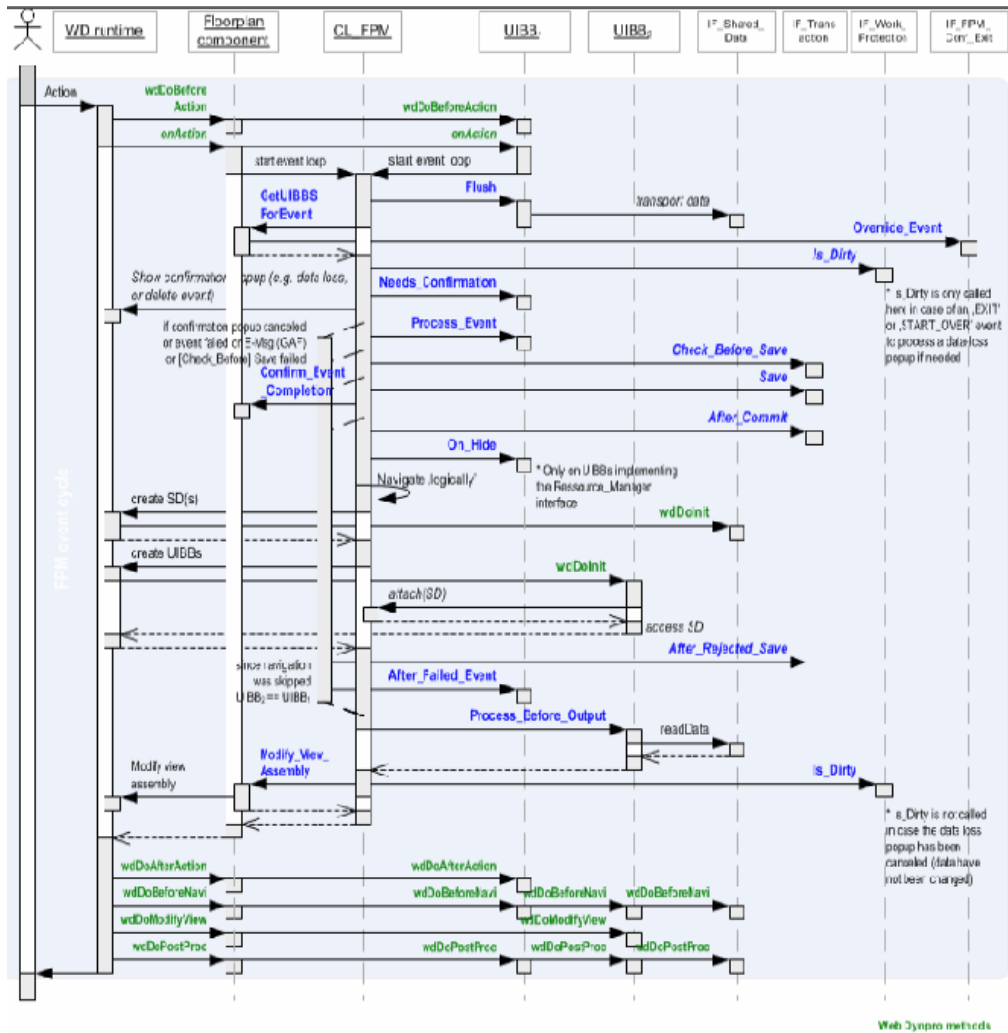


Fig. 2.10 Modelul pe etape (Phase Model) Floorplan manager [69]

2.2.2.2. Modelul de programare Floorplan Manager

O aplicație FPM este de fapt o aplicație WD obișnuită care are ca structură unul din cele patru modele de floorplan. În cadrul acestui model este apelat framework-ul FPM. La momentul execuției, acesta crează aplicația propriu-zisă în baza descrierii fișierelor de configurare ale componentelor UIBB și a altor componente din care este construită aplicația.

Părțile componente ale unei aplicații FPM sunt [69]:

- a) componentele floorplan ale framework-ului FPM, care servesc ca și cadru de proiectare pentru aplicațiile FPM și care au grad mare de configurabilitate;

- b) blocurile generice UI sau GUIBB puse la dispoziție de framework-ul FPM conțin măști de căutare, formulare sau tabele de afișare a datelor; cu ajutorul editorului de configurare ele pot fi adaptate conform cerințelor aplicației;
- c) UIBB sunt blocuri create de dezvoltator la libera alegere;
- d) Application Controller (AC) sunt componente WD utilizate pentru a controla blocurile UIBB într-o aplicație și care există maxim unul pentru fiecare aplicație FPM;
- e) Application Configuration Controller (ACC) sunt componente folosite la modificarea configurației componentei floorplan în timpul rulării, prin extinderea sau ascunderea blocurilor UIBB; la fel ca și AC există maxim unul pentru fiecare aplicație FPM;
- f) Shared Data Components sunt componente folosite pentru schimbul de date în aplicațiile FPM, servind ca sursă pentru legăturile contextelor din componentele WD;
- g) clasele Singleton "Globale" sunt utilizate alternativ pentru a face schimb de date între componente, întrucât datele din GUIBB nu pot fi mapate normal în context, ci doar în clase ABAP OO corespunzătoare.

Toate aceste componente fac parte dintr-un model pe etape (Phase Model) care este executat de către framework-ul FPM în cadrul modelului pe etape al Web Dynpro, funcționarea aplicației necesitând doar câțiva pași specifici FPM. Acești pași se referă la citirea, respectiv afișarea datelor din componentele specifice modelului FPM.

În continuare, modelul arhitectural FPM va fi prezentat numai în măsura necesară modelului pe etape. Realizarea unei aplicații FPM așa cum este descris în [72] și prezentat în figura 2.11 pornește de la componenta model floorplan aleasă, care are o configurație deja stabilită.

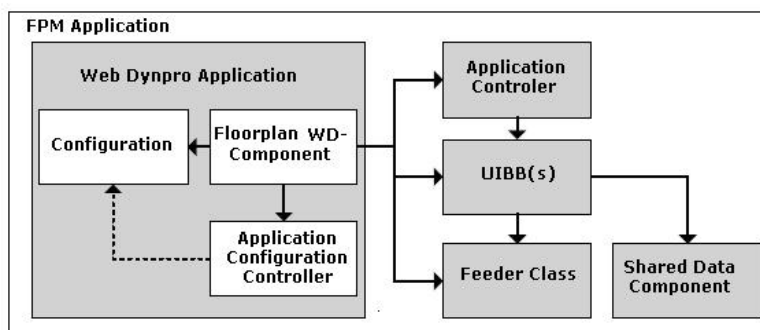


Fig. 2.11 Realizarea unei aplicații FPM [72]

Această configurație conține aplicația UI reală, realizată sub forma unui set de blocuri UIBB adecvat fiecărui floorplan. Pentru ca datele din blocurile individuale UIBB să fie afișate este necesară configurarea claselor Feeder corespunzătoare. Componenta ACC este utilizată pentru modificarea dinamică a configurației floorplan-ului, interfața UI reacționând astfel dinamic la datele de intrare. Componenta AC este utilizată pentru a controla aplicația. În acest context, se poate

citi și modifica configurația aplicației la fel ca și modelul structurat. Prin componenta Shared Data datele sunt partajate, fiind posibil schimbul și utilizarea în comun a acestora de către toate componentele.

Această modalitate de schimb a datelor [69] are însă dezavantajul că utilizatorul este dependent prin cod de această componentă. Pentru a evita acest lucru și a facilita punerea împreună a blocurilor UIBB într-o aplicație, există conceptul de *wiring*, care presupune implementarea unei interfețe speciale în clasa Feeder a blocului respectiv. Caracteristica specială a aceste interfețe este faptul că metodele sale pot fi invocate nu doar în timpul rulării.

Pentru a determina starea de wiring a interfeței, există editorul Flexible User Interface Designer (FLUID), care utilizează anumite metode la momentul proiectării [73]. Dacă clasele Feeder a două blocuri UIBB au interfețe comune, atunci ele se pot conecta prin wiring în acest editor. Transmitia datelor este realizată cu ajutorul unui conector de clasă. La inițializarea clasei Feeder implicate este generată o instanță a unui conector de clasă. După finalizarea procesării datelor de intrare de către UIBB, framework-ul FPM citește datele prin interfața wiring a clasei Feeder și o face disponibilă în clasa conector. Acolo, clasa Feeder poate citi scopul urmărit de UIBB [69].

Componentele aplicațiilor WD care includ blocuri GUIBB comunică cu framework-ul prin intermediul interfețelor FPM în cursul buclei de evenimente a floorplan-ului respectiv. Această buclă de evenimente include clasa de alimentare sau Feeder Class. Clasa conține o serie de metode predefinite care aparțin interfeței FPM și care sunt declanșate în bucla de evenimente la momentul execuției, atunci când floorplan-ul este instanțiat. De exemplu, pentru un GUIBB la momentul proiectării, metodele GET_DEFINITION() și GET_DATA() fac posibilă definirea catalogului de câmpuri și a valorilor concrete pentru câmpuri [74].

Ori de câte ori se utilizează un bloc generic GUIBB, configurarea componentei WD cere furnizarea unei clase feeder în care se poate defini propriul cod în metodele de interfață. Cele mai frecvent utilizate metode ale clasei feeder sunt:

AFTER_FAILD_EVENT(), INITIALIZE(), FLUSH(), NEEDS_CONFIRMATION(), PROCESS_BEFORE_OUTPUT() și PROCESS_EVENT().

Tehnologia FPM a fost concepută cu scopul de a crea componente reutilizabile, mult mai ușor decât tehnologia Web Dynpro. Framework-ul FPM oferă elementele de bază necesare pentru componentele și fluxurile de date, care prin configurare pot fi puse foarte ușor împreună. Cu ajutorul editorului grafic, aplicațiile FPM pot fi compilate doar prin configurare, nefiind necesar un efort de programare suplimentar. Tehnic însă, dacă interfețele FPM sunt realizate cu ușurință, blocurile UIBB și clasele Feeder sunt mai dificil de reutilizat. În majoritatea cazurilor nu este nevoie a fi create blocuri UIBB proprii, deoarece este posibil ca funcționalitățile acestora să fie îndeplinite de blocurile generice GUIBB [75].

Pentru FPM nu este necesar conceptul de extindere, întrucât majoritatea aplicațiilor FPM sunt definite prin configurare. La diferite nivele, aplicațiile pot fi suprascrise prin intermediul unei configurări specifice care să nu afecteze configurația de bază a componentei respective. Conceptul de extindere ABAP poate fi însă aplicat claselor Feeder și Conector care sunt specifice aplicației.

Chiar dacă FPM este un framework puternic, centrat pe design, el nu dispune de un Designer Workflow explicit. Se consideră că aplicația la baza căreia

stă un floorplan, are adăugate blocuri UIBB care deja există, iar blocurile generice GUIBB sunt construite pe un model standard bazat pe principiile SAP UI.

2.2.3. WebClient UI

WebClient UI este o interfață bloc (UIBB) oferită de SAP prin platforma Business Suite, care este concepută pentru utilizare în mediul afacerilor, mai exact pentru Customer Relationship Management. SAP CRM este soluția complexă pentru gestionarea relațiilor cu clienții [76], [77]. Fiind orientată spre client, ariile de activitate cuprind atât domeniul de marketing, vânzări și servicii, cât și canalele de interacțiune cu clienții.

Tehnologia WCUI se bazează în întregime pe stack-ul ABAP, dar nu pe Web Dynpro, ci pe o soluție brevetată, disponibilă numai prin CRM și anume BSP. Astfel, WebClient UI oferă posibilitatea combinării tehnologiei BSP cu funcționalitățile și confortul tehnologiei Web Dynpro de la care utilizează nu numai conceptele, ci și modelul de programare.

Ca tehnologie UI pentru realizarea interfețelor, BSP este o combinație între cod ABAP și extensii BSP. Aceste extensii constau din tag-uri HTMLB mult mai complexe decât tag-urile HTML clasice, care dau un aspect unitar aplicațiilor. Modelul BSP de programare este utilizat pentru dezvoltarea de aplicații BSP pe server-ul de aplicații SAP Web AS. Aceste aplicații pot accesa clase, module funcționale, interfețe și chiar baza de date și sunt legate de sistemul de transport (CTS), instrument care ajută la organizarea dezvoltării proiectelor în mediul ABAP Workbench.

Întrucât aplicațiile WebClient UI sunt implementate prin componente BSP, în cele ce urmează, se va face o descriere succintă a modelului arhitectural BSP și a componentei BSP atât cât este necesar pentru înțelegerea tehnologiei WCUI.

Începând cu versiunea 6.20 a server-ului de aplicații, modelul MVC s-a extins așa încât a permis implementarea componentelor BSP, necesare nu doar realizării aplicațiilor BSP, ci și a componentelor WebClient UI. Conform conceptului arhitectural al tehnologiei prezentat în figura 2.12, Controller-ul unei componente BSP reprezintă punctul de intrare în aplicație și este apelat printr-un URL. La nivelul său se prelucrează datele primite, se actualizează modelul de date (prin transferul datelor la nivel de View) și se comunică cu nivelul logic de afaceri (Business Logic) prin intermediul Application Class.

Controller-ul definește o clasă de bază, metode și atribute necesare procesării cererilor care, după instanțierea sistemului, sunt apelate într-o anumită ordine. Printre atribute există și o referință disponibilă pentru instanțierea clasei de aplicații Application Class [78].

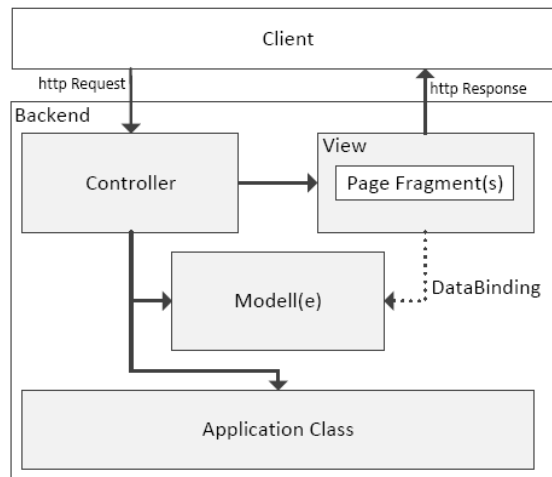


Fig. 2.12 MVC în arhitectura BSP [78]

Așa cum s-a menționat, logica de programare din spatele unei aplicații BSP este implementată prin clasa de aplicații sau interfața pe care aceasta o oferă. Modelul de date este și el la rândul său definit printr-o interfață care permite legarea elementelor HTML la atributele corespunzătoare din View prin care se realizează transferul automat de date (DataBinding) între aplicație și baza de date. View-urile nu pot fi apelate direct prin http ci doar prin intermediul Controller-ului. Ele sunt pagini BSP fără logică proprie și servesc doar reprezentării datelor care sunt transmise prin intermediul Controller-ului[78].

Pentru afișarea datelor se pot alege extensiile BSP ca HTMLB pe de o parte și pe de altă parte conținutul dorit poate fi creat manual, sau cu unelte și cu logica corespunzătoare.

Comparativ cu tehnologia Web Dynpro se disting câteva similarități și anume:

- abordare generativă;
- aplicațiile sunt realizate din componente;
- au la bază paradigma MVC;
- fiecare dintre controller-e (window, view și custom) posedă un context structurat sub forma unui arbore care poate fi legat de context-ul altor controller-e de componente;
- view-urile și window-urile posedă intrări și ieșiri necesare navigării;
- elemente ale context-ului construit în view pot fi utilizate în interfața cu utilizatorul.

În prezent, framework-ul WebClient UI este complet integrat în versiunea 7.02 a server-ului de aplicații ABAP. Aplicațiile WebClient UI, de obicei nu sunt executate direct, ci prin aplicații speciale de tipul Web CRM. Această aplicație are propriul său cadru de navigație, conceptul de rol și tehnică de programare pentru interfețele utilizator. O caracteristică de bază este navigarea și o interfață ușor de înțeles, (figura 2.13).



Fig. 2.13 Captură din pagina principală a aplicației CRM WebClient UI (versiunea în lb.română)

Zonele de navigare și antet sunt zone statice și conținutul lor poate fi configurat prin rolul de afaceri, pe când zona de lucru este dinamică, conținutul ei modificându-se în funcție de navigare sau de comenzile alese de utilizator. În CRM fiecărui utilizator i se atribuie un rol de afaceri în funcție de responsabilitățile și activitățile ce le are.

2.2.3.1. Modelul arhitectural WebClient UI

Arhitectural, framework-ului CRM WebClient UI este împărțit pe două secțiuni strict separate: aplicația CRM și baza de date (figura 2.14), însă din altă perspectivă se disting trei nivele. Acestea sunt: nivelul de persistență - Database Layer, nivelul logicii business - Business Layer și nivelul de prezentare - Presentation Layer [79].

Nivelul Database Layer constă din modelul relațional al bazei de date CRM, precum și toate interfețele API furnizate, pentru citirea ori scrierea datelor corespunzătoare obiectelor business (de exemplu: adresa, partner business, comanda, oferta, produs, etc.).

Nivelul Business Layer garantează separarea interfeței utilizator de logica de business implementată în nivelul Database Layer. Acest nivel constă din componenta BOL (Business Object Layer) și componenta GenIL (Generic Interaction Layer). Componenta BOL încapsulează obiectele de dată business persistente (Business Object Data and State) și le expune atât nivelului de prezentare, cât și interfețelor de programare API ale nivelului Business Logic. Transferul de date de la BOL la sistemul de persistență al bazei de date este asigurat de componenta GenIL. BSP-urile din nivelul de prezentare, nu au capacitatea de a controla logica business a obiectelor și proceselor CRM.

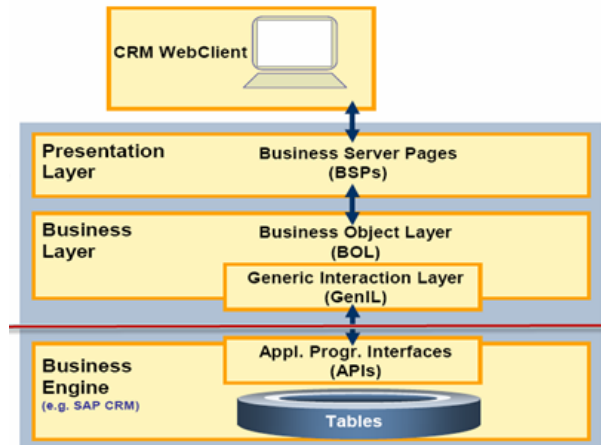


Fig. 2.14 Arhitectura CRM WebClient UI [79]

Legătura între Business Logic și framework-ul UI este realizată de către GenIL printr-o anumită interfață. Componenta GenIL este de fapt, o clasă ABAP care implementează interfața specifică pentru comunicarea cu framework-ul și constă dintr-o ierarhie de obiecte business. Fiecare componentă GenIL definește un model de date ierarhic, care este public și mapat în interfața corespunzătoare. Apelurile pentru executarea de căutări, creări și actualizări se declanșează tot prin componenta GenIL. Structurile de date utilizate în API nu sunt aceleași structuri cu cele din modelul de date al componentei GenIL [80].

Întodeauna când framework-ul este startat, el conține deja un set de componente GenIL. Atunci, modelul de date al fiecărei componente e chemat și stocat în memoria din GenIL. Setul de componente ajută la construirea unui context pentru o aplicație specifică CRM. Modelul de date al unui set de componente poate fi vizualizat în BOL Model Browser.

Presentation Layer sau UI, care este responsabil cu partea vizuala a aplicației, se compune din pagini web care sunt prestate la rulare pe bază de extensii BSP și clase care conțin logica de navigare și apelurile către business logic. Acest nivel este împărțit la rândul său în 3 nivele, cunoscute ca MVC (Model View Controller).

Modelul conține informații despre gama de structuri de date folosite, fiecare cunoscute ca un context. Modelul comunică cu BOL în Business Layer. La acest nivel, datele din context sunt mapate pentru a fi înțelese de către baza de date, însă GenIL va fi cel care va face legătura cu baza de date atunci când se citește din, sau se scrie în tabelele persistente.

View-ul conține un fișier HTML și este responsabil pentru afișarea informațiilor modelului pe ecran. Toate datele introduse de utilizator sunt primite de controler pentru prelucrare și apoi afișate prin view. De asemenea, view-ul este responsabil pentru definirea aspectului (layout) pentru fiecare pagină văzută de utilizator. Prin view se cunoaște designul paginii: butoanele, tabelele, câmpurile și etichetele ce vor fi afișate pentru fiecare context. Aceasta este partea cea mai frecvent personalizată de WebClient UI.

Accentul este pus pe nivelul de logică business sau Business Layer, prezentat în figura 2.15. Acest nivel este susținut prin două framework-uri și anume: Business Object Layer (BOL) și Generic Interaction Layer (GenIL) [81].

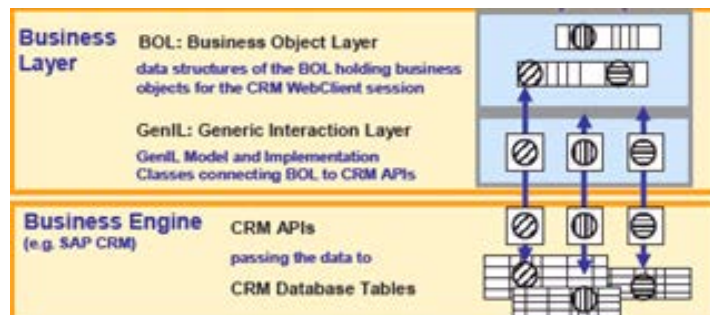


Fig. 2.15 Nivelul de logică business (Business Layer) [79]

BOL furnizează o interfață API uniformă, care este utilizată de componentele din nivelul de prezentare pentru accesarea datelor. API-urile din GenIL sunt utilizate atunci când se codifică în backend. Fiecare componentă GenIL este reprezentată de o clasă dedicată, care pune în aplicare un set de interfețe. Această clasă dispune de metode dedicate pentru căutare, citire, modificare și salvare pentru toate obiectele.

Deoarece cele trei nivele ale arhitecturii CRM WebClient UI sunt complet separate unul de celalalt, este permisă conectarea oricărei aplicații de business care se încadrează în framework-ul WebClient UI, prin nivelul (BOL) și stratul de interacțiune generic (GenIL) la nivelul de prezentare.

La baza nivelului de prezentare care face posibilă afișarea în Web browser a paginilor HTML generate de BSP, este CRM User Interface Framework (CRM UIF). Acest nivel este împărțit la rândul său în 3 nivele, cunoscute ca MVC, (figura 2.16) și anume [79]:

- *modelul* conține informații despre gama de structuri de date folosite, fiecare cunoscută ca și context și comunică cu BOL în Business Layer. La acest nivel, datele din context sunt mapate pentru a fi înțelese de către baza de date, însă GenIL va fi cel care va face legătura cu baza de date atunci când se citește sau se scrie în tabelele persistente;
- *view-ul* conține un fișier HTML și este responsabil pentru afișarea informațiilor modelului pe ecran. Toate datele introduse de utilizator sunt primite de controller pentru prelucrare și apoi afișate prin view. De asemenea, view-ul este responsabil pentru definirea aspectului (layout) pentru fiecare pagină văzută de utilizator. Prin view se cunoaște designul paginii: butoanele, tabelele, câmpurile și etichetele ce vor fi afișate pentru fiecare context. Aceasta este partea cea mai frecvent personalizată de WebClient UI;
- *controller-ul* gestionează partea de interacțiune logică și asigură totodată legătura dintre view și model.

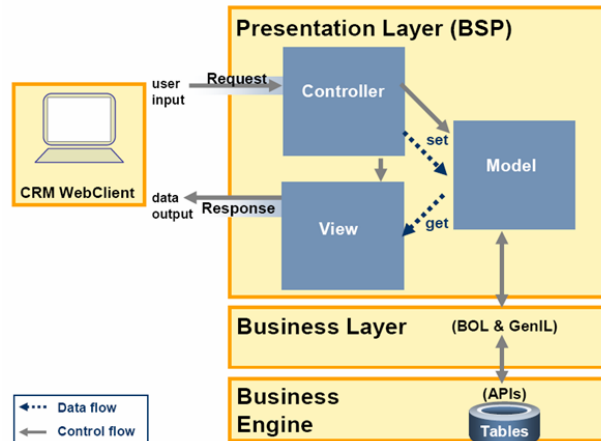


Fig. 2.16 Conceptul MVC în CRM WebClient UI [79]

2.2.3.2. Modelul de programare WebClient UI

La fel ca și în tehnologia Web Dynpro, componentele care alcătuiesc o aplicație WCUI, deși diferă prin realizarea lor tehnică, la implementare urmăresc același model, Model View Controller [82]. Aplicațiile sunt organizate în componente, care la rândul lor includ alte componente. Toate acestea sunt implementate prin componente BSP și afișate numai în IDE, care oferă asistență pentru realizarea lor. Framework-ul CRM WebClient UI generează scheletul de bază format din componente BSP și clase cu o ierarhie de moștenire corespunzătoare, care pot fi programate de dezvoltator.

Fiecare componentă are în alcătuire un controller care constă din două clase, care funcționează ca și controller BSP. Prima clasă moștenește clasa de bază pentru controller-ul tip Web Dynpro și conține cod generat WCUI. Ea este responsabilă cu procesarea aplicației și instanțierea context-ului corespunzător împreună cu legăturile sale. Cea de-a doua clasă este pusă la dispoziția dezvoltatorului pentru implementarea efectivă a aplicației.

Crearea unui controller presupune automat și crearea unei clase Context asociate, care moștenește context-ul clasei de bază Web Dynpro. Nodurile Context individuale sunt implementate ca și clase separate, instanțierea lor realizându-se în clasa Context. Ele moștenesc direct clasa modelului BSP putând fi folosite în View-uri. De implementarea claselor corespunzătoare nodurilor Context este răspunzător atât dezvoltatorul cât și nivelul BOL. Acesta oferă date, care prin metode suplimentare sunt furnizate în clasa nodului Context [83].

Toate aceste componente de bază sunt utilizate de View, Windows, Component Controller și Custom Controller. Window-ul unei componente este alcătuit doar dintr-un Window Controller în care pot fi legate mai multe View-uri la o aplicație, ca în figura 2.17.

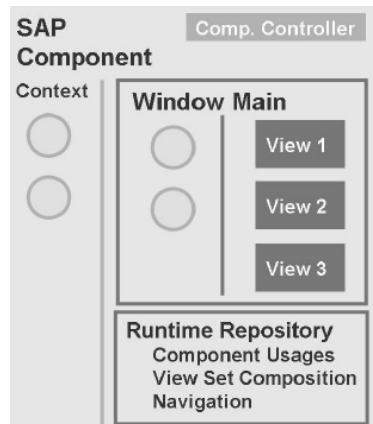


Fig. 2.17 Componenta WebClient UI [82]

View-ul componentei are în alcătuire un Controller și un View BSP. Controller-ul are la bază View Controller-ul Web Dynpro, iar View-ul BSP servește ca Layout, facilitând transmiterea datelor. Controller-urile ce aparțin Windows-ului și View-urilor implementează interfețe speciale prin care pot fi integrate în CRM, navigarea realizându-se prin intermediul Plug-urilor la fel ca în Web Dynpro. La nivelul componentei WebClient UI, interacțiunea dintre aceste componente de bază are loc în Runtime Repository (Navigation), figura 2.17.

Așa cum este prezentat și în figura 2.18, datele de intrare sunt transmise la nivel de View Controller care le procesează, dacă este necesar scrie datele modificate în Context și generează un răspuns prin componenta BSP [82].

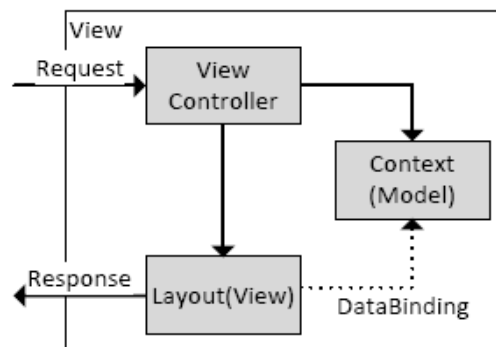


Fig. 2.18 MVC în aplicații WebClient UI [82]

Component, Custom și Window Controller joacă un rol, în măsura în care reprezintă sursa pentru mapările context-ului. Dacă într-un View Controller, datele dintr-un nod aflat într-o ierarhie se schimbă, acest fenomen trebuie să se reflecte și în nodul sursă, așa încât, alte View-uri care au mapate aceste noduri să le poată accesa. Component și Custom Controller se comportă analog, navigarea între componente are loc în tabelele mapate din Work Area Component Repository [82].

Componentele unei aplicații WCUI pot fi legate în două moduri [84], prin plug-uri și prin maparea Context-ului. Plug-urile nu împiedică reutilizarea componentelor, deoarece căile de navigație chiar dacă sunt dependente de mediul

apelant pot fi definite oricând noi căi de navigare. Maparea Context-ului realizată static face dificilă reutilizarea componentelor, însă realizată dinamic, deși necesită un efort suplimentar, permite o reutilizare a componentelor relativ flexibilă.

Conceptul de îmbunătățire sau extindere (enhancement) al WCUI, are la bază paradigma orientării pe obiect și poate fi aplicat pentru toate componentele [82]. Pentru aceasta este creată o componentă care conține o copie a Runtime Repository a componentei originale, ca în figura 2.19. Dacă se dorește extinderea Controller-ului sau Context-ului componentei, se generează o clasă în noua aplicație, înregistrată în Runtime Repository, în care vor fi realizate toate modificările. Trebuie specificat însă, că metoda care este responsabilă pentru inițializarea Controller-ului trebuie suprascrisă, astfel încât să poată fi folosit contextul noii clase.

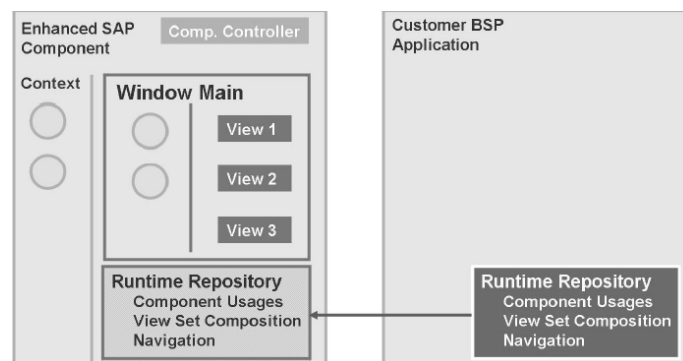


Fig. 2.19 Conceptul de extindere la nivel de componentă [82]

Pentru a extinde un View, Layout-ul său este copiat ca și Repository. Aceste copii sunt necesare, deoarece atât Layout-ul cât și Repository ca date-text nu preiau conceptul de moștenire, fiind necesară modificarea lor. Dacă se întâmplă ca fișierul original să fie schimbat, atunci în copie nu se va mai găsi schimbarea efectuată, lucru care constituie un punct slab al acestui concept de extindere.

Proprietățile UI ale framework-ului WCUI provin în mare parte din tehnologia BSP. Pentru aplicațiile WCUI există diferite extensii BSP (Tag Libraries) care aduc un plus de valoare la nivel UI. Printre ele, extensia thtmlb oferă pentru Layout elemente UI speciale, cum sunt controale de căutare sau elementele Road Map.

Întrucât tehnologia WebClient UI este dezvoltată pentru SAP CRM, aplicațiile de test vor avea la bază structura bazei de date utilizate în CRM, beneficiind astfel de o serie de avantaje pe care framework-ul le pune la dispoziție [81]:

- protecția datelor : autorizare pe roluri;
- tehnici de căutare avansate: interogări și rapoarte;
- adaptabilitate la necesitățile clientului: enhancement și customizing;
- multilingualitate .

2.3. Elemente de ansamblu specifice tehnologiilor WD ABAP, FPM și WebClient UI

Pentru că în paragrafele anterioare au fost descrise caracteristicile de bază și modelul de programare pentru fiecare dintre tehnologii, în continuare vor fi scoase în evidență principalele funcționalități și capabilități ale acestora [85], [86], [87].

Astfel, **tehnologia Web Dynpro ABAP**:

- furnizează un model de programare bazat pe dezvoltarea declarativă de interfețe;
- serviciile generice ale framework-ului oferă configurare și standardizare în prezentare;
- dispune de instrumente puternice pentru modelare, implementare și testare a aplicațiilor.

Includerea framework-ului **Floorplan Manager** presupune:

- utilizarea de șabloane și blocuri (UIBB's) pentru configurarea unei aplicații WD;
- adaptarea cu ușurință a interfețelor existente;
- armonizarea aspectului interfețelor cu ajutorul SAP Signature Design.

Utilizarea mediului PageBuilder și Side Panel presupune:

- realizarea de componente View prin combinarea mai multor interfețe CHIP ca părți dinamice de program;
- reutilizarea de componente View predefinite;
- adăugarea de elemente UI printr-un panou lateral –Side Panel.

Utilizarea graficelor și serviciilor interactive prin RIA (Rich Internet Application) Islands presupune:

- integrarea aplicațiilor Flash/Flex Adobe și Silverlight cu elemente UI specific;
- integrarea graficelor cu capabilități sofisticate (business graphics).

Principalele capabilități constau în:

- suport pentru HTML și ARIA (Accesibile Rich Internet Application) standard web;
- BOL (Business Object Layer) activ pentru framework-ul Floorplan Manager;
- design flexibil pentru dezvoltatori prin editorul FLUID pentru FPM;
- elemente UI diversificate (Form Layout bazat pe grid, client side table).

Tehnologia **WebClient UI** oferă:

- o interfață modernă concepută pentru SAP CRM prin framework-ul WebClient UI;
- arhitectură bazată pe navigare;
- flexibilitate în personalizarea domeniilor prin configurare;
- armonizarea interfețelor prin SAP Signature Design.

CRM WebClient UI presupune:

- navigare ușoară la nivel global datorită cadrului în formă de L;
- scurtături generice pentru o accesare ușoară a datelor;
- trecerea cu ușurință de la interfețele utilizator la necesitățile de business;
- adaptarea și configurarea interfețelor.

AET (Application Enhancement Tool) facilitează:

- crearea de vederi, tabele, câmpuri customizate;
- crearea de aplicații model pentru dezvoltarea platformelor test POC (point-of-care) prin Rapid Apps.

Principalele capabilități constau în:

- extinderea capacităților de configurare a interfețelor;
- încorporarea feedback-ului clientului;
- caracteristici suplimentare a utilizabilității framework-ului (ex. Tagging, Sharing).

2.4. Concluzii

Având în vedere cele prezentate anterior, referitor la aspectele legate de platforma tehnologică SAP NetWeaver și server-ul de aplicații AS care stă la baza acesteia, rezultă următoarele concluzii:

- SAP NetWeaver este fundamentul pentru platforma Business Suite pe care o integrează și prin care pune la dispoziție o gamă diversă de tehnologii UI pentru dezvoltarea de aplicații complexe;
- SAP NetWeaver reprezintă platforma tehnologică de integrare, care oferă posibilitatea gestionării tuturor etapelor de realizare a aplicațiilor web: proiectarea, dezvoltarea, implementarea și testarea;
- printr-o infrastructură scalabilă și sigură, formată din două stive tehnologice ABAP și Java, AS asigură suport nivelului de persistentă, serviciilor web independente de platformă și aplicațiilor business;
- la nivelul AS ABAP, tehnologiile specifice platformei Business Suite și recomandate de SAP pentru dezvoltarea de interfețe UI sunt: WD ABAP, Floorplan Manager și WebClient UI;
- toate modelele de programare ale tehnologiilor de pe platforma Business Suite sunt construite pe baza paradigmei MVC, iar arhitectura lor este identică cu cea a server-ului de aplicații ABAP (3 nivele).

Analizând tehnologiile UI ale AS ABAP specifice platformei Business Suite, atât din punct de vedere arhitectural, cât și din punct de vedere al modelului de programare, se poate considera:

- **Web Dynpro** ca tehnologie, reprezintă nu numai un model de programare pentru interfețele utilizator, ci și un set de instrumente necesare design-ului de interfețe, un mediu de execuție pentru aplicații și totodată o tehnologie pentru soft modularizat. Aplicațiile business pot fi dezvoltate utilizând mai multe tehnici, care au avantajele și dezavantajele lor, însă atunci când dimensiunea unui proiect este foarte mare, este de preferat realizarea sa prin metoda de componentizare cu clasă de asistență.
- **FPM** este un framework puternic centrat pe design, ce are la bază tehnologia Web Dynpro ABAP. FPM ajută la crearea și configurarea interfețelor utilizator oferind mai multe elemente: *modele de floorplan* (structuri de elemente UI aranjate într-o anumită ordine într-o aplicație) a căror spații de lucru sunt implementate automat de framework-ul FPM, blocuri generice GUIBB (șabloane UI pentru afișarea conținutului) și un

editor de configurare – FLUID, care permite configurarea structurii aplicației în ansamblu, precum și a modelului de floorplan și blocurilor UIBB.

- **WebClient UI** este interfața online bazată pe BSP, oferită de SAP doar pentru CRM. Arhitectura framework-ului permite realizarea și conectarea oricărei aplicații la nivelul de prezentare, prin intermediul celor două nivele BOL și GenIL. Conceptul de enhancement al componentelor joacă un rol important la adaptarea UI.

3. Implementarea și testarea aplicațiilor web la nivelul SAP AS

Obiectivele acestui capitol urmăresc două direcții principale. Prima, tratatează aspecte legate de implementarea aplicațiilor web, la nivelul server-ului ABAP, în tehnologiile UI recomandate de SAP pentru platforma Business Suite a NetWeaver. În acest context, sunt evidențiate caracteristicile specifice tehnologiilor Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI, la implementarea aceluiași model de aplicație web a cărui date provin din tabelele bazei de date SAP CRM.

Cea de-a doua direcție vizează testarea aceluiași aplicații, cu ajutorul instrumentelor specializate SAP Business Transaction Analysis și HTTP Watch pentru Internet Explorer, în vederea analizării performanțelor acestora. Testele presupun interogarea bazei de date cu afișarea rezultatelor pentru trei cazuri diferite.

3.1. Descrierea tipului de aplicație Web implementată în cele 3 tehnologii UI

Având în vedere că majoritatea aplicațiilor web care sunt legate de o bază de date efectuează operații de căutare, adăugare și ștergere la nivelul acesteia, s-a ales un model de aplicație business care să înglobeze toate aceste funcționalități.

Aplicația face parte dintr-un proiect dezvoltat de autor pentru compania SP Lion SA din Germania și constă în administrarea datelor referitoare la marketing-ul proiectelor firmei.

Interfața principală, prezentată în figura 3.1, presupune o însumare de elemente UI, fiecare specializat în îndeplinirea unei anumite funcționalități.



Fig. 3.1 Interfața principală a proiectului

Una dintre aceste funcționalități este legată de clienții firmei și constituie sursa pentru aplicația implementată în cele trei tehnologii SAP UI. Aplicația are la bază o interfață specializată în căutare, afișare, adăugare și ștergere a datelor clienților din tabelele corespunzătoare bazei de date CRM. Design-ul interfețelor și logica de funcționare a aplicației este aceeași pentru toate cele trei tehnologii SAP UI.

O captură a tabelor din baza de date SAP CRM, care sunt legate de interfața aplicației, este prezentată în figura 3.2. Aceste tabele conțin date generale despre clienți-parteneri business și adresele lor.

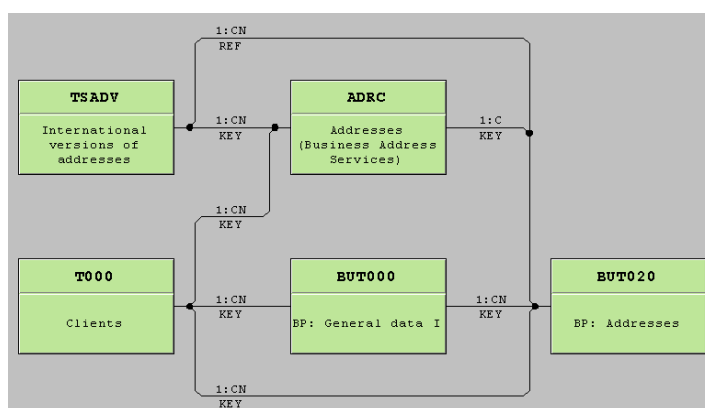


Fig. 3.2 Captură cu tabelele bazei de date SAP CRM corespunzătoare aplicației web

3.2. Implementarea aplicației web în tehnologiile Web Dynpro ABAP, Floorplan Manager și WebClient UI

Acest paragraf tratează modul în care o aplicație web, ce permite căutarea și afișarea datelor din baza de date SAP CRM, este implementată în trei dintre tehnologiile recomandate de SAP (Web Dynpro ABAP, Floorplan Manager și WebClient UI) pe platforma Business Suite a SAP NetWeaver, la nivelul server-ului de aplicații ABAP. Aplicația este analizată ca și componentă separată, care conform tehnicii de componentizare și conceptului de Usage face parte din proiect.

Design-ul interfeței permite pe lângă căutare și afișare, adăugarea, modificarea și ștergerea datelor. Scopul propus este de a măsura cu ajutorul instrumentului de analiză SAP Business Transaction Analysis performanța parametrilor aplicației la apelul unei tranzacții de căutare în baza de date a aplicației și de a analiza acești parametri în vederea comparării tehnologiilor UI recomandate.

3.2.1. Implementarea aplicației în tehnologia Web Dynpro ABAP

Aplicațiile Web Dynpro implementate la nivelul server-ului de aplicații ABAP sunt dezvoltate în mediul ABAP Workbench. Pe lângă faptul că este un mediu de execuție, ABAP Workbench [88], dispune de instrumente specializate de proiectare și implementare a aplicațiilor web. Așa cum este descris în paragraful 2.2.1.1, aplicația web a cărei interfață este analizată și prezentată în figura 3.3, are la bază o componentă și un model de date prezentat în figura 3.2.

