

---

# **Specificarea cerintelor software**

**pentru**

## **Sistem informatic pentru gestiunea automata a studentilor facultatii**

**Versiunea 1.0**

**Realizat de Alexandra-Mihaela Voicu,**

**Vlad-Ionut Greaca,**

**Alexandru Gegia**

**Facultatea de Automatica, Calculatoare si Electronica, Craiova**

**25.02.2023**

## **Table of Contents.**

<b>1. Introducere</b>	<b>3</b>
1.1 Scop	3
1.2 Conventiile documentului	3
1.3 Public vizat si sugestii de lectura	4
1.4 Scopul Produsului	4
1.5 Referinte	5
<b>2. Descriere de ansamblu</b>	<b>5</b>
2.1 Perspectiva Produsului	5
2.2 Functionalitatile Produsului	5
2.3 Tipurile de utilizatori si caracteristici	6
2.4 Mediul de operare	6
2.5 Constrangerii de proiectare si de implementare	6
2.6 Documentatia utilizatorului	6
2.7 Ipoteze si Dependente	6
<b>3. Cerintele interfetei externe</b>	<b>6</b>
3.1 Interfata Utilizatorului	6
3.2 Interfele Hardware	6
3.3 Interfetele Software	6
3.4 Interfata Comunicatiilor	18
<b>4. Caracteristicile sistemului</b>	<b>18</b>
4.1 Functia de mesagerie	18
4.2 Functia gestionare orar	18
4.3 Functia centralizarii notelor	19
4.4 Functia de organizare interna.	20
4.5 Functia avizier	22
4.6 Functia istoricul documentelor	22
<b>5. Alte Cerinte Nefunctionale</b>	<b>22</b>
5.1 Cerinte de performanta	22
5.2 Cerinte de siguranta	23
5.3 Cerinte de securitate	23
5.4 Atributele calitatii software	23
5.5 Reguli de afacere	24
<b>6. Alte Cerinte</b>	<b>24</b>

## Revision History

Nume	Data	Motivul modificarii	Versiunea
Greaca Vlad-Ionut, Gegia Alexandru, Voicu Mihaela-Alexandra	25.02.2023	Modificarea copertii	0.1
Greaca Vlad-Ionut	25.02.2023	Completarea sectiunii 1.4, alaturi de 3 reguli ale atributelor calitatii software	0.2
Gegia Alexandru	25.02.2023	Completarea sectiunii 1.3 si adaugarea a 4 reguli ale atributelor calitatii software	0.3
Voicu Mihaela-Alexandra	26.02.2023	Completarea sectiunii 1.1 si adaugarea a 3 reguli ale atributelor calitatii software	0.4
Greaca Vlad-Ionut, Gegia Alexandru, Voicu Mihaela-Alexandra	01.03.2023	Stabilirea cerintelor functionale a caracteristicilor sistemului	0.5
Gegia Alexandru	04.03.2023	Completarea sectiunilor 1.2 si 2.4	0.6
Greaca Vlad-Ionut	05.03.2023	Completarea sectiuniilor 2.1 si 2.6	0.7
Voicu Mihaela-Alexandra	05.03.2023	Completarea sectiunilor 2.2 si 2.4	0.8
Greaca Vlad-Ionut, Gegia Alexandru, Voicu Mihaela-Alexandra	05.03.2023	Completarea capitolului 4, Caracteristicile sistemului	0.9
Greaca Vlad-Ionut	09.03.2023	Corectarea cerintei "Documentatia utilizatorului"	0.10
Gegia Alexandru	11.03.2023	Completarea sectiunii 3.4	0.11
Voicu Alexandra Mihaela	11.03.2023	Completarea sectiunii 5.1	0.11
Greaca Vlad-Ionut	12.03.2023	Completarea sectiunii 3.3 Modificarea formatului scrisului de la 11 la 12	0.12
Gegia Alexandru	12.03.2023	Completarea sectiunii 3.3	0.13
Voicu Alexandra Mihaela	12.03.2023	Completarea sectiunii 5.1	0.14

Greaca Vlad-Ionut	12.03.2023	Completarea sectiunii 3.1	0.15
Greaca Vlad-Ionut, Gegia Alexandru, Voicu Mihaela-Alexandra	13.03.2023	Modificare Sectiunea 3.1 Modificare Sectiunea 4 Modificare Sectiunea 5.3	1.0

## 1. Introducere

### 1.1 Scop

Sistemul software pentru pentru “Gestiunea Automata a Studentilor Facultatii”, Versiunea 1.0, include următoarele subsisteme: colectarea si gestionarea datelor studentilor, cat si ale profesorilor, generarea de rapoarte privind situatia scolara a studentilor, planificarea pe specializari de studiu, furnizarea de informatii prin intermediul unui portal web si managementul fluxului de documente din cadrul secretariatului facultatii.

### 1.2 Conventiile documentului

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

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

**Managementul fluxului de documente** = reprezinta un sistem informatic care permite circulatia (pentru informari, aprobari sau modificari), stocarea si regasirea documentelor in format electronic, cu facilitati de conectare la alte sisteme informatice sau dispozitive electronice.

**GASF** = Gestiunea Automata a Studentilor Facultatii

**SSRS** = Specificarea cerintelor software

**WSGI** = Interfata Gateway Server Web

### 1.3 Public vizat si sugestii de lectură

Acest document se adreseaza tuturor utilizatorilor, cat si personalului ce va intretine pachetul software.

