**Document de specificatii pentru**

**Gestiunea**

**automată a studenților facultății**

**Echipa 7:**

**Aliman Roxana**

**Dumitrescu Loredana**

**Gidei Oana-Maria**

**Jianu Laura**

**Sirboiu Andreea**

**Cuprins**

**1. Introducere**

* Structura documentului……………………………………………………………………………3
* Scopul documentului…………………………………………………………………………………3
* Definitii, acronime, abrevieri si notatii……………………………………………………….3
* Scopul proiectului………………………………………………………………………………………4
* Referinte……………………………………………………………………………………………………4

**2. Descriere generala**

2.1 Perspectiva produsului…………………………………………………………………………….4

2.2 Cerintele functionale……………………………………………………………………………….5

2.3 Caracteristici de utilizator………………………………………………………………………..5

2.4 Mediu de lucru………………………………………………………………………………………..5

**3. Cerintele interfetei externe**

3.1 Interfata cu utilizatorul……………………………………………………………………………..5

3.2 Interfata hardware……………………………………………………………………………………5

3.3 Interfata software…………………………………………………………………………………….6

3.4 Interfata de comunicatii…………………………………………………………………………..6

**4. Cerinte non-functionale**

4.1 Cerinte de performanta…………………………………………………………………………….6

4.2 Cerinte de securitate…………………………………………………………………………………7

**5. Alte cerinte**

5.1 Diagrame Use Case…………………………………………………………………………………..8

**1. Introducere**

* **Structura documentului**

1. Introducere

Capitolul curent prezinta informatii despre scopul acestui document, modul in care este realizat si notatii folosite, cat si scopul proiectului in sine.

2. Descriere generala

Capitolul al doilea contine descrierea sistemului. Partea aceasta este scrisa in limbaj natural si este partea in care sunt exprimate cerintelor clientului.

3. Cerintele aplicatiei

Acest capitol prezinta cerintele sistemului software.

* **Scopul documentului**

Acest document descrie caietul de sarcini pentru proiectul **Gestiunea automata a studentilor facultatii (GASF).** GASFeste o aplicatie intranet, care poate fi accesata de pe întregul teritoriu al unei instituții sau al unui departament specificat. Acest sistem poate fi utilizat pentru monitorizarea studenților , dar si pentru a da posibilitatea studentilor de a accesa sau căuta orice informație cu privire la facultate.

**1.3 Definitii, acronime, abrevieri si notatii**

**e-Registrul matricol** reprezinta un document electronic ce contine pentru pentru fiecare student, toate disciplinele din planul de învăţământ cu numărul de credite si ore aferente fiecărei discipline.

**e-Centralizatorul de note** reprezinta un document electronic ce contine rezultatele obţinute de fiecare student, în urma sesiunilor de examene, într-un an universitar.

**Termen/Acronim Definitie**

**GASF** Gestiunea Automta a Studentilor Facultatii.

* **Scopul proiectului**

Scopul proiectului este dezvoltarea unui sistem software pentru automatizarea procesului de gestiune a studentilor unei facultatii care sa permita urmatoarele actiuni specifice:

- colectarea datelor corespunzatoare planurilor de invatamant, formatiunilor de studiu si a studenților acestora;

- gestionarea datelor;

- generarea de rapoarte;

- furnizarea de informatii prin intermediul unui portal web;

- interogarea sistemului;

- managementul fluxului de documente din cadrul secretariatului facultatii.

Acest sistem poate fi utilizat ca un sistem de gestionare a informațiilor

pentru o facultate. Pentru un student dat, administratorul creează ID de conectare și parola, astfel studentii pot avea acces la sistem si pot descărca informații dintr-o baza de date.