The screenshot displays the 'Search partner' interface. At the top, there is a search bar with a 'Business partner' dropdown and a 'Search' button. Below this is a section titled 'Display data from selection' with an 'Add' button. The main part of the interface is a table with the following data:

BusPartner	Partner Type	Name 1	Street	House Number	City	Postal Code	Region	Country
1141		Alpine Systems	Arnold Schwarzenegger Weg	54	Salzburg	5010	S	AT
1142		SP Lion AG	FridrichStr.	4	Muenchen	72765	02	DE
1143		Brussels	Europa Ava	9898	Brussel	9999		BE
1144		C.R.T.U1	Avenue du Mont-Royal Est	63	Montreal	H2Y 1N3	QC	CA
1145		Electrica Banat	Calea Aradului	7	Timisoara	2213243	37	RO

Below the table, there are navigation controls including 'Row 1 of 19413'. At the bottom, there is a 'Details - Partner and Address data' section with buttons for 'Change Mode', 'Delete', and 'Save'. The details for the selected partner (1141) are as follows:

Partner:	1141	Street/House number:	Arnold Schwarzenegger Weg 54
Partner Type:		City/Post code:	Salzburg 5010
Organization name:	Alpine Systems	Region:	S
		Country:	AT

Fig. 3.3 Interfața aplicației implementată în tehnologia Web Dynpro ABAP

Din figura 3.4 se poate observa alcătuirea componentei Y_WD_BUCOMP: ComponentController, Component Interface, Views și Windows. Toată această structură care are la bază paradigma MVC, facilitează decuplarea părții logice de interfața aplicației. Implementarea componentei presupune de fapt implementarea celor 5 View-uri, legarea și integrarea lor în Window-ul W_BUCOMP. Din punct de vedere tehnic, aceste componente au în mediul de dezvoltare sarcini și interacțiuni bine definite.

Utilizând instrumentele vizuale ale framework-ului, cele 5 View-uri programate static în modul design, conțin elemente UI care sunt inserate sub nodul ROOTUIELEMENTCONTAINER, corespunzător Layout-ului fiecărui View. Legătura între elementele UI și elementele corespunzătoare din baza de date este realizată prin binding, acest aspect fiind prezentat în figura 3.5.

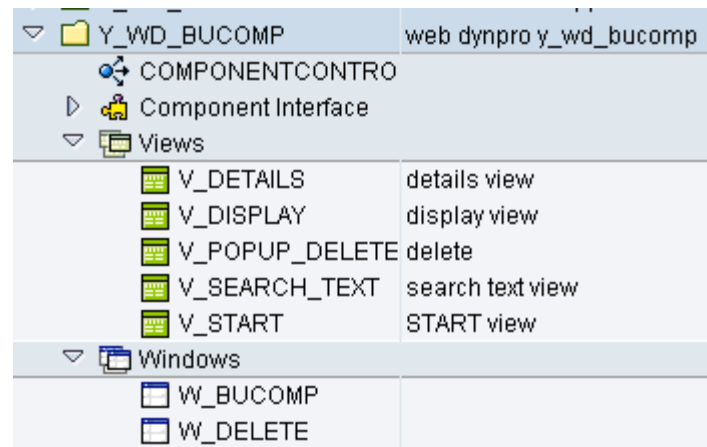


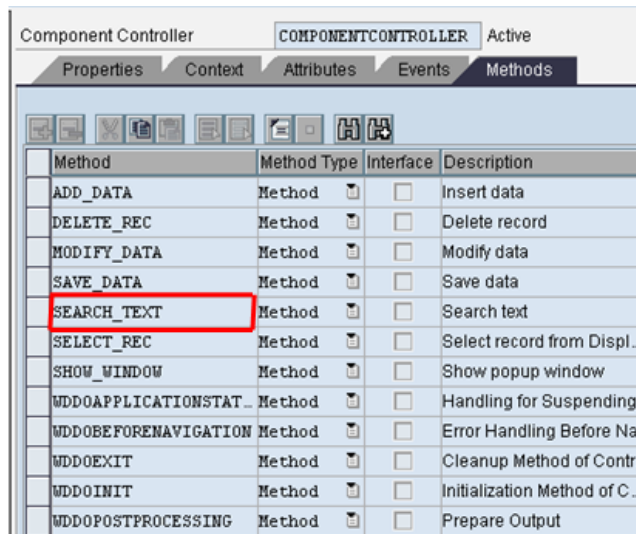
Fig. 3.4 Structura componentei Y_WD_BUCOMP



Fig. 3.5 Structura ROOTUIELEMENTCONTAINER corespunzătoare Layout-ului V_SEARCH_TEXT

Întrucât controller-ele componentei și Windows-ului sunt responsabile cu procesarea datelor, toate metodele corespunzătoare elementelor UI ce necesită cod (ex. butoane, link-uri, input field-uri) sunt programate în ComponentController, figura 3.6. La nivelul metodei din View mai rămâne doar un apel al metodei din ComponentController: `wd_comp_controller->search_text()`, al cărei cod este prezentat în Anexa 1.

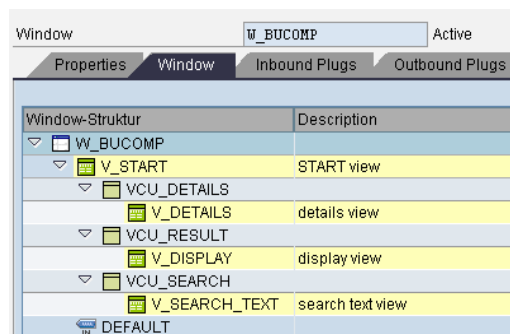
O analiză asupra codului acestei metode de căutare, indică faptul că instrucțiunea Open SQL SELECT efectuează o operație de citire asupra unei vederi din dicționarul ABAP, formată din câmpuri ale tabelor din baza de date SAP CRM. Rezultatele citirii sunt legate de un tabel intern care este afișat în tabelul din V_DISPLAY.



Method	Method Type	Interface	Description
ADD_DATA	Method	<input type="checkbox"/>	Insert data
DELETE_REC	Method	<input type="checkbox"/>	Delete record
MODIFY_DATA	Method	<input type="checkbox"/>	Modify data
SAVE_DATA	Method	<input type="checkbox"/>	Save data
SEARCH_TEXT	Method	<input type="checkbox"/>	Search text
SELECT_REC	Method	<input type="checkbox"/>	Select record from Displ...
SHOW_WINDOW	Method	<input type="checkbox"/>	Show popup window
WDDOAPPLICATIONSTAT...	Method	<input type="checkbox"/>	Handling for Suspending...
WDDOBEFORENAVIGATION	Method	<input type="checkbox"/>	Error Handling Before Na...
WDDOEXIT	Method	<input type="checkbox"/>	Cleanup Method of Contr...
WDDOINIT	Method	<input type="checkbox"/>	Initialization Method of C...
WDDOPOSTPROCESSING	Method	<input type="checkbox"/>	Prepare Output

Fig. 3.6 Metodele ComponentController

Toată structura de View-uri este integrată în containere UI speciale și acestea în Windows (figura 3.7), care face posibilă afișarea lor în momentul execuției aplicației.



Window-Struktur	Description
W_BUCOMP	
V_START	START view
VCU_DETAILS	
V_DETAILS	details view
VCU_RESULT	
V_DISPLAY	display view
VCU_SEARCH	
V_SEARCH_TEXT	search text view
DEFAULT	

Fig. 3.7 Integrarea Views în Windows

Așa cum Controller-ul este responsabil cu procesarea datelor, logica aplicației este legată de model care este realizat din contexte și face legătura (prin Context Binding) cu tabelele bazei de date SAP CRM. Ca și structură arborescentă formată din noduri și atribute, context-ul este declarat global la nivel de ComponentController și mapat de View-uri, (figura 3.8). Datele pot fi astfel transferate cu ușurință de la utilizator în baza de date și invers.

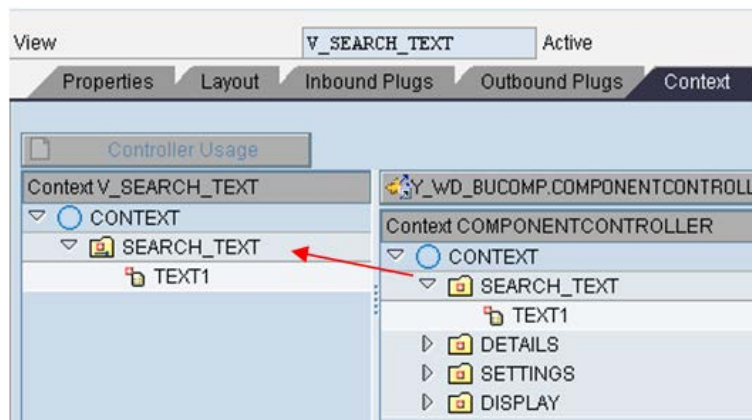


Fig. 3.8 Maparea Context-ului din ComponentController în View

Aplicația WD are ca punct de plecare spre execuție o adresă HTTP generată de către framework-ul Web Dynpro. Acesta realizează comunicarea între interfața aplicației și aplicația propriu-zisă. Modelarea întregii aplicații la nivel de prezentare, cât și la nivel de implementare a logicii se regăsește și în lucrarea autorului [89].

3.2.2. Implementarea aplicației în tehnologia Floorplan Manager

Cu ajutorul framework-ului FPM integrat în ABAP Workbench, aceeași structură de aplicație web dezvoltată în tehnologia Web Dynpro ABAP este implementată și în tehnologia FPM. Întrucât tehnologia FPM este dezvoltată sub tehnologia WD ABAP, aplicația realizată urmărește același concept MVC de separare a părții logice de cea de prezentare a datelor.

Dezvoltarea aplicației pornește de la alegerea modelului floorplan OVP, ce are ca suport componenta FPM_OVP_COMPONENT, care se va configura cu ajutorul editorului de configurare FLUID după structura aplicației WD, (figura 3.9). Inițial, această componentă OVP [90] afișează o imagine de ansamblu a celor mai importante date ale unui obiect business (de exemplu client) cu posibilitate de adăugare, ștergere, modificare și salvare a datelor.

În mediul de execuție, aplicația WD FPM adaugă în secțiunea de configurare, un bloc UIBB pentru căutare, a cărui componentă este FPM_SEARCH_UIBB. Această componentă UIBB, este specializată în selectarea instanței nodului pentru căutare și afișarea rezultatelor. Este necesar doar construirea layout-ului prin definirea câmpurilor și implementarea logicii de căutare.

În modul design, configurarea aplicației se realizează cu ajutorul editorului de configurare, astfel încât componenta FPM_SEARCH_UIBB este adăugată unei secțiuni noi, în care se definesc câmpurile pentru căutare și afișare a rezultatelor în acord cu câmpurile tabelor din CRM, (figura 3.10).

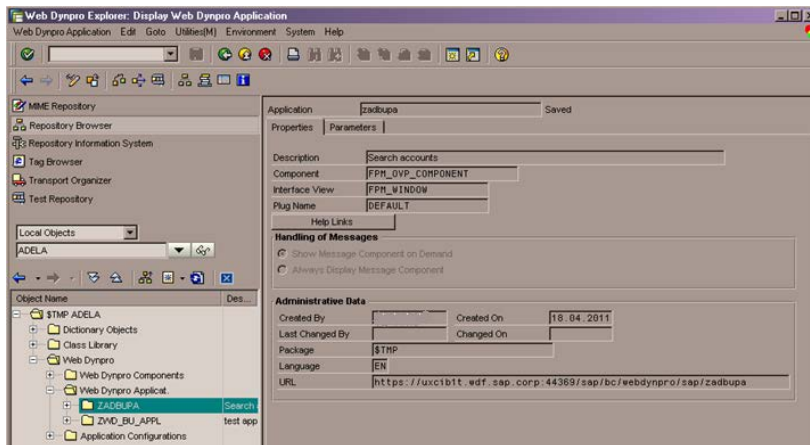


Fig. 3.9 Aplicația WD FPM ZADBUPA

La nivel logic, configurarea componentei se face în clasa feeder în care este definită interogarea. Clasa feeder este creată în biblioteca de clase din Data Dictionary și implementează două interfețe: IF_FPM_GUIBB_SEARCH și IF_FPM_GUIBB, (figura 3.11).

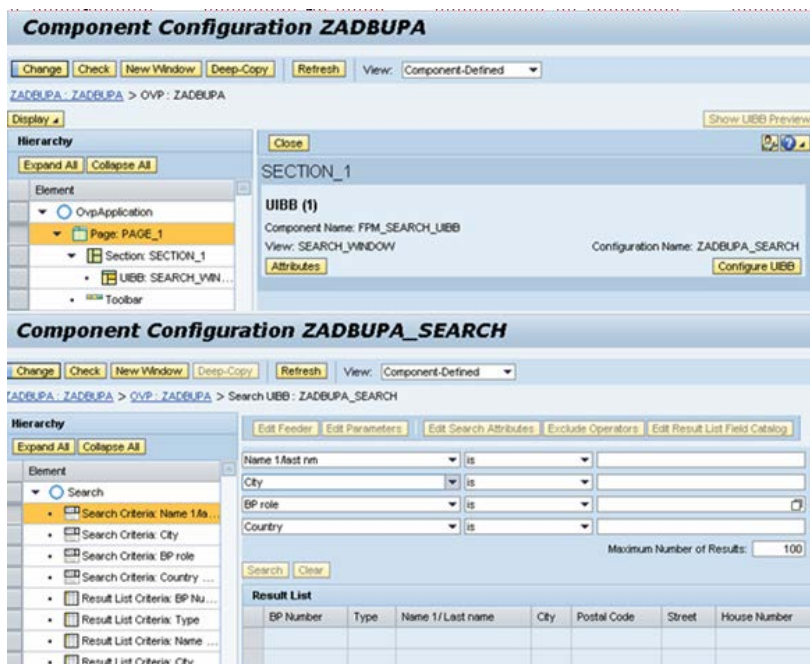


Fig. 3.10 Configurarea componentei ZADBUPA_SEARCH

Cele două interfețe ajută la definirea catalogului de câmpuri și a parametrilor de intrare necesari construirii părții vizuale a aplicației, mai exact a elementelor UI. Structura acestui catalog este una plată (flat structure), câmpurile fiind tipizate conform tipului de dată din tabelele respective. Comunicarea UABB cu catalogul de câmpuri [69] și parametrii de intrare se face prin intermediul

interfețelor implementate în cadrul buclei de evenimente a floorplan-ului și cuprinde definiția aplicației Web Dynpro cu definițiile de date și tabelele corespunzătoare.

Class Interface		ZAD_CL_SEARCH_FEEDER			Implemented / Active
Properties Interfaces Friends Attributes Methods Events Types Aliases					
Filter					
Interface	Abstract	Final	Modeled	Description	
IF_FPM_GUIBB	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Generic UI Building Block	
IF_FPM_GUIBB_SEARCH	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Generic Search UI Building Block	

Fig. 3.11 Implementarea interfețelor în clasa Feeder

Bucula de evenimente cuprinde o secvență de metode predefinite ce aparțin clasei feeder, care este startată în momentul instanțierii floorplan-ului. Acest model pe etape [71] cuprins în bucla de evenimente a floorplan-ului, face ca în timpul execuției, clasa feeder să declanșeze metodele predefinite GET_DEFINITION și GET_DATA. Aceste metode fac posibilă definirea câmpurilor, respectiv furnizarea valorilor pentru parametrii de intrare. Un model UML al diagramelor de clasă este reprezentat în figura 3.12.

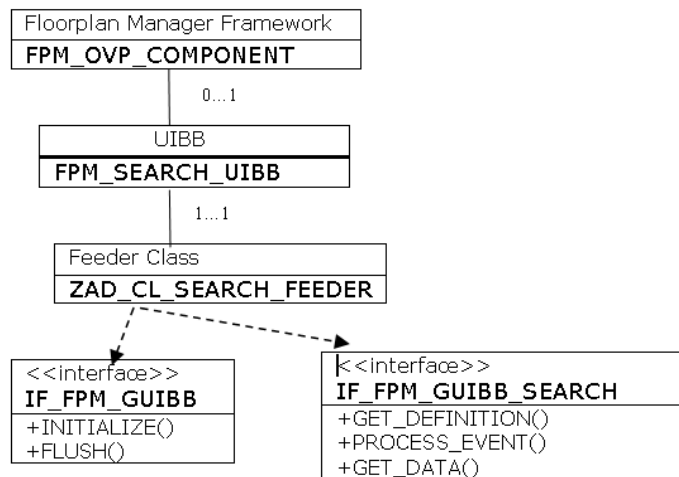


Fig. 3.12 Diagrama de clase UML

În acest context, la nivel logic, structura catalogului de câmpuri pentru UIBB se definește prin parametrii eo_field_catalog_attr și eo_field_catalog_result în metoda GET_DEFINITION (anexa 2a). Există două structuri globale din Data Dictionary care fac legătura cu acești parametri: ZBUPASEARCH și ZBUPARESLT, pentru căutarea, respectiv afișarea rezultatelor.

Informațiile preluate de pe interfața aplicației sunt astfel disponibile în clasa Feeder care le procesează și generează forma în care vor fi afișate rezultatele. În clasa Feeder, logica pentru metoda de căutare urmărește:

3.2. Implementarea aplicației web în tehnologiile WDA, FPM și WebClient UI 59

1. Crearea unui atribut care să păstreze rezultatul căutării (este necesară o stocare temporară, deoarece procesarea evenimentelor și extragerea datelor apar în două faze FPM diferite).

data MT_DATA type ZTBUPARESLT .

2. Implementarea metodei PROCESS_EVENT a interfeței IF_FPM_GUIBB_SEARCH pentru butonul SEARCH al UIBB. Aceasta este principala metodă în clasa feeder, care pentru UIBB-ul de căutare FPM_SEARCH_UIBB verifică valoarea de intrare pentru criteriul de căutare, (anexa 2b).
3. Implementarea metodei GET_DATA corespunzătoare interfeței IF_FPM_GUIBB_SEARCH (anexa 2c). Această metodă e apelată în bucla de evenimente, care redirecționează evenimentul PROCESS_BEFORE_OUTPUT al clasei de alimentare și trimite rezultatul căutării blocului UIBB.

Metodele INITIALIZE() și FLUSH() ale interfeței IF_FPM_GUIBB sunt apelate automat pentru inițializarea framework-ului FPM și în bucla de evenimente la declanșarea unui eveniment.

Ieșirea din buclă conduce la întoarcerea spre aplicația WD a cărei execuție este prezentată în figura 3.13.

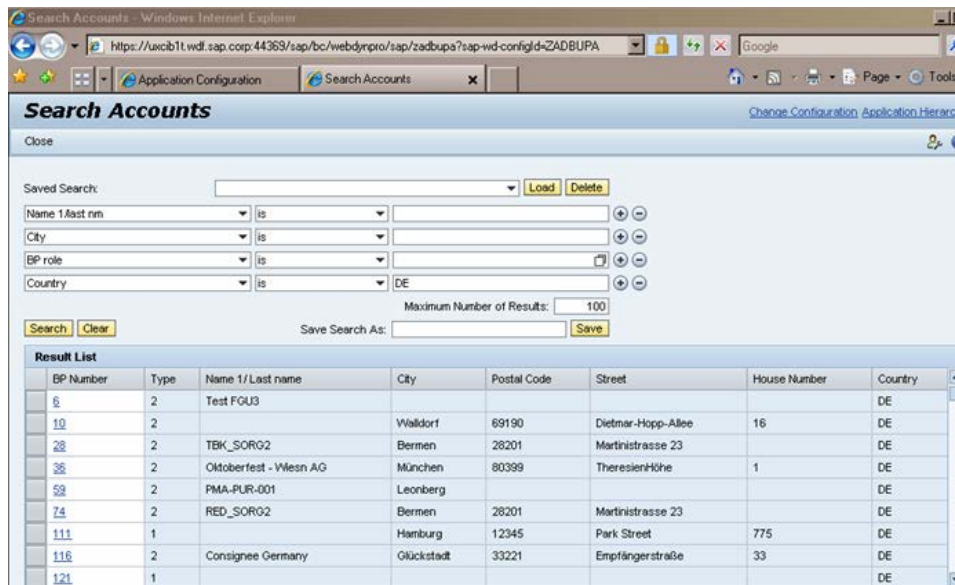


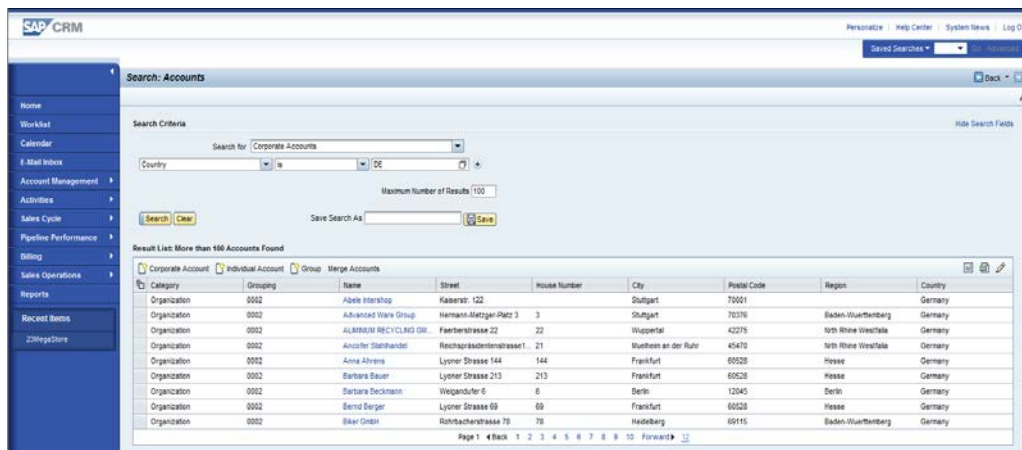
Fig. 3.13 Interfața aplicației implementată în tehnologia FPM

3.2.3. Implementarea aplicației în tehnologia CRM WebClient UI

Aplicația de test construită cu ajutorul framework-ului CRM WebClient UI are la bază una dintre componentele BSP și anume, BP_HEAD_SEARCH (search page). Această pagină oferă utilizatorilor posibilități de căutare pentru orice componentă CRM. Pagina constă din două zone distincte: una pentru căutare (care include o funcție de salvare pentru căutări) și una pentru afișarea rezultatelor. Utilizând link-urile din lista de rezultate ale căutării, utilizatorul poate naviga la paginile cu detalii a obiectelor corespunzătoare.

Prin tehnica de Enhancement și cu ajutorul UI Configuration Tool se vor specializa view-urile acestei componente, în sensul obținerii unei aplicații asemănătoare cu cea realizată în tehnologia Web Dynpro ABAP (un singur câmp de căutare și un tabel cu anumite câmpuri pentru rezultate). Înainte de configurare, se crează un profil UI personal prin tranzacția CRMC_UI_PROFILE. Configurarea aplicației în spațiul personal se realizează cu ajutorul cheii de configurare ROLE CONFIGURATION KEY prin tranzacția CRMC_UI_CONF_UI.

Noua aplicație BSP, YBP_SEARCH va moșteni toate funcționalitățile aplicației de bază BP_HEAD_SEARCH, inclusiv partea de implementare. Întrucât componenta este formată tot din două view-uri (MAINSEARCH și MAINSEARCHRESULT), configurarea componentei se face separat pentru fiecare dintre ele. Din listele cu criterii de căutare disponibile se selectează câmpurile dorite, restul se elimină și se obține astfel o aplicație ca cea din figura 3.14.



Category	Grouping	Name	Street	House Number	City	Postal Code	Region	Country
Organization	0002	Adèle Intershop	Kaiserstr. 122		Stuttgart	70001	Baden-Wuerttemberg	Germany
Organization	0002	Advanced Ware Group	Hermann-Metzger-Platz 3	3	Stuttgart	70376	Baden-Wuerttemberg	Germany
Organization	0002	ALBROUX RECYCLING DM	Faerberstrasse 22	22	Wuppertal	42275	Nord Rhine Westfale	Germany
Organization	0002	Ansofor Stahlhandel	Reichspräsidentenstrasse 1.	21	Wuppertal an der Ruhr	45470	Nord Rhine Westfale	Germany
Organization	0002	Anna Ahrens	Lyoner Strasse 144	144	Frankfurt	60528	Hesse	Germany
Organization	0002	Barbara Bauer	Lyoner Strasse 213	213	Frankfurt	60528	Hesse	Germany
Organization	0002	Barbara Beckmann	Wiegandstr 6	6	Sieff	12045	Berlin	Germany
Organization	0002	Bernd Berger	Lyoner Strasse 89	89	Frankfurt	60528	Hesse	Germany
Organization	0002	Bilal Ginter	Rohlfacherstrasse 78	78	Heidelberg	69115	Baden-Wuerttemberg	Germany

Fig. 3.14 Aplicația BSP YBP_SEARCH

Valoarea parametrului de căutare COUNTRY este preluată de pe interfața aplicației și prelucrată în metoda EH_ONSEARCH, figura 3.15, care stă la baza implementării butonului Search. Citirea din cele trei tabele se regăsește în cod, toată implementarea aparținând framework-ului.

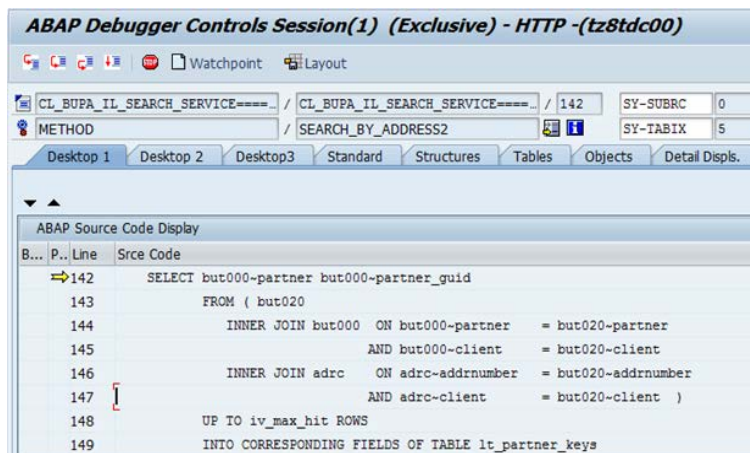


Fig. 3.15 Secvență din codul metodei EH_ONSERACH

O remarcă deosebit de importantă este faptul că toate elementele structurii componentei BP_HEAD_SEARCH sunt mapate în View Layout printr-o clasă și apoi afișate prin pagina htm, componenta fiind legată de Model Object din business layer.

3.3. Analiza performanțelor aplicațiilor web business

Analiza performanțelor unei aplicații este reflectată în înregistrările statistice ABAP. Din datele statistice se pot obține informații detaliate despre: performanța bazei de date, probleme de memorie, timp de procesare, etc. Aceste date statistice sunt create în procesul de lucru (WP – work processes) de server-ul de aplicații ABAP și salvate într-un fișier special. Pentru analiza performanțelor sistemului ABAP, mediul SAP dispune de instrumente specializate în acest scop și anume: ABAP Runtime Analysis, Memory Analysis și Business Transaction Analysis.

Analiza efectuată cu ajutorul *ABAP Runtime Analysis* este utilă pentru calculul și evaluarea codului scris la realizarea cerințelor de business. De obicei, această analiză este legată de testul de calitate și asigură performanța unui program, metodă sau modul funcțional executat pe server-ul de producție. Rezultatul acestei analize prezentat și în lucrările autorului [91], [92], [93] poate fi salvat sub forma unui fișier grafic, în care apare timpul (în microsec.) necesar: execuției programului ABAP, a lucrului cu baza de date și a timpului utilizat exclusiv în sistem. Cea mai mare pondere ar trebui să o aibă timpul necesar execuției programului ABAP. Aceste fișiere sunt utilizate pentru a realiza optimizările necesare.

Instrumentul de analiză *Memory Analysis* este utilizat pentru a afișa nivelul actual al consumului de memorie. În acest scop, instrumentul prevede două ecrane

de afișare cu informații diferite: unul pentru dimensiunile memoriei (în bytes) și altul pentru obiectele de memorie [93].

Cel mai utilizat instrument pentru analiza performanței sistemului SAP este *Business Transaction Analysis* (accesibil prin tranzacția STAD), care calculează utilizarea resurselor sistemului pentru sistemul local ABAP și oferă o analiză detaliată a unei tranzacții pentru o aplicație [94]. Datele sunt luate direct din fișierele statistice. Criteriile de selecție includ: utilizatorul, tranzacția, programul, tipul de activitate, data începerii și ora de începere. Analiza timpilor calculați și afișați în figura 3.16 sunt: timpul la nivel de procesor (CPUT) și timpul total în procesul de lucru, sau timpul de răspuns (RT) cu componentele sale: timp de așteptare pentru procesul de lucru, timp de procesare, de încărcare, de generare, timp de acces al bazei de date (DBRT) și timp de punere în coada de așteptare. Dintre aceștia, timpii cu impactul cel mai mare asupra performanței unei aplicații îl au: RT, DBRT și CPUT.

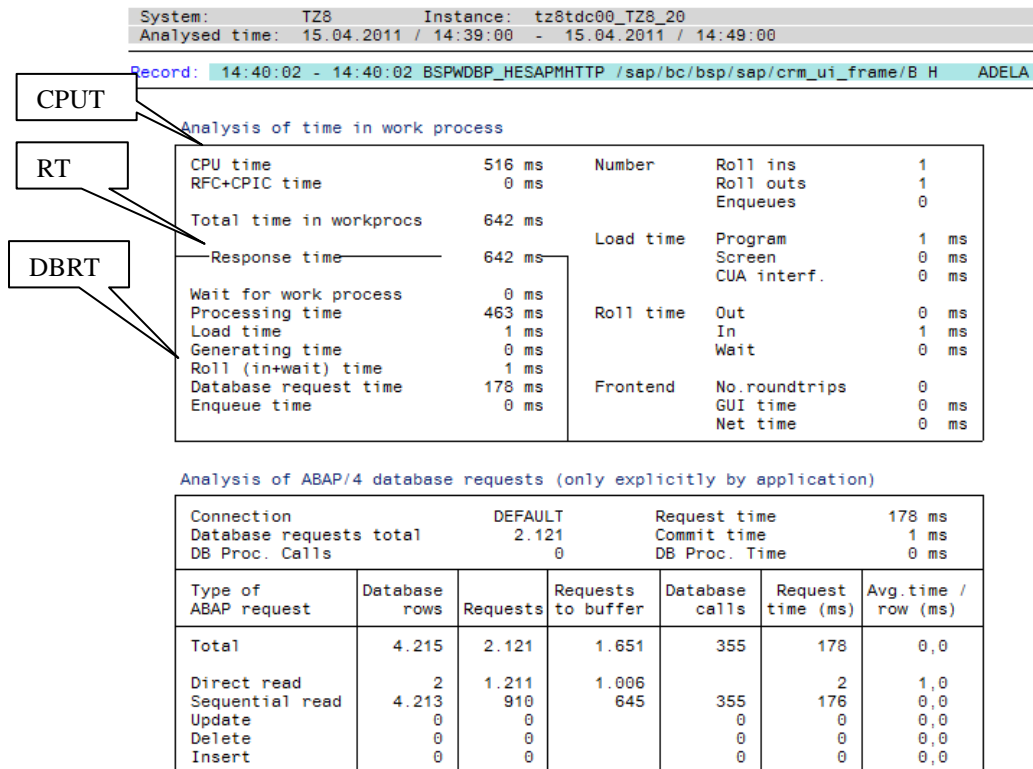


Fig. 3.16 Analiza timpului în procesul de lucru, afișat prin tranzacția STAD

Timpul de răspuns sau **Response Time (RT)**, se referă la timpul necesar pentru a procesa o cerere. Măsurarea timpului începe atunci când o cerere ajunge la serverul de aplicații ABAP și se termină atunci când prelucrarea este completă. Acest timp de răspuns este măsurat de ABAP Kernel și nu se referă la timpul de care este nevoie până când un utilizator primește un răspuns de la server.

CPU time reprezintă timpul consumat de procesor la sfârșitul unui pas de tranzacție. CPU time nu este determinat de sistemul de operare și nu este o componentă adițională a tranzacției timpului de răspuns, dar e consumator al timpului de încărcare, al timpului nominal (roll time) și timpului de procesare (processing time). Acest timp depinde de temporizatorul (cronometrul) sistemului de operare și poate fi identificat ca fiind timpul necesar pentru execuția instrucțiunilor ABAP [27].

DB request time este timpul necesar citirii sau schimbării datelor în baza de date. Acest timp este măsurat din momentul în care o solicitare este trimisă la server-ul bazei de date până în momentul în care datele sunt returnate server-ului de aplicații. Deoarece DB request time este măsurat de către server-ul de aplicații, el include nu numai timpul necesar operării în baza de date, ci și timpul necesar pentru transferul datelor în rețea. Valoarea acestui timp, depinde într-o mare măsură de timpul de execuție al rețelei și de încărcarea server-ului bazei de date [27].

În figura 3.17 și figura 3.18 sunt prezentate componentele timpului de răspuns la nivelul sistemului, cu evidențierea CPU time. În general, timpul de răspuns este aproximativ egal cu suma dintre CPU time și timpul DB request time [95], [96].

Astfel:

- dacă DB request time este considerabil mai mare decât CPU time, atenția trebuie concentrată asupra accesului bazei de date;
- dacă CPU time este considerabil mai mare decât DB request time, atunci este necesară o investigare a codului ABAP și apoi o analiză cu ajutorul instrumentului ABAP Runtime Analysis;
- dacă cele două valori ale DB request time și CPU time sunt apropiate, atunci trebuie analizate ambele.

Din studiile efectuate de către autor și experiența acumulată alături de colaboratorii SAP din Germania, în problemele legate de performanță, s-a observat că de cele mai multe ori este necesară o optimizare locală și nu o modificare în masă a programului realizat.

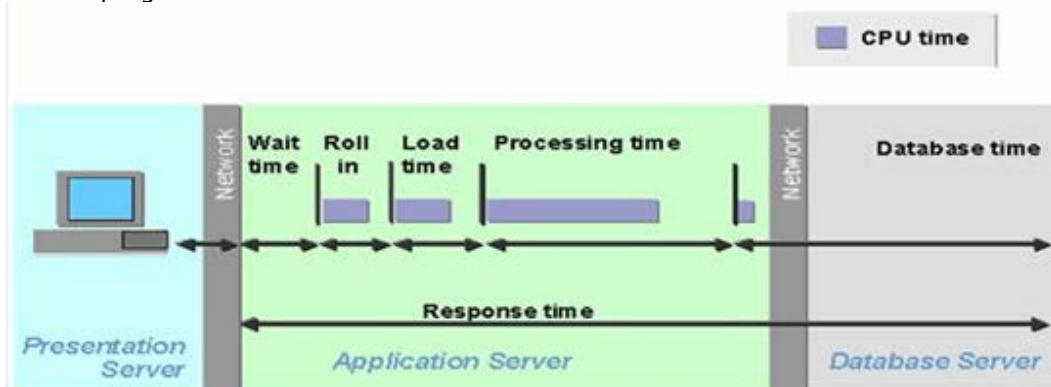


Fig. 3.17 Distribuția timpilor la nivelul server-ului SAP AS [27].

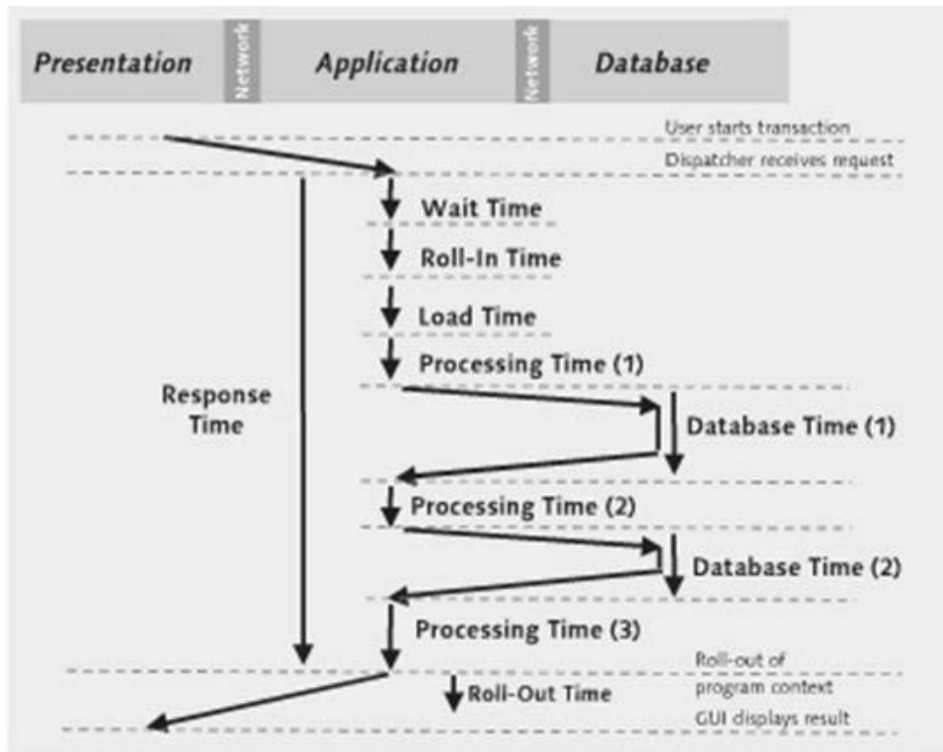


Fig. 3.18 Componentele Timpului de răspuns – Response Time [95]

Un alt instrument utilizat pentru monitorizare și analiză a traficului HTTP, care poate oferi informații detaliate despre o aplicație web care se încarcă în browser, este HTTP-Watch pentru Microsoft Internet Explorer și Firefox [97]. Cu ajutorul acestuia s-a obținut timpul total din front-end la încărcarea aplicației web, figura 3.19.

Funcție	Contor	Timp inclusi...	Timp exclusiv (ms)	URL	Număr linie
JScript - window script block	8	2.425,24	0,00		
crmFrmAdjustScrollAreaOnLoad	1	135,01	0,00		
thtmlbMenuIIInitialize	1	130,01	0,00		
onclick	1	105,01	0,00		

Fig. 3.19 Captură HTTP Watch pentru testarea aplicației implementate în CRM WebClient UI

Una dintre principalele caracteristici ale acestui plug-in este că afișează toate conexiunile și fișierele care sunt transferate atunci când aplicația web este încărcată în Internet Explorer.

În Anexele 3, 4 și 5 sunt prezentate măsurătorile parametrilor: Response Time (RT), CPU time (CPUT), DB request time (DBRT) și HTTP-Watch time (HTTPWT), pentru testele efectuate cu ajutorul instrumentelor Business Transactions Analysis (transaction STAD) și HTTP Watch. Testele sunt realizate pentru aplicațiile implementate în cele trei tehnologii SAP UI [94] pentru trei cazuri diferite, în care este interogată baza de date SAP CRM (mai exact tabelele BUT000, BUT020 și ADRC care conțin 9973 înregistrări), solicitându-se:

cazul a) afișarea primelor 100 rezultate;

cazul b) afișarea tuturor rezultatelor, tabelate pe câte 9 rânduri;

cazul c) afișarea tuturor rezultatelor, tabelate pe câte 100 rânduri.

Valorile măsurate sunt prezentate în tabelele din Anexele 3, 4 și 5.

3.4. Concluzii

Prin implementarea aplicației în toate cele trei tehnologii SAP UI la nivelul server-ului ABAP, se constată că:

- aplicația propusă face parte din categoria de aplicații utilizate în toate proiectele IT ce conțin interfețe specializate în căutare, afișare, modificare și ștergere a datelor din baza de date;
- **în tehnologia Web Dynpro**, aplicația este realizată pe baza unei componente care are la bază structura MVC. Acest model arhitectural presupune legarea elementelor UI corespunzătoare nivelului de prezentare (View Layout) cu modelul de date a cărui structură este reprezentată prin tabelele bazei de date și cu logica implementată la nivelul controller-ului;
- **în tehnologia FPM**, aplicația are la bază modelul floorplan OVP și un bloc UIBB, care la nivelul clasei feeder, ce joacă rolul de Controller, implementează logica corespunzătoare de căutare și afișare. Construirea părții vizuale și comunicarea UIBB cu modelul de date din baza de date se realizează prin intermediul interfețelor corespunzătoare clasei feeder, implementate în cadrul buclei de evenimente a floorplan-ului;
- **în tehnologia WebClient UI**, aplicația care are la bază componenta search page (pagina de căutare), configurează partea vizuală, căutarea și afișarea datelor conform modelului bazei de date se declanșează la nivelul framework-ului, prin recunoașterea parametrului preluat de pe interfața paginii de căutare.

Valorile parametrilor RT, CPU, DBRT și HTTPWT obținute prin testarea aplicațiilor în tehnologiile WD ABAP, FPM și CRM WebClient UI, cu instrumentele Business Transaction Analysis și HTTP Watch pentru Internet Explorer, vor fi prelucrate în capitolele următoare în vederea evaluării și ierarhizării tehnologiilor amintite.

4. Prelucrarea statistică a datelor experimentale

Obiectivul acestui capitol este de a prelucra și interpreta statistic datele experimentale obținute prin testarea aplicațiilor în cele trei tehnologii SAP UI: Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI, din capitolul anterior.

Prin prelucrarea datelor se au în vedere următoarele aspecte:

- determinarea principalilor parametri statistici și eliminarea valorilor afectate de erori grosiere;
- verificarea caracterului aleator al datelor;
- determinarea funcțiilor de distribuție corespunzătoare fiecărui eșantion de date;
- verificarea normalității distribuțiilor pe baza testelor de concordanță;
- generarea unor seturi de date mai mari, necesare validării existenței legăturilor dintre parametri;
- compararea principalilor parametri statistici ai datelor experimentale cu cei ai datelor generate.

4.1. Analiza preliminară a datelor

Deoarece valorile experimentale obținute prin testarea aplicațiilor sunt diferite în cadrul celor trei variante analizate, dar și pe fiecare tehnologie în parte, este necesară prelucrarea preliminară a acestora în sensul eliminării efectelor datorate diferitelor tipuri de erori, precum și a verificării caracterului aleator.

Prelucrarea statistică a parametrului X presupune următoarele etape prezentate în lucrarea autorului [98]:

- **Etapa I** Filtrarea erorilor
- **Etapa II** Determinarea estimației punctuale

4.1.1. Filtrarea erorilor

Tipurile de erori ce pot fi identificate și eliminate prin teste statistice sunt [99]:

- erorile *grosiere* cauzate de încălcarea unor principii generale de măsurare, sau ca urmare a funcționării defectuoase a sistemului de măsurare; valoarea lor diferă semnificativ față de celelalte, procesul de prelucrare statistică a măsurătorilor fiind influențat negativ de aceste rezultate, motiv pentru care trebuie eliminate;
- erorile *sistematice* cauzate de un număr mare de factori ce apar în procesul de măsurare ; depistarea lor necesită de obicei investigații speciale și pot fi eliminate prin aplicarea de corecții asupra rezultatelor măsurătorilor;
- erorile *aleatoare* cauzate de caracterul stohastic al oricărui proces; aceste erori rămân după eliminarea erorilor grosiere și sistematice, sunt inevitabile și nu este necesară înlăturarea lor din cadrul rezultatelor măsurătorilor.