Sectiunile cu informații tehnice referitoare la functionalitate, depanare si/sau intretinere sunt menite angajatilor ce vor intretine/depana pachetul software (sectiunile cu

acest rol sunt: sectiunea 2 in intregime, subsectiunile 3.2, 3.3, 3.4 si sectiunile 4 si 5 in intregime).

Sectiunile cu informatii generale despre rolul pachetului si modul de utilizare sunt menite utilizatorilor pachetului software. Atat profesorii cat si secretariatul institutiei trebuie sa citeasca subsectiunile 2.1, 2.2, 2.6, 3.1, 3.4, cat si sectiunile 4 si 5 in intregime.

Studentii ce vor utiliza pachetul, pot citi doar sectiunile 2.2, 2.6, 3.1, 4 in intregime cat si 5.2.

## **1.4 Scopul Produsului**

Scopul produsului este de a facilita comunicarea intre studenti si profesori prin intermediul unei aplicatii web, care permite studentilor sa acceseze informatii relevante despre datele personale, orar si notele obtinute, iar profesorilor sa le ofere posibilitatea de a acorda note tuturor studentilor, sa gestioneze orarul lor personal si sa transmita studentilor anunturi importante, cum ar fi modalitatile de evaluare.

In plus, in cazul aparitiei unei erori, studentii vor putea transmite problema profesorilor, iar acestia vor avea posibilitatea de aprobare sau respingere, in functie de situatia constatata.

De asemenea, unele informatii, cum ar fi orarul sau notele primite vor fi salvate sub forma unui raport pdf.

Drepturile vor fi acordate de catre secretariat, care va crea conturile atat pentru studenti, cat si pentru profesori, prin intermediul unei interfete usor de utilizat.

## **1.5 Referinte**

Surse: <https://caxapa.ru/thumbs/468328/misra-c-2004.pdf>

<http://www.ace.ucv.ro/>

<https://cis01.central.ucv.ro/evstud/>

<https://docs.djangoproject.com/en/4.1/ref/databases/>

[https://www.w3schools.com/django/django\\_intro.php](https://www.w3schools.com/django/django_intro.php)

<https://docs.djangoproject.com/en/4.1/howto/deployment/>

<https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>

<https://cis01.central.ucv.ro/evstud/site/index.php?sitecateg=1>

[https://cis01.central.ucv.ro/evstud/academic/situatia\\_scolara.php](https://cis01.central.ucv.ro/evstud/academic/situatia_scolara.php)

<http://www.ace.ucv.ro/invatamant/utile/orar.php>

[http://www.ace.ucv.ro/invatamant/utile/modalitati\\_de\\_evaluare.php](http://www.ace.ucv.ro/invatamant/utile/modalitati_de_evaluare.php)

## **2. Descriere de ansamblu**

### **2.1 Perspectiva Produsului**

Scopul produsului este de a facilita comunicarea intre studenti si profesori prin intermediul unei aplicatii web, care permite studentilor sa acceseze informatii relevante despre datele personale, orar si notele obtinute, iar profesorilor sa le ofere posibilitatea de a acorda note tuturor studentilor, sa gestioneze orarul lor personal si sa transmita studentilor anunturi importante, cum ar fi modalitatile de evaluare.

In plus, in cazul aparitiei unei erori, studentii vor putea transmite problema profesorilor, iar acestia vor avea posibilitatea de aprobare sau respingere, in functie de situatia constatata.

De asemenea, unele informatii, cum ar fi orarul sau notele primite vor fi salvate sub forma unui raport pdf.

Drepturile vor fi acordate de catre secretariat, care va crea conturile atat pentru studenti, cat si pentru profesori, prin intermediul unei interfete usor de utilizat.

### **2.2 Functionalitatile Produsului**

Funcțiile programului sunt:

- Functia de mesagerie;
- Functia de gestionare orar;
- Functia centralizare note;
- Functia de organizare interna;
- Functia avizier
- Functia istoricului documentelor

### **2.3 Tipurile de utilizatori si caracteristici**

Tipurile de utilizatori existente in pachetul software sunt:

- Profesor
- Student
- Admin(Secretariat)

Studentii isi vor putea vizualiza situatia scolara, orarul,cat si modalitatile de evaluare, au posibilitatea de trimitere si primire de mail-uri catre alti studenti, profesori si secretariat. De asemenea in cazul aparitiei unei erori de notare, au permisiunea semnalariei acesteia.

Profesorii vor putea vizualiza orarul studentilor cat si pe cel personal, sa acorde note studentilor. De asemenea, profesorii pot trimite mailuri catre ceilalti utilizatori.

Secretariatul poate crea alti utilizatori, trimite mailuri catre ceilalti utilizatori. Secretariatul modifica orarul si structura registrului matricol.