**1.5 Referinte**

- planuri de invatamant - <http://ace.ucv.ro/planuri.html>

- formatiuni de studiu - <http://ace.ucv.ro/orar.html>

- modalitati de evaluare - <http://ace.ucv.ro/evaluare.html>

**2.Descriere generala**

**2.1 Perspectiva produsului**  
 Produsul va fi o aplicație independentă și poate fi rulat pe mai multe sisteme în cadrul unei rețele Intranet. Produsul va necesita o tastatură, mouse și monitor pentru interfața cu utilizatorii. Cerințele minime de hardware pentru produs sunt specificate în acest document.

**2.2 Cerintele functionale**

Cerinte **generale** ale sistemului:

- Interfaţa cu utilizatorul trebuie sa fie accesibila web, securizat, cu acces diferentiat pentru rolurile

diferite de utilizator.

- Datele din cadrul GASF trebuie sa fie usor de vizualizat si de localizat; pentru aceasta se vor utiliza listele de tip selectie.

- Sistemul va contine urmatoarele module: modulul pentru colectarea datelor;

modulul pentru raportare si modulul pentru jurnalizarea operatiilor.

- Rapoartele trebuie sa poata fi vizualizate sub diverse forme; componenta de raportare trebuie sapermita salvarea rapoartelor in diferite formate cum ar fi CVS, XML, XLS si PDF.

- Managementul datelor trebuie sa fie asigurat prin functionalitati de extragere a datelor din diferite surse de date (baze date, fisiere Excel, fisier text), prin realizarea de filtre si diferite alte transformari asupra datelor necesare integrarii acestora in cadrul bazei de date.

Functionalitatile **portalului web** vor fi:

- Un utilizator al sistemului trebuie sa se poata inregistra prin intermediul interfatei web, avand initial un set predefinit de drepturi; drepturile de acces vor fi personalizate ulterior pentru permite un acces diferentiat la date.

- Colectarea datelor introduse de utilizator se va realiza prin intermediul interfatei web, iar utilizatorul va putea sa efectueze modificari, sa creeze si sa genereze rapoarte in concordanta cu drepturile alocate.

- Cautarea documentelor se va putea realiza dupa tipul fisierului cautat, dupa data crearii, dupa autorul documentului sau dupa continutul documentului.

- Datele vor fi afisate in cadrul paginii web sub forma tabelara, putand fi salvate intr-un format standard: CSV, XML sau XLS.

- Sistemul va permite definirea de grupuri de utilizatori si executia unor operatii la nivel de grup (de exemplu: trimiterea de mesaje de e-mail tuturor membrilor grupului).

- Validarea datelor se va realiza in cadrul modulului ce implementeaza interfata utilizator si va genera mesaje de eroare, iar cazurile de erori frecvente va sugera metode de rezolvare a acestora.

Cerintele pentru **baza de date**:

- Baza de date trebuie sa permita gestiunea unui volum mare de date referitoare la situatia

scolara a studentilor.

- Sistemul de gestiune al bazei de date trebuie sa permita: recuperarea datelor la un moment specificat de timp, suport de backup offline, administrarea din browser web, accesul concurrent al utilizatorilor la informatii, definirea de utilizatori si grupuri de utilizatori cu diferite drepturi de acces, monitorizarea tuturor tipurilor de interogări si implementarea structurilor de date complexe.

- Se va avea in vedere posibilitatea stocarii in cadrul fisierelor „log” a tuturor activitatilor din cadrul sistemului; se va identifica pentru fiecare activitate, utilizatorul care a realizat actiunea si toate datele relevante legate de informatia accesata/modificata.

**2.3 Caracteristici de utilizator**

Publicul țintă pentru GASF sunt studentii. Utilizatorii acestui sistem sunt:

- Administrator - Super utilizator al sistemului.

-Student - Un utilizator cu drepturi de acces limitat.

Vederea frontală a sistemului este formata din diferite servicii furnizate de sistem și o pagină de conectare unde un utilizator solicita conectarea la sistem prin numele său de utilizator și parola. După conectarea studentului, datele sale sunt actualizate în sistem. Dupa ce a introdus toate datele, studentul are posibilitatea de a vedea doar datele lui, dar, de asemenea, si de vizualiza examenele și raportul lor.