În cazul erorilor aleatoare care apar în majoritatea eșantioanelor de valori experimentale, testele statistice vor urmări gradul în care aceste erori afectează eșantionul respectiv, indicând dacă densitatea de probabilitate a acestuia mai coincide ca formă generală cu aceea a variabilei reale măsurate.

Pentru a afirma că o valoare din setul de valori testate este normală sau îndepărtată de celelalte valori, este necesară găsirea unei metode prin care se poate verifica acest lucru. Verificarea șirurilor de valori pentru eliminarea valorilor cu erori grosiere poartă denumirea de *analiză critică a datelor*. În literatura de specialitate sunt prezentate mai multe metode de filtrare a valorilor eronate cum ar fi: criteriul Chauvenet, testul Irvin (testul λ), testul Grubbs (testul g) și testul Romanowski (testul t) [99, 100, 101].

Pentru eliminarea valorilor ce prezintă erori grosiere s-a considerat relevant testul Grubbs.

Considerăm X parametrul ce urmează a fi analizat în una din cele trei variante ale unei tehnologii și x^* valoarea ce se intuiește că prezintă o eroare grosieră, din setul de valori $\{x_1, x_2, \dots, x_m, x_n\}$ măsurate.

Se parcurg următoarele etape de calcul:

- Se calculează *media aritmetică* a celor n valori:

$$\bar{x}_n = \frac{x_1 + x_2 + \dots + x_m + x_n}{n} = \frac{\sum_{i=1}^n x_i}{n} \quad (4.1)$$

- Se identifică valoarea *îndoielnică* $\{x^*\}$ (una dintre valorile cele mai depărtate de valoarea \bar{x}_n din șirul de valori testate).
- Se calculează abaterea standard pentru setul de date:

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x}_n)^2} \quad (4.2)$$

- Se calculează raportul:

$$g = \frac{|x^* - \bar{x}_n|}{s} \quad (4.3)$$

- Se compară valoarea raportului g cu valoarea $g_n(P)$ din Anexa 6, în funcție de numărul de valori n și nivelul de siguranță P . Pentru seturile de date analizate ce conțin 12 valori pentru fiecare variabilă, s-a considerat nivelul de siguranță $P = 0,95$, rezultând $t_{1,2}(P) = 2,29$.

Sunt posibile următoarele două cazuri:

1. $g > g_n(P)$, ceea ce indică faptul că valorile x^* sunt suspecte, conțin erori grosiere și se vor elimina din prelucrarea ulterioară. Dacă q este numărul valorilor suspecte x^* eliminate, atunci numărul valorilor parametrului X ce se va prelucra va fi:

$m = n - q$, cu condițiile $q > 0$ și $q \ll n$, rezultând valorile: $\{x_1, x_2, x_3, \dots, x_m\}$;

2. $g \leq g_n(P)$, ceea ce arată că toate valorile $\{x_1, x_2, x_3, \dots, x_m, \dots, x_n\}$ sunt valide și în acest caz $m=n$.

În tabelele 4.1, 4.2 și 4.3 se prezintă, ca exemplu, rezultatele obținute după aplicarea primei etape de filtrare și eliminare a erorilor grosiere. Sunt utilizate seturile de date prezentate în Anexele 3, 4, 5, pentru varianta 2.

Tabelul 4.1 Filtrarea și eliminarea valorilor afectate de erori grosiere, din setul de date măsurate în tehnologia WDA

Parametrul	CPUT	DBRT	HTTPWT	RT
Număr valori, n	12	12	12	12
Media, $\overline{x_n}$	179.16	290.75	1271.75	527.92
valoarea îndoielnică, x^*	190	589	2606	1986
Abaterea standard, s	6.69	94.48	424.66	459.30
$g_{12}(P)$	2.29	2.29	2.29	2.29
$g = \frac{ x^* - \overline{x_n} }{s} > g_{12}(P)$	1,62	3,16	3,14	3,17
Nr. valori eliminate	0	1	1	1
Nr. valori rămase m	12	11	11	11

Ținând cont de aceste rezultate, în anexa 3 (varianta 2) se elimină întreaga măsurătoare, ce conține valorile îndoielnice ale parametrilor RT, DBRT și HTTPWT. În prelucrările statistice ulterioare se va prelucra setul ce conține doar 11 valori pentru fiecare parametru.

Tabelul 4.2 Filtrarea și eliminarea valorilor afectate de erori grosiere, din setul de date măsurate în tehnologia FPM

Parametrul	CPUT	DBRT	HTTPWT	RT
Număr valori, n	12	12	12	12
Media, $\overline{x_n}$	264.17	301.33	1640.58	504.83
valoarea îndoielnică, x^*	230	306	1528	477
Abaterea standard, s	17,30	2,35	72,01	16,40
$g_{12}(P)$	2.29	2.29	2.29	2.29
$g = \frac{ x^* - \overline{x_n} }{s} < g_{12}(P)$	1,98	1,99	1,56	1,70
Nr. valori eliminate	0	0	0	0
Nr. valori rămase m	12	12	12	12

Tabelul 4.3 Filtrarea și eliminarea valorilor afectate de erori grosiere, din setul de date măsurate în tehnologia CRM WC UI

Parametrul	CPUT	DBRT	HTTPWT	RT
Număr valori, n	12	12	12	12
Media, $\overline{x_n}$	1966.17	998.25	3036.5	2574.17
valoarea îndoielnică, x^*	2156	1070	4824	2678
Abaterea standard, s	117,89	48,05	1063,87	70,67
$g_{12}(P)$	2.29	2.29	2.29	2.29
$g = \frac{ x^* - \overline{x_n} }{s} < g_{12}(P)$	1,61	1,49	1,68	1,47
Nr. valori eliminate	0	0	0	0
Nr. valori rămase m	12	12	12	12

4.1.2. Determinarea estimației punctuale

Prelucrarea eșantioanelor de date experimentale ce corespund cazurilor prezentate se realizează astfel:

- pentru cazul 1, corespunzător setului de valori $\{x_1, x_2, x_3, \dots, x_m\}$ a parametrului X (având numărul de valori $m = n-q$), se calculează media aritmetică $\overline{x_m}$ cu relația:

$$\overline{x_m} = \frac{x_1 + x_2 + \dots + x_m}{m} = \frac{\sum_{i=1}^m x_i}{m} \quad (4.4)$$

- pentru cazul 2, corespunzător mulțimii de valori $\{x_1, x_2, x_3, \dots, x_m, \dots, x_n\}$ a parametrului X (cu n valori), relația va fi:

$$m = n, \text{ deci } \overline{x_m} = \overline{x_n} \quad (4.5)$$

Analizând valorile parametrilor obținuți din măsurătorile experimentale, pentru fiecare tip de variantă și tehnologie, se calculează m_1 și $\overline{x_{m,1}}$ (pentru Web Dynpro ABAP), m_2 și $\overline{x_{m,2}}$ (pentru Floorplan Manager), m_3 și $\overline{x_{m,3}}$ (pentru CRM WebClient UI). Rezultatele obținute sunt prezentate în tabelele 4.4 ÷ 4.6.

Estimarea valorii x a parametrului X presupune:

- determinarea unei funcții dependente de valorile $\{x_1, x_2, x_3, \dots, x_m, \dots, x_n\}$, $f(x_1, x_2, x_3, \dots, x_m, \dots, x_n)$ suficient de apropiată de

valoarea x . Funcția astfel definită este denumită *estimație punctuală* sau *estimație* a valorii x ;

- b) determinarea intervalului de forma $(f-\varepsilon_1, f-\varepsilon_2)$ care să includă valoarea x cu o probabilitate fixată, denumită *estimație de încredere* și un nivel de încredere sau nivel de siguranță al estimației P [99], [100], [101]. Intervalul $(f-\varepsilon_1, f-\varepsilon_2)$ se numește *interval de încredere*, iar limitele sale sunt denumite *limite de încredere*.

Toate valorile $\{x_1, x_2, x_3, \dots, x_m, \dots, x_n\}$ ale parametrului X au același grad de precizie. În plus erorile aleatoare din cadrul valorilor $\{x_1, x_2, x_3, \dots, x_m, \dots, x_n\}$ prezintă o lege de distribuție normală.

Estimația valorii x a parametrului X analizat va fi de tip punctual [100], determinată pentru fiecare tehnologie în parte, pe baza mediei aritmetice ponderate a celor trei variante componente, cu relația:

$$x = \frac{m_1 \cdot \overline{x_{m,1}} + m_2 \cdot \overline{x_{m,2}} + m_3 \cdot \overline{x_{m,3}}}{m_1 + m_2 + m_3} = \frac{\sum_{i=1}^3 m_i \cdot \overline{x_{m,i}}}{\sum_{i=1}^3 m_i} \quad (4.6)$$

În tabelele 4.4 ÷ 4.6 sunt prezentate estimațiile punctuale pentru valorile parametrilor: Response time, CPU time, DB request time și HTTP Watch time, pe variantele de tehnologii studiate: Web Dynpro ABAP, Floorplan Manger și CRM WebClient UI.

Tabelul 4.4 Parametrii statistici obținuți prin prelucrarea datelor experimentale din tehnologia Web Dynpro ABAP

Varianta	Web Dynpro ABAP		CPU Time (ms)	DB Request Time (ms)	HTTP Watch Time (ms)	Response Time (ms)
1	Număr valori	n_1	12	12	12	12
		m_1	12	11	12	11
	Media aritmetică	\overline{x}_{n1}	135,00	35,42	907,42	162,67
		\overline{x}_{m1}	135,00	28,09	907,42	155,45
2	Număr valori	n_2	12	12	12	12
		m_2	12	11	11	11
	Media aritmetică	\overline{x}_{n2}	179,16	290,75	1271,75	527,92
		\overline{x}_{m2}	179,16	263,64	1150,45	395,36
3	Număr valori	n_3	12	12	12	12
		m_3	12	12	12	12
	Media aritmetică	\overline{x}_{n3}	630,83	263,08	3376,08	845,25
		\overline{x}_{m3}	630,83	263,08	3376,08	845,25
estimația punctuală x			315,00	187,24	1830,2	476,53

Tabelul 4.5 Parametrii statistici obținuți prin prelucrarea datelor experimentale din tehnologia Floorplan Manager

Varianta	Floorplan Manager		CPU Time (ms)	DB Request Time (ms)	HTTP Watch Time (ms)	Response Time (ms)
1	Număr valori	n_1	12	12	12	12
		m_1	12	11	12	11
	Media aritmetică	\bar{x}_{n1}	267,50	302,50	1630,42	509,75
		\bar{x}_{m1}	267,50	300,45	1630,42	507,82
2	Număr valori	n_2	12	12	12	12
		m_2	12	12	12	12
	Media aritmetică	\bar{x}_{n2}	264,17	301,33	1640,58	504,83
		\bar{x}_{m2}	264,17	301,33	1640,58	504,83
3	Număr valori	n_3	12	12	12	12
		m_3	11	12	12	11
	Media aritmetică	\bar{x}_{n3}	664,17	299,5	3492	904,75
		\bar{x}_{m3}	643,64	299,5	3492	884,91
estimația punctuală x			384,57	300,43	2254,33	628,76

Tabelul 4.6 Parametrii statistici obținuți prin prelucrarea datelor experimentale din tehnologia CRM WebClient UI

Varianta	CRM WebClient UI		CPU Time (ms)	DB Request Time (ms)	HTTP Watch Time (ms)	Response Time (ms)
1	Număr valori	n_1	12	12	12	12
		m_1	12	12	12	11
	Media aritmetică	\bar{x}_{n1}	457,17	135,5	3021	554,92
		\bar{x}_{m1}	457,17	135,5	3021	547,73
2	Număr valori	n_2	12	12	12	12
		m_2	12	12	12	12
	Media aritmetică	\bar{x}_{n2}	1966,17	998,25	3036,5	2574,17
		\bar{x}_{m2}	1966,17	998,25	3036,5	2574,17
3	Număr valori	n_3	12	12	12	12
		m_3	12	11	12	11
	Media aritmetică	\bar{x}_{n3}	4324,42	1365,75	11414,5	5128,42
		\bar{x}_{m3}	4324,42	1325,91	11414,5	5.058,27
estimația punctuală x			2249,25	805,43	5824,00	2722,24

Varianta 1 - Interogarea BD și afișarea primelor 100 rezultate;

Varianta 2 - Interogarea BD și afișarea tuturor rezultatelor pe câte 9 rânduri;

Varianta 3 - Interogarea BD și afișarea tuturor rezultatelor pe câte 100 rânduri.

4.1.2. Verificarea caracterului aleator al datelor – Testul Young

Pentru obținerea unor rezultate corecte este necesar ca datele experimentale să fie aleatoare, iar testul Young [102] este unul dintre testele frecvent utilizate pentru verificarea caracterului aleator.

Aplicarea testului pentru un eșantion de date urmărește:

- a) pentru șirul de valori $x_1, x_2, x_3, \dots, x_n$, se calculează mărimea δ^2 și M astfel:

$$\delta^2 = \frac{1}{n-1} \sum_{i=1}^{n-1} (x_{i+1} - x_i)^2 \quad (4.7)$$

$$M = \frac{\delta^2}{s^2} \quad (4.8)$$

în care s este abaterea standard a valorilor măsurate.

- b) mărimea M se compară cu valorile VCI (valoare critică inferioară) și VCS (valoare critică superioară), alese din tabelul 4.7. Dacă este îndeplinită condiția $VCI < M < VCS$, atunci șirul de valori experimentale are un caracter aleator, cu probabilitatea P.

Tabelul 4.7 Valori critice inferioare și superioare ale lui M

n	VCI		VCS		n	VCI		VCS	
	P=0,95	P=0,99	P=0,95	P=0,99		P=0,95	P=0,99	P=0,95	P=0,99
4	0,78	0,53	3,22	3,47	10	1,06	0,75	2,94	3,25
5	0,82	0,54	3,18	3,46	11	1,10	0,79	2,90	3,21
6	0,89	0,56	3,11	3,44	12	1,13	0,83	2,87	3,17
7	0,94	0,61	3,06	3,39	15	1,21	0,92	2,79	3,08
8	0,98	0,66	3,02	3,34	20	1,30	1,04	2,70	2,96
9	1,02	0,71	2,98	3,29	25	1,37	1,13	2,63	2,87

Aplicarea testului Young la valorile eșantioanelor de date (n = 11 respectiv n = 12), la un nivel de încredere P = 0,95, demonstrează caracterul aleator al datelor pentru fiecare variantă și tehnologie în parte, (tabelul 4.8).

Tabelul 4.8 Rezultatele testului Young aplicat eșantionului de date experimentale

Testul Young	M		
Tehnologia	WDA	FPM	CRM
Număr de valori	11	12	12
Valoare Critică Superioară	2.90	2.87	2.87
RT	1.35005	2.41784	1.379564
CPU	1.425926	2.794937	2.5696
DBRT	1.327466	1.994505	1.272253
HTTP	2.77117	2.079446	2.210134
Valoare Critică Inferioară	1.10	1.13	1.13

4.2. Funcții de distribuție

Întrucât valorile obținute prin măsurători experimentale sunt insuficiente pentru a obține informații relevante cu privire la legăturile dintre parametrii analizați (RT, DBRT, CPUT și HTTPWT), este necesară generarea unor seturi mai mari de date. În acest context se vor identifica, funcțiile de distribuție corespunzătoare seturilor de date măsurate și apoi pe baza lor se vor genera noi date.

4.2.1. Funcția de distribuție și densitatea de probabilitate

Funcția de distribuție (repartiție) [103] a variabilei aleatoare X este funcția reală, de variabilă reală, $F: \mathbb{R} \rightarrow \mathbb{R}$, definită prin:

$$F(x) = P(X \leq x) \quad (4.9)$$

în care $(X \leq x)$ reprezintă reuniunea evenimentelor pentru care variabila aleatoare ia valori mai mici sau egale cu x .

Funcția de distribuție este absolut continuă, dacă există o funcție reală, $f: \mathbb{R} \rightarrow \mathbb{R}$, astfel încât:

$$F(x) = \int_{-\infty}^x f(u) du \quad (4.10)$$

Funcția f se numește *densitate de probabilitate* a variabilei aleatoare X , sau probabilitatea cu care o variabilă aleatoare poate lua valori din domeniul ei de variație.

4.2.2. Verificarea normalității datelor experimentale

Deoarece valorile parametrilor mășurați (RT, CPUT, DBRT, HTTPWT) sunt variabile cantitative continue, repartiția acestora ar trebui să fie una normală (Gauss). Din diferite cauze, de cele mai multe ori, această repartiție nu este normală, datele fiind repartizate mai mult sau mai puțin normal. Verificarea normalității distribuției are la bază teste de concordantă. În general, testele de concordantă sunt teste care permit luarea unei decizii dacă un eșantion de date provine sau nu dintr-o populație statistică, ce se supune unui anumit tip de distribuție de frecvență. Pentru aceeași distribuție, aceste teste evidențiază diferențele dintre repartiția experimentală și cea teoretică.

În continuare se va face referire la trei dintre aceste teste de concordantă și anume: Kolmogorov-Smirnov, Anderson-Darling și Hi-pătrat (Chi-Squared).

4.2.2.1. Testul Kolmogorov-Smirnov (K-S)

Testul Kolmogorov-Smirnov, [104], [105], poate fi folosit pentru verificarea ipotezei că un eșantion de date urmează o anumită lege de distribuție, precum și pentru compararea legilor de distribuție ale populațiilor din care provin două eșantioane [106].

Testul K-S este bazat pe funcția de distribuție empirică.

Funcția de distribuție cumulativă empirică (ECDF) pentru distribuții continue este definită prin relația (4.10), iar pentru distribuții discrete cu n valori y_1, y_2, \dots, y_n , este definită prin relația:

$$E_n = \frac{n(i)}{n} \quad (4.11),$$

în care $n(i)$ este numărul de valori mai mici decât y_i , y_i fiind ordonate de la cea mai mică la cea mai mare valoare. E_n este o funcție care crește cu $1/N$ pentru fiecare valoare ordonată.

Distribuția statistică a acestui test nu depinde de funcția de distribuție cumulativă în curs de testare.

Există anumite limitări impuse acestui test și anume:

- se aplică numai pentru distribuții continue;
- tinde să fie mai sensibilă în apropierea centrului de distribuție decât la capete;
- distribuția trebuie să fie pe deplin specificată. Pentru cazul în care localizarea, scara și forma parametrilor se estimează pe baza datelor, regiunea critică a testului K-S nu mai este valabilă. De obicei trebuie determinată prin simulare.

Datorită limitărilor b și c, mulți dintre analiști preferă să utilizeze testul Anderson-Darling [107]. Acesta este valabil doar pentru anumite distribuții.

Testul statistic Kolmogorov-Smirnov D, se calculează cu relația:

$$D = \max_{1 \leq i \leq n} \left[F(x_i) - \frac{i-1}{n}, \frac{i}{n} - F(x_i) \right] \quad (4.12)$$

pentru distribuții continue, sau pentru distribuții discrete, cu relația:

$$D = \max_{1 \leq i \leq n} \left| F(y_i) - \frac{i}{n} \right| \quad (4.13)$$

unde F este distribuția teoretică cumulativă a distribuției în curs de testare, care trebuie să fie o distribuție normală, specificată în totalitate.

Ipoteza referitoare la forma distribuției este respinsă dacă testul statistic D este mai mare decât valoarea critică, tabelată. Valorile fixe (0.01, 0.05, etc.) sunt în general utilizate pentru a evalua ipoteza nulă H_0 (datele urmăresc o distribuție specifică) la diferite nivele de semnificație. Valoarea 0,05 este de obicei folosită pentru cele mai multe aplicații.

Valoarea "p" (p value), care exprimă gradul de credibilitate al ipotezei, se calculează luând în considerare valoarea D și ipoteza H_0 .

Dacă ipoteza H_0 este adevărată, valoarea p reprezintă probabilitatea ca statistica unui eșantion oarecare de date, D , să fie cel puțin egală cu statistica eșantionului curent de date. O valoare mică a lui p indică faptul că eșantionul curent este foarte improbabil, deci corespondența este eronată și trebuie respinsă. Invers, o valoare mare a lui "p" duce la acceptarea corespondenței, funcția de distribuție caracterizând datele analizate. Valoarea lui "p" este utilă, în special

atunci când ipoteza nulă este respinsă la toate nivelele de semnificație predefinite și trebuie cunoscut la ce nivel ar putea fi acceptată.

4.2.2.2. Testul Anderson - Darling (A-D)

Testul Anderson-Darling [107] este utilizat pentru a testa dacă un eșantion de date provine de la o populație cu o anumită distribuție. Comparativ cu testul Kolmogorov-Smirnov, acesta acordă o importanță mai mare pe capete. În calcularea valorilor critice, testul Anderson-Darling ține cont de distribuția testată. Avantajul este că această analiză permite un test mai sensibil, însă dezavantajul este că valorile critice trebuie calculate pentru fiecare distribuție.

Pentru distribuțiile cele mai frecvent utilizate: normală, lognormală, exponențială, Weibull, valoare extremă tipul I și distribuții logistice, există tabele de valori critice [108].

Testul statistic Anderson-Darling este definit prin relația:

$$A^2 = -n - \frac{1}{n} \sum_{i=1}^n (2i-1) * [\ln F(X_i) + \ln(1-F(X_{n-i+1}))] \quad (4.14)$$

în care F este funcția de distribuție cumulativă a distribuției analizate, iar X_i reprezintă datele ordonate. Valorile critice în acest caz, sunt dependente de distribuția testată.

La testul Anderson-Darling, ipoteza că distribuția are o anumită formă este respinsă, dacă valoarea testului A este mai mare decât valoarea critică.

Testul Anderson-Darling implementat de soft-ul EasyFit [109] utilizează aceleași valori critice pentru toate distribuțiile. Aceste valori depind numai de mărimea eșantionului.

4.2.2.3. Testul Hi-pătrat (Chi Squared)

La fel ca și testul D-A, testul Hi-pătrat [110] este utilizat pentru a testa dacă un eșantion de date provine de la o populație cu o anumită distribuție. Acesta poate fi aplicat la orice distribuție pentru care se poate calcula funcția de distribuție cumulativă. Testul de corespondență Hi-pătrat se aplică doar datelor grupate în clase.

Există însă câteva dezavantaje ale acestui test, și anume:

- valorile statistice ale testului sunt dependente de modul în care datele sunt grupate;
- eșantionul de n valori trebuie să fie suficient de mare pentru ca aproximarea Hi-pătrat să fie valabilă.

EasyFit folosește următoarele relații pentru numărul k de grupări și pentru calculul statistic:

$$k = 1 + \log_2 n \quad (4.15)$$

$$\chi^2 = \sum_{i=1}^k \frac{(O_i - E_i)^2}{E_i} \quad (4.16)$$

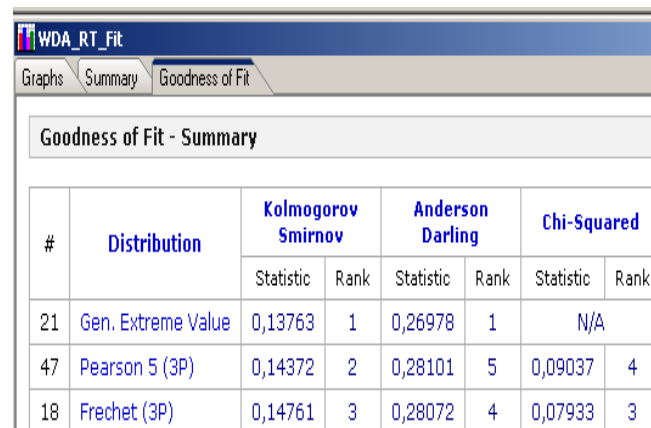
$$E_i = F(x_2) - F(x_1) \quad (4.17)$$

în care O_i și E_i sunt frecvențele observate respectiv așteptate ale grupării i , F este funcția de distribuție cumulativă a densității de probabilitate testate, iar x_1 și x_2 limitele grupării i .

4.2.3. Identificarea distribuțiilor de probabilitate aferente datelor analizate și generarea de valori pe baza acestora

În vederea identificării funcțiilor de distribuție corespunzătoare eșantioanelor de date analizate s-a utilizat soft-ul EasyFit [109], cu ajutorul căruia se pot combina metodele clasice de analiză statistică cu tehnici moderne de prelucrare a datelor, în vederea obținerii celor mai bune rezultate. Procesul de fitare a datelor oferă un sumar al rezultatelor analizelor sub formă de grafice generate și tabele interactive, atât pentru funcțiile de distribuție, cât și pentru valorile parametrilor acestor funcții.

Variantele de funcții de distribuție pe care acest soft le simulează pentru datele de test sunt prezentate în figurile 4.1÷4.12. Tabelele din figuri afișează primele trei distribuții obținute pentru valorile măsurătorilor (RT, DBRT, CPU, HTTPWT), în ordinea rank-urilor corespunzătoare testelor de concordanță K-S, A-D și Chi Squared din EasyFit. Testul Kolmogorov-Smirnov al cărui rol este de a măsura distanța minimă dintre funcția empirică de distribuție și funcția de distribuție cumulativă definită anterior, este considerat prioritar pentru majoritatea dintre funcțiile selectate. Acest test are la bază o teorie bine dezvoltată și poate fi utilizat cu succes pentru eșantioane cu număr mic de valori.



#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
21	Gen. Extreme Value	0,13763	1	0,26978	1	N/A	
47	Pearson 5 (3P)	0,14372	2	0,28101	5	0,09037	4
18	Frechet (3P)	0,14761	3	0,28072	4	0,07933	3

Fig. 4.1 Distribuții ale eșantionului de date RT - Web Dynpro ABAP

#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
18	Frechet (3P)	0,15722	1	0,26952	1	0,14502	13
47	Pearson 5 (3P)	0,15783	2	0,27357	2	0,13891	12
20	Gamma (3P)	0,17	3	3,9603	47	N/A	

Fig. 4.2 Distribuții ale eșantionului de date DBRT - Web Dynpro ABAP

#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
10	Error	0,27325	1	0,9422	3	N/A	
23	Gen. Pareto	0,27333	2	1,0402	27	N/A	
27	Inv. Gaussian	0,27876	3	0,96076	9	N/A	

Fig. 4.3 Distribuții ale eșantionului de date CPU - Web Dynpro ABAP

#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
37	Log-Logistic (3P)	0,11852	1	0,24545	5	0,06998	16
32	Laplace	0,12913	2	0,23942	2	0,0811	27
27	Hypersecant	0,13221	3	0,24008	4	0,08051	26

Fig. 4.4 Distribuții ale eșantionului de date HTTPWT - Web Dynpro ABAP

#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
21	Gen. Extreme Value	0,20581	1	0,6903	1	0,04785	1
24	Gen. Pareto	0,21621	2	0,71088	2	0,09697	2
37	Log-Pearson 3	0,25146	3	1,2369	3	1,2935	4

Fig. 4.5 Distribuții ale eșantionului de date RT - Floorplan Manager

#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
4	Cauchy	0,16393	1	0,29437	1	7,3271E-5	7
32	Laplace	0,16533	2	0,30013	2	5,3711E-4	15
37	Log-Logistic (3P)	0,16947	3	0,35054	8	1,9489E-4	13

Fig. 4.6 Distribuții ale eșantionului de date DBRT -Floorplan Manager

#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
4	Cauchy	0,25	1	15,4	56	N/A	
21	Gen. Extreme Value	0,25531	2	0,87377	1	N/A	
1	Beta	0,2726	3	10,66	50	N/A	

Fig. 4.7 Distribuții ale eșantionului de date CPUT - Floorplan Manager

The screenshot shows the 'Goodness of Fit - Summary' window for 'FPM_HTTP_Fit'. It contains a table with the following data:

#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
58	Uniform	0,11142	1	0,18339	2	0,00515	6
11	Error	0,11146	2	0,1832	1	0,00515	7
1	Beta	0,11582	3	1,5175	38	7,1290E-10	3

Fig. 4.8 Distribuții ale eșantionului de date HTTPWT - Floorplan Manager

The screenshot shows the 'Goodness of Fit - Summary' window for 'CRM_RT_Fit'. It contains a table with the following data:

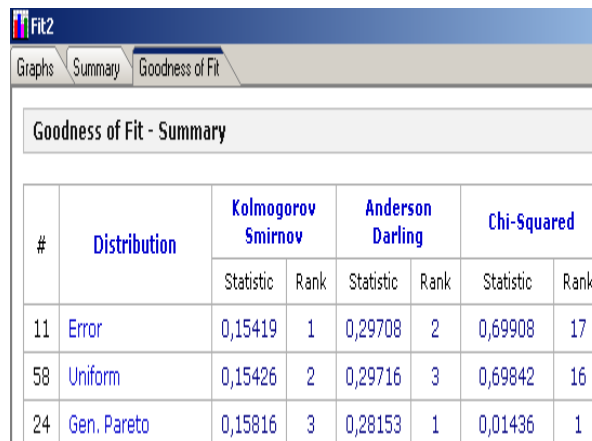
#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
30	Johnson SB	0,12567	1	0,29822	1	0,5203	35
24	Gen. Pareto	0,13293	2	0,31752	4	0,47883	33
21	Gen. Extreme Value	0,13375	3	0,33622	5	0,4242	17

Fig. 4.9 Distribuții ale eșantionului de date RT - CRM WebClient UI

The screenshot shows the 'Goodness of Fit - Summary' window for 'CRM_DB_Fit'. It contains a table with the following data:

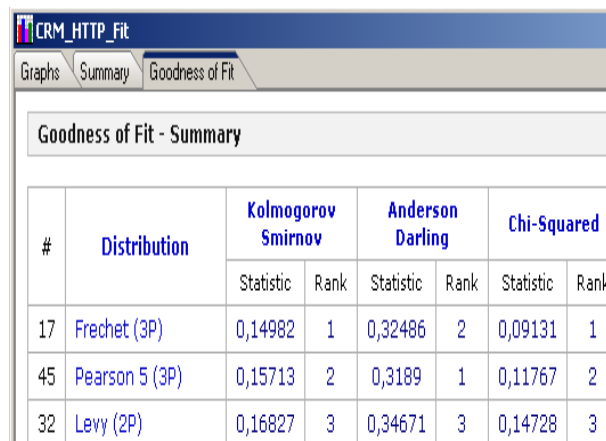
#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
24	Gen. Pareto	0,1539	1	0,37942	4	0,02615	9
30	Johnson SB	0,15435	2	0,3578	1	0,10857	15
1	Beta	0,15859	3	3,855	45	N/A	

Fig. 4.10 Distribuții ale eșantionului de date DBRT - CRM WebClient UI



#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
11	Error	0,15419	1	0,29708	2	0,69908	17
58	Uniform	0,15426	2	0,29716	3	0,69842	16
24	Gen. Pareto	0,15816	3	0,28153	1	0,01436	1

Fig. 4.11 Distribuții ale eșantionului de date CPUT - CRM WebClient UI



#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
17	Frechet (3P)	0,14982	1	0,32486	2	0,09131	1
45	Pearson 5 (3P)	0,15713	2	0,3189	1	0,11767	2
32	Levy (2P)	0,16827	3	0,34671	3	0,14728	3

Fig. 4.12 Distribuții ale eșantioanelor de date HTTPWT - CRM WebClient UI

În tabelul 4.9 sunt prezentate rezultatele testelor statistice, identificate ca fiind cele mai bune, "Goodness of Fit", pentru eșantioanele analizate, cu valorile parametrilor D (coloana statistic) și p (probabilitate) corespunzătoare.

Valorile mici ale lui p obținute pentru parametrul CPUT în tehnologiile WD ABAP și FPM indică faptul că eșantionul curent este foarte improbabil și este necesar să se analizeze un eșantion de date mult mai mare pentru obținerea unor rezultate concludente. Comparând valorile lui p obținute în toate cele trei tehnologii pentru parametrii analizați, se poate afirma că pentru valorile mai mari de 0,85, funcțiile de distribuție identificate caracterizează eșantionul de date analizat.

Tabelul 4.9 Valorile D și p pentru cele mai bune distribuții identificate a valorilor măsurate

Tehnologii		WebDynpro ABAP		Floorplan Manager		CRM WebClient UI	
Parametrii / Distribuții		statistic	p-value	statistic	p-value	statistic	p-value
RT	GEV	0,1376	0,9672	0,2058	0,6186	0,1337	0,9634
DBRT	Frechet (3P)	0,1572	0,9104				
	Cauchy			0,1639	0,8532		
	Gen. Pareto					0,1539	0,8981
CPUT	Gen.Pareto	0,2733	0,3225			0,1582	0,8798
	GEV			0,2553	0,3532		
HTTPWT	Laplace	0,1291	0,982				
	Uniform			0,1114	0,9943		
	Pearson 5 (3P)					0,1586	0,8778

Figurile 4.13÷4.18 prezintă graficele de tip histogramă PDF (Probability Density Function) a funcțiilor de repartiție, pentru fiecare dintre seturile de parametri, în cele trei tehnologii SAP UI. Funcția teoretică PDF este afișată ca o curbă continuă, scalată corespunzător în funcție de numărul de intervale, iar funcția PDF empirică este afișată ca o histogramă formată din bare verticale.

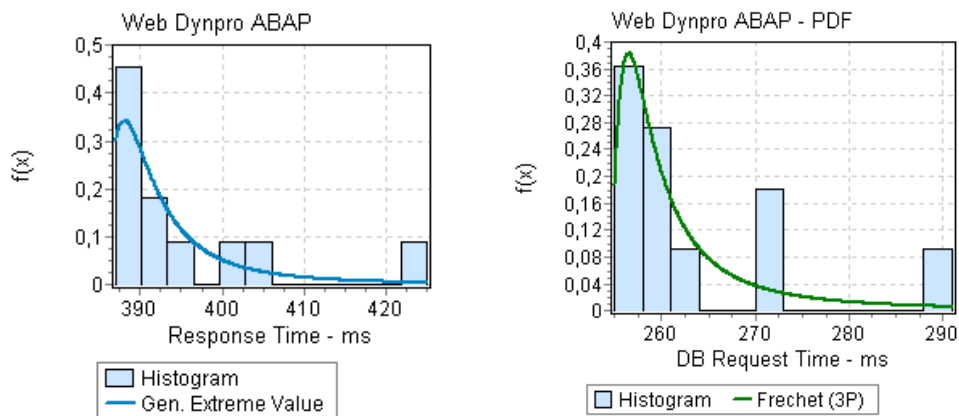


Fig. 4.13 Graficele PDF pentru RT și DBRT în Web Dynpro ABAP

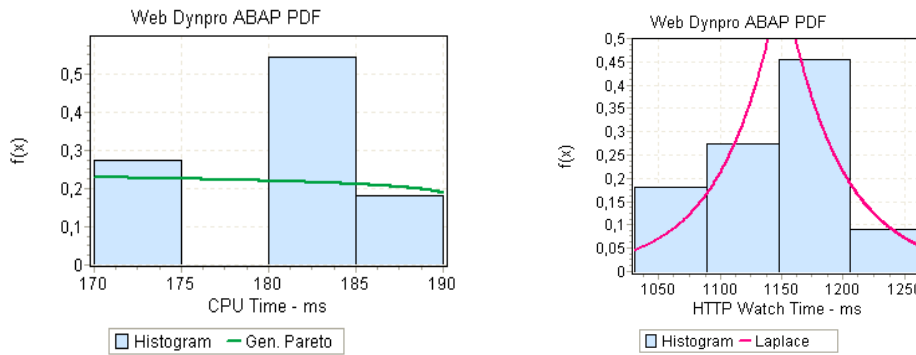


Fig. 4.14 Graficele PDF pentru CPUT și HTTPWT în Web Dynpro ABAP

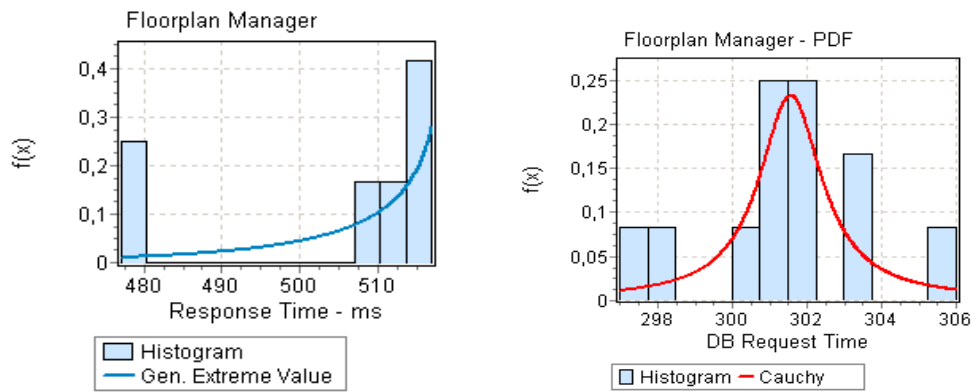


Fig. 4.15 Graficele PDF pentru RT și DBRT în Floorplan Manager

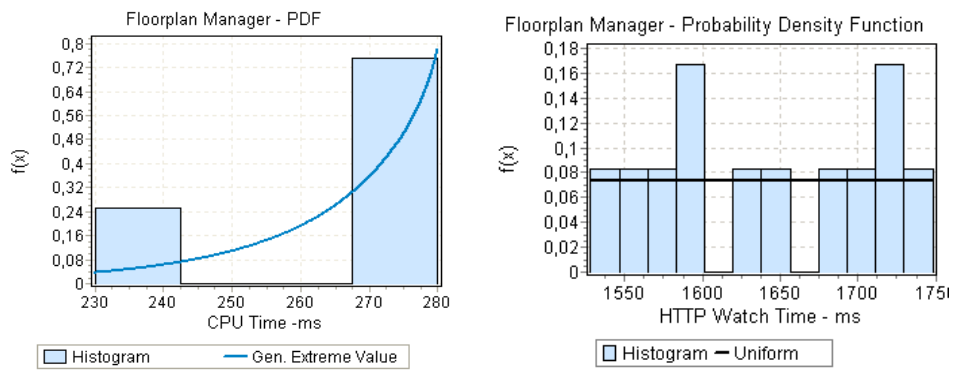


Fig. 4.16 Graficele PDF pentru CPUT și HTTPWT în Floorplan Manager

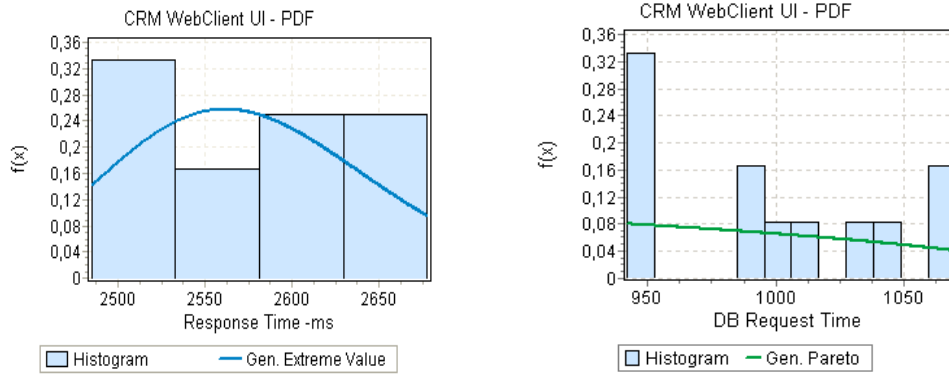


Fig. 4.17 Graficele PDF pentru RT și DBRT în CRM WebClient UI

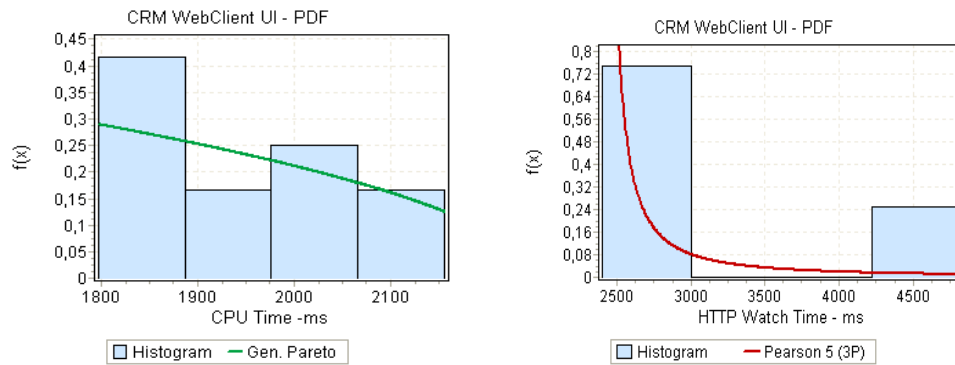


Fig. 4.18 Graficele PDF pentru CPU Time și HTTP Watch Time în CRM WebClient UI

Tabelul 4.10 prezintă domeniile și ecuațiile funcțiilor de distribuție pentru seriile de valori analizate.

Generarea de noi date pe baza funcțiilor de distribuție stabilite se realizează cu ajutorul instrumentului EasyFit integrat, Random Numbers. Seturile de date generate, prezentate în Anexele 7 ÷ 9, sunt prelucrate statistic prin același procedeu descris în paragraful 4.1 și apoi comparate cu seturile de date măsurate. În urma eliminării valorilor afectate de erori grosiere, din cele 100 valori generate rămân:

- pentru tehnologia Web Dynpro ABAP 96 valori;
- pentru tehnologia Floorplan Manager 99 valori;
- pentru tehnologia CRM WebClient UI 96 valori.

Valorile rezultate sunt prelucrate cu programul de calcul DataFit în capitolul următor, în vederea obținerii coeficienților de corelație dintre parametri și a stabilirii corelațiilor analitice și grafice dintre aceștia.