## **2.4 Mediul de operare**

Specificatii minime:

*Sistem de operare: Windows 10*

*Procesor: Intel(R) Core(TM) i5-6500 CPU @ 3.20GHz 3.19 GHz*

*RAM 8,0 GB*

*Tipul sistemului: 64-bit sistem de operare, x64-based processor*

*Tipul browserului: Microsoft Edge Versiunea 110.0.1587.63 (64-bit)*

*Tipul Python: Python versiunea 3.11*

*Framework: Django 4.1.4*

*Baza de date: MySQL 5.7*

## **2.5 Constrangerii de proiectare si de implementare**

*Nu se aplica.*

## **2.6 Documentatia utilizatorului**

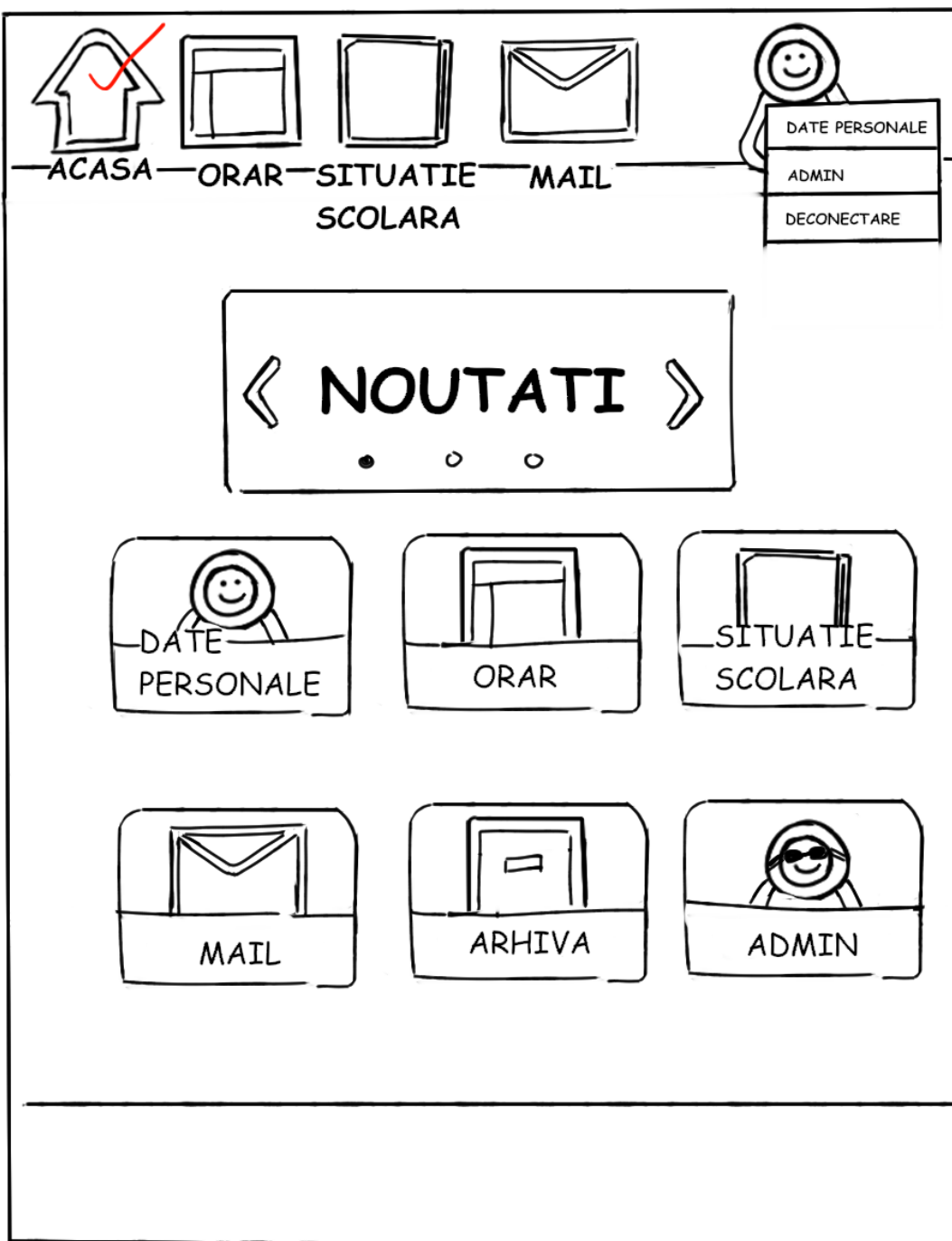
Documentatia utilizatorului este reprezentata de documentul "SRS pentru sistem de gestiune automata a studentilor".