**3.Cerintele interfetei externe**

**3.1 Interfata cu utilizatorul**

-simpla

-atractiva

-ușor de utilizat

-consecventă

-auto-explicativa

**3.2 Interfata Hardware**  
- partea de server: Aplicația web va fi găzduita pe un server web.

- partea de client:

Monitor - software-ul afișează informații pentru utilizator prin intermediul monitorului.

Mouse - software-ul interacționează cu mișcarea mouse-ului . Mouse-ul se activează zone pentru introducerea de date, butoane de comandă.

Tastatură – cu ajutorul tastaturii se vor introduce datele de intrare în zona activă a bazei de date.

**3.3 Interfata software**

-partea de server:

Un server de web Apache va accepta toate solicitările de la client și va transmite

în consecinţă. O bază de date va fi găzduita folosind PostgreSQL la nivel central.

-pe partea de client:

Un sistem de operare, care este capabil de a rula un browser web modern, care acceptă JavaScript și HTML.

**3.4 Interfata de comunicatii**

Protocolul HTTPS sau HTTP va fi folosit pentru a facilita comunicarea dintre client și server.

**4.Cerinte non-functionale**

**4.1 Cerinte de performanta**  
Sistemul ar trebui să sprijine cel puțin 200 de utilizatori simultan. Această afirmație oferă un sentiment general de siguranță atunci când sistemul este sub sarcină. Este important ca un număr substanțial de utilizatori să poată accesa sistemul în același timp timp, din moment ce un portal universitar este important pentru cursurile pe care le folosesc.

**4.2 Cerinte de securitate**  
1. Parolele vor fi salvate criptate in baza de date, în scopul de a asigura confidențialitatea utilizatorului.

2. IP-ul utilizatorului va fi inregistrat.

3. Sistemul va fi protejat împotriva vulnerabilităților, cum ar fi SQL injection.

**5. Diagrame Use Case**

**ADMINISTRATOR**

|  |  |
| --- | --- |
| ‘Login’ Use Case | |
| Use Case Name | Login |
| Use Case Description | Permite administratorului, pe baza unui username si a unei parole valide sa aiba acces la datele studentiilor. |
| Actor | User(Admin) |
| Pre-conditions | Existenta unor campuri pentru username si password unde administratorul va introduce datele. |
| Basic Flow | 1.Sistemul dispune de doua campuri pentru introducerea datelor  2.Userul completeaza cele doua campuri cu date valide  3.Userul va face click pe butonul “Login”  4.Userul va fi directionat catre o pagina principala in care vor exista cateva date generale . De aici el va alege ce doreste sa faca mai departe. |
| Post Condition | - |
| Alternate Flow | Daca username-ul si parola nu sunt valide:  1.Sistemul va semnala un mesaj de eroare |

|  |  |
| --- | --- |
| ‘Manage Accounts’ Use Case | |
| Use Case Name | Manage accounts |
| Use Case Description | Permite administratorului sa creeze conturi pentru studentii noi |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui link care va directiona administratorul cate o pagina html in care va putea intregistra un nou student prin intermediul unei ferestre de dialog |
| Basic Flow | 1.Sistemul dispune un link catre o pagina html destinata intregistrari de noi studenti  2.Userul va face click pe butonul Register student  3.Sistemul la dispune de o fereastra de dialog unde se afla campurile necesare inregistrari unui student  4.Userul va face click pe butonul “Save”  5.Sistemul va salva in baza de date contul creat de administrator |
| Post Condition | Informatiile adaugate vor aparea in baza de date. |
| Alternate Flow | Daca inregistrarile exista deja:  1.Sistemul va semnala un duplicat |