Tabelul 4.10 Ecuatiile funcțiilor de distribuție

Distribuția	Domeniu	Ecuatii
Generalized Extreme Value (GEV)	$1 + k \frac{(x - \mu)}{\sigma} > 0$ $-\infty < x < +\infty$	PDF $f(x) = \begin{cases} \frac{1}{\sigma} \exp(-(1+kz)^{-1/k}) (1+kz)^{-1-1/k} & k \neq 0 \\ \frac{1}{\sigma} \exp(-z - \exp(-z)) & k = 0 \end{cases}$ CDF $F(x) = \begin{cases} \exp(-(1+kz)^{-1/k}) & k \neq 0 \\ \exp(-\exp(-z)) & k = 0 \end{cases} \quad z \equiv \frac{x - \mu}{\sigma}$
Cauchy	$-\infty < x < +\infty$	PDF $f(x) = \left(\pi \sigma \left(1 + \left(\frac{x - \mu}{\sigma} \right)^2 \right) \right)^{-1}$ CDF $F(x) = \frac{1}{\pi} \arctan\left(\frac{x - \mu}{\sigma}\right) + 0,5$
Frechet (3P)	$\gamma \leq x < +\infty$	PDF $f(x) = \frac{\alpha}{\beta} \left(\frac{\beta}{x - \gamma} \right)^{\alpha+1} \exp\left(-\left(\frac{\beta}{x - \gamma}\right)^\alpha\right)$ CDF $F(x) = \exp\left(-\left(\frac{\beta}{x - \gamma}\right)^\alpha\right)$
Generalized Pareto	$\mu \leq x < +\infty$	PDF $f(x) = \begin{cases} \frac{1}{\sigma} \left(1 + k \frac{(x - \mu)^{-1/k}}{\sigma} \right) & k \neq 0 \\ \frac{1}{\sigma} \exp\left(-\frac{(x - \mu)}{\sigma}\right) & k = 0 \end{cases}$ CDF $F(x) = \begin{cases} 1 - \left(1 + k \frac{(x - \mu)^{-1/k}}{\sigma} \right) & k \neq 0 \\ 1 - \exp\left(-\frac{(x - \mu)}{\sigma}\right) & k = 0 \end{cases}$
Uniform	$a \leq x \leq b$	PDF $f(x) = \frac{1}{b - a}$ CDF $F(x) = \frac{x - a}{b - a}$
Pearson 5 (3P)	$\gamma < x < +\infty$	PDF $f(x) = \frac{\exp(-\beta/(x - \gamma))}{\beta \Gamma(\alpha) ((x - \gamma)/\beta)^{\alpha+1}}$ CDF $F(x) = 1 - \frac{\Gamma_{\beta/(x - \gamma)}(\alpha)}{\Gamma(\alpha)}$
Laplace	$-\infty < x < +\infty$	PDF $f(x) = \frac{\lambda}{2} \exp(-\lambda x - \mu)$ CDF $F(x) = \begin{cases} \frac{1}{2} \exp(-\lambda(\mu - x)) & x \leq \mu \\ 1 - \frac{1}{2} \exp(-\lambda(x - \mu)) & x > \mu \end{cases}$

În tabelul 4.11 sunt prezentate valorile parametrilor D (coloana statistic) și p (probabilitate) corespunzătoare distribuțiilor pentru care s-au generat datele.

Tabelul 4.11 Valorile parametrilor statistici D și p pentru seturile de date generate

Tehnologii		Web Dynpro ABAP		Floorplan Manager		CRM WebClient UI	
		statistic	p	statistic	p	statistic	p
RT	GEV	0,053	0,9273	0,0593	0,852	0,0458	0,9785
DBRT	Frechet (3P)	0,0590	0,8556				
	Cauchy			0,0469	0,9727		
	Gen.Pareto					0,0506	0,9485
CPUT	Gen.Pareto	0,0517	0,9394			0,0505	0,9490
	GEV			0,0534	0,9230		
HTTPWT	Laplace	0,0501	0,9521				
	Uniform			0,0520	0,93638		
	Pearson 5					0,0762	0,5799

Din analiza statistică obținută pentru datele generate, (tabelul 4.11), comparativ cu analiza efectuată asupra datelor măsurate, (tabelul 4.9), se poate observa că în tehnologia Web Dynpro ABAP și FPM, un eșantion mai mare de date analizate, face ca valoarea probabilității "p" pentru parametrul CPUT să crească semnificativ.

4.3. Compararea informațiilor statistice a datelor experimentale și a datelor generate

4.3.1. Informații statistice despre datele măsurate

În tabelele 4.12, 4.13 și 4.14 sunt prezentate date statistice care s-au obținut prin prelucrarea statistică a parametrilor din capitolul 3, utilizând programul DataFit.

Tab. 4.12 Date statistice obținute pentru setul de măsurători aparținând aplicației realizate prin tehnologia Web Dynpro ABAP (11 valori)

Variable	CPUT	DBRT	HTTPWT	RT
Number of Points	11	11	11	11
Missing Points	0	0	0	0
Maximum Value	190	291	1264	425
Minimum Value	170	255	1031	387
Range	20	36	233	38
Average	179.0909091	263.6363636	1150.454545	395.3636364
Standard Deviation	7.006490497	10.70768628	64.52807705	11.28071564

Tab.4.13 Date statistice obținute pentru setul de măsurători aparținând aplicației realizate prin tehnologia Floorplan Manager

Variable	CPUT	DBRT	HTTPWT	RT
Number of Points	12	12	12	12
Missing Points	0	0	0	0
Maximum Value	280	306	1748	517
Minimum Value	230	297	1528	477
Range	50	9	220	40
Average	264.1666667	301.3333333	1640.583333	504.8333333
Standard Deviation	17.29862492	2.348435972	72.01446614	16.40306697

Tab. 4.14 Date statistice obținute pentru setul de măsurători aparținând aplicației realizate prin tehnologia CRM WebClient UI

Variable	CPUT	DBRT	HTTPWT	RT
Number of Points	12	12	12	12
Missing Points	0	0	0	0
Maximum Value	2156	1070	4824	2678
Minimum Value	1797	942	2402	2485
Range	359	128	2422	193
Average	1966.166667	998.25	3036.5	2574.166667
Standard Deviation	117.892735	48.05134564	1063.874864	70.67059737

4.3.2. Informații statistice despre datele generate

În tabelele 4.15, 4.16 și 4.17 sunt prezentate date statistice obținute cu programul DataFit pentru setul de măsurători generate corespunzătoare celor 3 tehnologii analizate.

Tabelul 4.15 Date statistice obținute pentru setul de măsurători generate, utilizând tehnologia Web Dynpro ABAP (96 valori pentru fiecare variabilă)

Variable	CPUT	DBRT	HTTPWT	RT
Number of Points	96	96	96	96
Missing Points	0	0	0	0
Maximum Value	190	303	1276	424
Minimum Value	168	255	1000	386
Range	22	48	276	38
Average	178.65625	262.375	1146.34375	393.1770833
Standard Deviation	6.610721953	8.736192113	54.3607498	7.438158787

Tabelul 4.16 Date statistice obținute cu programul DataFit pentru setul de măsurători generate utilizând tehnologia Floorplan Manager (99 valori pentru fiecare variabilă)

Variable	CPUT	DBRT	HTTPWT	RT
Number of Points	99	99	99	99
Missing Points	0	0	0	0
Maximum Value	281	642	1765	519
Minimum Value	207	284	1520	445
Range	74	358	245	74
Average	264.4040404	306.1010101	1639.525253	505.2727273
Standard Deviation	16.36816172	36.3400872	74.06110872	15.41587982

Tabelul 4.17 Date statistice obținute cu programul DataFit pentru setul de măsurători generate utilizând tehnologia Customer Relationship Management WebClient UI (96 valori pentru fiecare variabilă)

Variable	CPUT	DBRT	HTTPWT	RT
Number of Points	96	96	96	96
Missing Points	0	0	0	0
Maximum Value	2223	1102	125405	2752
Minimum Value	1788	925	2398	2432
Range	435	177	123007	320
Average	1961.46875	996.3645833	5703.364583	2571.583333
Standard Deviation	120.4368324	49.15499006	14817.21616	70.55399823

În tabelul 4.18 sunt prezentate comparativ valorile abaterilor standard ale parametrilor mășurați și generați corespunzătoare celor 3 tehnologii utilizate.

Tabelul 4.18 Abateri standard ale parametrilor mășurați și generați

Nr.valori	Variable	CPUT	DBRT	HTTPWT	RT
	Tehnologia	<i>Parametri mășurați</i>			
11	WDA	7.006490497	10.70768628	64.52807705	11.28071564
12	FPM	17.29862492	2.348435972	72.01446614	16.40306697
12	CRM WCUI	117.892735	48.05134564	1063.874864	70.67059737
		<i>Parametri generați</i>			
96	WDA	6.610721953	8.736192113	54.3607498	7.438158787
99	FPM	16.36816172	36.3400872	74.06110872	15.41587982
96	CRM WCUI	120.4368324	49.15499006	14817.21616	70.55399823

Observații:

- În general se observă o scădere a abaterii standard pentru parametri generați față de parametrii mășurați. Această observație arată că distribuțiile utilizate la generarea valorilor parametrilor sunt cele corecte.
- tehnologia WDA este cea pentru care se obțin cele mai mici valori ale abaterii standard a valorilor măsurate sau generate.

4.4. Concluzii

Analiza statistică a datelor conduce la formularea următoarelor concluzii:

- Este necesară verificarea valabilității și a corectitudinii seturilor de date măsurate. Acest aspect este realizat prin eliminarea valorilor afectate de erori grosiere și prin verificarea caracterului aleator al datelor.
- Este necesară generarea de seturi mai mari de date pentru verificarea existenței și a corectitudinii unor posibile corelații între parametrii experimentali obținuți prin testarea aplicațiilor. Acest aspect este realizat prin alegerea corectă a funcțiilor de distribuție corespunzătoare datelor măsurate.
- Este necesară verificarea corectitudinii funcțiilor de distribuție alese. Acest aspect este reliefat prin compararea parametrilor statistici ai datelor generate cu cei ai datelor măsurate, din care se observă o îmbunătățire a acestora.
- Noile valori ale parametrilor generați sunt utilizate ulterior pentru a verifica dacă se obțin aceleași dependențe între aceștia ca și dependențele dintre parametri mășurați.
- Funcțiile de distribuție alese pentru generarea de noi date sunt cele corecte deoarece valoarea parametrului "p" se îmbunătățește.

Având în vedere aceste aspecte se consideră că analiza statistică a datelor este necesară pentru stabilirea corectă a dependențelor dintre parametrii, abordată în capitolul următor.

5. Analiza dependențelor dintre parametrii RT, CPU, DBRT și HTTPWT

Pe baza măsurătorilor efectuate cu ajutorul Business Transaction Analysis pentru aplicațiile implementate în cele trei tehnologii SAP UI, a prelucrării statistice prezentate în capitolul anterior și a lucrării autorului [111], se efectuează o analiză corelațională și de regresie pentru seturile de date experimentale și generate.

Obiectivele acestei analize presupun:

- obținerea de informații referitoare la corelațiile dintre parametrii datelor experimentale și dintre parametrii datelor generate prin analizarea matricelor coeficienților de corelație;
- stabilirea funcțiilor analitice și a graficelor de dependență dintre parametrii (analiza de regresie);
- compararea funcțiilor analitice de corelație în vederea stabilirii parametrilor dependenți și independenți.

Scopul analizei corelaționale și de regresie este de a determina relațiile de dependență dintre parametrii, informațiile obținute fiind necesare la implementarea algoritmilor pentru ierarhizarea tehnologiilor în capitolul următor.

5.1. Analiza datelor experimentale

5.1.1. Coeficienții de corelație dintre parametrii măsurați

Corelația este o metodă statistică utilizată pentru determinarea relațiilor dintre două sau mai multe variabile.

Coeficientul de corelație reprezintă o valoare cantitativă care dă o măsură a relației dintre două sau mai multe variabile. El variază între [-1 și +1], unde valorile extreme presupun o relație perfectă între variabile, în timp ce 0 înseamnă o lipsă totală a relației de dependență.

Coeficientul de corelație simplă măsoară gradul de dependență dintre două variabile.

În cazul corelației liniare simple, pentru care ecuația dreptei de regresie este $y = a + b \cdot x$, se pot determina prin analiza de regresie panta b și ordonata la origine a prin relațiile:

$$b = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} \quad (5.1)$$

$$a = \bar{y} - b \cdot \bar{x} \quad (5.2),$$

În aceste relații \bar{x} și \bar{y} sunt valorile medii ale variabilei independente x și ale variabilei dependente y .

O mărime importantă ce dă informații despre împrăștierea punctelor față de dreapta de regresie este coeficientul de corelație, exprimat de relația [101]:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \cdot \sum (y_i - \bar{y})^2}} \quad (5.3)$$

Pătratul coeficientului de corelație este coeficientul de determinare r^2 .

Coeficientul de corelație multiplă măsoară gradul de dependență dintre variabilele cauză și variabilele efect, fiind din acest punct de vedere un indice de calitate al modelului.

Coeficientul de corelație multiplă se calculează cu relația:

$$R = \frac{\sqrt{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}}{\sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (5.4)$$

în care sunt folosite valorile măsurate y_i , valorile calculate \hat{y}_i și valoarea medie \bar{y} a variabilei dependente.

Coeficientul de determinare multiplă R^2 este definit ca fiind pătratul coeficientului de corelație multiplă.

În cazul corelației duble descrisă de ecuația:

$$y = a + b \cdot x_1 + c \cdot x_2 \quad (5.5),$$

coeficientul de determinare multiplă poate fi calculat și cu ajutorul coeficienților de corelație simplă dintre variabilele y și x_1 , y și x_2 , respectiv x_1 și x_2 , conform relației [112]:

$$R^2 = \frac{r_{yx_1}^2 + r_{yx_2}^2 - 2r_{yx_1} \cdot r_{yx_2} \cdot r_{x_1x_2}}{1 - r_{x_1x_2}^2} \quad (5.6)$$

în care r_{yx_1} , r_{yx_2} și $r_{x_1x_2}$ sunt coeficienții de corelație simplă dintre variabilele y , x_1 și x_2 , din matricea coeficienților de corelație.

Tabelele 5.1, 5.2 și 5.3 prezintă matricele coeficienților de corelație obținute prin prelucrarea statistică a seturilor de date prezentate în varianta 2 a anexelor 3, 4, 5, utilizând programul DataFit.

Tab. 5.1. Matricea coeficienților de corelație pentru setul de date aparținând aplicației realizate prin tehnologia Web Dynpro ABAP

	CPUT	DBRT	HTTPWT	RT
CPUT	1	0.115115749	-0.45463055	0.118469717
DBRT	0.115115749	1	0.023419788	0.995488799
HTTPWT	-0.45463055	0.023419788	1	-0.024840293
RT	0.118469717	0.995488799	-0.02484029	1

Tab. 5.2. Matricea coeficienților de corelație pentru setul de date aparținând aplicației realizate prin tehnologia Floorplan Manager

	CPUT	DBRT	HTTPWT	RT
CPUT	1	0,164103723	-0,64066246	0,931783156
DBRT	0,164103723	1	0,004658659	0,348486537
HTTPWT	-0,640662466	0,004658659	1	-0,650372318
RT	0,931783156	0,348486537	-0,65037231	1

Tab. 5.3. Matricea coeficienților de corelație pentru setul de date aparținând aplicației realizate prin tehnologia CRM WebClient UI

	CPUT	DBRT	HTTPWT	RT
CPUT	1	0,176244645	-0,522588161	0,226866731
DBRT	0,176244645	1	0,186564999	0,98711007
HTTPWT	-0,522588161	0,186564999	1	0,117922972
RT	0,226866731	0,98711007	0,117922972	1

Observații

Analiza coeficienților de corelație din tabelele 5.1, 5.2 și 5.3 arată că:

1. În cazul utilizării tehnologiei Web Dynpro ABAP (cu 11 valori ale fiecărei variabile) există o corelație semnificativă doar între RT și DBRT, cu coeficientul de corelație $r=0,995$.
2. În cazul utilizării tehnologiei Floorplan Manager există o corelație semnificativă doar între RT și CPUT. Valoarea coeficientului de corelație este $r=0,932$.
3. În cazul utilizării tehnologiei CRM WebClient UI există o corelație semnificativă doar între RT și DBRT. Valoarea coeficientului de corelație este $r=0,987$.
4. Indiferent de tehnologia utilizată nu există o corelație semnificativă între HTTPWT și parametrii CPUT, DBRT sau RT.
5. Indiferent de tehnologia utilizată nu există o corelație semnificativă între CPUT și DBRT,

De aceste observații se ține seama la stabilirea corelațiilor analitice și grafice dintre parametrii măsurați.

5.1.2. Stabilirea corelațiilor analitice și grafice dintre parametrii măsurați

Analiza regresională construiește un model al dependenței dintre variabile. Această dependență, implicit cea dintre fenomenele care au avut ca efect apariția unor anumite valori pentru variabilele în cauză, se descrie printr-o relație analitică care poartă numele de *funcție de regresie, sau model de regresie*.

Determinarea coeficienților de corelație dintre variabilele RT, CPUT, DBRT și HTTPWT permite analiza regresională a acestor date.

Datorită faptului că în general la stabilirea dependențelor nu se cunosc toți factorii implicați, construirea funcției de regresie se realizează în anumite ipoteze simplificatoare.

Analiza seturilor de date prezentate în Anexele 3, 4, 5 pentru varianta 2, se poate efectua din următoarele puncte de vedere:

1. *din punctul de vedere a unei singure variabile dependente, care poate fi de exemplu timpul de răspuns RT din setul de date, sau*
2. *din punctul de vedere al ansamblului de attribute ce caracterizează un element, de exemplu: RT, CPUT, DBRT, HTTPWT.*

Analiza de regresie a acestor date conduce la obținerea unor modele analitice (funcții de corelație) care definesc setul de date [113].

a) În cazul utilizării tehnologiei Web Dynpro ABAP cu 11 valori pentru fiecare variabilă, există o dependență, cu coeficient de corelație foarte bun, doar între variabilele RT și DBRT.

Funcția de regresie liniară simplă dintre variabila dependentă RT și variabila independentă DBRT, determinată cu ajutorul programului DataFit, este descrisă de relația:

$$RT = 1,049DBRT + 118,871 \quad (5.7)$$

Coeficientul de corelație simplă este $r=0,995$, iar coeficientul de determinare este $r^2 = 0,991$.

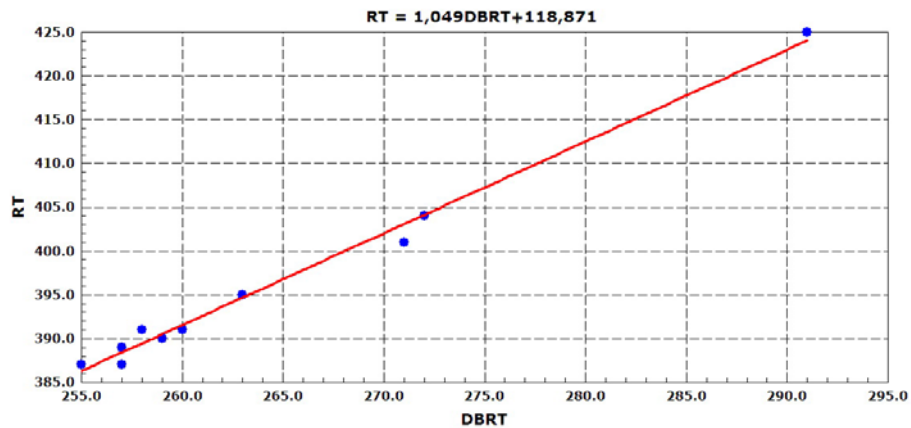
Funcția de regresie liniară simplă arată că o scădere a timpului consumat pentru accesarea bazei de date cu 100 ms determină o scădere a timpului de răspuns cu 104,9 ms.

În figura 5.1 este prezentată dependența liniară a timpului de răspuns RT în funcție de timpul necesar citirii sau schimbării datelor în baza de date, DBRT.

Dacă se analizează dependența RT de DBRT și CPUT se obține, utilizând programul DataFit, *funcția de regresie multiplă*:

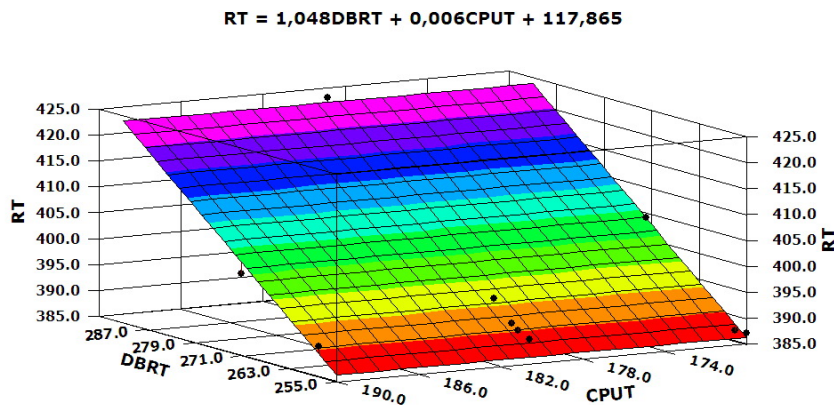
$$RT = 1,048DBRT + 0,006CPUT + 117,865 \quad (5.8)$$

Coeficientul de determinare multiplă este $R^2=0,991$.

Fig. 5.1. Funcția de regresie liniară $RT=f(DBRT)$ pentru tehnologia Web Dynpro ABAP

Funcția de regresie multiplă arată că o scădere a timpului consumat de procesor pentru efectuarea tranzacției (CPUT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 0,6 ms, în timp ce o scădere a timpului consumat pentru accesarea bazei de date (DBRT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 104,8 ms. Timpul de răspuns este mai sensibil la scăderea DBRT decât la scăderea CPUT. Se poate afirma că RT depinde semnificativ doar de DBRT.

În figura 5.2 este prezentată suprafața $RT=f(DBRT,CPUT)$ pentru tehnologia *Web Dynpro ABAP*.

Fig. 5.2. Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia Web Dynpro ABAP

b) **În cazul utilizării tehnologiei Floorplan Manager** există o dependență doar între variabilele RT și CPUT.

Funcția de regresie liniară simplă dintre variabila dependentă RT și variabila independentă CPUT, determinată cu ajutorul programului DataFit, este descrisă de relația:

$$RT = 0,883CPUT + 271,430 \quad (5.9)$$

Coeficientul de corelație simplă este $r=0,932$, iar coeficientul de determinare este $r^2 = 0,869$.

Funcția de regresie liniară simplă arată că o scădere a timpului consumat de procesor pentru efectuarea tranzacție cu 100 ms determină o scădere a timpului de răspuns cu 88,3 ms.

În figura 5.3 este prezentată dependența liniară a timpului de răspuns RT în funcție de timpul consumat de procesor la sfârșitul unui pas de tranzacție CPUT.

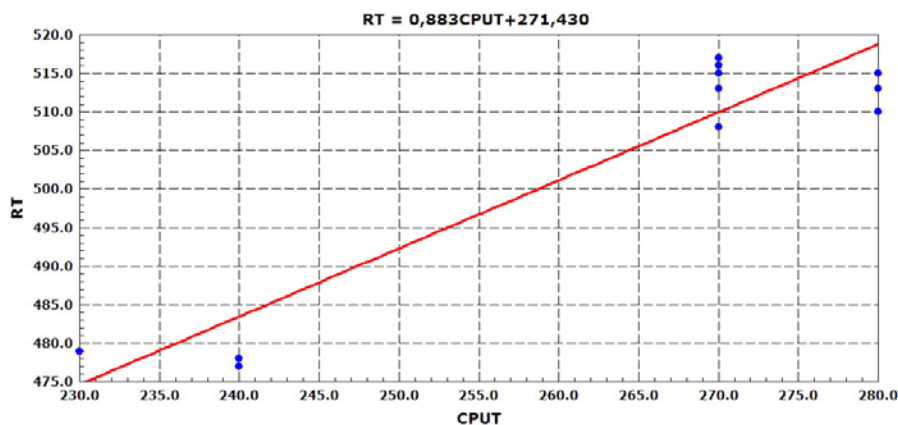


Fig. 5.3. Funcția de regresie $RT=f(CPUT)$ pentru tehnologia Floorplan Manager

Dacă se analizează dependența RT de CPUT și DBRT se obține, utilizând programul DataFit, *funcția de regresie multiplă*:

$$RT = 0,852CPUT + 1,404DBRT - 143,335 \quad (5.10)$$

Coeficientul de determinare multiplă este $R^2=0,907$.

Funcția de regresie multiplă arată că o scădere a timpului consumat de procesor pentru efectuarea tranzacției (CPUT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 85,2 ms, în timp ce o scădere a timpului consumat pentru accesarea bazei de date (DBRT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 140,4 ms. Timpul de răspuns este mai sensibil la scăderea DBRT decât la scăderea CPUT.

În figura 5.4 este prezentată suprafața $RT=f(CPUT, DBRT)$ pentru tehnologia **Floorplan Manager**.

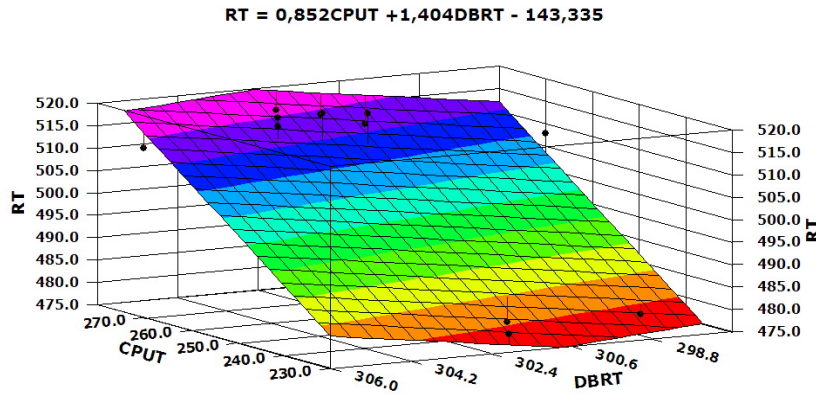


Fig. 5.4. Funcția de regresie $RT=f(CPUT, DBRT)$ pentru tehnologia Floorplan Manager

c) **În cazul utilizării tehnologiei CRM WebClient UI** există o dependență, cu coeficient de corelație semnificativ, doar între variabilele RT și DBRT. *Funcția de regresie liniară simplă* dintre variabila dependentă RT și variabila independentă DBRT, determinată cu ajutorul programului DataFit, este descrisă de relația:

$$RT = 1,452DBRT + 1124,934 \tag{5.11}$$

Coeficientul de corelație simplă este $r=0,987$, iar coeficientul de determinare este $r^2 = 0,974$.

Funcția de regresie liniară simplă arată că o scădere a timpului consumat pentru accesarea bazei de date cu 100 ms determină o scădere a timpului de răspuns cu 145,2 ms.

În figura 5.5 este prezentată dependența liniară a timpului de răspuns RT în funcție de timpul necesar citirii sau schimbării datelor în baza de date DBRT.

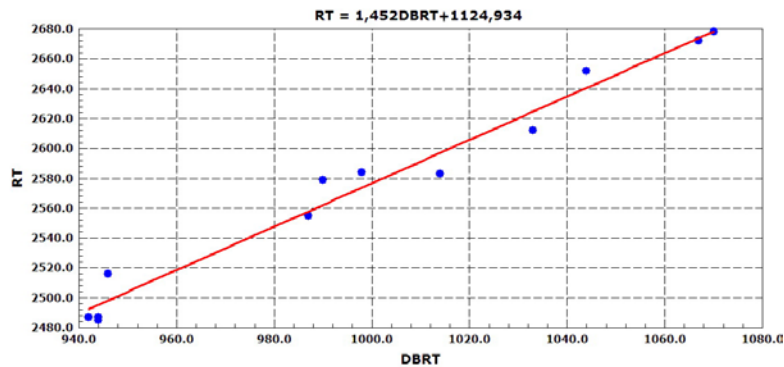


Fig. 5.5. Funcția de regresie $RT=f(DBRT)$ pentru tehnologia CRM WebClient UI

Dacă se analizează dependența RT de DBRT și CPUT se obține, utilizând programul DataFit, *funcția de regresie multiplă*:

$$RT = 1,438DBRT + 0,033CPUT + 1074,719 \quad (5.12)$$

Coeficientul de determinare multiplă este $R^2=0,977$.

Funcția de regresie multiplă arată că o scădere a timpului consumat de procesor pentru efectuarea tranzacției (CPUT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 3,3 ms, în timp ce o scădere a timpului consumat pentru accesarea bazei de date (DBRT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 143,8 ms. Timpul de răspuns este mai sensibil la scăderea DBRT decât la scăderea CPUT. Se poate afirma că RT depinde semnificativ doar de DBRT.

În figura 5.6 este prezentată suprafața $RT=f(DBRT,CPUT)$ pentru tehnologia **CRM WebClient UI**.

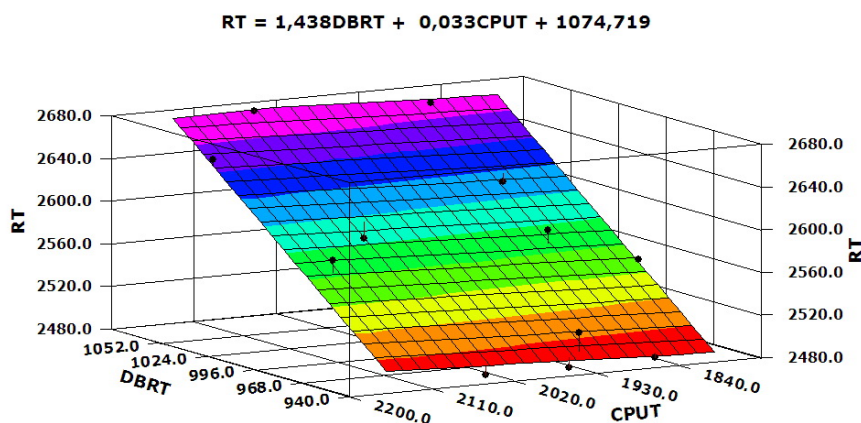


Fig. 5.6. Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia CRM WebClient UI

5.2. Analiza datelor generate

5.2.1. Coeficienții de corelație dintre parametrii generați

În tabelele 5.4, 5.5 și 5.6 sunt prezentate matricele coeficienților de corelație obținuți, cu ajutorul programului DataFit, prin prelucrarea statistică a datelor generate (după eliminarea erorilor grosiere), prezentate în anexele 7 ÷ 9.

Tabelul 5.4. Matricea coeficienților de corelație pentru setul de date generate (96 valori), utilizând tehnologia Web Dynpro ABAP

	CPUT	DBRT	HTTPWT	RT
CPUT	1	0.82190511	0.962500472	0.868675111
DBRT	0.82190511	1	0.844679706	0.994397165
HTTPWT	0.962500472	0.844679706	1	0.882343163
RT	0.868675111	0.994397165	0.882343163	1

Tabelul 5.5. Matricea coeficienților de corelație pentru setul de date generate (99 valori), utilizând tehnologia Floorplan Manager

	CPUT	DBRT	HTTPWT	RT
CPUT	1	0.22064556	0.898249587	0.994734322
DBRT	0.22064556	1	0.302998853	0.204172421
HTTPWT	0.898249587	0.302998853	1	0.850444903
RT	0.994734322	0.204172421	0.850444903	1

Tabelul 5.6. Matricea coeficienților de corelație pentru setul de date generate (96 valori), utilizând tehnologia CRM WebClient User Interface

	CPUT	DBRT	HTTPWT	RT
CPUT	1	0.999953451	-0.1646406	0.991851584
DBRT	0.999953451	1	-0.1648184	0.991889586
HTTPWT	-0.164640663	-0.164818441	1	-0.151906252
RT	0.991851584	0.991889586	-0.1519062	1

Observații

Analiza coeficienților de corelație din tabelele 5.4, 5.5 și 5.6 arată că:

1. În cazul utilizării tehnologiei Web Dynpro ABAP cu 96 valori pentru fiecare variabilă, există o dependență doar între variabilele RT și DBRT.
2. În cazul utilizării tehnologiei Floorplan Manager cu 99 valori pentru fiecare variabilă, există o dependență doar între variabilele RT și CPUT. Există și o corelație mai slabă, cu coeficientul de corelație $R=0,85$, între RT și HTTPWT.
3. În cazul utilizării tehnologiei CRM WebClient UI cu 96 valori pentru fiecare variabilă, există dependențe cu coeficienți de corelație semnificativi între variabilele RT și CPUT respectiv DBRT.

De aceste observații trebuie să se țină seama la stabilirea corelațiilor analitice și grafice dintre parametrii generați.

5.2.2. Stabilirea corelațiilor analitice și grafice dintre parametri generați

a) În cazul utilizării tehnologiei Web Dynpro ABAP cu 96 valori pentru fiecare variabilă, există o dependență doar între variabilele RT și DBRT. Funcția de regresie liniară simplă dintre variabila dependentă RT și variabila independentă DBRT, determinată cu ajutorul programului DataFit, este descrisă de relația:

$$RT = 0,847DBRT + 171,038 \quad (5.13)$$

Coeficientul de corelație simplă este $r=0,994$, iar coeficientul de determinare este $r^2=0,989$.

Funcția de regresie liniară simplă indică faptul că o scădere a timpului consumat pentru accesarea bazei de date cu 100 ms determină o scădere a timpului de răspuns cu 84,7 ms.

În figura 5.7 este prezentată dependența liniară a timpului de răspuns RT în funcție de timpul necesar citirii sau schimbării datelor în baza de date, DBRT.

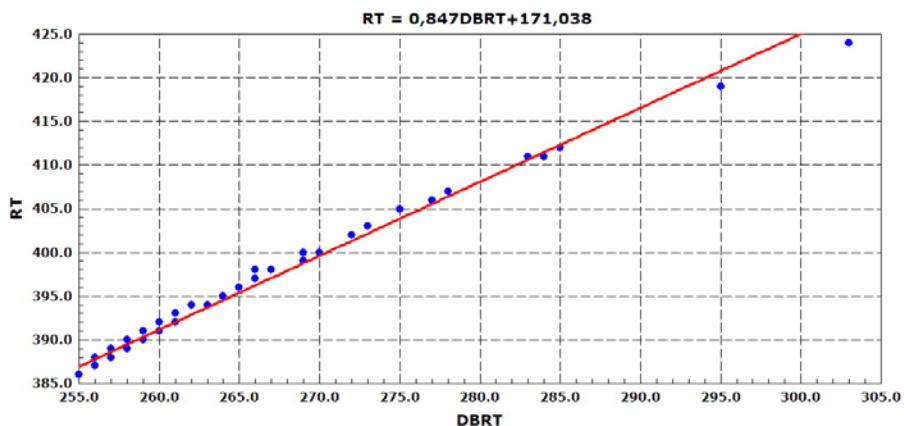


Fig. 5.7. Funcția de regresie liniară $RT=f(DBRT)$ pentru tehnologia Web Dynpro ABAP (96 valori)

Dacă se analizează dependența RT de DBRT și CPUT se obține, utilizând programul DataFit, funcția de regresie multiplă:

$$RT = 0,736DBRT + 0,178CPUT + 168,281 \quad (5.14)$$

Coeficientul de determinare multiplă este $R^2=0,997$.

Funcția de regresie multiplă indică faptul că o scădere a timpului consumat de procesor pentru efectuarea tranzacției (CPUT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 17,8 ms, în timp ce o scădere a timpului consumat pentru accesarea bazei de date (DBRT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 73,6 ms. Timpul de răspuns este mai sensibil la scăderea DBRT decât la scăderea CPUT. Se poate afirma că RT depinde semnificativ doar de DBRT.

În figura 5.8 este prezentată suprafața $RT=f(DBRT,CPUT)$ pentru tehnologia **Web Dynpro ABAP**.

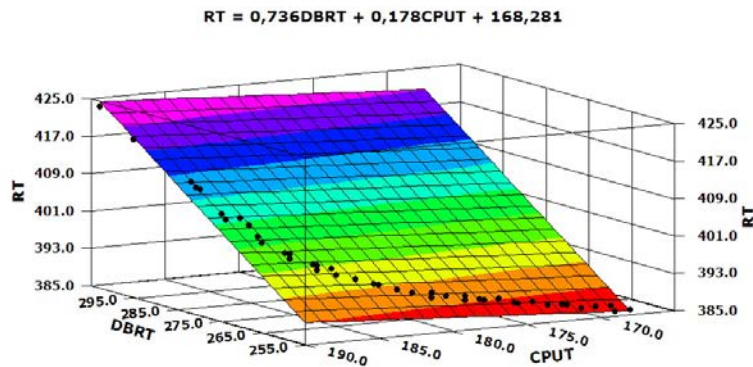


Fig. 5.8. Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia Web Dynpro ABAP (96 valori)

b) În cazul utilizării tehnologiei Floorplan Manager cu 99 valori pentru fiecare variabilă, există o dependență doar între variabilele RT și CPUT.

Funcția de regresie liniară simplă dintre variabila dependentă RT și variabila independentă CPUT, determinată cu ajutorul programului DataFit, este descrisă de relația:

$$RT = 0,937CPUT + 257,563 \quad (5.15)$$

Coeficientul de corelație simplă este $r=0,995$, iar coeficientul de determinare este $r^2=0,989$.

Funcția de regresie liniară simplă indică faptul că o scădere a timpului consumat de procesor pentru efectuarea tranzacție cu 100 ms determină o scădere a timpului de răspuns cu 93,7 ms.

În figura 5.9 este prezentată dependența liniară a timpului de răspuns RT în funcție de timpul consumat de procesor la sfârșitul unui pas de tranzacție CPUT.

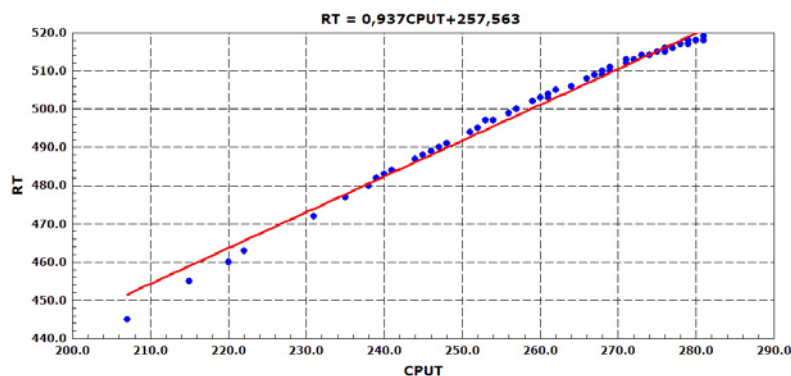


Fig. 5.9. Funcția de regresie $RT=f(CPUT)$ pentru tehnologia Floorplan Manager (99 valori)

Dacă se analizează dependența RT de CPUT și DBRT se obține, utilizând programul DataFit, *funcția de regresie multiplă*:

$$RT = 0,940CPUT - 0,007DBRT + 258,768 \quad (5.16)$$

Coeficientul de determinare multiplă este $R^2=0,990$.

Funcția de regresie multiplă indică faptul că o scădere a timpului consumat de procesor pentru efectuarea tranzacției (CPUT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 94 ms, în timp ce o scădere a timpului consumat pentru accesarea bazei de date (DBRT) cu 100 ms determină o creștere a timpului de răspuns (RT) cu 0,7 ms. Timpul de răspuns este mai sensibil la scăderea CPUT decât la creșterea DBRT. Se poate afirma că RT depinde semnificativ doar de CPUT.

În figura 5.10 este prezentată suprafața $RT=f(CPUT, DBRT)$ pentru tehnologia *Floorplan Manager*.

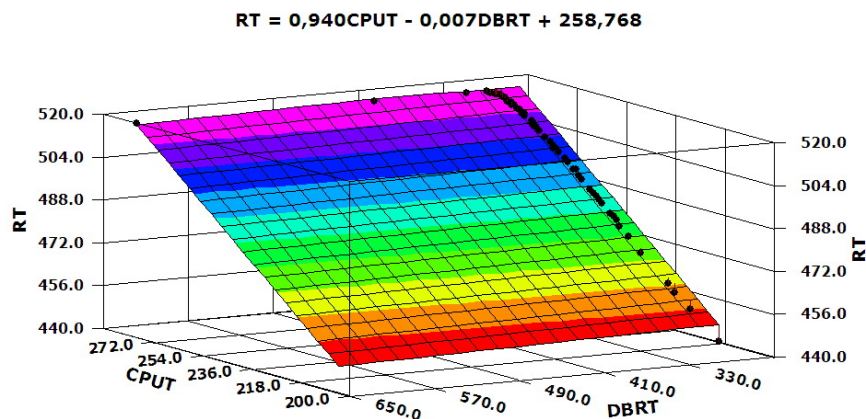


Fig. 5.10. Funcția de regresie $RT=f(CPUT, DBRT)$ pentru tehnologia Floorplan Manager (99 valori)

Dacă se analizează dependența RT de CPUT și HTTPWT se obține, utilizând programul DataFit, *funcția de regresie multiplă*:

$$RT = 1,125CPUT - 0,046HTTPWT + 283,786 \quad (5.17)$$

Coeficientul de determinare multiplă este $R^2=0,999$.

Funcția de regresie multiplă indică faptul că o scădere a timpului consumat de procesor pentru efectuarea tranzacției (CPUT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 112,5 ms, în timp ce o scădere a timpului consumat pentru accesarea bazei de date (DBRT) cu 100 ms determină o creștere a timpului HTTPWT cu 4,6 ms. Timpul de răspuns este mai sensibil la scăderea DBRT decât la

creșterea HTTPWT. Și în acest caz se poate afirma că RT depinde semnificativ doar de CPUT.

În figura 5.11 este prezentată suprafața $RT=f(CPUT,HTTPWT)$ pentru tehnologia *Floorplan Manager*.

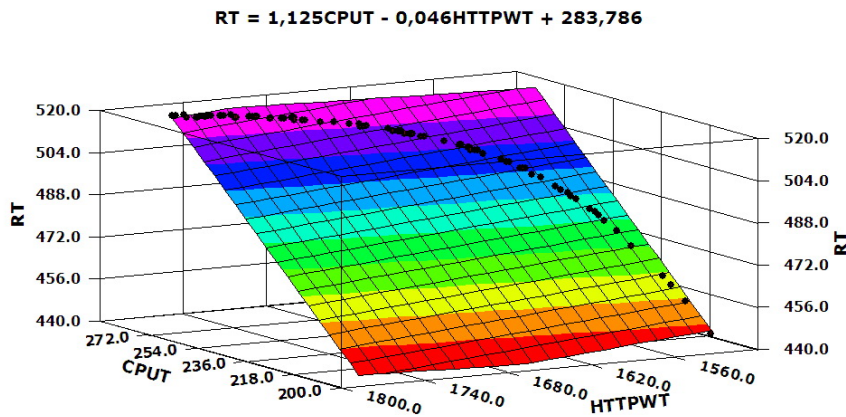


Fig. 5.11. Funcția de regresie $RT=f(CPUT,HTTPWT)$ pentru tehnologia Floorplan Manager (99 valori)

c) **În cazul utilizării tehnologiei CRM WebClient UI cu 96 valori** pentru fiecare variabilă, există dependențe cu coeficienți de corelație semnificativi între variabilele RT și CPUT respectiv DBRT.