## **2.7 Ipoteze si Dependente**

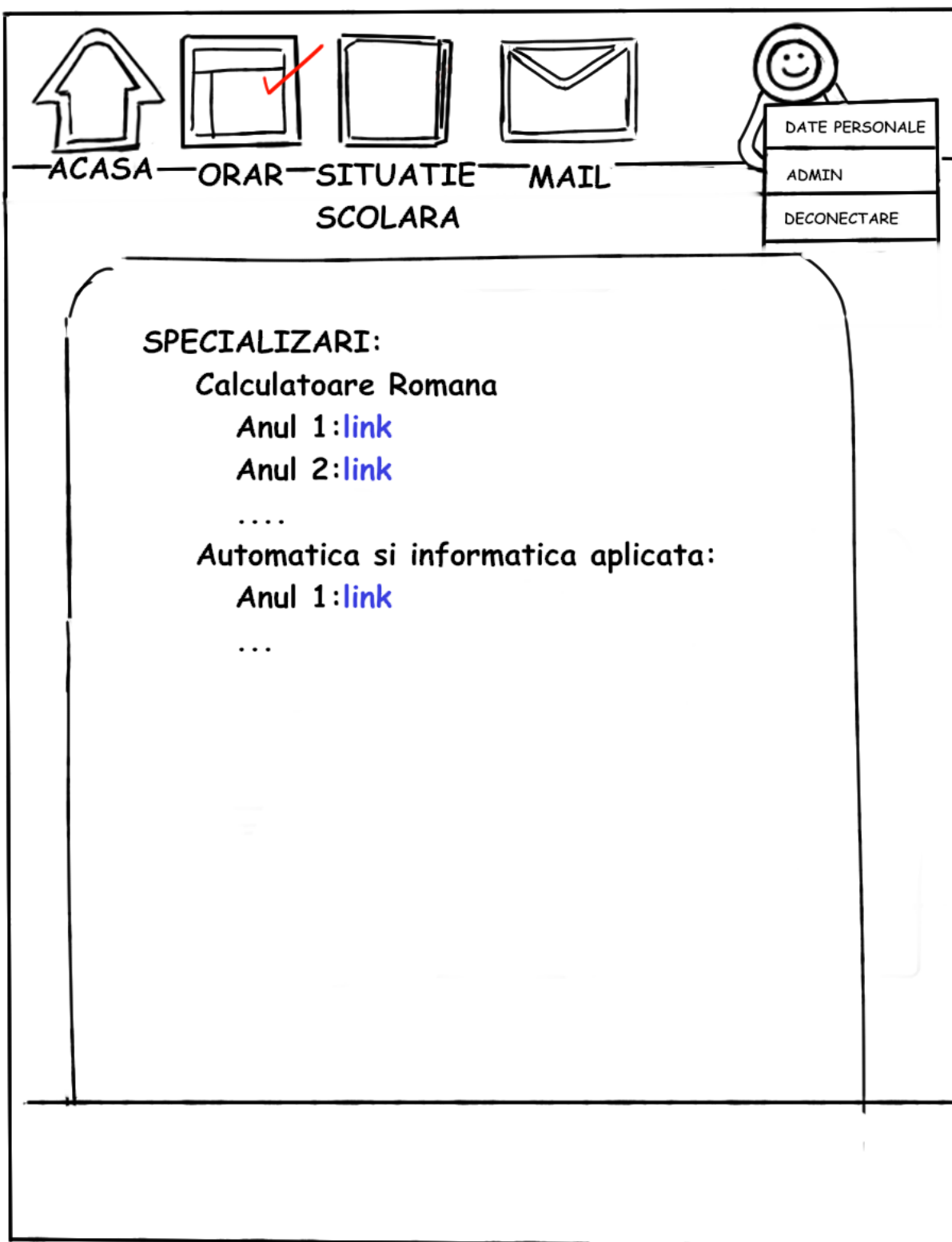
*Nu se aplica.*

### 3. Cerintele interfetei externe

#### 3.1 Interfata Utilizatorului







The wireframe depicts a web application interface for a student management system. At the top, there is a navigation bar with four icons: a house, a calendar, a book, and an envelope with a red checkmark. Below these icons are the labels 'ACASA', 'ORAR', 'SITUATIE', and 'MAIL'. The 'MAIL' label is underlined. To the right of the navigation bar is a user profile icon (a circle with a smiley face) and a vertical menu with three options: 'DATE PERSONALE', 'ADMIN', and 'DECONNECTARE'. Below the navigation bar, the text 'SCOLARA' is centered. The main content area is a large rectangle containing four identical message boxes. Each box has a label 'Nume' and a text field with the placeholder 'O informatie importanta care nu trebuie ratata'. Below the fourth message box is an ellipsis '...'. At the bottom right of the interface is a 'COMPOSE' button with a pencil icon.