|  |  |
| --- | --- |
| ‘Generate Reports’ Use Case | |
| Use Case Name | Generate Reports |
| Use Case Description | Permite utilizatorilor sa genereze un raport cu datele cerute. Datele pot fi filtrate pe an,specializare. |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui buton pentru generarea raportului |
| Basic Flow | 1.Sistemul dispune de un camp folosit pentru filtrarea datelor  2.Userul completeaza campul cu ceea ce doreste sa obtina in raport  3.Sistemul va aduce datele cerute din baza de date  4.Userul va da click pe butonul “Print”  5. Informatiile cerute vor fi printate in format pdf si salvate in computer |
| Post Condition | - |
| Alternate Flow | Daca utilizatorul doreste alt format pentru generarea raportului:  1.Userul poate alege si alt format pentru generarea raportului |

|  |  |
| --- | --- |
| ‘Insert Student’ Use Case | |
| Use Case Name | Insert Student |
| Use Case Description | Permite utilizatorilor sa insereze in baza de date un nou student |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui camp in care userul introduce datele |
| Basic Flow | 1.Sistemul dispune de un link care va conduce utilizatorul din pagina html de “home” in pagina html “student”  2. Aici sistemul va dispune de un buton “New Student”  3.Userul va face click pe butonul “New Student”  4.Va aparea o fereastra de dialog in care utilizatorul va introduce datele dorite  3.Userul va face click pe butonul “Save”  4.Sistemul va salva modificarile realizate de user |
| Post Condition | Informatiile adaugate vor aparea in baza de date. |
| Alternate Flow | Daca inregistrarile exista deja:  1.Sistemul va semnala un mesaj prin care anunta utilizatorul ca datele exista deja |

|  |  |
| --- | --- |
| ‘Delete Student’ Use Case | |
| Use Case Name | Delete Student |
| Use Case Description | Permite utilizatorilor sa stearga din baza de date a evidentei studentilor |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui buton pentru executarea optiunii de stergere |
| Basic Flow | 1.Sistemul va dispune de un camp de cautare a datelor  2.Userul completeaza completeaza campul cu datele dorite  3.Userul va face click pe studentul pe care doreste sa il stearga  4.Userul va face click pe butonul “Delete”  5.Sistemul va salva modificarile realizate de user |
| Post Condition | Informatiile vor fi sterse din baza de date. |
| Alternate Flow | - |

|  |  |
| --- | --- |
| ‘Update Student’ Use Case | |
| Use Case Name | Update Student |
| Use Case Description | Permite utilizatorilor sa actualizeze datele studentiilor |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui buton de Update |
| Basic Flow | 1.Sistemul va dispune de un camp de cautare a dateor  2.Userul completeaza completeaza campul cu datele dorite  3.Userul va face click pe studentul pe care doreste sa il actualizeze  4.Userul va face click pe butonul “Update”  5. Va aparea o fereastra de dialog in care utilizatorul va actualizate datele dorite  6.Userul va face click pe butonul “Save”  7.Sistemul va salva modificarile realizate de user |
| Post Condition | Informatiile adaugatevor fi actualizate in baza de date. |
| Alternate Flow | - |

|  |  |
| --- | --- |
| ‘Insert Course’ Use Case | |
| Use Case Name | Insert Course |
| Use Case Description | Permite utilizatorilor sa insereze un nou curs unui student |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui camp in care userul introduce datele |
| Basic Flow | 1.Userul va putea afla sau adauga mai multe informatii despre un student facand click pe numele studentului respectiv  2.Utilizatorul va fi directionat catre o pagina un sistemul va aduce din baza de date toate informatiile despre studentul respectiv  3.Userul va face click pe butonul “New Course”  4.Userul va completa campurile cu informatiile pe care doreste sa le adauge  5.Userul va face click pe butonul “Save”  6.Sistemul va salva modificarile realizate de user |
| Post Condition | Informatiile adaugate vor fi salvate in baza de date. |
| Alternate Flow | Daca inregistrarile exista deja:  1.Sistemul va semnala un mesaj prin care anunta utilizatorul ca datele exista deja |