Funcția de regresie liniară simplă dintre variabila dependentă RT și variabila independentă DBRT, determinată cu ajutorul programului DataFit, este descrisă de relația:

$$RT = 1,424DBRT + 1153,063 \quad (5.18)$$

Coeficientul de corelație simplă este $r=0,992$, iar coeficientul de determinare este $r^2=0,984$.

Funcția de regresie liniară simplă indică faptul că o scădere a timpului consumat pentru accesarea bazei de date cu 100 ms determină o scădere a timpului de răspuns cu 142,4 ms.

În figura 5.12 este prezentată dependența liniară a timpului de răspuns RT în funcție de timpul necesar citirii sau schimbării datelor în baza de date DBRT.

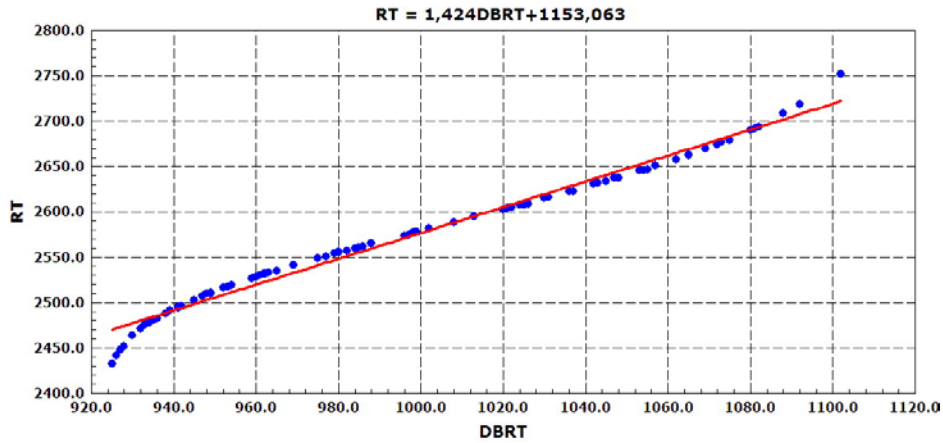


Fig. 5.12 Funcția de regresie $RT=f(DBRT)$ pentru tehnologia CRM WebClient UI (96 valori)

Dacă se analizează dependența RT de DBRT și CPUT se obține, utilizând programul DataFit, *funcția de regresie multiplă*:

$$RT = 1,298DBRT + 0,051CPUT + 1177,723 \quad (5.19)$$

Coeficientul de determinare multiplă este $R^2=0,984$.

Funcția de regresie multiplă arată că o scădere a timpului consumat de procesor pentru efectuarea tranzacției (CPUT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 5,1 ms, în timp ce o scădere a timpului consumat pentru accesarea bazei de date (DBRT) cu 100 ms determină o scădere a timpului de răspuns (RT) cu 129,8 ms. Timpul de răspuns este mai sensibil la scăderea DBRT decât la scăderea CPUT. Se poate afirma că RT depinde semnificativ doar de DBRT.

În figura 5.13 este prezentată suprafața $RT=f(DBRT,CPUT)$ pentru tehnologia CRM WebClient UI.

$$RT = 1,298DBRT + 0,051CPUT + 1177,723$$

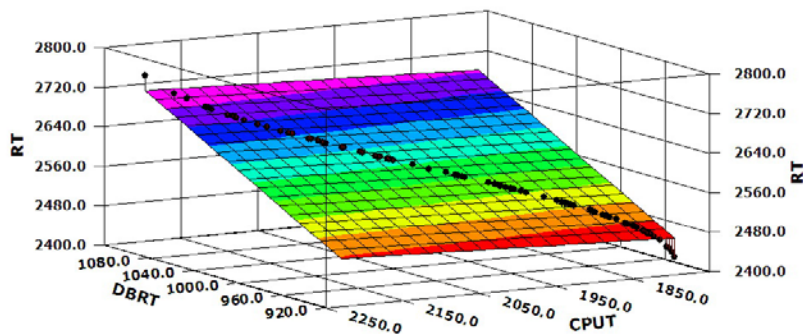


Fig. 5.13. Funcția de regresie $RT=f(DBRT,CPUT)$ pentru tehnologia CRM WebClient UI (96 valori)

5.3. Analiza comparativă a corelațiilor dintre parametri

a) Corelația simplă

Se realizează o comparare atât a corelațiilor simple, corespunzătoare diferitelor tehnologii, cât și o comparare a corelațiilor simple pentru aceeași tehnologie, când se folosesc valori măsurate sau valori generate.

În tabelul 5.7 sunt prezentate relațiile de dependență ale timpului de răspuns RT funcție de parametri măsurați, DBRT și CPUT, pentru care există coeficienți de corelație semnificativi.

Tab. 5.7 Funcțiile de regresie liniară și coeficienții de corelație corespunzători, pentru cele trei tehnologii utilizate la interogarea bazei de date SAP

Tehnologia utilizată	Funcția de regresie liniară	Coeficientul de corelație r
Web Dynpro ABAP 11 valori măsurate	$RT = 1,049DBRT + 118,871$	0,995
Web Dynpro ABAP 96 valori generate	$RT = 0,847DBRT + 171,038$	0,994
Floorplan Manager 12 valori măsurate	$RT = 0,883CPUT + 271,430$	0,932
Floorplan Manager 99 valori generate	$RT = 0,937CPUT + 257,563$	0,995
CRM WebClient UI 12 valori măsurate	$RT = 1,452DBRT + 1124,934$	0,987
CRM WebClient UI 96 valori generate	$RT = 1,424DBRT + 1153,063$	0,992

Timpul de răspuns RT depinde doar de timpul de răspuns la interogarea bazei de date DBRT în cazul tehnologiilor Web Dynpro ABAP și CRM WebClient UI iar în cazul tehnologiei Floorplan Manager RT depinde doar de CPUT.

b) Corelația multiplă

Se realizează o comparare a corelațiilor multiple între parametri, corespunzători diferitelor tehnologii (separat pentru valori măsurate și pentru valori generate).

În tabelul 5.8 sunt prezentate relațiile de dependență ale timpului de răspuns de ceilalți parametri măsurați.

Tab. 5.8. Funcțiile de regresie multiplă și coeficienții de determinare corespunzători parametrilor mășurați, pentru cele trei tehnologii utilizate la interogarea bazei de date SAP

Tehnologia	Funcția de regresie			R ²
WD ABAP 11 valori	RT=	+1,049DBRT	+118,871	0,991
	RT=	0,006CPUT +1,048DBRT	+117,865	0,991
	RT=	-0,038CPUT +1,053DBRT -0,010HTTPWT	+136,317	0,994
FPM 12 valori	RT=	0,883CPUT	+271,430	0,869
	RT=	0,852CPUT +1,404DBRT	-143,335	0,907
	RT=	0,771CPUT +1,507DBRT -0,029HTTPWT	+103,928	0,917
CRM WCU1 12 valori	RT=	+1,452DBRT	+1124,934	0,974
	RT=	0,033CPUT +1,438DBRT	+1074,719	0,977
	RT=	0,014CPUT +1,461DBRT -0,004HTTPWT	+1099,993	0,979

În figura 5.14 este prezentată dependența parametrului dependent RT măsurat de ceilalți parametri mășurați. Grosimea săgeților simbolizează o măsură a sensibilității parametrului dependent RT de ceilalți parametri, iar valorile coeficienților de corelație simplă dintre parametri, o măsură a corectitudinii corelației.

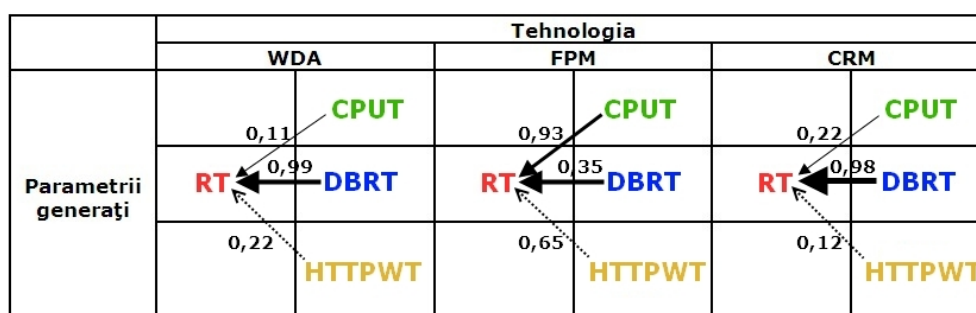


Fig. 5.14 Dependența parametrului dependent RT de ceilalți parametri (pentru datele măsurate)

În tabelul 5.9 sunt prezentate relațiile de dependență ale timpului de răspuns RT de ceilalți parametri generați.

Tab. 5.9. Funcțiile de regresie multiplă și coeficienții de determinare corespunzători parametrilor generați, pentru cele trei tehnologii utilizate

Tehnologia	Funcția de regresie			R ²
WD ABAP 96 valori	RT=	+0,847DBRT	+171,038	0,989
	RT=	0,178CPUT +0,736DBRT	+168,281	0,997
	RT=	0,179CPUT +0,736DBRT -0,001HTTPWT	+168,225	0,997
FPM 99 valori	RT=	0,937CPUT	+257,563	0,990
	RT=	0,940CPUT -0,007DBRT	+258,768	0,990
	RT=	1,128CPUT +0,004DBRT -0,047HTTPWT	+283,686	0,999
CRM WCU1 96 valori	RT=	+1,424DBRT	+1153,063	0,984
	RT=	0,051CPUT +1,298DBRT	+1177,723	0,984
	RT=	0,038CPUT +1,332DBRT -0,000HTTPWT	+1168,495	0,984

În figura 5.15 este prezentată dependența parametrului dependent RT generat de ceilalți parametri generați. Grosimea săgeților simbolizează o măsură a sensibilității parametrului dependent RT la variația celorlalți parametri, iar valorile coeficienților de corelație simplă dintre parametri, o măsură a corectitudinii corelației.

	Tehnologia					
	WDA		FPM		CRM	
Parametrii generați		CPUT		CPUT		CPUT
	RT	0,87	RT	0,99	RT	0,99
	DBRT	0,99	DBRT	0,20	DBRT	0,99
		HTTPWT		HTTPWT		HTTPWT
		0,88		0,85		0,15

Fig. 5.15 Dependența parametrului dependent RT de ceilalți parametri (pentru datele generate)

5.4. Concluzii

Din analiza corelațională și de regresie a datelor măsurate și a celor generate, prezentată în acest capitol, rezultă o serie de concluzii:

- Corelația simplă între RT și DBRT sau CPUT (pentru care s-au obținut coeficienți de corelație liniară simplă semnificativi) arată că:
 - Numai pentru tehnologiile WD ABAP și CRM WebClient UI există o dependență $RT = f(DBRT)$.
 - Pentru tehnologia FPM nu există o astfel de corelație. În schimb pentru această tehnologie $RT = f(CPUT)$.
 - Concluziile anterioare sunt valabile atât pentru valorile măsurate ale parametrilor cât și pentru valorile generate ale acestora.
- Corelația multiplă între timpul de răspuns RT și ceilalți parametri măsurați arată că:
 - Pentru toate tehnologiile folosite RT depinde semnificativ de DBRT.
 - Dependența RT de HTTPWT este neglijabilă.
 - De asemenea numai pentru tehnologia FPM, RT depinde de CPUT dar într-o măsură mai mică decât dependența lui RT de DBRT.
- O corelație multiplă între timpul de răspuns RT și ceilalți parametri generați demonstrează că:
 - Numai pentru tehnologiile WD ABAP și CRM WebClient UI folosite, RT depinde semnificativ de DBRT.
 - Dependența RT de HTTPWT este neglijabilă.
 - Numai pentru tehnologia FPM, RT depinde de CPUT.

-
4. Pe baza analizării matricelor coeficienților de corelație rezultă că HTTPWT este singurul parametru independent (care nu depinde de ceilalți 3 parametri: RT, DBRT și CPUT).
 5. Rezultă pe baza analizei de regresie prezentată în acest capitol și pe baza analizei din cap.3.3 și prezentată în figura 3.18 că RT este un parametru dependent. El depinde de DBRT sau de CPUT astfel:
 - pentru tehnologia WD ABAP, RT depinde semnificativ numai de DBRT;
 - pentru tehnologia FPM, RT depinde semnificativ numai de CPUT;
 - pentru tehnologia CRM WCUI, RT depinde semnificativ numai de DBRT;
 6. Pe baza analizei corelaționale și de regresie a datelor măsurate sau generate s-a stabilit că nu există o dependență semnificativă între parametrii CPUT și DBRT (sau între DBRT și CPUT), cu o singură excepție. Doar în cazul datelor generate, pentru tehnologia CRM WCUI există o dependență semnificativă între CPUT și DBRT. Cei doi parametri, CPUT și DBRT pot fi și ei considerați ca parametri independenți.

6. Metodologie de analiză și evaluare multicriterială a tehnologiilor ABAP UI

În baza analizei efectuate în capitolele precedente, obiectivul acestui capitol este de a evalua tehnologiile WD ABAP, FPM și CRM WebClient UI, utilizând metode de analiză multicriterială, în vederea ierarhizării tehnologiilor. Scopul evaluării este de a identifica caracteristicile comune specifice tehnologiilor analizate și a stabili acele ponderi de importanță pentru care ierarhizarea lor să fie cât mai aproape de adevăr. Sunt formulate soluțiile alternative, sunt definite seturile de criterii și sunt implementați algoritmi metodologici multicriteriale.

6.1. Metode de analiză multicriterială

Evaluarea este un proces care servește atât pentru prioritizarea unor acțiuni sau procese, cât și pentru clarificarea proceselor de decizie. Utilizat în domenii variate, conceptul de evaluare presupune existența unui set de tehnici și metode de evaluare și ierarhizare. Aceste metode prezintă unele elemente metodologice comune și se referă la :

- abordarea multicriterială;
- studierea comparativă (cu un model real sau cu unul virtual/teoretic);
- utilizarea unor scale și grile de evaluare;
- folosirea instrumentelor matematice și/sau statistice.

Evaluarea și ierarhizarea complexă are la bază o analiză multicriterială. În cadrul acestei analize, o mulțime de obiecte este clasificată în funcție de un număr de factori cărora li se acordă o anumită importanță.

Ca și concept, *analiza multicriterială* stabilește o ierarhizare a alternativelor unei probleme ce are la bază o mulțime de obiective pentru care s-au stabilit criterii măsurabile de evaluare. Caracteristica esențială a analizei multicriteriale se bazează pe puterea de judecată a decidentului pentru stabilirea obiectivelor și criteriilor pentru estimarea ponderilor relative și, parțial, pentru evaluarea contribuției fiecărei alternative la realizarea fiecărui criteriu [114][115][116].

Există câteva concepte cheie [117], [118] utilizate în procesele de analiză multicriterială, și anume:

- a) *alternativele* - reprezintă acele variante prin care obiectivele unei probleme sunt îndeplinite;
- b) *criteriile* - reprezintă acele măsuri în funcție de care sunt evaluate și comparate alternativele;
- c) *matricea de performanță* – reprezintă instrumentul standard al procesului de analiză multicriterială, în care fiecare rând descrie o alternativă și fiecare coloană reprezintă un criteriu decizional. Per ansamblu, matricea decizională descrie consecințele sau performanțele alternativelor, referitor la satisfacerea criteriilor decizionale;
- d) *punctajele și ponderile* – sunt stabilite în diverse etape ale prelucrării matricei de performanță. Punctajele se acordă consecințelor preconizate, în funcție de o anumită scală. Ponderile reprezintă măsuri cantitative asociate

criteriilor, care stabilesc importanța acestora. Ponderile sunt stabilite subiectiv de către decident, sau prin metode speciale;

- e) *metodele de normalizare* a matricei de performanță – presupun ca valorile criteriilor exprimate prin diferite unitați de măsură, să fie transformate într-o scală comună.

În literatura de specialitate [117], [119], principalele etape ale analizei multicriteriale sunt:

Etapa I Stabilirea contextului decizional. Are loc identificarea obiectivului și sub-obiectivelor aferente.

Etapa II Definirea alternativelor sau opțiunilor. Constă în identificarea și descrierea detaliată a ceea ce reprezintă fiecare dintre acestea.

Etapa III Definirea criteriilor. Presupune identificarea și definirea criteriilor relevante pentru problema decizională. Aceste criterii pot fi de natură cantitativă, dar și de natură calitativă.

Etapa IV Crearea matricei de performanță. Se stabilesc performanțele fiecărei alternative în funcție de criterii.

Etapa V Standardizarea punctajelor. Fiecărui criteriu i se acordă un punctaj la nivelul unei scale comune (de exemplu valori cuprinse între 0÷1 sau 0÷100).

Etapa VI Ponderarea criteriilor. Presupune cuantificarea relativă a fiecărui criteriu în cadrul procesului decizional. Ponderile pot fi stabilite din datele existente sau pot fi determinate prin aplicarea metodelor specifice (de exemplu metodele AHP, FAHP).

Etapa VII Efectuarea analizei de sensibilitate. Presupune modificarea consecințelor și a ponderilor.

Diversitatea de metode prezentate în literatura de specialitate [116], [120], utilizează combinații diverse de factori în funcție de particularitățile obiectelor evaluate. Astfel, problemele de analiză multicriterială pot fi abordate prin:

- Metode bazate pe determinarea scorului favorabil (de exemplu metoda utilității globale (MUG), procesul ierarhic de analiză AHP, metoda SMART – Simple Multi-Attribute Rating Technique) [121], [122], [123].
- Metode de compromis, caracterizate de alegerea drept optim multicriterial a acelei variante decizionale care se apropie cel mai mult de soluția ideală (de exemplu metoda TOPSIS) [124], [125].
- Metode de surclasare, bazate pe conceptele de concordanță și discordanță, care rezidă în ordonarea alternativelor în funcție de concordanța alternativelor cu criteriile (de exemplu metodele ELECTRE, PROMETHEE) [126], [127].

În prezent, nu există cercetări referitoare la evaluarea tehnologiilor WD ABAP, FPM și CRM WebClient UI utilizate în aplicațiile web business implementate pentru căutări și afișări ale datelor. În acest sens, prezenta teză propune o metodologie de analiză multicriterială, care sprijină procesul decizional în ceea ce privește evaluarea și ierarhizarea acestor tehnologii.

Modele de procese decizionale există încă din deceniul al șaptelea al secolului trecut. Literatura de specialitate [128] evidențiază câteva caracteristici comune ale acestor modele prezentate în tabelul 6.1, și anume: descriu mediul în mod selectiv, stabilesc variabilele deciziei, stabilesc obiectivele.

Tabelul 6.1. Modele ale procesului decizional [129]

Nr.	Autor(ii) – Anul	Etapele procesului
1	Pounds, 1969	1. Alegerea modelelor; 2. Confruntarea cu realitatea; 3. Identificarea diferențelor; 4. Selectarea diferențelor.
2	Cleland – William, 1972	1. Identificarea problemei; 2. Analiza informațiilor; 3. Formularea variantelor; 4. Instrucțiuni de aplicare; 5. Aplicarea deciziei; 6. Compararea rezultatelor cu așteptările.
3	Simon, 1977	1. Informarea – căutarea, identificarea, formularea problemei; 2. Proiectarea – stabilirea obiectivelor și elaborarea variantelor; 3. Alegerea variantei dorite; 4. Evaluarea factorilor de influență.
4	Kotter, 1983	1. Fixarea agendei – obiective urmărite; 2. Construirea rețelei de colaboratori pentru implementarea agendei.
5	Pokras, 1989	1. Recunoașterea problemei 2. Caracterizarea } - definirea problemei (1, 2, 3) 3. Analiza 4. Variante posibile 5. Evaluarea } - luarea deciziei (4, 5, 6) 6. Plan de acțiune pentru aplicarea deciziei
6	Nutt, 1990	1. Explorarea posibilităților; 2. Stabilirea opțiunilor; 3. Testarea ipotezelor formulate anterior; 4. Lansarea deciziei.
7	Paterson, 1996	1. Identificarea problemei; 2. Generarea soluțiilor alternative; 3. Alegerea soluției optime; 4. Autorizarea; 5. Implementarea.
8	Cougar, 1996	1. Delimitarea oportunităților și definirea problemei, 2. Culegerea informațiilor relevante; 3. Generarea ideilor; 4. Evaluarea – ierarhizarea variantelor; 5. Elaborarea planului de implementare.

9	Bazerman1 1998	1. Definirea problemei; 2. Identificarea criteriilor; 3. Ierarhizarea criteriilor; 4. Generarea alternativelor; 5. Aprecierea după criterii a fiecărei alternative; 6. Calculul deciziei optime.
10	Hammond, 1998	1. Definirea problemei; 2. Obiectivele urmărite; 3. Alternative; 4. Consecințele fiecărei alternative; 5. Compromisuri necesare; 6. Incertitudini; 7. Toleranța la riscuri; 8. Corelarea cu alte decizii.
11	Stăncioiu – Militaru, 1998	1. Pregătirea informațională (problema și informațiile utile); 2. Adoptarea deciziei optime (variante - varianta optimă); 3. Implementarea deciziei (instrucțiuni și aplicare).
12	Nicolescu– Verboncu, 1999	1. Identificarea și definirea problemei; 2. Precizarea obiectivelor; 3. Stabilirea variantelor decizionale; 4. Alegerea variantei convenabile și realiste (decizia); 5. Aplicarea deciziei; 6. Evaluarea rezultatelor.
13	Burduș – Căprărescu, 1999	1. Identificarea și definirea problemei; 2. Stabilirea variantelor posibile; 3. Stabilirea criteriilor și obiectivelor decizionale; 4. Caracterizarea variantelor; 5. Alegerea variantei optime; 6. Aplicarea variantei alese; 7. Evaluarea rezultatelor.
14	Certo, 2002	1. Identificarea problemei; 2. Enunțarea alternativelor; 3. Alegerea celei mai avantajoase alternative; 4. Implementarea variantei alese; 5. Culegerea reacțiilor pentru aprecierea soluționării problemei.
15	David (DECIDE), 2003	1. Definirea problemei; 2. Evidențierea variantelor; 3. Culegerea informațiilor și estimarea consecințelor; 4. Identificarea variantei convenabile; 5. Dezvoltarea și implementarea variantei convenabile; 6. Evaluarea rezultatelor.

C. Zamfir [130] oferă o perspectivă parțial diferită asupra procesului decizional, identificând următoarele faze:

1. definirea problemei;
2. formularea soluțiilor alternative;
3. evaluarea și ierarhizarea soluțiilor alternative, cu următoarele 4 subetape:
 - a) determinarea criteriilor de evaluare;
 - b) ierarhizarea criteriilor de evaluare;
 - c) evaluarea soluțiilor alternative pe baza criteriilor anterior considerate;
 - d) ierarhizarea soluțiilor alternative;

4. alegerea soluției (decizia propriu-zisă);
5. execuția deciziei.

Comparând modelele decizionale existente prin perspectiva oferită de Zamfir, metodologia propusă se referă la faza predecizională și poate fi structurată astfel:

1. definirea problemei;
2. formularea alternativelor;
3. identificarea criteriilor de evaluare;
4. ierarhizarea alternativelor.

Etapă de ierarhizare a alternativelor implică implementarea unor algoritmi care au la bază metodele multicriteriale MUG, AHP, TOPSIS și ELECTRE prezentate sintetic în continuare.

6.1.1. Metoda utilității globale

Conceptul de *utilitate*, propus de matematicianul John von Neumann și economistul Oskar Morgenstern, măsoară importanța pe care o are pentru decident o anumită variantă decizională care aparține unei mulțimi de variante [131], [132].

Utilitatea este o mărime adimensională, cuprinsă între zero și unu, în care se convertesc diferitele unități de măsură a criteriilor de apreciere, pentru a ușura compararea acestora, în vederea facilitării luării deciziei.

Această metodă poate fi aplicată atât în situația în care criteriile decizionale sunt la fel de importante pentru decident, dar și în situația în care ele au o importanță diferită.

Aplicarea acestei metode constă în parcurgerea a **două etape**, decizia optimă fiind cea care are utilitatea globală cea mai mare [131].

1. Prima etapă presupune înlocuirea tabelului decizional cu tabelul utilităților, convertind fiecare valoare $X_{i,j}$ în valoarea utilității corespunzătoare $U_{i,j}$. Conversia se face diferit pentru caracteristicile maximizante respectiv cele minimizante:

- pentru caracteristicile maximizante:

$$U_{i,j} = \frac{X_{i,j} - \min C_{i,j}}{\max C_{i,j} - \min C_{i,j}} \text{ pentru } j = \overline{1, n} \quad (6.1)$$

- pentru caracteristicile minimizante:

$$U_{i,j} = \frac{\max C_{i,j} - X_{i,j}}{\max C_{i,j} - \min C_{i,j}} \text{ pentru } j = \overline{1, n} \quad (6.2)$$

2. Cea de-a doua etapă presupune calcularea utilității globale pentru fiecare variantă decizională. Ea se poate determina în două variante:

- A. se face media aritmetică a utilităților fiecărei variante decizionale:

$$UG_i = \frac{\sum_{j=1}^n U_{ij}}{n} \quad \text{pentru } i = \overline{1, m} \quad (6.3)$$

- B. se face media ponderată a utilităților fiecărei variante decizionale, în care decidentul stabilește ponderi de importanță pentru fiecare criteriu în funcție de specificul deciziei respective:

$$\text{pentru } j = \overline{1, n} \text{ se atribuie } p_j \text{ cu condițiile } 0 \leq p_j \leq 1 \text{ și } \sum_{j=1}^n p_j = 1 \quad (6.4)$$

$$UG_i = \sum_{j=1}^n p_j U_{ij} \quad \text{pentru } i = \overline{1, m} \quad (6.5)$$

6.1.2. Metoda AHP

Una dintre abordările metodelor de analiză multicriterială, oferită ca și instrument de ierarhizare a alternativelor unei probleme, este AHP (Analytic Hierarchy Process). Dezvoltată de către Thomas L.Saaty [133], această metodă se bazează pe divizarea problemei în subprobleme mai simple de ierarhizare, ce pot fi analizate independent una de alta.

Principiul de bază al metodei [134], [135] se bazează pe divizarea ierarhică a problemei, în subprobleme ce pot fi analizate independent una de alta. Elementele ierarhiei sunt evaluate subiectiv prin compararea lor succesivă, două câte două, iar ordonările sunt transformate în valori numerice, care permit agregarea rezultatelor obținute în vederea obținerii soluției .

Utilizarea acestei metode presupune parcurgerea următoarelor etape [136]:

Etapa I Construirea structurii ierarhice a problemei de decizie, (figura 6.1) presupune:

- definirea scopului sau obiectivului fundamental al problemei;
- definirea criteriilor decizionale, eventual descompunerea acestora în subcriterii;
- definirea variantelor sau alternativelor decizionale care trebuie ordonate în funcție de preferințele decidentului.

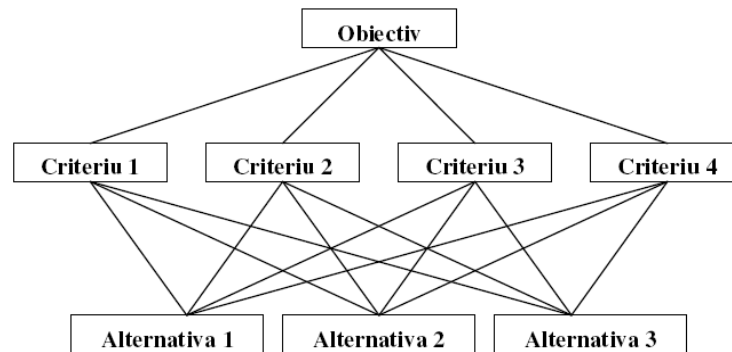


Fig. 6.1 Ierarhie AHP

Etapa II Determinarea ponderilor relative ale criteriilor, prin compararea acestora două câte două, în raport cu obiectivul aflat la nivelul imediat superior al ierarhiei. Comparațiile pe perechi se realizează prin evaluarea subiectivă a decidentului, pe baza unei scale de la 1 la 9, ce reprezintă gradul de importanță al unui criteriu față de altul. O astfel de scală este prezentată în tabelul 6.2.

Etapa III Determinarea ponderilor relative ale alternativelor pe baza criteriilor după aceeași procedură descrisă la punctul 2 și înregistrarea rezultatelor într-o matrice pătratică.

Tabelul 6.2. Scala preferințelor relative - Saaty [137]

Intensitatea preferinței	Definiție	Explicații
1	Egal preferate	Două elemente contribuie egal la realizarea obiectivului
3	Moderat preferat	Experiența și judecata favorizează ușor un element față de celălalt
5	Preferat	Experiența și judecata favorizează un element față de celălalt
7	Puternic preferat	Un element este puternic favorizat și dominanța sa este demonstrată practic
9	Extrem de preferat	Elementul este favorizat evident față de celălalt
2,4,6,8		Valori intermediare
1/2, 1/3, 1/4, 1/5, 1/6, 1/7, 1/8, 1/9		Dacă un element "i" are una din valorile de mai sus când e comparat cu elementul "j", atunci "j" are valoarea reciprocă în raport cu "i"

Etapa IV Normalizarea comparațiilor între criterii și a celor între alternative în raport cu fiecare criteriu și verificarea consistenței comparațiilor efectuate presupune patru sub-etape:

- a) Normalizarea matricelor de comparație se realizează prin împărțirea valorilor la suma elementelor de pe fiecare coloană.

- b) Obținerea vectorului normalizat al ponderilor se realizează prin împărțirea fiecărui element al vectorului nenormalizat la suma tuturor elementelor vectorului. Acest vector va permite stabilirea priorităților globale pentru fiecare alternativă decizională.
- c) Determinarea valorii proprii a vectorului priorităților. Elementele vectorului priorităților sunt determinate ca medii aritmetice ale valorilor de pe fiecare rând.
- d) Calcularea indicelui de consistență CR a matricelor se obține ca raport între inconsecvența factorului de decizie și un indice generat aleatoriu (dacă $CR < 0.10$, atunci matricea se consideră a fi consistentă, adică vectorul ponderilor este bine determinat, altfel, se recomandă reconstruirea matricei de comparație pentru a obține rezultate viabile).

Etapa V Determinarea scorului total de prioritate al fiecărei alternative se obține ca suma a produselor dintre ponderea relativă pentru un anumit criteriu și ponderea relativă a unei alternative în raport cu acel criteriu.

Etapa VI Elaborarea deciziei finale prin alegerea variantei cu scor maxim și analiza de sensibilitate a soluției obținute.

Prima și ultima etapă din procesul AHP sunt proceduri relativ simple, însă evaluarea priorităților locale pe baza perechilor de comparații presupune aplicarea unor metode de prioritizare [138], [139], [140].

Câteva dintre avantajele metodei AHP:

- Teoretic - AHP este una dintre metodele multicriteriale care oferă o tehnică axiomatică.
- Practic - tehnica AHP oferă un model practic ușor de înțeles.
- Complexitate - AHP integrează abordări deductive și tehnici pentru rezolvarea problemelor complexe.
- Structura ierarhică – AHP reflectă tendința naturală a minții de a sorta elementele unui sistem la diferite nivele și a grupa obiectele similare pe același nivel.
- Acțiune – AHP oferă o scară pentru evaluarea caracteristicilor și o metodă de clasificare a priorităților.
- Proces – AHP ia în considerare prioritățile relative ale factorilor într-un sistem, pentru a selecta cea mai bună alternativă în virtutea obiectivelor.
- Judecată – sintetizează un rezultat reprezentativ pentru diferite procese.

Metoda AHP prezintă și o serie de dezavantaje:

- scala utilizată pentru exprimarea judecăților relative asociate comparațiilor este limitată;
- coerența evaluării deciziei, deoarece depinde de stabilirea subiectivă a gradului de importanță pentru elementele ierarhiei ;
- alegerea unei noi alternative poate influența preferința decidentului sau evidenția neconcordanțe în luarea deciziei.

Cu toate acestea, standardul AHP de elaborare a unei decizii nu poate fi utilizat atunci când factorul de decizie se confruntă cu o problemă complexă și incertă, atunci când comparația se bazează pe informații imprecise. În acest sens există abordarea fuzzy, care permite valorificarea informației exprimate prin limbajul natural și asimilarea elementelor specifice judecății umane.

6.1.3. Metoda TOPSIS

Metoda TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) dezvoltată de C.L. Hwang și K. Yoon [141], se confruntă cu dilema de a lucra cu soluția ideală, anti-ideală sau amestecul celor două. Principiul de bază [142] este acela că decizia optimă trebuie să fie cât mai aproape de soluția cea mai avantajoasă și cât mai depărtată de soluția cea mai dezavantajoasă. Apropierea sau depărtarea este luată în considerare sub forma distanței geometrice dintre caracteristici sau criterii.

Aplicarea metodei presupune următoarele *etape* [131]:

Etapa I Înlocuirea tabelului decizional cu tabelul consecințelor, în care criteriile calitative sunt convertite în utilități.

Etapa II Calcularea matricei normalizate prin convertirea consecințelor C_{ij} în valorile normalizate CN_{ij} conform relației:

$$CN_{ij} = \frac{C_{ij}}{\sqrt{\sum_{i=1}^m C_{ij}}} \quad \text{sau} \quad (6.6)$$

$$CN_{ij} = \frac{C_{ij}}{\sqrt{\sum_{i=1}^m C_{ij}^2}} \quad (6.7)$$

Etapa III Calcularea matricei normalizate ponderate prin înmulțirea valorilor normalizate cu ponderile de importanță (p_j) acordate de decident fiecărui criteriu:

$$CNP_{ij} = p_j \cdot CN_{ij} \quad (6.8)$$

Etapa IV Determinarea pentru fiecare criteriu ($j=1\dots n$) a soluției celei mai avantajoase (ideal pozitivă) C_j^+ , respectiv celei mai dezavantajoase (ideal negativă) C_j^- . Pentru aceasta se ține cont de tipul criteriului respectiv (maximizant sau minimizant):

- pentru caracteristici maximizante:

$$C_j^+ = \max_{1 \leq i \leq m} \{CNP_{ij}\} \quad (6.9)$$

$$C_j^- = \min_{1 \leq i \leq m} \{CNP_{ij}\}$$

- pentru caracteristici minimizante:

$$\begin{aligned} C_j^+ &= \min_{1 \leq i \leq m} \{CNP_{ij}\} \\ C_j^- &= \max_{1 \leq i \leq m} \{CNP_{ij}\} \end{aligned} \quad (6.10)$$

Etapa V Determinarea depărtării fiecărei alternative față de soluția cea mai avantajoasă, respectiv cea mai dezavantajoasă:

$$S_i^+ = \sqrt{\sum_{j=1}^n (CNP_{ij} - C_j^+)^2} \quad (6.11)$$

$$S_i^- = \sqrt{\sum_{j=1}^n (CNP_{ij} - C_j^-)^2} \quad (6.12)$$

Etapa VI Ierarhizarea fiecărei variante decizionale față de soluția ideală. Varianta cea mai bună este cea care obține scorul S^* cel mai mare:

$$S_i^* = \frac{S_i^-}{S_i^- + S_i^+} \quad (6.13)$$

Această metodă are o procedură de aplicare simplă, iar numărul de pași rămâne același indiferent de numărul criteriilor alese. Un dezavantaj este distanța euclidiană, care nu ia în considerare corelația dintre criterii [143], [144].

Aceste etape pot fi sintetizate într-un algoritm a cărui structură este prezentată în figura 6.2.

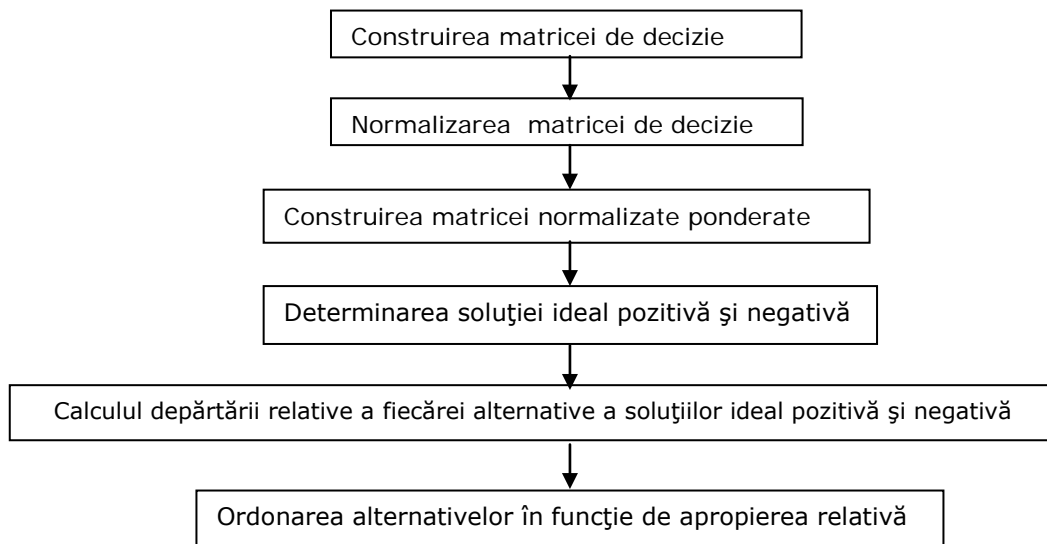


Fig. 6.2. Algoritmul metodei TOPSIS

6.1.4. Metoda ELECTRE

Această metodă a fost propusă de Bernard Roy [145] în 1968, iar acronimul ELECTRE provine de la inițialele denumirii metodei: **EL**imination **Et** **Ch**oix **T**raduisent la **RE**alité (eliminare și alegere reflectând realitatea). În timp, metoda a evoluat într-o serie de alte variante: ELECTRE I, ELECTRE II, ELECTRE III, ELECTRE IV, ELECTRE IS, ELECTRE TRI.

Metoda *ELECTRE* este o metodă de surclasare și alegere în prezența unor puncte de vedere multiple, care permite ordonarea variantelor după criterii complexe prin comparații succesive, două câte două. Logica metodei are la bază conceptul relației de surclasare [146], în care, între două alternative A și B care nu sunt în relație de dominanță, decidentul acceptă riscul de a considera alternativa A mai bună decât alternativa B.

Aplicarea metodei presupune parcurgerea următorilor pași [131]:

- 1) Stabilirea variantelor decizionale, a criteriilor aferente și a tipului aprecierii care poate fi calitativă sau cantitativă.
- 2) Construirea matricei de decizie și înlocuirea sa cu matricea utilităților, convertind fiecare valoare C_{ij} în valoarea utilității corespunzătoare U_{ij} .
- 3) Atribuirea de ponderi de importanță (k_j) fiecărui criteriu, respectând condițiile:

$$0 < k_j < 1 \quad \text{și} \quad \sum_{j=1}^n k_j = 1 \quad (6.14)$$

- 4) Calcularea indicilor de concordanță între perechi de variante decizionale, pe baza ponderilor de importanță acordate fiecărui criteriu.

$$p(V_x, V_y) = p_{xy} = \frac{\sum_{j=1}^n k_j^*}{\sum_{j=1}^n k_j} \quad \text{unde: } k_j^* \in \{k_j | u_x \geq u_y\} \quad (6.15)$$

Indicii de concordanță se determină în raport cu ponderile de importanță ai criteriilor.

- 5) Calcularea indicilor de discordanță între perechi de variante decizionale, pe baza utilităților acestora.

$$q(V_x, V_y) = q_{xy} = \frac{1}{E} \cdot \max \begin{cases} 0, & \text{dacă } u_y \leq u_x \\ |u_y - u_x|, & \text{dacă } u_y > u_x \end{cases} \quad (6.16)$$

unde E este diferența maximă dintre cea mai mare (1), respectiv cea mai mică (0) utilitate $E=1$. Indicii de discordanță se determină în raport cu utilitățile variantelor care se compară.

- 6) Alcătuirea tabelului indicilor de concordanță și de discordanță.

7) Alcătuirea tabelului diferențelor utilizând relația:

$$d_{xy} = p_{xy} - q_{xy} \quad (6.17)$$

8) Alcătuirea tabelului surclasărilor prin compararea reciprocă d_{xy} cu d_{yx} . Valoarea cea mai mare dintre d_{xy} și d_{yx} este înlocuită cu 1 iar cea mai mică cu 0. Dacă cele două valori sunt egale atunci ambele se înlocuiesc cu 1.

$$\begin{array}{llll} \text{dacă } d_{xy} > d_{yx} & \text{atunci} & s_{xy}=1 & \text{și } s_{yx}=0 \\ \text{dacă } d_{xy} = d_{yx} & \text{atunci} & s_{xy}=1 & \text{și } s_{yx}=1 \\ \text{dacă } d_{xy} < d_{yx} & \text{atunci} & s_{xy}=0 & \text{și } s_{yx}=1 \end{array}$$

9) Ierarhizarea variantelor decizionale prin însumarea, pe linii, a surclasărilor:

$$S_x = \sum_{y=1}^m S_{xy} \text{ pentru } x = \overline{1, m} \quad (6.18)$$

Pentru cea mai bună variantă decizională, valoarea lui S_x trebuie să fie cât mai mare.

6.2. Metodologie de evaluare și ierarhizare a tehnologiilor ABAP UI

În acest paragraf sunt prezentate etapele de realizare a metodologiei propuse pentru evaluarea multicriterială a tehnologiilor WD ABAP, FPM și CRM WebClient UI. Etapa de ierarhizare a tehnologiilor constă în implementarea algoritmilor AITOP și AIEL dezvoltăți de autor pe baza metodelor TOPSIS, ELECTRE și MUG.

Evaluarea tehnologiilor ține cont de seturile de caracteristici definite în etapa de identificare a criteriilor și ponderile acordate acestora, determinate prin modelul ierarhic MEIAHP dezvoltat de autor. În funcție de obiectivele propuse, aceste criterii pot fi oricând actualizate de către cei care au în vedere evaluarea acestor tehnologii.

6.2.1. Etapele metodologiei procesului de evaluare

1. Definirea problemei

Se urmărește evaluarea și ierarhizarea tehnologiilor ABAP UI dezvoltate pe platforma SAP NetWeaver, considerând implementarea aceleiași aplicații web la nivelul server-ului SAP AS în tehnologiile Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI. Aplicațiile web presupun interogarea bazei de date SAP CRM și afișarea rezultatelor pentru trei situații diferite. Datele obținute se prelucrează statistic, iar rezultatele se utilizează la implementarea algoritmilor de ierarhizare, care au la bază metode diferite de analiză multicriterială. Pentru evaluare se iau în considerare seturile de criterii definite, a căror ponderi se determină prin metoda AHP pentru un model ierarhic dezvoltat de autor, utilizând instrumentul SuperDecision.

2. Formularea alternativelor

În contextul enunțat prin definirea problemei, alternativele supuse procesului de evaluare sunt tehnologiile: **Web Dynpro ABAP**, **Floorplan Manager** și **CRM WebClient UI**.