ACASA ORAR SITUATIE MAIL

SCOLARA

DATE PERSONALE  
ADMIN  
DECONNECTARE

Nume  
O informatie importanta care nu trebuie ratata

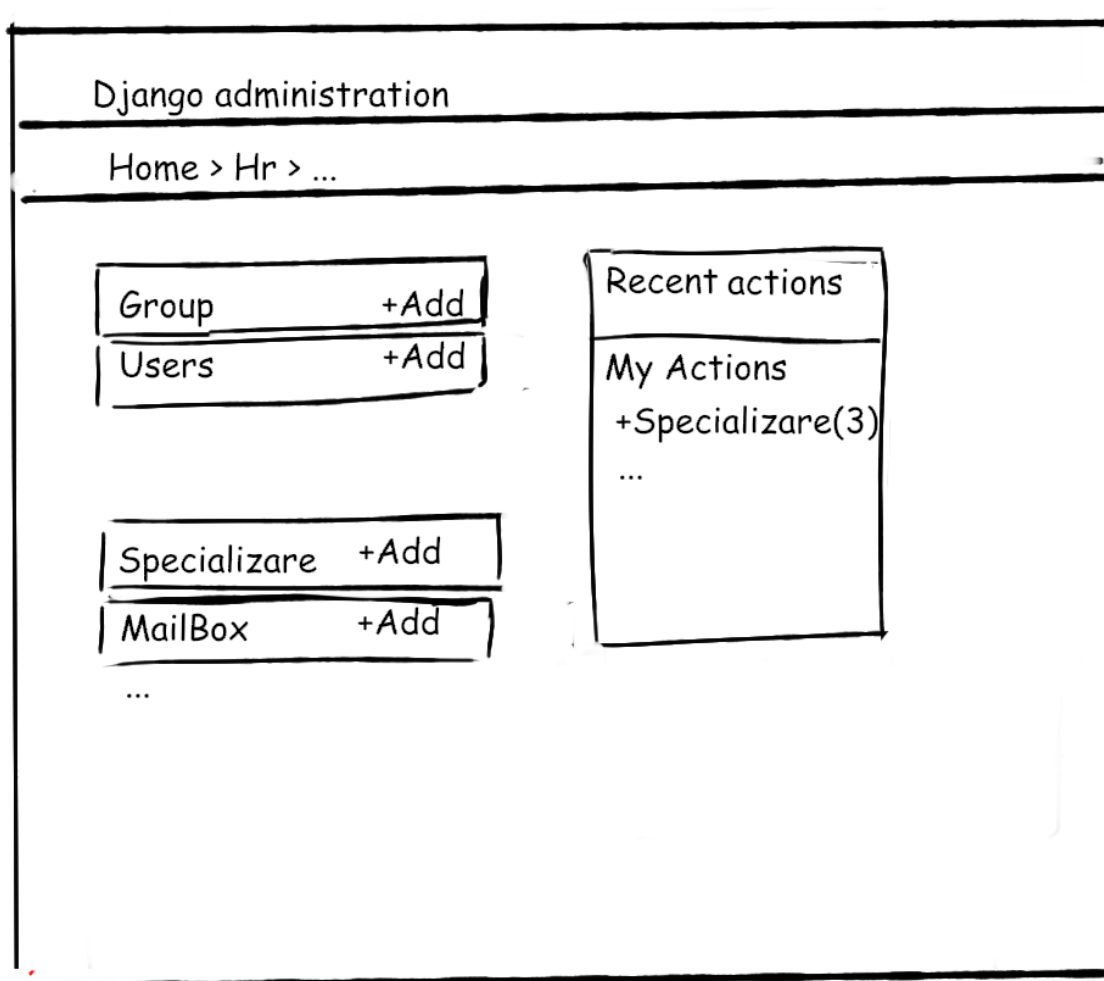
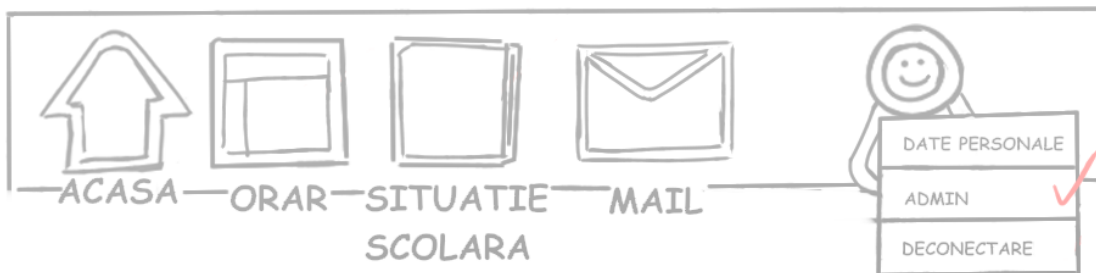
Nume  
O informatie importanta care nu trebuie ratata

Nume  
O informatie importanta care nu trebuie ratata

Nume  
O informatie importanta care nu trebuie ratata

...

COMPOSE





Django administration

Home > Hr > ...

Group	+Add
Users	+Add

Change Mailbox / Add Mailbox  
Mailbox object(11)

Specializare	+Add
MailBox	+Add
...	

Transmitator  (v)

Receptor  (v)