|  |  |
| --- | --- |
| ‘Insert Mark ’ Use Case | |
| Use Case Name | Insert Mark |
| Use Case Description | Permite utilizatorilor sa insereze o nota la un anumit curs unui student |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui camp in care userul introduce datele |
| Basic Flow | 1.Userul va putea afla sau adauga mai multe informatii despre un student facand click pe numele studentului respectiv  2.Utilizatorul va fi directionat catre o pagina un sistemul va aduce din baza de date toate informatiile despre studentul respectiv  3.Userul va face click pe butonul “New Course”  4.Userul va completa campurile cu informatiile pe care doreste sa le adauge  5.Userul va face click pe butonul “Save”  6.Sistemul va salva modificarile realizate de user |
| Post Condition | Informatiile adaugate vor fi salvate in baza de date. |
| Alternate Flow | Daca inregistrarile exista deja:  1.Sistemul va semnala un mesaj prin care anunta utilizatorul ca datele exista deja |

|  |  |
| --- | --- |
| ‘View Data’ Use Case | |
| Use Case Name | View Data |
| Use Case Description | Permite utilizatorilor sa vizualizeze informatii din baza de date |
| Actor | User(Admin) |
| Pre-conditions | Existenta unui link care va directiona utilizatorul catre o noua pagina |
| Basic Flow | 1.Sistemul dispune de un link care va conduce utilizatorul din pagina html de “home” in pagina html “student”  2.Userul va putea cauta un student dupa nume, sau va putea vizualiza toti studentii dintr-un an, sau/si de la o anumita specializare |
| Post Condition | - |
| Alternate Flow | - |

**STUDENT**

|  |  |
| --- | --- |
| ‘Login’ Use Case | |
| Use Case Name | Login |
| Use Case Description | Permite studentului, pe baza unui username si a unei parole valide sa aibe acces la datele personale. |
| Actor | User(Student) |
| Pre-conditions | Existenta unor campuri pentru username si password unde studentul va introduce datele. |
| Basic Flow | 1.Sistemul dispune de doua campuri pentru introducerea datelor  2.Userul completeaza cele doua campuri cu date valide  3.Userul va face click pe butonul “Login”  4.Userul va fi directionat catre o pagina principala in care vor exista cateva date generale . De aici el va alege ce doreste sa faca mai departe. |
| Post Condition | - |
| Alternate Flow | Daca username-ul si parola nu sunt valide:  1.Sistemul va semnala un mesaj de eroare |

|  |  |
| --- | --- |
| ‘Request Document’ Use Case | |
| Use Case Name | Request Document |
| Use Case Description | Permite utilizatorilor sa genereze un document in format pdf cu datele personale |
| Actor | User(Student) |
| Pre-conditions | Existenta unui buton pentru generarea raportului |
| Basic Flow | 1.sistemul va dispune o pagina principala in care vor exista cateva date generale . De aici el va alege ce doreste sa faca mai departe.  2.Userul va da click pe butonul “Academic Situation”  3.Sistemul va aduce datele cerute din baza de date  4.Userul va da click pe butonul “Print”  5. Informatiie cerute vor fi printate in format pdf si salvate in computer |
| Post Condition | - |
| Alternate Flow | - |

|  |  |
| --- | --- |
| ‘View personal Data’ Use Case | |
| Use Case Name | View personal Data |
| Use Case Description | Permite utilizatorilor sa vizualizeze informatii din baza de date |
| Actor | User(Student) |
| Pre-conditions | Existenta unui link care va directiona utilizatorul catre o noua pagina |
| Basic Flow | 1.Sistemul dispune de un link care va conduce utilizatorul din pagina html de “home” in pagina html “student”  2.Userul va putea vizualiza datele personale facand click pe butonul “Personal Data”  3.Userul va fi directionat catre o pagina html unde sistemul va aduce din baza de date toate datele personale ale studentului |
| Post Condition | - |
| Alternate Flow | - |