3. Identificarea criteriilor de evaluare

Pentru a afla cum poate un model decizional să susțină cât mai optim alegerea unei tehnologii și care sunt informațiile pe care acesta trebuie să le conțină, este necesar să se cunoască cum poate fi aleasă o tehnologie UI și care sunt factorii care o influențează.

Punctul de plecare este de obicei cererea unui client pentru o aplicație, care trebuie să fie integrată în peisajul sistemului, mai ales atunci când aceasta urmează să fie extinsă. În plus, față de cerințele funcționale specifice scenariului, se află dorința clientului pentru a utiliza o interfață cât mai modernă, eficientă și care să prezinte o anumită siguranță în viitor [147]. Deci, pe de o parte, aplicația trebuie să fie cât mai atrăgătoare, să permită îndeplinirea sarcinilor cât mai eficient și într-un timp cât mai optim, iar pe de altă parte, să fie structurată intern astfel încât să poată fi adaptată pe durata ciclului de viață la nevoile de schimbare. În paralel, aplicația trebuie să respecte standardele de produs, principiile și procedurile deținute de SAP, care este proprietarul.

În aceste condiții, teza va focusa spre realizarea unei metodologii pentru evaluarea și ierarhizarea tehnologiilor ABAP UI, care să țină cont nu numai de arhitectura și caracteristicile tehnologiilor, dar și de acei parametri de performanță obținuți prin testarea aplicațiilor implementate în tehnologiile respective.

Luarea unei decizii, în ceea ce privește alegerea criteriilor pentru realizarea unei aplicații software cu o anumită tehnologie, poate fi făcută numai de dezvoltator [148], care trebuie să țină seama de faptul că aplicația trebuie să se integreze în mediul sistemului, aspectul și utilizarea sa să corespundă cerințelor utilizatorului și să fie cât mai eficientă.

Deoarece un model standard pentru evaluarea tehnologiilor UI pentru un anumit tip de aplicație web nu există, vor fi luate în considerare toate aspectele enumerate, propunându-se următoarele criterii după care să poată fi realizată evaluarea tehnologiilor: arhitectura aplicațiilor dezvoltate prin aceste tehnologii, caracteristici proprii tehnologiilor, precum și un set de parametri care influențează performanța aplicațiilor realizate în baza tehnologiilor. Caracteristicile proprii se referă la aspectul comportamental și utilizabilitatea acestor tehnologii, care dispun de o calitate proprie în mediul de afaceri.

Înainte ca aceste aspecte să fie dezvoltate și detaliate, criteriile care sunt propuse a fi luate în considerare, trebuie conturat și mai precis conținutul lucrării. Astfel, la realizarea metodologiei de evaluare, ar putea fi avuți în vedere și alți factori [149], ca de exemplu eventualele costuri pentru dezvoltarea și întreținerea aplicațiilor, însă se consideră că aceștia ar putea fi ignorați. În cadrul tezei de față se urmăresc doar acei factori care au o influență decisivă asupra tehnologiilor din punct de vedere UI și a caracteristicilor arhitecturale.

a. Criterii bazate pe considerente arhitecturale

În principiu, toate tehnologiile recomandate de SAP pentru dezvoltatorii ABAP sunt integrate pe platforma NetWeaver. În acest caz, în funcție de tehnologie, pot fi puse în aplicare diferite modele arhitecturale [150]. Acestea influențează modul în care aplicația este dezvoltată, afectând întregul său ciclu de viață. Conceptul MVC a fost dezvoltat pe platforma NetWeaver în cadrul server-ului de aplicații, în așa manieră încât a devenit un standard pentru aplicațiile GUI [151]. În acest context, într-o tehnologie, aplicațiile pot fi realizate printr-o diversitate de interpretări ale modelului ales.

De cele mai multe ori, utilizarea unui model arhitectural este susținută din punct de vedere tehnic [152]. Acest lucru poate fi realizat, pe de o parte prin framework-ul asociat tehnologiei respective, care pune la dispoziție componentele și ajută la aplicarea modelului, iar pe de altă parte, prin framework-uri arhitecturale sau instrumente externe care nu sunt direct parte a tehnologiilor.

O influență semnificativă în procesul de dezvoltare al unei aplicații într-o anumită tehnologie, îl are alegerea modelului arhitectural care, așa cum susțin dezvoltatorii [153], [154], poate conduce la un cod sursă de calitate superioară, poate simplifica proiectarea și reduce semnificativ efortul. Din aceste considerente, un criteriu definitoriu pentru alegerea unei tehnologii este **efortul de dezvoltare (DE)** pentru realizarea unei aplicații.

Alegerea unui model arhitectural într-o anumită tehnologie are un impact puternic asupra capacității de reutilizare a componentelor aplicațiilor. Astfel, tehnologii ca Web Dynpro ABAP și Floorplan Manager pun în aplicare definiții clare de interfață și folosesc paradigme de interacțiune între componentele definite individual, pe când altele, ca Business Server Page și HTMLB rezolvă mai puțin acest aspect, rezultând componente dependente și puternic legate unele de celelalte [155].

Dezvoltarea aplicațiilor cu componente reutilizate oferă o serie de avantaje și dezavantaje, care trebuie luate în calcul la momentul proiectării și implementării acestora [76]. În acest sens, un dezavantaj major îl constituie depanarea acestor aplicații. De aceea contează nu numai timpul de dezvoltare al aplicației, ci și efortul de modificare al acesteia. De aici rezultă un alt indicator, care va constitui criteriu în alegerea unei tehnologii, și anume, **efortul de modificare și întreținere sau de mentenanță (ME)**.

Impactul arhitecturii aplicației asupra procesului de dezvoltare depinde foarte mult de modelul ales și componentele reutilizate. Măsura în care o aplicație este performantă depinde în cele din urmă de cât de puternică este legătura între componentele sale și cât de flexibilă este arhitectura sa referitor la ulterioarele extinderi [32].

În contrast cu dezvoltarea clasică, îmbunătățirea unei aplicații cu noi funcționalități nu permite dezvoltatorului schimbarea codului original (al producătorului). Pentru aceasta, există zone specifice în care clientul poate să intervină cu propriul său cod și să extindă aplicația după nevoile sale, însă problema rămâne atunci când proprietarul aplicației face actualizări. O remarcă importantă în acest sens, este de a nu confunda personalizarea unei aplicații cu extinderea ei [156]. Extinderea presupune o îmbunătățire a aplicației din punct de vedere al codului, care poate fi uneori reflectată și în interfața UI, pe când personalizarea unei aplicații se referă doar la adaptarea UI, ceea ce se petrece grafic cu interfața aplicației. În acest sens, rezultă încă un indicator care poate fi ales ca și criteriu în evaluarea unei tehnologii: **efortul necesar extinderii (EE)** unei aplicații.

Tehnologia aleasă trebuie să îndeplinească cerințele actuale de business, dar în același timp trebuie să ofere posibilitatea de extindere prin alte module sau funcționalități, pentru a îndeplini nevoile viitoare, generate de dezvoltarea afacerii.

Cele trei criterii bazate pe considerente arhitecturale propuse pentru evaluarea și ierarhizarea tehnologiilor sunt: DE, ME și EE.

b. Criterii de performanță

Performanța unui sistem de prelucrare a datelor este definită ca fiind abilitatea sistemului pentru a îndeplini cerințele date, referitor la timpul de răspuns și transferul de date. Performanța poate fi măsurată după diverși indicatori, ca de exemplu: durata unui proces, timpul de răspuns, operații de calcul complexe pe unitate de timp, etc. [157].

Analiza performanței aplicațiilor web business descrisă în paragraful 3.3 evaluează utilizarea resurselor sistemului pentru sistemul local ABAP. Parametrii obținuți sunt timpul consumat de procesor la sfârșitul unui pas de tranzacție (CPUT) și timpul total în procesul de lucru sau timpul de răspuns (RT) cu componentele sale: timp de așteptare pentru procesul de lucru, timp de procesare, de încărcare, de generare, timp de acces al bazei de date (DBRT) și timp de punere în coada de așteptare. Impactul cel mai mare asupra performanței unei aplicații revine timpului de răspuns, timpului de acces al bazei de date și timpului consumat de procesor.

Pe lângă parametrii menționați, un rol important îl are și timpul obținut de utilizator în front-end, care pentru aplicațiile web implementate în cele trei tehnologii ABAP UI a fost măsurat cu ajutorul instrumentului HTTP-Watch pentru Microsoft Internet Explorer și Firefox.

În acest context, cei patru parametri: RT, DBRT, CPUT și HTTPWT sunt definiți ca și criterii de performanță pentru realizarea modelelor decizionale.

4. Ierarhizarea alternativelor

În vederea ierarhizării alternativelor formulate, sunt dezvoltați de autor algoritmi AITOP și AIEL pe baza metodelor de analiză multicriterială TOPSIS, ELECTRE și MUG.

Deoarece criteriile definite sunt atât numerice, provenite din măsurători experimentale, cât și non-numerice, rezultate din analiza tehnologiilor la nivelul server-ului de aplicații, procentajul stabilit pentru acordarea ponderilor acestora va fi diferit. Astfel, pentru:

- criteriile a căror date provin din măsurători (RT, CPUT, DBRT și HTTPWT) și conțin valori concrete, se stabilește un procent de 65% din suma totală acordată ponderilor, iar pentru
- criteriile a căror date conțin informații imprecise, de natură lingvistică (DE, ME, EE), se stabilește un procent de 35% din suma totală a ponderilor.

Ponderile acordate fiecărui criteriu în parte sunt determinate de un model ierarhic dezvoltat de autor, utilizând instrumentul SuperDecision.

6.2.2. Modelul MEIAHP

Acest model este construit utilizând instrumentul Super Decision, bazat pe metoda de analiză multicriterială AHP. Ponderile de importanță obținute pentru criteriile vor fi utilizate în cadrul algoritmilor AITOP și AIEL în vederea ierarhizării tehnologiilor SAP UI evaluate.

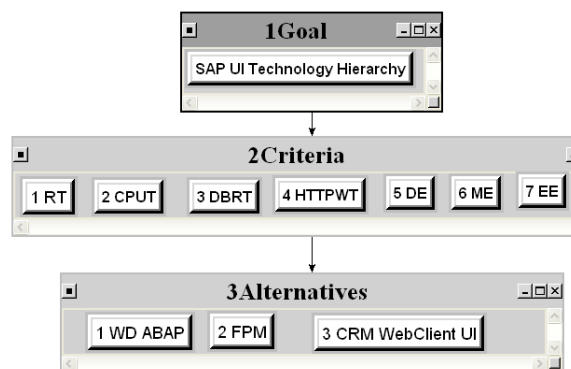


Fig. 6.3 Modelul ierarhic MEIAHP

Cele trei nivele ale modelului conțin noduri care definesc elementele problemei și anume: scopul, criteriile și alternativele posibile și sunt conectate între ele în sens top-down. Nivelul care include criteriile este evaluat în funcție de importanța lor față de scop, iar alternativele în funcție de preferințe în raport cu criteriile. Odată ce această structură ierarhică este definită, elementele de pe fiecare nivel sunt analizate în cadrul unei proceduri de prioritizare, stabilindu-se importanța relativă a acestora pe fiecare nivel.

În cadrul acestei analize se ține cont pentru:

- criteriile numerice (RT, CPU, DBRT și HTTPWT), de analiza parametrilor rezultați din măsurători și prelucrați în capitolele 4 și 5;
- criteriile non-numerice (DE, ME, EE), de analiza prezentată în capitolele 2 și 3.

Având la bază metodologia AHP, determinarea ponderilor relative ale criteriilor se face prin compararea acestora două câte două, în raport cu obiectivul aflat la nivelul imediat superior ierarhiei. Comparațiile pe perechi se realizează prin evaluarea acestora pe baza unei scale de la 1 la 9, (tabelul 6.2), ce reprezintă gradul de importanță al unui criteriu față de altul. Chestionarul este prezentat în figura 6.4.

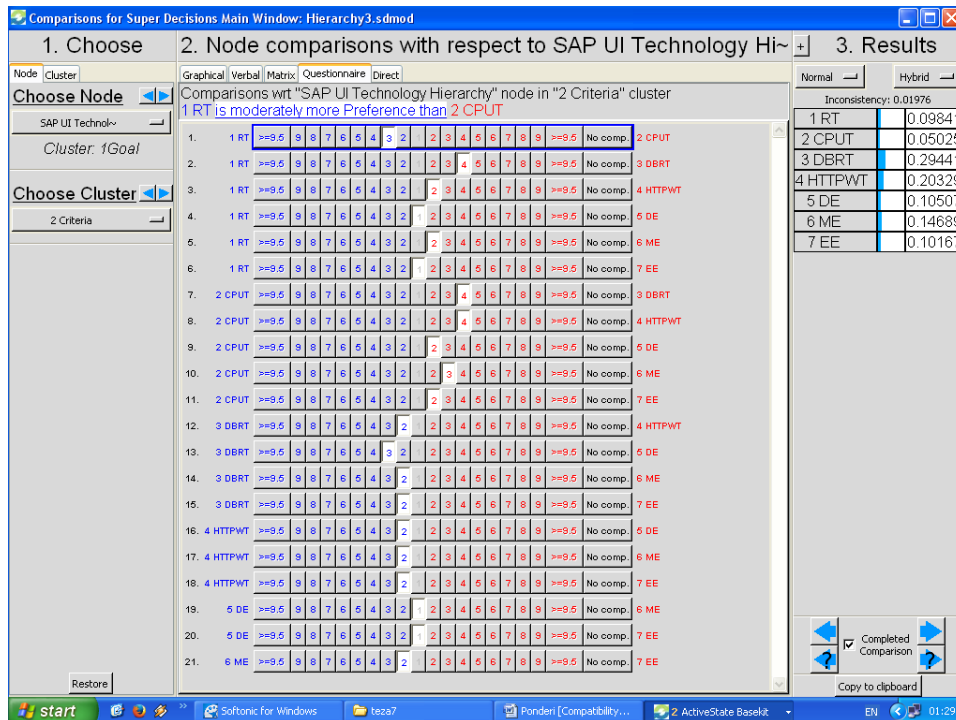


Fig. 6.4 Chestionarul comparării perechilor de criterii

Rezultatul cuantificării prezentat și în lucrarea [158] este afișat în matricea decizională a criteriilor din figura 6.5.

Inconsistency	2 CPUT ~	3 DBRT ~	4 HTTPWT ~	5 DE ~	6 ME ~	7 EE ~
1 RT ~	← 3	↑ 4	↑ 2	← 1	↑ 2	← 1
2 CPUT ~		↑ 4	↑ 4	↑ 2	↑ 3.0000	↑ 2
3 DBRT ~			← 2	← 3	← 2	← 2
4 HTTPWT ~				← 2	← 2	← 2
5 DE ~					← 1	← 1
6 ME ~						← 2

Fig. 6.5 Matricea decizională a criteriilor

Raportul referitor la inconsistența comparațiilor criteriilor unul față de altul este prezentat în figura 6.6. O inconsistență mai mică de 10% este tolerabilă, dar tot ce depășește această valoare e necesar a se ajusta. Din raport și figura 6.4 se observă că inconsistența matricei este de 0,01976, ceea ce indică faptul că vectorul ponderilor este bine determinat.

Rank	Row	Col	Current Val	Best Val	Old Inconsist	New Inconsist	% Improvement
1.	1 RT	2 CPUT	3.000000	1.647105	0.019757	0.015108	23.53 %
2.	2 CPUT	3 DBRT	4.000000	6.824648	0.019757	0.016283	17.58 %
3.	4 HTTPWT	6 ME	2.000000	1.194200	0.019757	0.016607	15.95 %
4.	3 DBRT	7 EE	2.000000	3.358864	0.019757	0.016628	15.84 %
5.	3 DBRT	4 HTTPWT	2.000000	1.274152	0.019757	0.016896	14.48 %
6.	6 ME	7 EE	2.000000	1.269891	0.019757	0.016920	14.36 %
7.	5 DE	6 ME	1.000000	1.597545	0.019757	0.017348	12.19 %
8.	1 RT	3 DBRT	4.000000	2.670065	0.019757	0.017540	11.22 %
9.	1 RT	6 ME	2.000000	1.330221	0.019757	0.017563	11.10 %
10.	3 DBRT	5 DE	3.000000	2.729212	0.019757	0.019560	1.00 %
11.	1 RT	5 DE	1.000000	1.094983	0.019757	0.019596	0.81 %
12.	2 CPUT	5 DE	2.000000	2.126985	0.019757	0.019677	0.40 %
13.	4 HTTPWT	5 DE	2.000000	1.910090	0.019757	0.019716	0.21 %
14.	1 RT	7 EE	1.000000	1.046166	0.019757	0.019717	0.20 %
15.	5 DE	7 EE	1.000000	1.046652	0.019757	0.019717	0.20 %
16.	1 RT	4 HTTPWT	2.000000	2.091699	0.019757	0.019717	0.20 %
17.	2 CPUT	6 ME	3.000003	2.894047	0.019757	0.019732	0.13 %
18.	2 CPUT	7 EE	2.000000	2.032132	0.019757	0.019750	0.03 %
19.	2 CPUT	4 HTTPWT	4.000000	4.062905	0.019757	0.019751	0.03 %
20.	4 HTTPWT	7 EE	2.000000	1.999400	0.019757	0.019757	0.00 %
21.	3 DBRT	6 ME	2.000000	2.005939	0.019757	0.019757	-0.00 %

Fig. 6.6 Raportul de inconsistență al comparației criteriilor

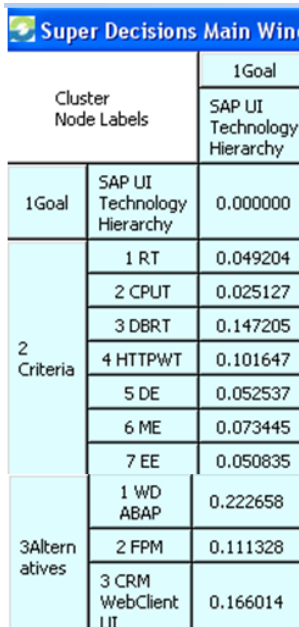
Matricea neponderată rezultată din comparațiile de perechi pentru criterii și alternative redată în figura 6.7 este numită Super Matrice și conține prioritățile din grupurile de comparații a criteriilor în raport cu scopul și a alternativelor în raport cu criteriile. Matricea ponderată într-o structură ierarhică este aceeași cu matricea neponderată.

Determinarea scorului total de prioritate al fiecărei alternative se obține ca sumă a produselor dintre ponderea relativă pentru un anumit criteriu și ponderea relativă a unei alternative în raport cu acel criteriu.

Cluster Node Labels		1 Goal	2 Criteria						
		SAP UI Technology Hierarchy	1 RT	2 CPUT	3 DBRT	4 HTTPWT	5 DE	6 ME	7 EE
1 Goal	SAP UI Technology Hierarchy	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
2 Criteria	1 RT	0.098408	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	2 CPUT	0.050254	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	3 DBRT	0.294410	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	4 HTTPWT	0.203295	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	5 DE	0.105074	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	6 ME	0.146889	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	7 EE	0.101669	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3 Alternatives	1 WD ABAP	0.000000	0.707117	0.063252	0.181821	0.704936	0.071927	0.648329	0.717235
	2 FPM	0.000000	0.070155	0.193882	0.272723	0.210920	0.278954	0.229651	0.194688
	3 CRM WebClient UI	0.000000	0.222728	0.742867	0.545455	0.084144	0.649118	0.122020	0.088077

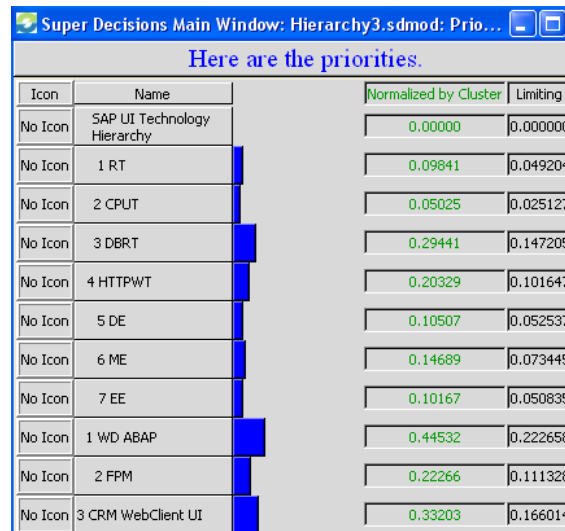
Fig. 6.7 Matricea normalizată a ponderilor (Super Matrice)

Super Matricea limită, figura 6.8, este versiunea finală sau răspunsul pentru modelul de evaluare și ierarhizare. Prioritățile finale ale alternativelor se găsesc în coloana 1 Goal. Rezultatele priorităților sunt afișate în figura 6.9.



Cluster Node Labels		1Goal
1Goal	SAP UI Technology Hierarchy	0.000000
2 Criteria	1 RT	0.049204
	2 CPUT	0.025127
	3 DBRT	0.147205
	4 HTTPWT	0.101647
	5 DE	0.052537
	6 ME	0.073445
	7 EE	0.050835
3Alternatives	1 WD ABAP	0.222658
	2 FPM	0.111328
	3 CRM WebClient UI	0.166014

Fig. 6.8 Matricea limită

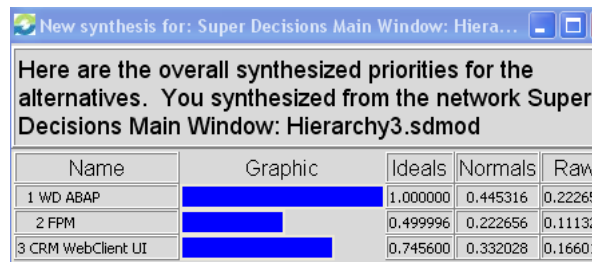


Here are the priorities.

Icon	Name	Normalized by Cluster	Limiting
No Icon	SAP UI Technology Hierarchy	0.00000	0.000000
No Icon	1 RT	0.09841	0.049204
No Icon	2 CPUT	0.05025	0.025127
No Icon	3 DBRT	0.29441	0.147205
No Icon	4 HTTPWT	0.20329	0.101647
No Icon	5 DE	0.10507	0.052537
No Icon	6 ME	0.14689	0.073445
No Icon	7 EE	0.10167	0.050835
No Icon	1 WD ABAP	0.44532	0.222658
No Icon	2 FPM	0.22266	0.111328
No Icon	3 CRM WebClient UI	0.33203	0.166014

Fig. 6.9 Prioritățile elementelor modelului ierarhic

Rezultatul final se regăsește și în lucrarea autorului [159] și este dat de sinteza priorităților pentru alternative, sau ierarhizarea alternativelor, figura 6.10, în care valorile Raw provin din matricea limită, valorile Normals din normalizarea valorilor Raw și valorile Ideals rezultă din împărțirea tuturor valorilor Raw la cea mai mare dintre ele.



Here are the overall synthesized priorities for the alternatives. You synthesized from the network Super Decisions Main Window: Hierarchy3.sdmod

Name	Graphic	Ideals	Normals	Raw
1 WD ABAP		1.000000	0.445316	0.22265
2 FPM		0.499996	0.222656	0.11132
3 CRM WebClient UI		0.745600	0.332028	0.16601

Fig. 6.10 Ierarhizarea alternativelor

Pentru a realiza stabilitatea și compatibilitatea rezultatului obținut, se aplică analiza de sensibilitate. Această analiză este utilă pentru a demonstra modul în care decizia luată în urma ierarhizării tehnologiilor poate fi modificată. Modificarea ponderii criteriului mai sensibil duce la schimbări în prioritățile alternativelor.

Analiza de sensibilitate permite selectarea oricărei combinații de variabile independente. Ele pot fi noduri, intrările super matricei, sau chiar combinații de noduri. Prioritățile alternativelor sunt reprezentate grafic pentru fiecare punct de pe axa x din grafic, etichetate "Experiments".

Din analiza senzitivă pentru parametrul DBRT (figura 6.11a), care este mai sensibil la modificarea ponderii, se observă că prioritățile tehnologiilor se schimbă astfel:

- WDA, CRM WebClient Ui și FPM, pentru o pondere mai mică de 0,45;
- CRM WebClient UI, WDA și FPM pentru o pondere cuprinsă între 0,45 și 0,79;
- CRM WebClient UI, FPM și WDA pentru o pondere mai mare de 0,79.

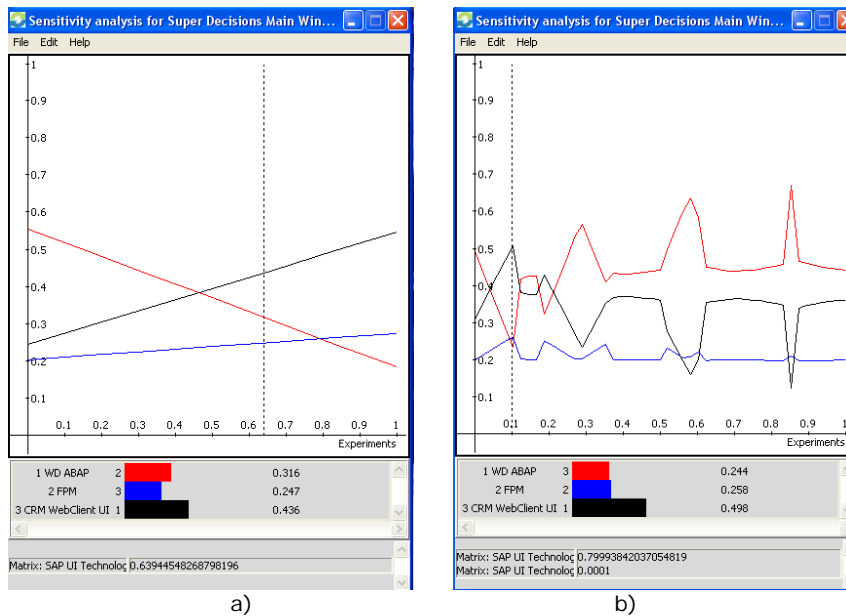


Fig. 6.11 Analiza senzitivă în raport cu: a) DBRT și b) HTTPWT

Din analiza senzitivă pentru parametrul HTTPWT (figura 6.11b), care este un parametru independent, prin modificarea ponderii, se observă că în general pentru valori mai mari decât 0.1, prioritățile tehnologiilor se modifică astfel: WDA, CRM WebClient UI și FPM.

Dacă se efectuează analiza senzitivă comparativă la nivelul tehnologiilor pentru doi parametri, de exemplu RT și HTTPWT (figura 12a), se observă că o modificare a ponderilor nu produce modificări pentru prioritatea tehnologiei FPM, care este totdeauna pe locul 3, ci doar pentru tehnologiile WDA și CRM WebClient UI, care se schimbă alternativ.

Figura 12b prezintă analiza de sensibilitate comparativă pentru parametrii HTTP și DBRT pentru cele trei tehnologii. Se observă cu ușurință, cum se modifică prioritățile alternativelor atunci când ponderile se schimbă.

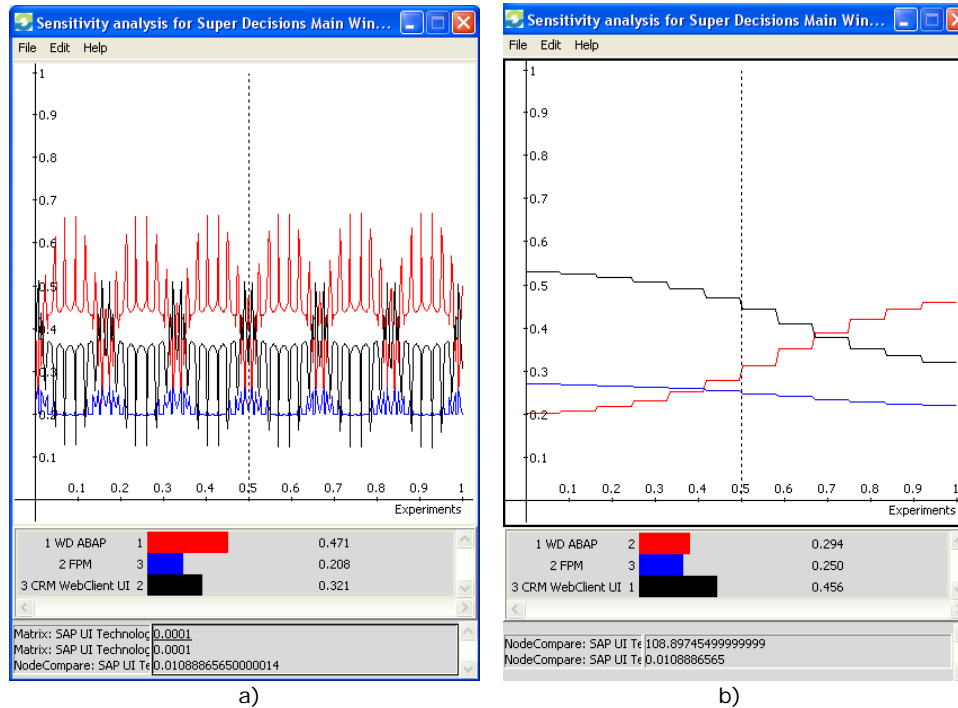


Fig. 6.12 Analiza senzitivă în raport cu: a) RT și HTTP; b) HTTP și DBRT

6.2.3 Algoritmul AITOP

Acest algoritm urmărește etapele de analiză multicriterială ale metodei TOPSIS (figura 6.13), în care ponderile pentru criterii sunt determinate în cadrul modelului MEIAHP. În acest sens, algoritmul ia în considerare:

- alternativele, corespunzătoare celor 3 tehnologii analizate: Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI;
- setul de criterii: CPUT, DBRT, HTTPWT, RT, DE, ME și EE;
- setul de ponderi de importanță pentru fiecare criteriu.

Fiecare criteriu din matricea decizională este fie monoton crescător fie monoton descrescător. Pentru alternativa optimă toate caracteristicile analizate trebuie să fie minime.

Orice rezultat care este exprimat într-un mod non-numeric trebuie cuantificat prin tehnica de scalare adecvată. Scala propusă pentru criteriile non-numerice DE, ME și EE este considerată cea prezentată în tabelul 6.2.

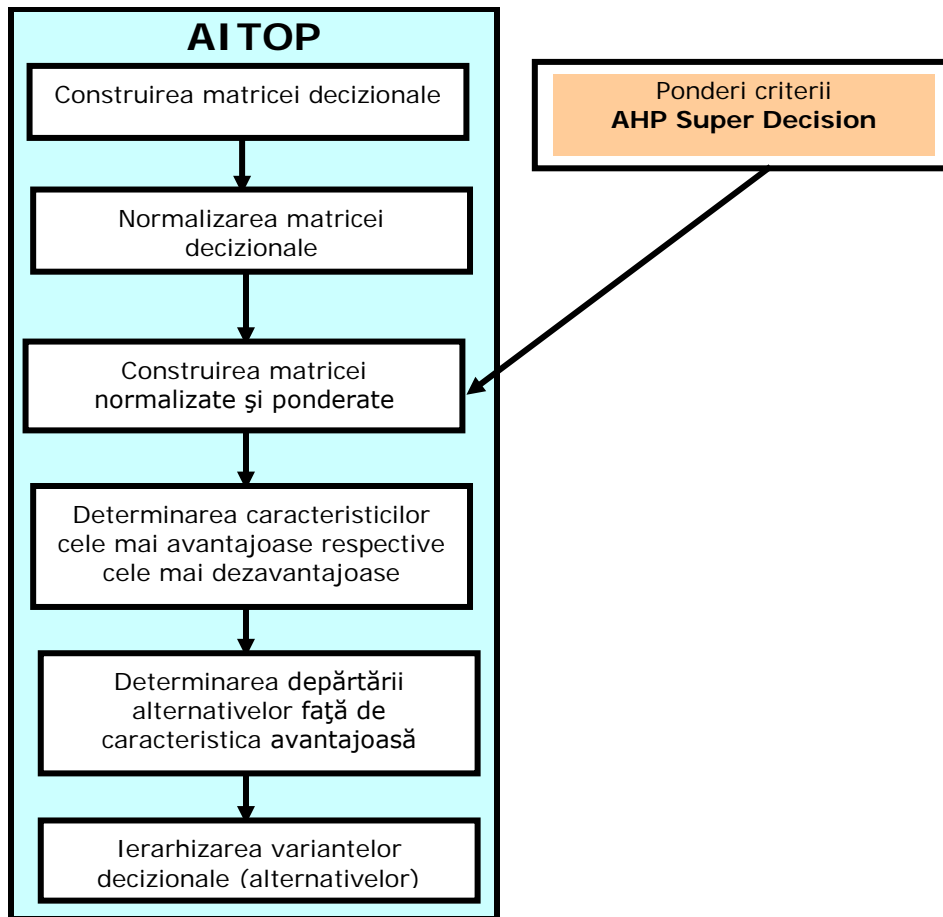


Fig.6.13. Algoritmul AITOP

Valorile numerice ale criteriilor: CPUT, DBRT, HTTPWT și RT prezentate în tabelul 6.3 corespund valorilor estimațiilor punctuale ale parametrilor mășurați pentru cele 3 tehnologii [98]. În același tabel sunt prezentate valorile eforturilor: DE, ME și EE, convertite conform scalei de conversie din tabelul 6.2.

Tab. 6.3. Matricea decizională cu variantele decizionale și criteriile de decizie

Tehn.	Var. dec. v_i	Criterii de decizie c_i						
		c_1 CPUT ∇ minim	c_2 DBRT ∇ minim	c_3 HTTPWT ∇ minim	c_4 RT ∇ minim	c_5 DE ∇ minim	c_6 ME ∇ minim	c_7 EE ∇ minim
WDA	v_1	315.00	187.24	1830.2	476.53	7	3	3
FPM	v_2	384.57	300.43	2254.33	628.76	5	5	5
CRM WCUI	v_3	2249.25	805.43	5824.00	2722.24	3	7	7
	$\sqrt{\sum_{i=1}^m c_{ij}^2}$	2303.529	879.792	6507.735	2834,257	9.110	9.110	9.110

Pentru a putea compara criteriile, matricea decizională este transformată în matricea normalizată. Scopul normalizării este de a transforma diferitele atribute dimensionale în atribute adimensionale.

În literatură sunt descrise mai multe procedee de normalizare, sintetizate în tabelul 6.4 [160], [161], [162], [163].

Tab. 6.4. Procedee de normalizare

	Tip	Definiție	Domeniu	Conservă proporționalitatea	Interpretare
Procedeu 1	Normalizare vectorială	$CN_{ij} = \frac{C_{ij}}{\sqrt{\sum_{i=1}^m C_{ij}^2}}$	$0 < r_{ij} < 1$	Da	Componenta i a vectorului unitate
Procedeu 2	Normalizare liniară	$CN_{ij} = \frac{C_{ij}}{\sum_{i=1}^m C_{ij}}$	$0 < r_{ij} < 1$	Da	% din total
Procedeu 3	Normalizare liniară	$CN_{ij} = \frac{C_{ij}}{C_{ij}^{max}}$ Criteriu maximizant	$0 \leq r_{ij} \leq 1$	Da	% din maxim
		$CN_{ij} = \frac{C_{ij}^{min}}{C_{ij}}$ Criteriu minimizant			% din minim
Procedeu 4	Normalizare liniară	$CN_{i,j} = \frac{C_{i,j} - \min C_{i,j}}{\max C_{i,j} - \min C_{i,j}}$ Criteriu maximizant	$0 \leq r_{ij} \leq 1$	Da	% din domeniu
		$U_{i,j} = \frac{\max C_{i,j} - C_{i,j}}{\max C_{i,j} - \min C_{i,j}}$ Criteriu minimizant			

Pentru normalizarea matricei decizionale s-a utilizat procedeul 1, normalizare vectorială, descris prin relația (6.7). De exemplu:

$$CN_{11} = \frac{C_{11}}{\sqrt{\sum_{i=1}^3 C_{i1}^2}} = \frac{315.00}{\sqrt{315.00^2 + 384.57^2 + 2249.25^2}} = \frac{315.00}{2303.529} = 0.1367$$

În tabelul 6.5 sunt prezentate valorile normalizate corespunzătoare criteriilor stabilite pentru tehnologiile analizate, la care s-au adăugat ponderile de importanță corespunzătoare fiecărui criteriu, determinate în paragraful 6.2.2.

Tab.6.5. Matricea normalizată și a ponderilor de importanță.

Tehn.	Var. v_i	Valori normalizate ale consecințelor CN_i						
		C_1	C_2	C_3	C_4	C_5	C_6	C_7
		CPUT	DBRT	HTTPWT	RT	DE	ME	EE
WDA	v_1	0.1367	0.2128	0.2812	0.1681	0.7683	0.3293	0.3293
FPM	v_2	0.1669	0.3415	0.3464	0.2218	0.5488	0.5488	0.5488
CRM WCUI	v_3	0.9764	0.9155	0.8949	0.9605	0.3293	0.7683	0.7683
Ponderi de importanță p_i		0.05	0.3	0.2	0.1	0.01	0.31	0.03
		0.65				0.35		

Matricea normalizată ponderată se obține prin înmulțirea valorilor normalizate cu ponderile de importanță (p_j) ale fiecărui criteriu. Ea este prezentată în tabelul 6.6.

Tab. 6.6. Matricea normalizată ponderată

Tehn.	Var. v_i	Valori normalizate ponderate ale consecințelor CNP_i						
		C_1	C_2	C_3	C_4	C_5	C_6	C_7
		CPUT minim	DBRT minim	HTTPWT minim	RT minim	DE minim	ME minim	EE minim
WDA	v_1	0.0068	0.0638	0.0562	0.0168	0.0085	0.1021	0.0089
FPM	v_2	0.0083	0.1024	0.0693	0.0222	0.0060	0.1701	0.0148
CRM WCUI	v_3	0.0488	0.2746	0.1790	0.0960	0.0036	0.2382	0.0207

Din tabelul 6.6 sunt selectate cu ajutorul relațiilor (6.9) și (6.10), pentru fiecare criteriu ($j=1\dots n$), valoarea cea mai avantajoasă (ideal pozitivă) C_j^+ , respectiv cea mai dezavantajoasă (ideal negativă) C_j^- . Pentru aceasta se ține cont de tipul criteriului respectiv (maximizant sau minimizant). Valorile calculate sunt prezentate în tabelul 6.7.

Tab. 6.7. Valorile cele mai avantajoase și cele mai dajavantajoase ale criteriilor analizate

	C_1	C_2	C_3	C_4	C_5	C_6	C_7
	CPUT	DBRT	HTTPWT	RT	DE	ME	EE
C_j^+	0.0068	0.0638	0.0562	0.0168	0.0036	0.1021	0.0089
C_j^-	0.0488	0.2746	0.1790	0.0960	0.0085	0.2382	0.0207

Depărtarea fiecărei alternative față de valoarea cea mai avantajoasă (S_i^+), respectiv cea mai dezavantajoasă (S_i^-), calculate cu ajutorul relațiilor (6.11) și (6.12) sunt prezentate în tabelul 6.8.

Apropierea relativă față de soluția ideală (scorul) este exprimată de mărimea S^* calculată cu relația (6.13). Pe baza valorilor obținute sunt ierarhizate variantele decizionale față de soluția ideală.

Tab. 6.8. Valorile depărtării fiecărei alternative față de valoarea cea mai avantajoasă, respectiv cea mai dezavantajoasă și ierarhizarea variantelor decizionale față de soluția ideală

	S^+	S^-	Scorul S^*	Ordinea de preferință
WDA	0.0048	0.2936	0.9838	I
FPM	0.0798	0.2312	0.7435	II
CRM WCUI	0.2936	0.0048	0.0162	III

În urma aplicării algoritmului, rezultă că tehnologia clasată pe primul loc din punct de vedere al performanțelor este Web Dynpro ABAP. Această ierarhizare este valabilă dacă sunt luate în considerare cele 7 criterii și cele 7 ponderi de importanță acordate acestor criterii.

Dacă sunt luate în considerare alte criterii sau alte ponderi de importanță ale acestora este posibil ca ordinea de preferință să se modifice.

6.2.4 Algoritmul AIEL

Algoritmul AIEL dezvoltat pentru ierarhizarea celor trei tehnologii urmărește etapele de rezolvare multicriterială ale metodei ELECTRE (figura 6.14), în care utilitățile pentru calculul indicilor de concordanță se determină la fel ca și pentru metoda MUG, iar ponderile criteriilor sunt determinate în cadrul modelului ierarhic MEIAHP.

Pentru acest algoritm se consideră:

- alternativele corespunzătoare celor 3 tehnologii SAP: Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI;
- setul de criterii: CPUT, DBRT, HTTPWT, RT, DE, ME și EE;
- setul de ponderi de importanță pentru fiecare criteriu.

Valorile numerice ale caracteristicilor: CPUT, DBRT, HTTPWT și RT prezentate în tabelul 6.9 corespund valorilor estimațiilor punctuale ale parametrilor mășurați, pentru cele 3 tehnologii [98]. În același tabel sunt prezentate valorile eforturilor: DE, ME și EE convertite conform scalei de conversie din tabelul 6.2.

Tab. 6.9. Matricea decizională cu variantele decizionale și criteriile de decizie

Tehn.	Var. dec. v_i	Criterii de decizie c_j						
		c_1	c_2	c_3	c_4	c_5	c_6	c_7
		CPUT ∇ minim	DBRT ∇ minim	HTTPWT ∇ minim	RT ∇ minim	DE ∇ minim	ME ∇ minim	EE ∇ minim
WDA	v_1	315.00	187.24	1830.2	476.53	7	3	3
FPM	v_2	384.57	300.43	2254.33	628.76	5	5	5
CRM WCUI	v_3	2249.25	805.43	5824.00	2722.24	3	7	7

Cu ajutorul matricei de decizie se construiește matricea utilităților. Deoarece toate caracteristicile sunt minimizante, se convertește fiecare valoare din matricea decizională în valoarea utilității corespunzătoare folosind relația (6.2).