Mail

ACASA ORAR SITUATIE SCOLARA MAIL






DATE PERSONALE  
ADMIN  
DECONNECTARE

Catre PROFESOR/GRUP/SECRETARIAT

Subiect Titlu

Ceva mesaj pompos legat de o problema

SEND COMPOSE



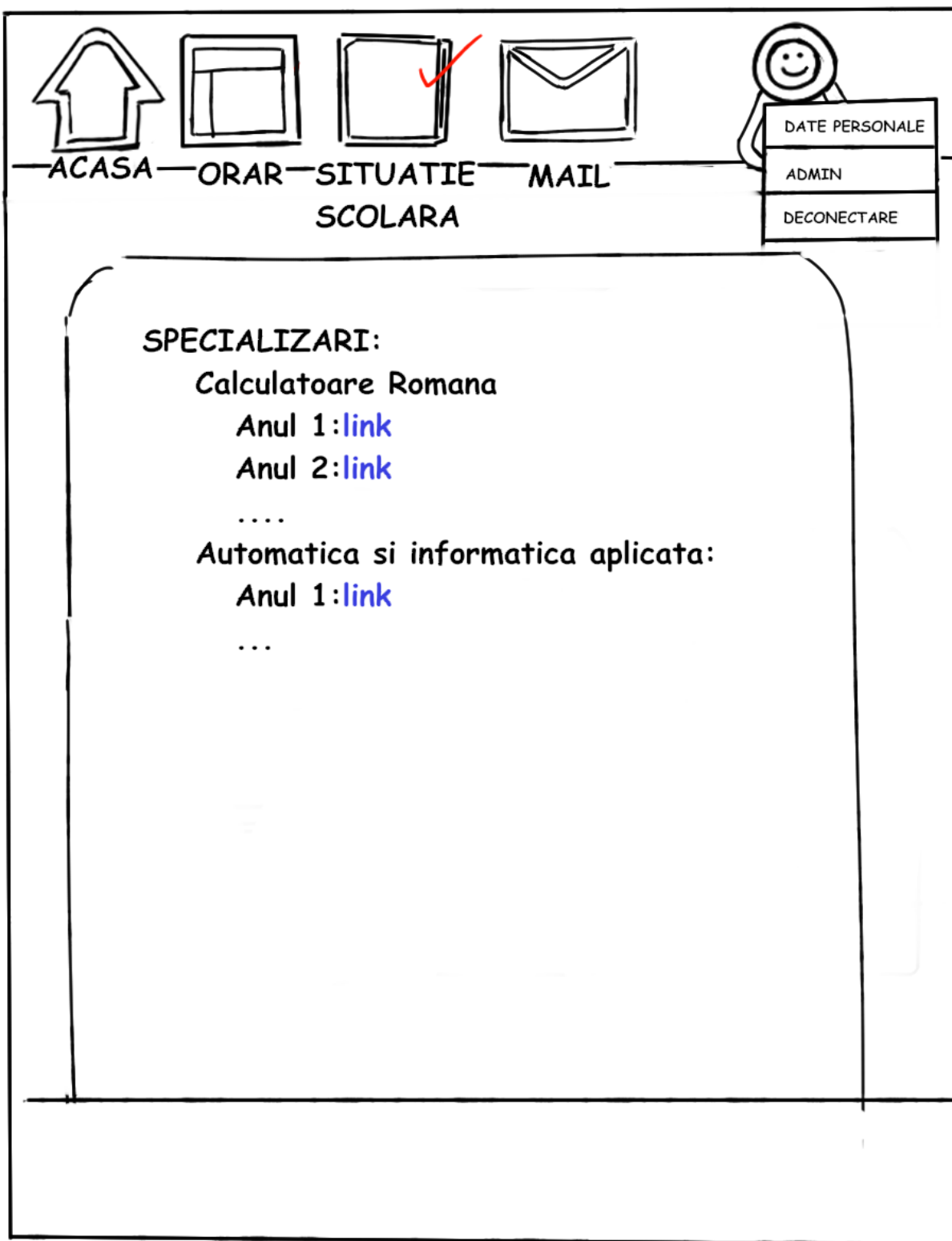
ACASA — ORAR — SITUATIE — MAIL —  
SCOLARA

DATE PERSONALE ✓

ADMIN

DECONNECTARE

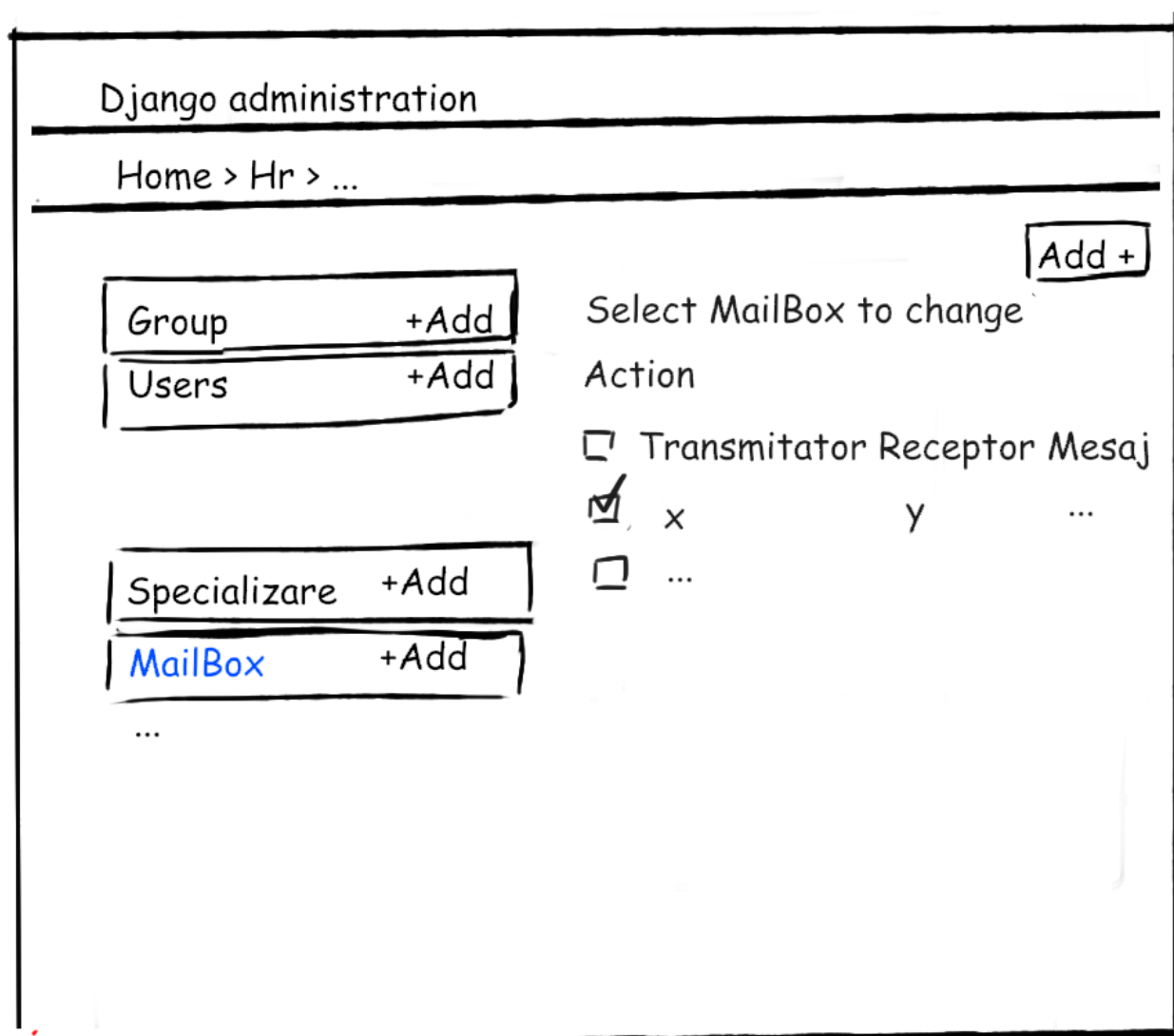
Cont institutional	...@ucv.ro
CNP	99999999999999
Nume	Nume Initiala Prenume
Sex	Masculin/Feminin
Nationalitate	...
Cetatenie	Roman
Data Nasterii	22/09/2001
Telefon	0799999999
Adresa	Romania





A hand-drawn login form with a black border. It contains two input fields: 'Username' and 'Parola'. Below these fields is a 'Log in' button.





### 3.2 Interfele Hardware

Nu se aplica.

### **3.3 Interfetele Software**

Comunicarea intre baza de date si server se realizeaza prin intermediul Python. Django este un framework de tip Python care face comunicarea mai usoara intre paginile Web si asigura intergritatea de transfer de date.

Django respecta urmatorul sablon de proiectare:

- Model: Data pe care doriti sa o reprezentati, uzual datele din baza de date
- Vedere: Un handler pentru cereri care returneaza sablonul si continutul relevant, bazat pe cererea utilizatorului
- Sablon: Un fisier text(precum un fisier HTML) ce contine aspectul paginii web, alaturi de logica afisarii datelor

Django deschide o linie de comunicare cu baza de date(MySQL) cand se indeplineste prima interogare.

Django este un framework web ce necesita un server de date pentru a putea opera.

Interfata folosita in realizarea pachetului software este WSGI:

- WSGI este modalitatea standard de comunicare intre servere web si aplicatii in Python

De asemenea django ofera si posibilitatea de a naviga intre diferite pagini web. Cand un utilizator cere un URL, Django decide ce vedere sa foloseasca. Aceasta operatie se realizeaza printr-un fisier denumit "urls.py".

### **3.4 Interfata Comunicatiilor**

Pachetul software se va folosi de framework-ul Django, alaturi de o baza de date MySQL. Limbajul folosit este Python.

## **4. Caracteristicile sistemului**

### **4.1 Functia de mesagerie**

#### **4.1.1 Descriere si prioritate**

Aceasta functie permite trimiterea si primirea de mailurilor catre diferiti utilizatori, cat si crearea de grupuri de comunicare. Mail-urile vin sub forma de mesaje.

Prioritate acestei functii este ridicata.

#### **4.1.2 Stimul/Secvente de raspuns**

Input:

- Adresa receptor;
- Subiect mail;
- Corp mail

Output:

- Mail(mesajul);

#### 4.1.3 Cerinte functionale

Sistemul de mesagerie va permite trimiterea de mail-uri. Mail-ul este structurat prin destinatar, subiect si context.. In cazul neadaugarii mesajului catre trimitere acesta va fi transmis ca eroare in log.

- REQ-1: Utilizatorii de tip profesor pot trimite mail-uri tuturor celorlalti utilizatori
- REQ-2: Utilizatorii de tip student pot trimite mail-uri tuturor celorlalti utilizatori
- REQ-3: Utilizatorii de tip admin(secretariat) pot trimite mail-uri tuturor celorlalti utilizatori
- REQ-4: Utilizatorii de tip admin(secretariat) pot crea grupuri de comunicare pentru ceilalti utilizatori
- REQ-5: Utilizatorii de tip profesor pot crea grupuri de comunicare pentru pentru ceilalti utilizatori

## 4.2 Functia gestionare orar

#### 4.2.1 Descriere si prioritate(Descriere si prioritate)

Aceasta functie permite vizualizarea orarului specializarilor facultatii, cat si posibilitatea modificarii acestuia prin intermediul secreteariatului. De asemenea acesta poate fi salvat sub forma unui raport in format PDF.

Prioritatea acestei functii este de ridicata.

#### 4.2.2 Stimul/Secvente de raspuns

Input:

- Specializarea;
- An de studiu
- Materii;
- Profesori titulari;

Output:

- Orar(format PDF);

#### 4.2.3 Cerinte functionale

Orarul va fi completat de catre secretariat prin intermediul unui tabel predefinit, caruia i se va atribui o durata, materie si profesor, care se poate selecta din lista profesorilor definiti pentru acel an si materie. Acesta va fi vizualizat atat de profesori, cat si de studenti insa acestia nu vor avea posibilitatea sa il modifice. In cazul in care un orar a fost adaugat, se va transmite in log.