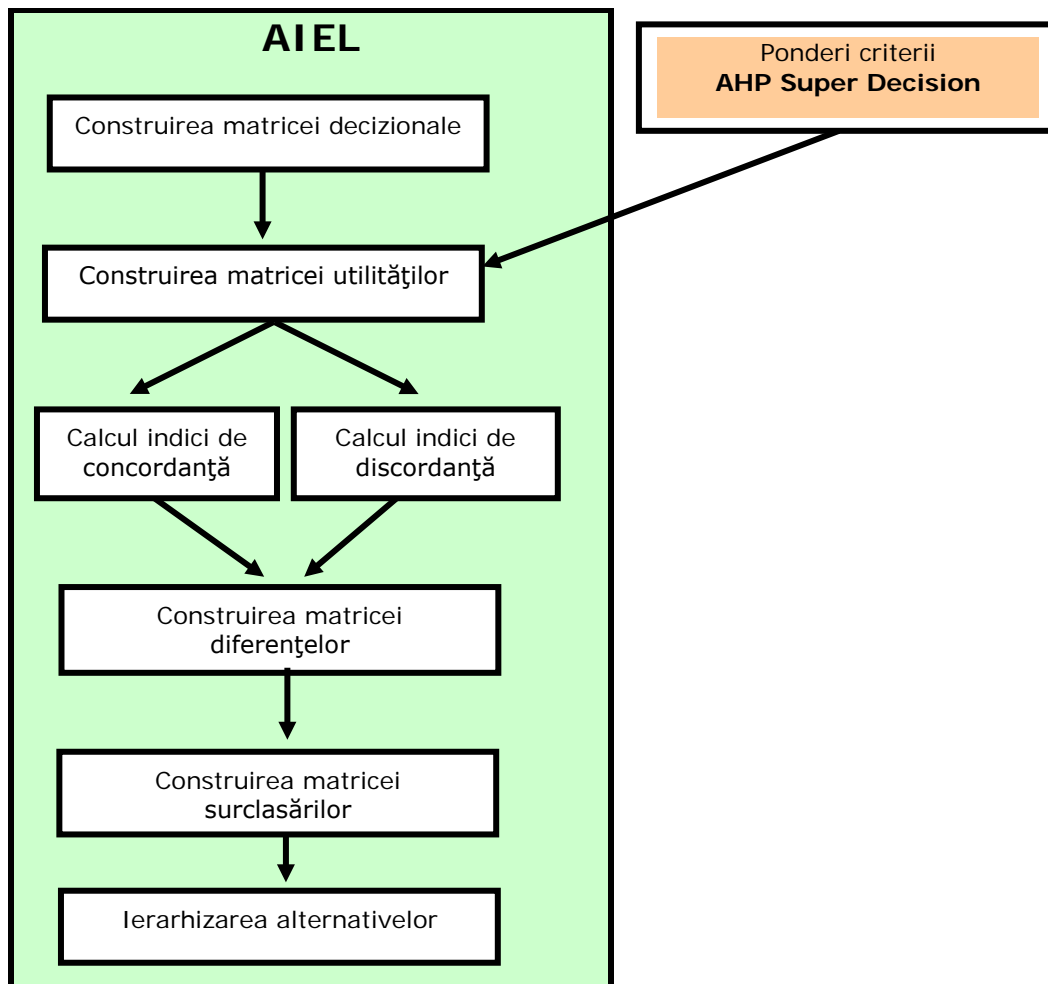


Fig.6.14. Algoritmul AIEL

De exemplu:

$$U_{1,1} = \frac{\max C_{i,j} - X_{1,1}}{\max C_{i,j} - \min C_{i,j}} = \frac{2249.25 - 315.00}{2249.25 - 315.00} = 1$$

$$U_{2,1} = \frac{\max C_{i,j} - X_{2,1}}{\max C_{i,j} - \min C_{i,j}} = \frac{2249.25 - 384.57}{2249.25 - 315.00} = 0.96 \dots\dots$$

În tabelul 6.10 sunt prezentate valorile utilităților corespunzătoare criteriilor de decizie și a tehnologiilor analizate, la care s-au adăugat ponderile de importanță corespunzătoare criteriilor.

Tab. 6.10. Matricea utilităților și a ponderilor de importanță.

Tehnologia	Variante v_i	Criterii de decizie c_j						
		c_1 CPUT	c_2 DBRT	c_3 HTTPWT	c_4 RT	c_5 DE	c_6 ME	c_7 EE
WDA	v_1	1,00	1,00	1,00	1,00	0,00	1,00	1,00
FPM	v_2	0,96	0,82	0,89	0,93	0,50	0,50	0,50
CRM WCUI	v_3	0,00	0,00	0,00	0,00	1,00	0,00	0,00
Ponderi de importanță	k_j	0.05	0.3	0.2	0.1	0,01	0,31	0,03
		0.65				0.35		

Utilitatea globală pentru fiecare variantă decizională se calculează ca medie ponderată a utilităților conform relației (6.5).

$$UG_1 = \sum_{j=1}^n k_j \cdot u_{1,j} = k_1 \cdot u_{1,1} + k_2 \cdot u_{1,2} + \dots = 0.05 \cdot 1 + 0.3 \cdot 1 + 0.2 \cdot 1 + \\ + 0.1 \cdot 1 + 0.01 \cdot 0 + 0.31 \cdot 1 + 0.03 \cdot 1 = 0.99$$

$$UG_2 = \sum_{j=1}^n k_j \cdot u_{2,j} = k_1 \cdot u_{2,1} + k_2 \cdot u_{2,2} + \dots = 0.05 \cdot 0.96 + 0.3 \cdot 0.84 + 0.2 \cdot 0.89 + \\ + 0.1 \cdot 0.93 + 0.01 \cdot 0.5 + 0.31 \cdot 0.5 + 0.03 \cdot 0.5 = 0.74$$

$$UG_3 = \sum_{j=1}^n k_j \cdot u_{3,j} = k_1 \cdot u_{3,1} + k_2 \cdot u_{3,2} + \dots = 0.05 \cdot 0 + 0.3 \cdot 0 + 0.2 \cdot 0 + 0.1 \cdot 0 + \\ + 0.01 \cdot 1 + 0.31 \cdot 0 + 0.03 \cdot 0 = 0.01$$

Valorile utilităților globale calculate și ordinea de preferință pentru cele 3 tehnologii analizate sunt prezentate în tabelul 6.11.

Tab. 6.11. Valorile utilității globale și ordinea de preferință pentru cele 3 tehnologii

Tehnologia	Criterii de decizie c_j							Utilitatea globală	Ordinea de preferință
	CPUT	DBRT	HTTPWT	RT	DE	ME	EE		
WDA	0,05	0,30	0,20	0,10	0,00	0,31	0,03	0,99	I
FPM	0,05	0,25	0,18	0,09	0,01	0,16	0,01	0,74	II
CRM WCUI	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,01	III

Metoda ELECTRE servește la compararea variantelor V_1, V_2, \dots, V_m din punct de vedere al criteriilor x_1, x_2, \dots, x_n .

Aplicarea metodei ELECTRE se bazează pe două grupe de indicatori și anume: *indicatori de concordanță* $p(V_x, V_y)$ și *indicatori de discordanță* $q(V_x, V_y)$.

Comparând două variante, V_x și V_y , indicatorii de concordanță scot în evidență aspectele favorabile ale variantei V_x față de varianta V_y , iar indicatorii de discordanță evidențiază aspectele nefavorabile ale variantei V_x față de V_y [152].

Cei doi indici sunt calculați cu relațiile (6.15) respectiv (6.16).

a) **Calcularea indicilor de concordanță**

$$\begin{array}{lll} p(V_1, V_1) = p_{1,1} = 1 & p(V_1, V_2) = p_{1,2} = 0.99 & p(V_1, V_3) = p_{1,3} = 0.99 \\ p(V_2, V_1) = p_{2,1} = 0.01 & p(V_2, V_2) = p_{2,2} = 1 & p(V_2, V_3) = p_{2,3} = 0.99 \\ p(V_3, V_1) = p_{3,1} = 0.01 & p(V_3, V_2) = p_{3,2} = 0.01 & p(V_3, V_3) = p_{3,3} = 1 \end{array}$$

b) **Calcularea indicilor de discordanță**

Pentru calcularea indicilor de discordanță se utilizează relația (6.16):

$$q(V_x, V_y) = q_{xy} = \frac{1}{E} \cdot \max \begin{cases} 0, & \text{dacă } u_y \leq u_x \\ |u_y - u_x|, & \text{dacă } u_y > u_x \end{cases}$$

$$\begin{array}{l} q(V_1, V_2) = \max\{0; 0; 0; 0; 0.5; 0; 0\} = 0.5 \\ q(V_1, V_3) = \max\{0; 0; 0; 0; 1; 0; 0\} = 1 \\ q(V_2, V_1) = \max\{0,03318; 0,16446; 0,10619; 0,06779; 0; 0.5; 0.5\} = 0.5 \\ q(V_2, V_3) = \max\{0; 0; 0; 0; 0.5; 0; 0\} = 0.5 \\ q(V_3, V_1) = \max\{1; 1; 1; 1; 0; 1; 1\} = 1 \\ q(V_3, V_2) = \max\{0,96403; 0,8169; 0,8938; 0,9322; 0; 0.5; 0.5\} = 0,96403 \end{array}$$

În tabelul 6.12 este prezentată matricea indicilor de concordanță și a indicilor de discordanță calculați.

Tab. 6.12. Matricea indicilor de concordanță și de discordanță

p_{xy}		v1	v2	v3	Indici de concordanță
	v1	1	0,99	0,99	
	v2	0,01	1	0,99	
	v3	0,01	0,01	1	
q_{xy}		v1	v2	v3	Indici de discordanță
	v1	0	0,5	1	
	v2	0,5	0	0,5	
	v3	1	0,964	0	

După determinarea acestor două matrici se calculează matricea diferențelor conform relației (6.17).

În tabelul 6.13 este prezentată matricea diferențelor dintre indicii de concordanță și cei de discordanță.

Tab. 6.13. Matricea diferențelor dintre indicii de concordanță și cei de discordanță

$d_{xy} = p_{xy} - q_{xy}$		v1	v2	v3
	v1	1,000	0,490	-0,010
	v2	-0,490	1,000	0,490
	v3	-0,990	-0,954	1,000

Pe baza matricei diferențelor se calculează matricea surclasărilor [164], [165], prezentată în tabelul 6.14. Se compară elementele $(V_x; V_y)$ cu $(V_y; V_x)$. În

locul elementelor cu valoare mai mare se trece 1 în matricea surclasărilor, iar în locul elementelor cu valoare mai mică se trece 0; pe diagonala se trece 1, iar când elementele au valori egale se trece tot 1.

Tab. 6.14. Matricea surclasărilor

	v1	v2	V3	S	ordinea
v1	1	1	1	3	I
v2	0	1	1	2	II
v3	0	0	1	1	III

Comparând variantele, fiecare cu toate celelalte, se obține numărul de dominante pentru fiecare dintre acestea. Numărul de dominante pentru fiecare variantă este $V_1= 3$, $V_2= 2$, $V_3= 1$, de unde rezultă următoarea ordine de preferință: $V_1 > V_2 > V_3$.

În urma implementării modelului ierarhic și a celor doi algoritmi de evaluare, se constată că:

- În funcție de criterii și de ponderile acordate, rezultatul procesului de ierarhizare poate fi diferit.
- Implementarea modelului ierarhic utilizând instrumentul Super Decision permite ca rezultatul să fie interpretat și sub forma analizei senzitive.
- Normalizarea valorilor consecințelor matricei decizionale pentru algoritmi AITOP și AIIEEL se poate realiza prin procedee diferite, rezultatul obținut fiind același.
- Cele trei modele de evaluare conduc la rezultate diferite în ceea ce privește ierarhizarea celor trei tehnologii SAP UI.

Algoritmii astfel construiți sunt flexibili și oferă următoarele avantaje:

- Ponderile acordate criteriilor se pot stabili în funcție de obiectivele evaluării.
- Permit compararea diferitelor alternative, oferind suport în alegerea alternativei optime în funcție de scenariul care se dorește a se utiliza.
- Permit extensia prin adăugarea de noi categorii/subcategorii/criterii de evaluare a calității.
- Permit actualizarea metricilor de măsurare a criteriilor de evaluare în funcție de evoluția cerințelor și de obiectivele evaluării.

6.3. Concluzii

În capitolul de față, cele trei tehnologii SAP: WD ABAP, FPM și CRM WebClient UI, au fost evaluate și ierarhizate pe baza metodologiei propuse și a algoritmilor AITOP și AIEL. Algoritmii implementați combină metodele de analiză multicriterială TOPSIS, ELECTRE și MUG, în care ponderile sunt determinate în cadrul modelului ierarhic MEIAHP, dezvoltat de autor, utilizând instrumentul SuperDecision. Metodologia de evaluare propusă evidențiază etapele de rezolvare și modul de acordare a ponderilor în cadrul algoritmilor implementați.

Concluzionând, se poate afirma că:

- Analiza și prelucrarea seturilor de date experimentale efectuată în capitolele 4 și 5 a fost necesară pentru stabilirea corectă a importanței criteriilor și a determinării ponderilor corespunzătoare acestora.
- Ținând cont de seturile de criterii alese și ponderile acordate acestora, tehnologia WD ABAP este indicată ca fiind cea mai performantă.
- Modelul ierarhic MEIAHP dezvoltat de autor, utilizând instrumentul Super Decizion, permite efectuarea unei analize complete în ceea ce privește evaluarea multicriterială și ierarhizarea celor trei tehnologii.
- Analiza senzitivă pentru modelul MEIAHP demonstrează că modificarea ponderilor acordate criteriilor poate produce modificări semnificative în ierarhizarea acestora.

7. Concluzii, contribuții personale și dezvoltări viitoare

În cadrul acestui capitol sunt prezentate concluziile finale ale cercetărilor întreprinse în cadrul prezentei teze, principalele contribuții aduse de autor, precum și dezvoltările pentru viitor la nivel de extindere și optimizare a metodologiei de evaluare.

Prezenta teză se adresează dezvoltatorilor de aplicații web, abordând probleme teoretice și aplicative de o reală importanță, direcționate în principal pe analiza și evaluarea multicriterială a tehnologiilor UI la nivelul platformei de aplicații SAP.

Soluțiile propuse pentru dezvoltările pe viitor iau în considerare faptul că analiza efectuată nu a cuprins toată gama de tehnologii SAP UI și nici toate criteriile pentru care s-a construit modelul decizional.

7.1. Concluzii finale

Prezenta teză este concepută ca o dezvoltare succesivă de problematici specifice analizei tehnologiilor ABAP UI de pe platforma SAP NetWeaver, sfârșitul fiecărui capitol evidențiind aspectele particulare sau generale ale acestora.

Sunt prezentate într-un mod sistematizat conceptele de bază specifice tehnologiilor UI, recomandate de SAP pentru platforma Business Suite a server-ului de aplicații ABAP și sunt evidențiate proprietăților și calităților acestora.

Obiectivul principal al tezei constă în dezvoltarea unei metodologii pentru evaluarea și ierarhizarea tehnologiilor Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI. La baza metodologiei de evaluare multicriterială stau cercetările efectuate, care susțin alegerile referitoare la definirea criteriilor și ponderilor necesare implementării algoritmilor pentru ierarhizare.

În contextul acestui obiectiv, se pot evidenția următoarele direcții de cercetare abordate în teză:

- analiza modelelor arhitecturale și de programare a tehnologiilor ABAP UI pe platforma SAP NetWeaver;
- analiza aplicațiilor web business implementate prin tehnologiile ABAP UI;
- analiza statistică, corelațională și de regresie a datelor experimentale provenite din testarea aplicațiilor web business implementate prin tehnologiile ABAP UI;
- analiza proceselor decizionale și a metodelor de analiza multicriterială.

Pentru a fundamenta această evaluare, cititorului i-au fost prezentate conceptele de bază și paradigmele ce stau la baza construirii aplicațiilor pe

platforma NerWeaver a SAP, care prin intermediul server-ului de aplicații AS ABAP oferă toate uneltele necesare implementării aplicațiilor web, oferind soluții diverse și complexe pentru fiecare dintre tehnologiile sale. Uneltele standard puse la dispoziție de această platformă asigură integritatea datelor în cadrul unor infrastructuri IT eterogene, oferind totodată servicii pentru consolidarea și gestionarea centrală a datelor. Baza de date generată de modulul SAP CRM și utilizată ca suport pentru aplicațiile testate, poate fi legată și de baze de date non-SAP, sau poate fi extinsă cu o bază de date proprie.

Informațiile adunate permit să se verifice dacă cerințele unui client referitoare la un anumit gen de aplicații pot fi îndeplinite și dacă tehnologia aleasă este în principiu adecvată sau nu. În plus, devine clar care sunt avantajele și dezavantajele tehnologiei respective și care îi sunt punctele tari și slabe. Acest lucru face posibil estimarea efortului în legătură cu utilizarea tehnologiei respective.

Prezentarea rezultatelor, de la o imagine de ansamblu a tehnologiilor până la furnizarea detaliată a informațiilor la nivel arhitectural și de implementare a aplicațiilor, permite cititorilor familiarizarea cu aceste tehnologii. Prin aceasta se reduce posibilitatea alegerii unei tehnologii neadecvate, dar cunoscute, și se mărește posibilitatea alegerii unei tehnologii mai puțin cunoscute, dar adecvate.

Seturile de măsurători experimentale obținute prin testarea aplicațiilor cu ajutorul instrumentelor Business Transaction Analysis și HTTP Watch și prelucrarea lor statistică utilizând instrumentul EasyFit, au făcut posibilă determinarea funcțiilor de distribuție pentru fiecare dintre parametrii. Pe baza acestor funcții au fost generate noi date de test, necesare interpretării corecte a legăturilor dintre parametrii RT, CPUT, DBRT și HTTPWT. În baza acestor interpretări s-au stabilit indicatorii de importanță pentru fiecare dintre criterii, în cadrul algoritmilor de ierarhizare.

Legăturile între parametrii s-au evidențiat clar în analiza de regresie a datelor măsurate și generate, realizată utilizând instrumentul DataFit și prezentată în capitolul 5. Rezultatele obținute demonstrează că:

- pentru corelația simplă între RT și DBRT sau CPUT (pentru care se obțin coeficienți de corelație liniară simplă semnificativi), există o dependență $RT = f(DBRT)$ numai pentru tehnologiile WD ABAP și CRM WebClient UI; pentru tehnologia FPM, $RT = f(CPUT)$;
- corelația multiplă între RT și ceilalți parametrii măsurați arată că în toate tehnologiile RT depinde semnificativ de DBRT, dependența RT de HTTPWT este neglijabilă, iar pentru tehnologia FPM, RT depinde de CPUT dar într-o măsură mai mică decât dependența lui RT de DBRT.

Tot pe baza analizei de regresie din analiza matricelor coeficienților de corelație, rezultă că HTTPWT este singurul parametru independent. RT este un parametru dependent care depinde de DBRT sau de CPUT, iar între parametrii CPUT și DBRT nu există o dependență semnificativă. Toate rezultate obținute au fost luate în calcul la stabilirea ponderilor de importanță a criteriilor utilizate la implementarea algoritmilor pentru ierarhizarea tehnologiilor.

Algoritmii pentru ierarhizare AITOP și AIEL, au fost dezvoltați pe baza metodelor de analiză multicriterială: TOPSIS, ELECTRE și MUG. Metoda propusă de

autor pentru stabilirea ponderilor criteriilor a urmărit implementarea unui model ierarhic bazat pe metoda AHP.

Concluzia care poate fi desprinsă la finele lucrării este că metodologia propusă pentru evaluarea multicriterială a tehnologiilor ABAP UI de pe platforma de aplicații SAP NetWeaver, constituie o soluție sigură și performantă pentru facilitarea procesului de decizie și consilierea eficientă a celor interesați.

7.2. Contribuții personale

Pornind de la obiectivele acestei lucrări, sunt prezentate în continuare principalele contribuții originale ale autorului:

- Elaborarea unei sinteze critice asupra stadiului actual în domeniul tehnologiilor UI ale platformei de aplicații SAP a cărei infrastructură complexă facilitează dezvoltarea și implementarea aplicațiilor web business pentru stocarea și procesarea datelor.
- Analiza modelelor arhitecturale și de programare a tehnologiilor UI: Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI, recomandate de SAP pentru dezvoltatorii de aplicații web ABAP pe platforma Business Suite, cu evidențierea funcționalităților și capabilităților fiecăreia dintre tehnologii.
- Analiza comparativă a tehnologiilor Web Dynpro (ABAP și Java), precum și a conceptelor și tehnicilor care stau la baza dezvoltării aplicațiilor Web Dynpro ABAP la nivelul platformei SAP NetWeaver.
- Stabilirea la nivelul server-ului SAP AS ABAP a bazei de date și a tuturor elementelor de dezvoltare necesare implementării aplicațiilor web în toate cele trei tehnologii (elemente de dată, tabele, vederi, componente web, clase, etc.).
- Modelarea și implementarea aplicațiilor web în cele trei tehnologii: Web Dynpro ABAP, Floorplan Manager și CRM WebClient UI.
- Prelucrarea statistică a datelor experimentale pentru parametrii RT, CPUT, DBRT și HTTPWT în vederea:
 1. identificării valorilor suspecte și a eliminării lor din eşantioanele de date experimentale;
 2. determinării unor parametrii statistici cum ar fi: valoarea medie, abaterea standard, estimația punctuală, în scopul realizării algoritmilor pentru ierarhizare;
 3. verificării caracterului aleator -utilizând testul Young.
- Analiza seturilor de date obținute prin măsurători experimentale, în vederea identificării funcțiilor de distribuție pentru fiecare dintre parametrii RT, CPUT, DBRT și HTTPWT.

- Identificarea distribuției celei mai semnificative pentru datele prelucrate, pe baza rezultatelor testelor de concordanță Kolmogorov-Smirnov, Anderson-Darling și Hi-pătrat și a analizei acestora.
- Generarea pe baza funcțiilor de distribuție identificate, a unor seturi de date mai mari, necesare validării existenței legăturilor dintre parametrii, precum și a stabilirii relațiilor dintre aceștia
- Stabilirea dependențelor dintre parametrii măsurați și generați realizată prin:
 1. generarea matricelor coeficienților de corelație dintre parametrii obținuți în toate cele 3 tehnologii UI: WD ABAP, FPM și CRM WebClient UI;
 2. determinarea sub formă analitică și grafică a dependențelor liniare simple, duble sau triple numai dintre parametri pentru care coeficienții de corelație au valori semnificative (apropriate de 1).
- Stabilirea parametrilor dependenți și independenți pe baza analizei matricelor coeficienților de corelație.
- Stabilirea funcțiilor analitice și a graficelor de dependență dintre parametrii (analiza de regresie).
- Stabilirea seturilor de criterii, definite pe baza caracteristicilor arhitecturale și de performanță, pentru aplicațiile implementate în cele 3 tehnologii UI.
- Conceperea strategiei de dezvoltare a metodologiei de evaluare multicriterială și a algoritmilor pentru ierarhizare.
- Realizarea și implementarea unui model ierarhic MEIAHP, pentru determinarea ponderilor criteriilor necesare implementării algoritmilor de ierarhizare.
- Realizarea analizei senzitive a ierarhizării tehnologiilor, pentru modelul MEIAHP
- Dezvoltarea algoritmilor AITOP și AIEL pentru ierarhizarea tehnologiilor WD ABAP, FPM și CRM WebClient UI.
- Conceperea și construirea matricei de decizie a modelelor dezvoltate, pe baza valorilor estimațiilor punctuale ale parametrilor măsurați și a valorilor convertite a caracteristicilor arhitecturale.

7.3. Dezvoltări de viitor

În contextul continuării rezultatelor obținute în cadrul acestei teze, trebuie luat în considerare că deși tehnologiile UI analizate sunt actuale pentru dezvoltarea aplicațiilor web business în limbajul ABAP, există și alte tehnologii. Ce nu a fost abordat, sunt de exemplu tehnologiile WEB Dynpro Java și Adobes Flex, care prezintă o asemănare concepțională foarte puternică cu Web Dynpro ABAP sau Microsoft Silverlight. Pot fi examinate și tehnologiile specializate pentru dispozitivele mobile terminale, cum ar fi Smartphones sau tehnologiile disponibile în cloud.

În prezent, aplicațiile SAP Business Suite ale platformei SAP NetWeaver versiunea 7.4 sunt optimizate pentru platforma SAP HANA în cloud, prin intermediul serviciului SAP HANA Enterprise Cloud. SAP pune astfel împreună, într-un model simplu și puternic, facilitatea utilizării tehnologiei cloud cu inovația și performanța aduse de platforma analitică SAP HANA. Există deci, încă un șir de posibilități pentru a extinde conținutul temei, prin analiza acestor tehnologii suplimentare.

Utilizarea surselor VRML în aplicații web business dezvoltate cu ajutorul tehnologiilor ABAP UI reprezintă încă o cale pentru continuarea cercetărilor în viitor, lucrările autorului [166], [167] prezentând încercări în acest sens.

O altă direcție pentru continuarea cercetării acestei teze poate fi abordarea logicii fuzzy pentru interpretarea caracteristicilor lingvistice, bazate pe informații imprecise, prin care algoritmi de evaluare și ierarhizare pot fi extinși și optimizați.

Bibliografie

- [1] <http://www.sap.com/romania/index.epx>
- [2] SAP AG : *SAP Deutschland – Geschichte der SAP*,
<http://www.sap.com/germany/about/company/geschichte/index.epx>
- [3] Muir, N., Kimbell, I., *Discover SAP*, Second Edition, Editura SAP Press, Bonn, Boston, 2010, pp.30-50
- [4] Ofenloch, D., Schweiger, R., *Getting started with Web Dynpro ABAP*, Editura SAP Press, 2010, ISBN 978-1-59229-311-7
- [5] Whealy, C., *Inside Web Dynpro for Java*, Editura SAP Press, 2007, ISBN 978-3-89842-092-5
- [6] Heinemann, F., Rau, C., *SAP Web Application Server*, Editura SAP Press, 2003
- [7] SAP AG, *What's Inside Your SAP Web Application Server?*,
<https://www.sdn.sap.com/irj/servlet/prt/portal/prtroot/docs/library/uuid/13d02057-0601-0010-abba-be4a86d47f9e>
- [8] Heinemann, F., Rau, C., *Web Programming in ABAP with the SAP Web Application Server*, Galileo Press, Bonn, 2005, ISBN 3-89842-523-1
- [9] *Die SAP NetWeaver Komponenten*,
<http://www.oio.de/sap-netweaver-components.htm>
- [10] Muir, N., Kimbell, I. , *Discover SAP*, SAP Press, 2008, ISBN 978-1-59229-320-9
- [11] Heilig, L., Karch, St., Bottcher, O., Mutzig, C., Weber, J., Pfennig, R., *SAP NetWeaver: The Official Guide*, ISBN 978-1-59229-193-9, 2008
- [12] *Development Concepts and Architecture*, www.help.sap.com
- [13] SAP AG : *SAP INFO – Enterprise Service Architecture*,
http://www.sap.info/public/INT/int/glossary/de/glossarysearch/Word50633e881378e6d28_glossary/ESA#Word-50633e881378e6d28_glossary
- [14] http://www.uniorg.com/page/sap_beratung_netweaver_soa
- [15] http://www.purdue.edu/onepurdue/about/documents/SAPNetWeaverbrochure_000.pdf
- [16] Kessler, K., Tillert, P., Dobrikov, P., *Java-Programmierung mit dem SAP Web Application Server*. 1 Auflage, Bonn, Galileo Press GmbH, 2005, ISBN 3-89842-317-4
- [17] Woods, D., Word, J., *SAP NetWeaver for Dummies*, Wiley Publishing, USA, Indianapolis (Indiana) , 2004, ISBN-13: 978-0-7645-6883-3
- [18] McDonald, K., Wilmsmeier, A., Dixon, D. C., Inmon, W. H., *Mastering the SAP Business Information Warehouse: Leveraging the Business Intelligence Capabilities of SAP NetWeaver, Cap.2*, Second Edition, Editura John Wiley & Sons, 2006, ISBN-13: 978-0-7645-9637-7
- [19] Woods, D., Word, J., *SAP NetWeaver For Dummies*, Editura John Wiley & Sons, 2011, ISBN 0-7645-6883-3
- [20] Keller, H., Thümmel, W. H., *Official ABAP Programming Guidelines*, 2009, SAP Press, ISBN 978-1-59229-290-5
- [21] Kessler, K., Tillert, P., Dobrikov, P., *Java Programming with the SAP Web Application Server*, SAP Press, 2005, ISBN-13: 978-1592290208
- [22] Gupta, T., *ABAP Data Dictionary*, SAP Press, 2011, ISBN: 978-1-59229-379-7
- [23] Wood, J., *ABAP Cookbook*, SAP Press, 2010, ISBN: 978-1-59229-326-1

- [24] Gahm, H., *ABAP Performance Tuning*, SAP Press, 2009, ISBN: 978-1-59229-289-9
- [25] Fuchs, K., „SAP NetWeaver in der Praxis – Wie gut bewährt sich der Technologie-Stack in der praktischen Arbeit?“, Informatik Spektrum. Heidelberg Springer Verlag, 2007, Band 30, Heft 6, pag. 428-433. – DOI 10.1007/s00287-007-0191-3, ISSN: 0170-6012
- [26] Wegelin, M., Englbrecht M., *SAP Interface Programming*, SAP Press, 2009, ISBN: 1-59229-318-2
- [27] www.help.sap.com
- [28] Keller, H., Kruger S., *Abap Objects – Abap Programming in SAP NetWeaver. 2. Auflage*, Bonn, Galileo Press, 2007. – ISBN 1-5922-9079-5
- [29] Site SAP SDN URL: <http://www.sdn.sap.com/irj/sdn>
- [30] Kangas, M., *Introduction to the SAP Web Application Server*, online SAP Publication, 2005,
<http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/e1515a37-0301-0010-b2bb-99780cb4cafa?QuickLink=index&overridelayout=true>
- [31] Heinemann, F., Rau, C., *Web Programming in ABAP with the SAP Web Application Server*, SAP Press, 2005, ISBN 978-1-59229-060-4
- [32] Wiegenstein, A., Schumacher, M., Schinzel, S., Weidemann F., *Sichere ABAP-Programmierung*, SAP Press, 2009, pag.54-62, ISBN 978-3-8362-1357-8
- [33] Nocon, O., Zirkel V., *Building Web Infrastructures with SAP NetWeaver Application Server*, SAP PRESS Essentials 33, ISBN 978-1-59229-161-8
- [34] *NET200 SAP Web Application: Developing BSP Applications*,
<http://sapient.files.wordpress.com/2007/08/bsp-training-i.pdf>,
- [35] *SAP Guidelines for Best-Built Applications That Integrate with SAP Business Suite*, The SAP Architectural Community 2011,
www.sdn.sap.com – SAP Developer Network
- [36] *Getting Started with Web Dynpro*, <http://scn.sap.com/docs/DOC-18499>
- [37] *UI Technologies in SAP NetWeaver*, <http://help.sap.com>
- [38] *Fundamentals of Web Dynpro For ABAP*. SAP AG.,
https://websmp106.sap-ag.de/~form/sapnet?_SHORTKEY=01100035870000441461&_SCENARIO=01100035870000000112&_OBJECT=011000358700006256352006E
- [39] Szücs, T., *Dynamic Programming in Web Dynpro ABAP*,
<http://scn.sap.com>
- [40] Osaci, M., **Berdie, A. D.**, Szemciuk, I., „Studies on the Web Dynpro ABAP Dynamic Programming of the Graphical Interface versus the Programming in Design Mode“, 3rd World Conference on Information Technology, 2012, Barcelona, Spania
- [41] Barzewski, A., Bönner, C. , & all, *Java Programming with SAP NetWeaver*, SAP Press, 2008, ISBN: 978-1-59229-181-6
- [42] *The Model View Controller Paradigm*, <http://help.sap.de>
- [43] Cristea, A.D., **Berdie, A.D.**, Osaci, M., „Parallel between two SAP Frameworks that use MVC paradigm“, IEEE INTERNATIONAL JOINT CONFERENCES IN COMPUTATIONAL CYBERNETICS AND TECHNICAL INFORMATICS (ICCC-CONTI), 2010, Timișoara, ISBN 978-1-4244-7432-5, p. 397-400, indexată B+
- [44] Ofenloch, D., Schwaiger, R., *Getting Started with Web Dynpro ABAP*, Publishing House SAP, Galileo Press, Bonn, Germany, 2009

- [45] Meiners, J., Nüßer, W., *SAP Interface Programming*, SAP Press, 2004, ISBN-13: 978-1592290345
- [46] *Curs ABAP Web Dynpro*, <http://www.slideshare.net/jky1234/abap-web-dynpro>
- [47] *An introduction to Web Dynpro – Part 2*, <http://www.itpsap.com/blog/2012/11/25/an-introduction-to-web-dynpro-part-2/>
- [48] Cristea, A.D., **Berdie, A.D.**, Osaci, M., „Application Componentization and component usages with Web Dynpro ABAP and Web Dynpro Java”, BULETINUL STIINTIFIC al Universităţii “Politehnica” din Timișoara, ROMANIA, Seria AUTOMATICA si CALCULATOARE SCIENTIFIC BULLETIN of “Politehnica” University of Timișoara, ROMANIA, Transactions on AUTOMATIC CONTROL and COMPUTER SCIENCE, Vol. 55 (69), Fasc. 1, 2010, ISSN 1224-600X
- [49] Cristea, A.D., **Berdie, A.D.**, Osaci, M., „The use and importance of Hook Methods in Web Dynpro ABAP and Web Dynpro Java”, Buletinul Șt.al UPT, B+, Vol. 54 (68), Fasc. 3, 2009
- [50] Osaci, M., **Berdie, A.D.**, Cristea, A.D., Molnar, E.D., „Modalities to Implement the Multilinguality in Web DYNPRO ABAP”, Analele “Dunărea de Jos” Universitatea Galați, 2010 indexată B+
- [51] Cristea, A.D., **Berdie, A.D.**, Osaci, M., Chirtoc, D., „The advantages of using Mind Map for learning web Dynpro”, Computer Application in Engineering Education, vol.19, 2011, pg.201-207, Jurnal ISI, Factor impact 0,33,
- [52] Wood, J., Parvaze, S., *Web Dynpro ABAP: The Comprehensive Guide*, SAP Press, 2012, ISBN: 978-1-59229-416-9
- [53] Gellert, U., Cristea, A.D., *Web Dynpro ABAP for Practitioners*, Editura Springer, 2010, ISBN-13: 978-3642113840
- [54] *SAP Library*, 2010. SAP Library: Creating Web Applications with BSPs, http://help.sap.com/saphelp_470/helpdata/en/c8/101c3a1cf1c54be1000000a114084/content.htm
- [55] *Web Dynpro Architecture*, http://help.sap.com/saphelp_nw04/helpdata/en/a5/1a1e3e7181b60ae1000000a114084/content.htm
- [56] *SAP NetWeaver Application Concepts*, Part 1, Spring 2010, http://einstein.etsu.edu/~pittares/podcast/slides/Spring2010/02_SAP%20NetWeaver%20Architecture,%20part%201.pdf
- [57] *Web Dynpro and the Model View Controller Design*, http://help.sap.com/saphelp_nwce72/helpdata/en/47/b467c028912972e10000000a42189b/content.htm
- [58] Varadadesigan, V., *Web Dynpro for ABAP*, SAP AG, 2006, <http://www.ayantech.com/pdf/WebDynproForABAP.pdf>
- [59] Hoffmann, U. , *Praxisbuch WebDynpro for ABAP*. 1. Auflage, Bonn, Galileo Press 2006. – ISBN 3-89842-690-4
- [60] **Berdie, A.D.**, Osaci, M., Prostean, G., Cristea, A.D., „Web Programming features on integrated system SAP”, Conferința SACI 2011, B+
- [61] Osaci, M., **Berdie, A. D.**, Lup, A. E., „Educational soft presenting particularities through special functions of ABAP List Viewer in Web Dynpro technology”, 4th World Conference on Educational Sciences (WCES), 2012, ELSEVIER SCIENCE BV, ISSN 1877-0428, WOS:000314465903116

- [62] Bourque, P., Dupuis, R., Abran, A., Moore, J. W., Tripp, L., Wolff, S., *The Journal of Systems and Software*, 62, 2002, pp.59–70
- [63] Levine, L., „Business Agility and Information Technology Diffusion”, IFIP International Federation for Information Processing, 2005, Volume 180/2005, pp.353-365
- [64] *Web Dynpro for ABAP*, SAP Online Help, 2006, <http://www.sdn.sap.com>
- [65] Osaci, M., **Berdie, A. D.**, Prostean, G., Pop, A. A., „Faceless Componentization Versus Componentization with Assistance Class in Web Dynpro ABAP”, Conferința SACI 2012, indexată B+,
- [66] NET311, *Advanced ABAP Web Dynpro, Unit 3, Enhancements*, curs SAP, versiune 2006
- [67] *Floorplan Manager for Web Dynpro ABAP - Developer's Guide*, 2011, <http://www.sdn.sap.com>
- [68] SAP homepage. URL: www.sap.com
- [69] *SAP NetWeaver. SAP Schulungshandbuch: Net313, Floor Plan Manager for Web Dynpro ABAP*. SAP AG. Documentație studiată în cadrul stagiului de practică la firma SP Lion AG Heddeshheim, martie 2011
- [70] *Floorplan Manager for ABAP - Developer's Guide*, Release 701, 2008, <http://www.sdn.sap.com>
- [71] Saundattikar, R., *SAP Wiki: FPM Phase Model*, SAP AG, 2009, <https://wiki.wdf.sap.corp/wiki/download/attachments/626624202/FPM+Phase+Model.pdf?version=1>
- [72] SAP AG. *Floorplan Manager for ABAP Developer Guide*, <http://scn.sap.com/docs/DOC-30668>
- [73] *Floorplan Manager for Web Dynpro ABAP*, <http://help.sap.com>
- [74] Bender, J., Hoeg, S. & all, *FPM Cookbook, Version 2.1*, 2008, Documentație studiată în cadrul stagiului de practică la compania SP Lion AG Heddeshheim, Germania, martie 2011
- [75] Saundattikar, R., *SAP Wiki: GUIBB*, SAP AG, 2009, <https://wiki.wdf.sap.corp/wiki/display/FPM/GUIBB>
- [76] Tzanko, St., Sezikeye, A., Mal S., *SAP WebClient, A Comprehensive Guide for Developers*, SAP Press, 2011, ISBN 978-1-59229-389-6
- [77] WebClient UI Framework, <http://help.sap.com>
- [78] *SAP Web Application Server: Developing BSP Application*, SAP AG, 2006, Literatura de specialitate studiată în cadrul stagiului de practică ERASMUS la compania SP Lion AG Heddeshheim, Germania, martie 2011
- [79] *CRM WebClient User Interface Architecture*, SAP CRM 2007
- [80] *CRM WebClient UI 2007*, <http://www.slideshare.net/FAISALSPARTA/sap-crm-ui-cookbook>
- [81] *Framework Architecture*, <http://help.sap.com>
- [82] *SAP CRM. SAP Schulungshandbuch: CR580 - CRM User Interface*. SAP AG, 2010, Literatura de specialitate studiată în cadrul stagiului de practică ERASMUS la compania SP Lion AG Heddeshheim, Germania, februarie 2011
- [83] Fücksle, M., Zierke, M.E., *SAP CRM Web Client— Customizing and Development*, SAP Press, 2009, ISBN 978-1-59229-297-4
- [84] Karuppanan, A.P., *WebDynpro ABAP in CRM 2007: Dynamic Context Nodes and Component Usages*, <http://scn.sap.com/docs/DOC-2329>
- [85] *SAP Product Road Map, SAP User Interface*, Ian.2012, www.sdn.sap.com – SAP Developer Network

- [86] **Berdie, A.D.**, Osaci, M., Muscalagiu, I., Proștean, G., „A case-study about a web business application implemented in different SAP UI technologies”, Lucrare prezentată la conferința SACI 2012, indexată B+
- [87] Falk M., *Product Strategy and Roadmaps*, SAP User Interfaces, 2011, <http://sapportalexpert.com/wp-content/uploads/2011/08/SAP-UI-Roadmap-2011.pdf>
- [88] *SAP Library*, 2010, SAP Library. ABAP Workbench. http://help.sap.com/saphelp_glossary/en/35/26b1a7afab52b9e1000009b38f974/content.htm
- [89] **Berdie, A.D.**, Osaci, M., Proștean, G., Haipl, R., „Analysis and design in databases modeling using SAP integrated systems”, Lucrare prezentată la conferința SACI 2009
- [90] *Floorplan for the Overview Page (OVP)*, <http://help.sap.com>
- [91] Osaci, M., **Berdie, A.D.**, Hammes, A.D., „Studies on Efficiency of the Data Access by ABAP Programming”, 4th World Conference on Educational Sciences (SOFA-2012), august 2012, Szeged, Hungary, SPRINGER-VERLAG BERLIN, ISBN 978-3-642-33940-0, ISSN 2194-5357, WOS:000314077300063
- [92] Osaci, M., **Berdie, A.D.**, Cristea, A.D., „Studies about optimizing ABAP Codes for access to data from persistence level”, Lucrare prezentată la conferința 2nd World Conference on Innovation and Computer Sciences INSODE, AWERProcedia Information Technology and Computer, mai 2012, Izmir, Turcia, ISSN 2147-5369, p. 67-72
- [93] Osaci, M., **Berdie, A.D.**, Cristea, A.D., „Control Tools for Optimizing ABAP Codes”, *Acta Technica Corviniensis –Bulletin of Engineering*, 5 (12), 2012, p.107-110
- [94] **Berdie, A.D.**, Osaci, M., „Comparing the web business applications using the SAP Business Transaction Analysis”, Lucrare prezentată la conferința World Conference on Business, Economics and Management WCBEM, mai 2012, Antalya, Turcia, indexată ISI
- [95] Schneider, T., *SAP Performance Optimization Guide*, Galileo Press, 2008, ISBN:978-1-59229-202-8, pp.163
- [96] Schwarz, W., *Performance Problems in ABAP Programs: How to Find Them*, SAP Professional Journal May/June 2003
- [97] <http://help.httpwatch.com/#introduction.html>
- [98] **Berdie, A.D.**, Osaci, M., Opreșă, G., „Study about the estimation of the system parameters at the implementation of SAP web applications”, 3rd World Conference on Information Technology, nov. 2012, Barcelona, Spania.
- [99] Liteanu, C., Rîcă, I., *Teoria și metodologia statistică a analizei urmelor*, Ed. Scrisul Românesc, Craiova, 1979
- [100] Georgescu, M., Zaharia, R., *Introducere în tehnica proiectării asistate de experiment a construcțiilor metalice*, Editura Orizonturi Universitare, Timișoara, 1999, pag.67-68
- [101] Julean, I., Holban, Șt., *Incertitudini la prelucrarea datelor experimentale și în exprimarea rezultatelor*, Ed. Politehnica, Timișoara, 2009
- [102] Vasilescu, N., Costescu, M.R., Ionascu, C., Babucea, G., Tomita, V., Stuparu, D., *Statistica*, Ed. Universitaria, Craiova, 2003.
- [103] Vertan, C., Gavăț, I., Stoian, R., *Variabile și procese aleatoare:Principii și aplicații*, 1999, pag.15

- [104] Kolmogorov, A., *Confidence Limits for an Unknown Distribution Function*. *The Annals of Mathematical Statistics*, 1941, 12(4), 461-463.
- [105] http://www.mathwave.com/help/easyfit/html/analyses/goodness_of_fit/kolmogorov-smirnov.html
- [106] Smirnov, N. V., „Table for estimating the goodness of fit of empirical distributions”, *The Annals of Mathematical Statistics*, 1948, 19(2), 279-281.
- [107] *Engineering Statistics Handbook*,
<http://www.itl.nist.gov/div898/handbook/eda/section3/eda35e.htm>
- [108] Anderson, T.W., Darling, D.A., „A Test of Goodness of Fit”, *Journal of American Statistics Association*, (1954), pp. 765-767
- [109] <http://www.mathwave.com/help/easyfit/index.html>
- [110] Mihoc, I., Fătu, C. I., *Calculul probabilităților și statistică matematică*, Transilvania Press, 2003, pag.551-553
- [111] **Berdie, A.D.**, Osaci, M., Lemle, L.D., „Comparative statistical study of some SAP UI technologies”, *Lucrare prezentată la conferința ICNAAM, 2011 Grecia*, indexată ISI
- [112] STATS @ MTSU website, Middle Tennessee State University, *Regression Tutorial Project, Multiple Correlation Coefficient*,
<http://mtweb.mtsu.edu/stats/regression/level3/multicorrel/multicorrcoef.htm>
- [113] Holban, Șt., *Data Mining. Modele corelaționale*, cap. 2, curs UPTimișoara, (format electronic)
- [114] Dobre, I., Badescu A. V., Pauna, L., *Teoria deciziei*, Ed. ASE, (2007), București
- [115] Leon, F., *Managementul proiectelor software*,
http://eureka.cs.tuiasi.ro/~fleon/curs_mps.htm
- [116] Filip, F.G., *Sisteme suport pentru decizii*, Ed. Tehnică, ed II completată și revizuită, 2007, București.
- [117] Roman, M., *Analiza multicriterială - manual*, 2012
- [118] *Teoria deciziilor curs*, 2012 http://www.cs.ubbcluj.ro/~per/Dss/Dss_5.pdf
- [119] *Adapter from Multi-criteria analysis: a manual*, January 2009, Department for Communities and Local Government: London, ISBN: 978-1-4098-1023-0,
https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/7612/1132618.pdf
- [120] Zaer, M. S., & all, “Application of multi criteria decision making technique to evaluation suppliers in supply chain management”, *African Journal of Mathematics and Computer Science Research*, Vol. 4 (3), pp. 100-106, March, 2011
- [121] Triantaphyllou, E. Stuart H. Mann, „Using the Analytic Hierarchy Process for Decision Making in Engineering Applications: Some Challenges”, *Inter'l Journal of Industrial Engineering: Applications and Practice*, Vol. 2, No. 1, pp. 35-44, 1995
- [122] Filho, Adiel T. de Almeida, „Multicriteria decision making on selection of decision analysis software”, *Journal of Academy of Business and Economics* Publisher: International Academy of Business and Economics, 2005, Volume 5, Issue 3
- [123] Edwards, W., Barton, F.H., "Smarts and Smarter: Improved Simple Methods for Multi Attribute Utility Measurement", *Organizational Behavior and Human Decision Processes*, Vol. 60, 1994, 306-325

- [124] Giove, S., „Interval TOPSIS for Multicriteria Decision Making”, 2002, Neural Nets, Springer, Berlin, Heidelberg, 978-3-540-44265-3, pp 56-63
- [125] Hunjak, T., „Mathematical foundations of the methods for multicriterial decision making”, Mathematical Communications 2(1997), 161-169
- [126] Rogers, M.G, Bruen, M., Maystre, L.Y., *Chapter 3: The ELECTRE Methodology, ELECTRE and Decision Support*, Kluwer Academic Publishers, Boston, MA, U.S.A, 1999
- [127] Brans, J.P., Mareschal, B., *Promethee Methods, Multiple Criteria Decision Analysis: State of the Art Surveys*, Springer New York, 2005, ISBN 978-0-387-23067-2, pp 163-186
- [128] Eppen, G.D., *Quantitative concepts for management – Decision making without algorithms*, Prentice-Hall, SUA, p. 7, (1985)
- [129] Bontaș, D., *New Models of the Decision Making Process. Economy Transdisciplinarity Cognition*, George Bacovia University Printing House, Bacău, Vol. XII, nr. 1, 2009, ISSN 1454- 5675, pp. 10 – 14
- [130] Zamfir, C., *Incertitudinea. O perspectivă psihosociologică*, București, Ed. Economică, (1990/1995), p. 15.
- [131] Șuteu, S., *Modele de fundamentare a deciziilor*, Suport curs
- [132] Fulop, J., *Introduction to Decision Making Methods*, <http://academic.evergreen.edu/projects/bdei/documents/decisionmakingmethods.pdf>
- [133] Saaty, T. L., *The analytic hierarchy process*, New York, McGraw-Hill, (1980)
- [134] Choi H., „Analytic hierarchy process - It can work for group decision support systems”, Computers & industrial engineering, vol. 27, nr. 1-4, pp. 167-171, 1994
- [135] Jantea, L., Sacal, B., Bădescu, A.V., „A fuzzy approach of multi-criteria decisions”, http://www.asecib.ase.ro/simpozion/2009/full_papers/pdf/58_Jantea_ro.pdf
- [136] Ciocoiu, N., *Modelarea proceselor decizionale multicriteriale și de grup*, Curs, <http://www.scribd.com/doc/148025632/Curs7-8-Modelare-Economica-2013-Nadia-Ciocoiu>
- [137] Saaty, T., „An exposition of the AHP. In reply to the paper Remarks on the analytic hierarchy process”, Management science, vol. 36, nr. 3, pp. 259-268, 1990
- [138] Akarte, M. M., „Web based casting supplier evaluation using analytical hierarchy process”, The journal of the Operational Research Society, vol. 52, nr. 5, pp. 511-522, 2001
- [139] Al Harbi, K. M. A., „Application of the AHP in project management”, International Journal of Project Management, vol. 19, no. 1, pp. 19-27, Jan.2001
- [140] Wei, C. C., Chien, C. F., Wang, M. J., „An AHP-based approach to ERP system selection”, International Journal of Production Economics, vol. 96, nr. 1, pp. 47-62, 2005
- [141] Yoon, K., Hwang, C.L., *Multiple Attribute Decision Making, Methods and Application*, Berlin, Heidelberg, Springer, 1981
- [142] Wang, R. T., *Performance Evaluation Method - Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)*, www.researcher.nsc.gov.tw/public/caroljoe/Data/O2182133671.ppt

- [143] Bulgurcu-Kiran, B., „Application of TOPSIS Technique for Financial Performance Evaluation of Technology Firms in Istanbul Stock Exchange Market”, WCBEM 2012, Procedia Social and Behavioral Science, 62, pp.1033-1040
- [144] Velasquez, M., Hester, P. T., „An Analysis of Multi-Criteria Decision Making Methods”, International Journal of Operations Research, Vol. 10, Nr.2, pp.56-66, 2013
- [145] Roy, B., „Classement et choix en presence de points de vue multiples: La method ELECTRE”, R.I.R.O 8, 1968, pp. 57 – 75.
- [146] Pang, J., Zhang, G., Chen, G., *ELECTRE I Decision Model of Reliability Design Scheme for Computer Numerical Control Machine*
- [147] Meineke, L., *Wirtschaftlichkeitsanalyse – Basis zur Durchführung von IT-Projekten. IT Management.* Sauerlach IT-Verlag für Informationstechnik 2003, Heft 12, p. 12-17
- [148] Aer, St., Schonherr, M., *Evaluating Enterprise Architectures – A Scenario-Based Evaluation of Integration Architectures. Trends in Enterprise Application Architecture: VLDB Workshop, TEAA 2005*, Editura Springer, Berlin, Heidelberg, New York, ISBN 3-540-32734
- [149] Gartner Group : *Decision Tools for Cost Management*, http://www.gartner.com/4_decision_tools/measurement/decision_tools/tco/tco.html
- [150] Albin, St. T., *The Art of Software Architecture - Design Methods and Techniques.* Indianapolis, Indiana Wiley Publishing, Inc.2003 - ISBN 0-471-22886-9
- [151] SAP NetWeaver Help, *Model View Controller (MVC)*, http://help.sap.com/saphelp_nw70/helpdata/en/0f/ab3a3c9ca75402e1000000a114084/content.htm
- [152] *Implementing a Custom Model*, <http://www.veui5infra>
- [153] UI Governance Group, *Dynpro-based Frontend Technologies*, www.wiki.wdf.sap.corp/wiki/display/UIG/UI+Technology+Dynprobased+front+end+technologies
- [154] Booch, G., *Object Oriented Analysis and Design With Applications*, Amsterdam, Addison-Wesley Longman, 1993, ISBN 0-8053-5340-2
- [155] McKellar, B., Jung T., *Advanced BSP Programming*, Editura SAP Press, 2006, ISBN 978-1592290499
- [156] *SAP NetWeaver Help, Enhancement Technologies*, http://help.sap.com/saphelp_nwpi71/helpdata/EN/70/63da4023a28631e1000000a1550b0/content.htm
- [157] Gorton, I., *Essential Software Architecture*, Editura Springer. 2006.
- [158] **Berdie, A.D.**, Osaci, M., Lemle, D.L., „An AHP evaluation between WD ABAP, FPM and CRM WebClient UI technologies”, *Lucre prezentată la conferința 2nd World Conference on Innovation and Computer Sciences INSODE*, mai 2012, Izmir, Turcia, în curs de indexare
- [159] **Berdie, A.D.**, Osaci, M., Budișan, N., „A Multicriteria Approach in the Selection of a SAP UI Technology”, *Lucre prezentată la conferința SOFA 2012*, indexată ISI
- [160] Rao, R. V., Davim, J. P., *Int J Adv Manuf Technol*, (2008), 35, p.751–760
- [161] Milani, A. S., Shanian, A., Madoliat, R., Nemes, J. A., „The effect of normalization norms in multiple attribute decision making models: a case study in gear material selection”, *Industrial Applications*, vol. 29, p. 312, 2005.

-
- [162] Shih, H. S., Shyr, H. J., Lee, E. S., „An extension of TOPSIS for group decision making”, *Mathematical and Computer Modelling*, vol. 45, no. 7-8, pp. 801-813, Apr.2007.
- [163] Pomerol, J., Barba-Romero, S., *Multicriterion Decision in Management: Principles and Practice*, Kluwer Academic Publishers, 2000.
- [164] *Metode de fundamentare a deciziilor multicriteriale*,
<http://www.scribd.com/doc/44225467/Probleme-Management-Rezolvate>
- [165] Proștean, G., *Bazele managementului*,
www.mpt.upt.ro/doc/curs/gp/Bazele.../Elaborarea_deciziilor_cap7.pdf
- [166] Osaci M., **Berdie A.D.**, Proștean G., Cristea A.D., *Virtual Programming in SAP technologies*, *Scientific Bulletin of the „Politehnica” University of Timișoara, România, Transactions on Automatic Control and Computer Control*, **56**(70), no. 3, 2011, p.105-108.
- [167] Osaci M., **Berdie A.D.**, Proștean G., Belea D., Cristea A.D., *VRML (Virtual Reality Modeling Language) Programming in SAP technologies*, 6th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI), May 19–21, 2011, Timișoara, Romania, 2011, p.223-226

Anexa 1. Codul sursă al metodei SEARCH_TEXT din ComponentController

```
METHOD search_text .
DATA:
node_search_text  TYPE REF TO if_wd_context_node,
elem_search_text  TYPE REF TO if_wd_context_element,
stru_search_text  TYPE if_componentcontroller=>element_search_text ,
item_text1        LIKE stru_search_text-text1.
*navigate from <CONTEXT> to<SEARCH_TEXT> via lead selection
node_search_text = wd_context->get_child_node( name =
if_componentcontroller=>wdctx_search_text ).
*   get element via lead selection
elem_search_text = node_search_text->get_element( ).

*   get single attribute
elem_search_text->get_attribute(
EXPORTING
name = 'TEXT1'
IMPORTING
value = item_text1 ).
DATA:
node_display TYPE REF TO if_wd_context_node,
elem_display TYPE REF TO if_wd_context_element,
stru_display TYPE if_componentcontroller=>element_display
*   navigate from <CONTEXT> to <DISPLAY> via lead selection
node_display = wd_context->get_child_node( name =
if_componentcontroller=>wdctx_display ).
DATA it_display LIKE TABLE OF stru_display.
REPLACE ALL OCCURRENCES OF '*' IN item_text1
WITH '%' IN CHARACTER MODE.

CONCATENATE '%' item_text1 '%' INTO item_text1.
SELECT * FROM yv_buadr INTO CORRESPONDING FIELDS OF
TABLE it_display
WHERE partner LIKE item_text1
OR name_org1 LIKE item_text1
OR country LIKE item_text1
OR city1 LIKE item_text1
OR street LIKE item_text1
OR region LIKE item_text1
OR post_code1 LIKE item_text1.
node_display->bind_table( new_items = it_display
set_initial_elements = abap_true ).
wd_this->select_rec( ).
ENDMETHOD.
```


Anexa 2. Codurile sursă ale metodelor din aplicația implementată în tehnologia FPM

a)

```
METHOD if_fpm_guihb_search~get_definition.
eo_field_catalog_attr ?=
cl_abap_structdescr=>describe_by_name(,ZBUPASEARCH').
eo_field_catalog_result ?=
cl_abap_structdescr=>describe_by_name(,ZTBUPARESLT').
ENDMETHOD.
```

b)

```
METHOD if_fpm_guihb_search~process_event.
DATA:
  ls_sel_option TYPE rsdsselopt,
  lt_sel_option TYPE TABLE OF rsdsselopt.
FIELD-SYMBOLS:
  <criteria> TYPE fpmgb_s_search_criteria.
IF io_event->mv_event_id NE if_fpm_guihb_search=>fpm_execute_search
RETURN.
ENDIF.
LOOP AT it_fpm_search_criteria ASSIGNING <criteria>.
TRY.
ls_sel_option = cl_fpm_guihb_search_conversion=>to_abap_select_option(is_fpm_
search_row = <criteria> ).
CATCH cx_fpmgb.
ENDTRY.
IF <criteria>-search_attribute = 'COUNTRY'.
INSERT ls_sel_option INTO TABLE lt_sel_option.
ENDIF.
ENDLOOP.
SELECT
  b~partner AS partner
  b~type AS type
  b~name_org1 AS mc_name1
  b~name_org2 AS mc_name2
  a~city1 AS mc_city1
  a~post_code1 AS post_code1
  a~street AS mc_street
  a~house_num1 AS house_num1
  a~country AS country
FROM but000 AS b
INNER JOIN but020 AS x ON b~partner = x~partner
INNER JOIN adrc AS a ON x~addrnumber = a~addrnumber
INTO CORRESPONDING FIELDS OF TABLE mt_data
UP TO iv_max_num_results ROWS
WHERE a~country IN lt_sel_option.
ENDMETHOD.
```

c) `METHOD` if_fpm_guihb_search~get_data.
et_result_list = mt_data.
`ENDMETHOD`.

Anexa 3 Valorile parametrilor obținuți prin interogarea aplicației Web Dynpro ABAP pentru cele 3 cazuri

Varianta 1) Afișarea primelor 100 rezultate

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch Time (ms)
1	242	242	130	116	8.345	43,1	1.077
2	158	158	140	30	8.345	43,1	811
3	160	160	130	31	8.345	43,1	843
4	153	153	140	28	8.345	43,1	874
5	161	161	140	29	8.345	43,1	904
6	152	152	130	28	8.345	43,1	999
7	157	157	130	28	8.345	43,1	1.091
8	156	156	120	27	8.345	43,1	889
9	153	153	140	27	8.345	43,1	859
10	153	153	140	27	8.345	43,1	858
11	155	155	140	27	8.345	43,1	795
12	152	152	140	27	8.345	43,1	889

Varianta 2) Afișarea tuturor rezultatelor, tabelate pe câte 9 rânduri

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch Time (ms)
1	425	425	180	291	8.345	728	1.138
2	391	391	180	260	8.345	728	1.156
3	387	387	170	257	8.345	728	1.264
4	389	389	180	257	8.345	728	1.139
5	389	389	180	257	8.345	728	1.185
6	387	387	170	255	8.345	728	1.061
7	404	404	170	272	8.345	728	1.186
8	1.986	721	180	589	8.345	728	2.606
9	390	390	180	259	8.345	728	1.169
10	391	391	190	258	8.345	728	1.031
11	395	395	180	263	8.345	728	1.202
12	401	401	190	271	8.345	728	1.124

Varianta 3) Afișarea tuturor rezultatelor, tabelate pe câte 100 rânduri

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch Time (ms)
1	951	951	730	259	8.345	728	3.245
2	872	872	670	256	8.345	728	3.135
3	824	824	600	267	8.345	728	3.026
4	769	769	560	258	8.345	728	3.026
5	839	839	620	266	8.345	728	4.508
6	803	803	590	257	8.345	728	2.980
7	786	786	570	267	8.345	728	4.602
8	829	829	610	266	8.345	728	2.948
9	898	898	680	266	8.345	728	3.183
10	898	898	690	266	8.345	728	3.089
11	804	804	590	262	8.345	728	3.619
12	870	870	660	267	8.345	728	3.152

Anexa 4 Valorile parametrilor obținuți prin interogarea aplicației Floorplan Manager pentru cele 3 cazuri

Varianta 1) Afișarea primelor 100 rezultate

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch Time (ms)
1	531	530	270	325	8.345	864	1.701
2	510	510	260	302	8.345	864	1.653
3	507	507	260	299	8.345	864	1.638
4	510	510	270	301	8.345	864	1.653
5	507	507	270	299	8.345	864	1.591
6	505	505	270	298	8.345	864	1.482
7	502	502	270	296	8.345	864	1.686
8	504	504	270	296	8.345	864	1.794
9	514	514	260	306	8.345	864	1.701
10	508	508	270	302	8.345	864	1.530
11	516	516	270	309	8.345	864	1.560
12	503	503	270	297	8.345	864	1.576

Varianta 2) Afișarea tuturor rezultatelor, tabelate pe câte 9 rânduri

Index	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch Time (ms)
1	513	513	270	306	12.437	864	1.576
2	477	477	240	298	12.437	864	1.716
3	510	510	280	300	12.437	864	1.591
4	478	478	240	301	12.437	864	1.717
5	517	517	270	303	12.437	864	1.699
6	513	513	280	301	12.437	864	1.684
7	515	515	280	302	12.437	864	1.654
8	479	479	230	302	12.437	864	1.748
9	508	508	270	297	12.437	864	1.528
10	516	516	270	301	12.437	864	1.592
11	517	517	270	302	12.437	864	1.622
12	515	515	270	303	12.437	864	1.560

Varianta 3) Afișarea tuturor rezultatelor, tabelate pe câte 100 rânduri

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch Time (ms)
1	845	845	600	299	12.437	864	3.558
2	851	851	610	298	12.437	864	3.433
3	1.123	1.123	890	297	12.437	864	3.713
4	878	878	630	299	12.437	864	3.432
5	913	912	670	304	12.437	864	3.479
6	894	894	650	299	12.437	864	3.448
7	837	837	600	294	12.437	864	3.447
8	867	867	640	299	12.437	864	3.291
9	890	890	640	299	12.437	864	3.495
10	914	914	670	301	12.437	864	3.462
11	979	979	750	298	12.437	864	3.697
12	866	866	620	307	12.437	864	3.449

Anexa 5 Valorile parametrilor obținuți prin interogarea aplicației CRM WebClient UI pentru cele 3 cazuri

Varianta 1) Afișarea primelor 100 rezultate

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch (ms)
1	499	499	422	91	37.006	1.113	2.334
2	553	553	484	136	37.006	1.144	4.803
3	554	554	422	138	37.006	1.139	5.044
4	634	634	500	180	37.006	1.139	2.404
5	549	549	500	137	37.006	1.139	2.355
6	545	545	438	130	37.006	1.139	2.373
7	556	556	438	135	37.006	1.139	2.403
8	535	535	469	123	37.006	1.139	4.883
9	561	561	391	140	37.006	1.139	2.376
10	575	575	484	150	37.006	1.139	2.412
11	543	543	438	130	37.006	1.139	2.446
12	555	555	500	136	37.006	1.139	2.419

Varianta 2) Afișarea tuturor rezultatelor, tabelate pe câte 9 rânduri

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch (ms)
1	2.612	2.612	1.859	1.033	65.648	11.872	4.796
2	2.678	2.678	2.063	1.070	65.648	11.720	2.407
3	2.555	2.555	1.797	987	65.648	11.715	2.402
4	2.584	2.584	1.875	998	65.648	11.715	2.456
5	2.583	2.583	2.047	1.014	65.648	11.716	2.425
6	2.487	2.487	1.859	944	65.648	11.716	4.824
7	2.485	2.485	1.953	944	65.648	11.866	2.559
8	2.652	2.652	2.156	1.044	65.648	11.716	2.471
9	2.672	2.672	1.875	1.067	65.648	11.716	4.778
10	2.579	2.579	2.125	990	65.648	11.716	2.414
11	2.487	2.487	2.047	942	65.648	11.716	2.441
12	2.516	2.500	1.938	946	65.648	11.716	2.465

Varianta 3) Afișarea tuturor rezultatelor, tabelate pe câte 100 rânduri

Nr. măs.	Response Time(ms)	Time in WP(ms)	CPU time(ms)	DB req. Time(ms)	Memory (kB)	Transferred (kB)	HTTP-Watch (ms)
1	4.964	4.964	4.078	1.273	163.847	12.420	9.927
2	5.003	5.003	4.234	1.276	163.847	12.424	12.167
3	5.900	5.900	4.625	1.804	163.847	12.419	12.272
4	5.150	5.150	4.344	1.379	163.847	12.419	12.125
5	5.000	5.000	4.422	1.312	163.847	12.419	12.132
6	5.011	5.011	4.313	1.300	163.847	12.419	12.155
7	5.016	5.016	4.313	1.309	163.847	12.569	12.153
8	4.999	4.999	3.969	1.286	163.847	12.419	12.159
9	4.960	4.960	4.094	1.273	163.847	12.419	9.957
10	5.300	5.300	4.563	1.462	163.847	12.419	12.172
11	5.068	5.068	4.422	1.327	163.847	12.419	9.855
12	5.170	5.170	4.516	1.388	163.847	12.419	9.900

Anexa 6 Valori critice Grubbs, $g_n(P)$ [100]

P= n(↓)	0,95	0,98	0,99	0,999	P= n(↓)	0,95	0,98	0,99	0,999
	5	3,04	4,11	5,04		9,43	20	2,15	2,60
6	2,78	3,64	4,36	7,41	25	2,11	2,54	2,85	3,819
7	2,62	3,36	3,96	6,37	30	2,08	2,50	2,80	3,719
8	2,51	3,18	3,71	5,73	35	2,06	2,48	2,79	3,652
9	2,43	3,05	3,54	5,31	40	2,05	2,46	2,74	3,602
10	2,37	2,96	3,41	5,01	45	2,04	2,44	2,72	3,565
11	2,33	2,89	3,31	4,79	50	2,03	2,43	2,71	3,532
12	2,29	2,83	3,23	4,62	60	2,02	2,41	2,68	3,492
13	2,26	2,78	3,17	4,48	70	2,01	2,39	2,67	3,462
14	2,24	2,74	3,12	4,37	80	2,00	2,38	2,65	3,439
15	2,22	2,71	3,08	4,28	90	1,99	2,38	2,64	3,423
16	2,20	2,68	3,04	4,20	100	1,99	2,38	2,64	3,409
17	2,18	2,66	3,01	4,13	∞	1,96	2,33	2,57	3,291
18	2,17	2,64	2,98	4,07					

Anexa 7 Valorile parametrilor generați Web Dynpro ABAP

Nr. Crt.	CPUT	DBRT	HTTPWT	RT	Nr. Crt.	CPUT	DBRT	HTTPWT	RT
	GP	Frechet	Laplace	GEV		GP	Frechet	Laplace	GEV
1	177	259	1142	390	51	188	275	1222	405
2	191	507	1387	525	52	187	269	1199	399
3	184	264	1177	395	53	170	256	1070	387
4	189	283	1242	411	54	187	269	1200	399
5	168	254	768	384	55	169	255	1034	386
6	171	256	1088	387	56	174	257	1123	389
7	175	257	1128	389	57	172	256	1101	388
8	191	771	1436	628	58	169	255	1053	386
9	171	256	1095	387	59	188	273	1215	403
10	173	257	1116	388	60	183	262	1170	394
11	170	256	1073	387	61	170	256	1076	387
12	177	258	1140	390	62	181	261	1160	392
13	172	257	1105	388	63	177	259	1143	390
14	176	258	1139	390	64	183	262	1170	394
15	175	258	1134	389	65	190	295	1263	419
16	183	262	1169	394	66	177	258	1142	390
17	177	258	1140	390	67	180	260	1154	391
18	189	284	1244	411	68	172	257	1108	388
19	180	260	1154	392	69	183	263	1173	394
20	187	270	1204	400	70	174	257	1126	389
21	177	259	1142	390	71	175	258	1129	389
22	175	258	1129	389	72	171	256	1093	387
23	183	263	1172	394	73	183	263	1172	394
24	180	260	1153	391	74	186	266	1189	397
25	172	257	1110	388	75	187	269	1201	400
26	178	259	1145	390	76	177	258	1142	390
27	188	273	1215	403	77	168	255	1000	386
28	173	257	1115	388	78	169	255	1028	386
29	168	255	1018	386	79	185	265	1182	396
30	180	260	1154	391	80	182	261	1164	393
31	183	262	1170	394	81	191	355	1324	454
32	189	278	1231	407	82	183	262	1168	394
33	177	259	1142	390	83	185	265	1182	396
34	178	259	1146	390	84	174	257	1127	389
35	180	260	1156	392	85	174	257	1124	389
36	177	259	1144	390	86	178	259	1145	390
37	171	256	1093	387	87	186	266	1190	398
38	189	285	1247	412	88	173	257	1113	388
39	172	257	1108	388	89	170	256	1079	387

40	185	266	1188	397	90	169	256	1063	387
41	186	267	1192	398	91	178	259	1145	390
42	184	264	1176	395	92	178	259	1148	391
43	190	303	1276	424	93	189	277	1228	406
44	186	267	1193	398	94	170	256	1075	387
45	175	258	1129	389	95	174	257	1126	389
46	170	256	1074	387	96	188	277	1225	406
47	183	263	1173	394	97	174	257	1125	389
48	179	260	1152	391	98	170	256	1085	387
49	188	273	1214	403	99	171	256	1089	387
50	188	272	1210	402	100	180	260	1153	391

Anexa 8 Valorile parametrilor generați Floorplan Manager

Nr. Crt.	CPUT GEV	DBRT Cauchy	HTTPWT Unif	RT GEV	Nr. Crt.	CPUT GEV	DBRT Cauchy	HTTPWT Unif	RT GEV
1	267	301	1620	509	51	280	305	1739	518
2	281	418	1765	519	52	279	303	1723	517
3	276	302	1696	516	53	238	298	1537	480
4	280	306	1748	518	54	279	303	1723	517
5	102	-2553	1516	298	55	222	293	1526	463
6	245	299	1548	488	56	259	301	1584	502
7	261	301	1591	504	57	251	300	1558	494
8	281	642	1765	519	58	231	296	1531	472
9	248	300	1553	491	59	279	304	1735	518
10	257	300	1575	500	60	275	302	1683	515
11	239	298	1539	482	61	240	298	1540	483
12	266	301	1615	508	62	273	302	1664	514
13	252	300	1562	495	63	267	301	1621	509
14	266	301	1613	508	64	275	302	1683	515
15	264	301	1602	506	65	281	309	1755	518
16	275	302	1683	515	66	267	301	1619	509
17	266	301	1615	508	67	271	302	1649	512
18	280	307	1749	518	68	253	300	1565	497
19	271	302	1650	513	69	276	302	1688	516
20	279	304	1727	518	70	260	301	1588	503
21	267	301	1620	509	71	262	301	1595	505
22	262	301	1594	505	72	247	299	1551	490
23	276	302	1687	515	73	276	302	1687	516
24	271	302	1647	512	74	278	303	1711	517
25	254	300	1567	497	75	279	303	1724	517
26	268	301	1626	510	76	267	301	1619	509
27	279	304	1735	518	77	207	284	1520	445
28	256	300	1573	499	78	220	292	1524	460
29	215	290	1523	455	79	277	303	1703	516
30	271	302	1649	512	80	274	302	1672	514
31	275	302	1683	515	81	281	331	1763	519
32	280	305	1744	518	82	275	302	1681	515
33	267	301	1620	509	83	277	303	1702	516
34	269	301	1630	510	84	261	301	1590	504
35	272	302	1655	513	85	260	301	1586	503
36	268	301	1623	509	86	268	301	1627	510
37	247	299	1551	490	87	278	303	1713	517
38	280	307	1750	518	88	256	300	1571	499
39	254	300	1565	497	89	241	299	1542	484

40	278	303	1710	517	90	235	297	1534	477
41	278	303	1716	517	91	268	301	1628	510
42	276	302	1694	516	92	269	301	1633	511
43	281	312	1757	519	93	280	305	1743	518
44	278	303	1716	517	94	240	298	1540	483
45	262	301	1594	505	95	261	301	1589	503
46	239	298	1539	482	96	280	305	1741	518
47	276	302	1689	516	97	260	301	1588	503
48	271	302	1645	512	98	244	299	1546	487
49	279	304	1734	518	99	246	299	1548	489
50	279	304	1732	518	100	271	302	1647	512

Anexa 9 Valorile parametrilor generați CRM UI

Nr. Crt.	CPUT	DBRT	HTTPWT	RT	Nr. Crt.	CPUT	DBRT	HTTPWT	RT
	GP	GP	P5	GEV		GP	GP	P5	GEV
1	1921	980	2398	2555	51	2141	1069	2880	2670
2	2238	1107	2409	2786	52	2101	1053	2422	2646
3	2045	1031	4205	2616	53	1808	933	2659	2476
4	2168	1080	2484	2690	54	2101	1054	2419	2646
5	1782	923	2414	2348	55	1794	928	3362	2452
6	1822	939	2423	2491	56	1870	959	2480	2527
7	1880	963	26032	2533	57	1835	945	24439	2503
8	2243	1109	2411	2806	58	1800	930	3858	2464
9	1828	942	2437	2496	59	2130	1065	2441	2663
10	1857	954	2411	2519	60	2022	1022	3363	2605
11	1811	934	2407	2478	61	1812	935	4238	2481
12	1913	977	2423	2551	62	1989	1008	2494	2588
13	1841	947	2407	2507	63	1922	980	2416	2556
14	1910	975	4549	2549	64	2023	1022	2484	2605
15	1895	969	2580	2541	65	2189	1088	2406	2709
16	2022	1021	2433	2604	66	1919	979	2438	2554
17	1913	977	2759	2551	67	1965	998	2498	2577
18	2170	1081	2532	2692	68	1844	948	3412	2510
19	1967	999	57923	2578	69	2032	1026	2411	2609
20	2111	1057	2419	2651	70	1875	961	2670	2530
21	1921	980	2462	2555	71	1884	965	15653	2535
22	1884	965	3825	2535	72	1826	941	125405	2495
23	2029	1024	2554	2608	73	2030	1025	2436	2608
24	1961	997	3445	2575	74	2077	1043	5287	2632
25	1847	949	51585	2511	75	2104	1055	2419	2647
26	1930	984	2411	2560	76	1919	979	2410	2554
27	2130	1065	2402	2663	77	1788	925	2419	2432
28	1855	953	2434	2517	78	1792	927	2449	2448
29	1790	926	2435	2442	79	2060	1037	2426	2623
30	1965	998	2422	2577	80	2002	1013	2418	2595
31	2022	1022	2441	2605	81	2223	1102	2530	2752
32	2154	1075	2406	2679	82	2018	1020	2488	2602
33	1921	980	2440	2555	83	2059	1036	2698	2623
34	1935	986	2441	2562	84	1878	962	2406	2532
35	1975	1002	3147	2582	85	1872	960	3243	2528
36	1926	982	2410	2557	86	1931	984	2493	2560
37	1826	941	3176	2494	87	2080	1045	5264	2634
38	2173	1082	2480	2694	88	1853	952	2419	2516
39	1845	949	2507	2510	89	1814	936	2501	2482

40	2074	1042	2506	2631	90	1805	932	2416	2471
41	2086	1047	2429	2637	91	1932	985	2405	2561
42	2043	1030	2461	2615	92	1940	988	6080	2565
43	2199	1092	2411	2719	93	2150	1073	2443	2677
44	2087	1048	2643	2637	94	1812	935	2412	2480
45	1884	965	2407	2535	95	1877	962	2406	2531
46	1811	934	2791837	2479	96	2147	1072	2649	2674
47	2032	1026	3019	2609	97	1875	961	2737	2530
48	1959	996	2443	2574	98	1819	938	2553	2488
49	2129	1065	2406	2662	99	1823	939	2431	2491
50	2122	1062	2456	2658	100	1961	997	2411	2575

**LISTA PUBLICAȚIILOR REZULTATE ÎN URMA TEZEI DE
DOCTORAT,
PUBLICATE SUB AFILIERE UPT**

1. Lucrări științifice publicate în reviste indexate ISI

1. Cristea A.D., **Berdie A.D.**, Osaci M., Chirtoc D., *The advantages of using Mind Map for learning web Dynpro*, Computer Application in Engineering Education, vol.19,(1), 2011, pp.201-207, Jurnal ISI, Factor impact 0,33.

**2. Lucrări științifice publicate în volumele unor manifestări științifice
(Proceedings), indexate ISI Proceedings**

1. **Berdie A.D.**, Osaci M., Budișan N., *A Multicriteria Approach in the Selection of a SAP UI Technology*, Soft Computing Application, Proceedings of the 5th International Workshop Soft Computing Applications (SOFA) 2012, pp. 699
2. Osaci M., **Berdie A.D.**, Muscalagiu I., *Faceless Componentization in Web Dynpro ABAP –IT Project Management Strategy*, World Conference on Business, Economics and Management (BEM-2012), May 4–6, 2012, Antalya, Turkey, Procedia Social and Behavioral Sciences, vol. 62, oct. 2012, pp. 585-589.
3. Osaci M., **Berdie A.D.**, Lup A.E., *Educational soft presenting particularities through special functions of ABAP List Viewer in Web Dynpro technology*, 4th World Conference on Educational Sciences (WCES 2012), Barcelona, Spain, Procedia Social and Behavioral Sciences, Vol. 46, p.3450-3455, 2012, ISSN 1877-0428
4. **Berdie A.D.**, Osaci M., *Comparing the web business applications using the SAP Business Transaction Analysis*, World Conference on Business, Economics and Management (BEM-2012), May 4–6, 2012, Antalya, Turkey, Procedia Social and Behavioral Sciences, vol. 62, oct. 2012, p.45-51,
5. **Berdie A.D.**, Osaci M., Lemle L.D., *Comparative statistical study of some SAP UI technologies*, International Conference of Numerical Analysis and Applied Mathematics, Grecia, Vol. 1389, p.555-558, ISBN 978-0-7354-0956-9
6. **Berdie A.D.**, Osaci M., Proștean G., Haipl R., *Analysis and design in databases modeling using SAP integrated systems*, Proceedings of the 5th International Symposium on Applied Computational Intelligence and Informatics (SACI 2009), Timișoara, România, 2009, p. 247-260, ISBN 978-1-4244-4478-6

7. Osaci M., **Berdie A.D.**, Hammes A.D., *Studies on Efficiency of the Data Access by ABAP Programming*, Soft Computing Application, Proceedings of the 5th International Workshop Soft Computing Applications (SOFA) 2012, p. 707-714

3. Lucrări științifice publicate în reviste de specialitate indexate BDI

1. Cristea A.D., **Berdie A.D.**, Osaci M., *Application Componentization and component usages with Web Dynpro ABAP and Web Dynpro Java*, BULETINUL ȘTIINȚIFIC al Universitatii "Politehnica" din Timisoara, ROMANIA, Seria AUTOMATICA si CALCULATOARE SCIENTIFIC BULLETIN of "Politehnica" University of Timișoara, ROMANIA, Transactions on AUTOMATIC CONTROL and COMPUTER SCIENCE, Vol. 55 (69), Fasc. 1, 2010, ISSN 1224-600X
2. Osaci M., **Berdie A.D.**, Cristea A.D., *Control Tools for Optimizing ABAP Codes*, Acta Technica Corviniensis –Bulletin of Engineering, 5 (12), 2112, p.107-110
3. Cristea A.D., **Berdie A.D.**, Osaci M., *The use and importance of Hook Methods in Web Dynpro ABAP and Web Dynpro Java*, BULETINUL ȘTIINȚIFIC al Universitatii "Politehnica" din Timisoara, ROMANIA, Seria AUTOMATICA si CALCULATOARE SCIENTIFIC BULLETIN of "Politehnica" University of Timișoara, ROMANIA, Transactions on AUTOMATIC CONTROL and COMPUTER SCIENCE, Vol. 54 (68), Fasc. 3, 2009, ISSN 1224-600X
4. Osaci M., **Berdie A.D.**, Cristea A.D., Molnar E.D., *Modalities to Implement the Multilinguality in Web DYNPRO ABAP*, The Annals of "Dunărea de Jos" University of Galați, Fascicle I – 2010. Economics and Applied Informatics. XVI, no 1, p. 137-143, ISSN 1584-0409
5. Osaci M., **Berdie A.D.**, Proștean G., Cristea A.D., *Virtual Programming in SAP technologies*, Scientific Bulletin of the „Politehnica” University of Timișoara, România, Transactions on Automatic Control and Computer Control, **56**(70), no. 3, 2011, p.105-108.

4. Lucrări științifice publicate în volumele unor manifestări științifice (Proceedings) indexate BDI

1. Osaci M., **Berdie A.D.**, Proștean G., Pop A.A., *Faceless Componentization Versus Componentization with Assistance Class in Web Dynpro ABAP*, 7th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI), Timișoara, România, 2012, p. 107-110.
2. **Berdie A.D.**, Osaci M., Muscalagiu I., Proștean G., *A case-study about a web business application implemented in different SAP UI technologies*, 7th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI), Timișoara, România, 2012, p. 111-114

3. Cristea A.D., **Berdie A.D.**, Osaci M., *Parallel between two SAP Frameworks that use MVC paradigm*, IEEE International Joint Conferences on Computational Cybernetics and Technical Informatics (ICCC-CONTI 2010), Timișoara, 2010, p.397-400.
4. **Berdie A.D.**, Osaci M., Prostean G., Cristea A.D., *Web Programming features on integrated system SAP*, 6th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI), Timișoara, România, 2011, p. 227-230.
5. Osaci M., **Berdie A.D.**, Prostean G., Belea D., Cristea A.D., *VRML (Virtual Reality Modeling Language) Programming in SAP technologies*, 6th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI), May 19–21, 2011, Timișoara, Romania, **2011**, p.223-226

5. Lucrări științifice publicate în volumele unor manifestări științifice internaționale (Proceedings) din străinătate

1. **Berdie A.D.**, Osaci M., Oprea G., *Study about the estimation of the system parameters at the implementation of SAP web applications*, 3rd World Conference on Information Technology, nov. 2012, Barcelona, Spania, AWERPROCEDIA INFORMATION TECHNOLOGY AND COMPUTER SCIENCE, vol.3, 2013, p.394-399.
2. Osaci M., **Berdie A.D.**, Cristea A.D., *Studies about optimizing ABAP Codes for access to data from persistence level*, 2nd World Conference on Innovation and Computer Sciences INSODE, mai 2012, Izmir, Turcia, AWERPROCEDIA INFORMATION TECHNOLOGY AND COMPUTER SCIENCE, vol 2, p. 67-72.
3. Osaci M., **Berdie A.D.**, Szemciuck I., *Studies on the Web Dynpro ABAP Dynamic Programming of the Graphical Interface versus the Programming in Design Mode*, 3rd World Conference on Information Technology 2012, Barcelona, Spania, AWERPROCEDIA INFORMATION TECHNOLOGY AND COMPUTER SCIENCE, vol.3, 2013, p.890-894.
4. **Berdie, A.D.**, Osaci, M., Lemle, D.L., „An AHP evaluation between WD ABAP, FPM and CRM WebClient UI technologies”, 2nd World Conference on Innovation and Computer Sciences INSODE, mai 2012, Izmir, Turcia, AWERPROCEDIA INFORMATION TECHNOLOGY AND COMPUTER SCIENCE, vol. 2, p. 9-16.

6. Lucrări științifice publicate în volumele unor manifestări științifice

1. Cristea, A.D., **Berdie, A.D.**, Osaci, M., *Tipuri de date în Web Dynpro ABAP*, Lucrările celei de-a IX-a Conferințe Naționale Multidisciplinare cu Participare Internațională "Profesorul Dorin Pavel-Fondatorul Hidroenergeticii Românești", Știință și Inginerie, vol. 15, Sebeș, 2009, p. 681-688, ISSN 973-8130-82-4.

2. **Berdie, A.D.**, Osaci, M., Cristea, A.D., *Formulare interactive ADOBE în aplicații pe sisteme integrate SAP*, Lucrările celei de-a X-a Conferințe Naționale Multidisciplinare cu Participare Internațională "Profesorul Dorin Pavel-Fondatorul Hidroenergeticii Românești", Știință și Inginerie, vol. 17, Sebeș, 2010, p.364-370, ISSN 2067-7139.
3. Osaci, M., **Berdie, A.D.**, Supuran, A., *Arhitecturi soft pentru întreprinderi*, Lucrările celei de-a XI-a Conferințe Naționale Multidisciplinare cu Participare Internațională "Profesorul Dorin Pavel-Fondatorul Hidroenergeticii Românești", Știință și Inginerie, vol. 19, Sebeș, 2011, p.175-180, ISSN 973-8130-82-4.
4. Cristea, A.D., **Berdie, A.D.**, Osaci, M., *Error handling and messages with Application Server ABAP*, ISIRR 2009, Hunedoara, 2009, CD, ISSN 1584-2665.
5. Cristea, A.D., **Berdie A.D.**, Osaci, M., *Working with ABAP persistent data*, ISIRR 2009, Hunedoara, 2009, CD, ISSN 1584-2666.