- REQ-1: Utilizatorii de tip profesor isi pot vizualiza orarul;
- REQ-2: Utilizatorii de tip student isi pot vizualiza orarul;
- REQ-3: Utilizatorii de tip profesor pot vizualiza orarul studentilor;
- REQ-4: Utilizatorii de tip profesor pot preda la mai multe materii;
- REQ-5: Utilizatorii de tip profesor pot preda la mai multi ani de studiu;
- REQ-6: Utilizatorii de tip student pot vizualiza orarul altor ani de studii;
  
- REQ-8: Utilizatorii de tip student pot sa vizualizeze materiile a 2 specializari simultan, in aceasi pagina;
- REQ-9: Utilizatorii de tip secretariat pot crea si modifica orare;
- REQ-10: Utilizatorii de tip secretariat pot adauga planuri de invatamant;
- REQ-11: Orarul va avea pagina sa proprie;

## **4.3 Functia centralizarii notelor**

### **4.3.1 Descriere si prioritate**

Aceasta functie permite adaugarea si modificarea notelor de catre profesori si vizualizarea acestora de catre studenti.

Prioritate acestei functii este ridicata.

### **4.3.2 Stimul/Secvente de raspuns**

Input:

- Nota;
- Materia;
- Profesor titular;
- Studentul evaluat;

Output:

- Situatia scolara a studentului;

### **4.3.3 Cerinte functionale**

Dupa finalizare notarii, se va transmite in log, ca a fost transmis cu succes. In cadrul modificarii, este necesara interventia secretariatului, aceasta informatie va fi transmisa in log.

Studentul va putea sa isi vizualizeze situatia scolara, indiferent de sectiile in care se afla, avand posibilitatea generarii unui raport sub format PDF.

- REQ-1: Utilizatorii de tip profesor pot sa noteze studentii;
- REQ-2: Utilizatorii de tip student isi pot vizualiza notele;
- REQ-3: Utilizatorii de tip secretariat pot genera centralizatorul de note;
- REQ-4: Utilizatorii de tip profesor pot genera centralizatorul de note;

- REQ-5: Utilizatorii de tip student isi pot salva situatia scolara in format offline sub format PDF;
- REQ-6: Utilizatorii de tip secretariat au permsiunea modifacarii notelor;
- REQ-7: Notarea se va face prin intermediul unei tabele ce contine numele studentilor si nota in sine;
- REQ-8: Profesorii vor avea posibilitatea selectarii notelor ca numere intregi de la 1 la 10, precum si calificative: „admis” sau „absent”;
- REQ-9: Dupa notare, actiunea este trecuta in log;

## **4.4 Functia de organizare interna.**

### **4.4.1 Descriere si prioritate**

Aceasta functie permite definirea structurii universitatii prin adaugarea de noi utilizatori in cadrul:

- secretariatului (care se ocupa cu gestionarea si adaugarea sau stergerea specializarilor de studiu si materiilor de studiu );
- profesorilor (care predau);
- studentilor (care studieaza la momentul respectiv).

Prioritatea acestei functii este extrem de ridicata.

### **4.4.2 Stimul/Secvente de raspuns**

Input:

- Specializare de studiu;
- Profesori titulari;
- Studenti;

Output:

- Structura anului universitar;

### **4.4.3 Cerinte functionale**

Aceasta actiune se va realiza prin intermediul framework-ului Django, care ne permite adaugare, stergerea si modificarea datelor, prin introducerea datelor in format brut. Odata adaugat, stearsa sau modificata o data se va transmite in log.

- REQ-1: Utilizatorii de tip secretariat pot sa adauge materii;
- REQ-2: Utilizatorii de tip secretariat pot sa elimine materii;
- REQ-3: Utilizatorii de tip secretariat pot sa modifice materii;
- REQ-4: Utilizatorii de tip secretariat pot sa adauge specializari;
- REQ-5: Utilizatorii de tip secretariat pot sa elimine specializari;
- REQ-6: Utilizatorii de tip secretariat pot sa modifice specializari;
- REQ-7: Utilizatorii de tip secretariat pot sa adauge utilizatori;
- REQ-8: Utilizatorii de tip secretariat pot sa elimine utilizatori;
- REQ-9: Utilizatorii de tip secretariat pot sa modifice utilizatori;

- REQ10: Utilizatorii de tip secretariat pot sa adauge grupe;
- REQ11: Utilizatorii de tip secretariat pot sa elimine grupe;
- REQ12: Utilizatorii de tip secretariat pot sa modifice grupe;
- REQ13: Utilizatorii de tip secretariat pot sa adauge grupe;
- REQ14: Utilizatorii de tip secretariat pot sa elimine semestre;
- REQ15: Utilizatorii de tip secretariat pot sa modifice semestre;
- REQ16: Utilizatorii de tip secretariat pot sa adauge semestre;
- REQ17: Utilizatorii de tip secretariat pot sa adauge numarul de credite alocate materiilor;
- REQ18: Utilizatorii de tip secretariat pot sa stearga numarul de credite alocate materiilor;
- REQ19: Utilizatorii de tip secretariat pot sa modifice numarul de credite alocate materiilor;
- REQ20: Utilizatorii de tip secretariat pot sa adauge modalitati de desfasurare a orelor sub forma de noutate;
- REQ21: Utilizatorii de tip secretariat pot sa stearga modalitati de desfasurare a orelor sub forma de noutate;
- REQ22: Utilizatorii de tip secretariat pot sa modifice modalitati de desfasurare a orelor sub forma de notat;
- REQ-23: Utilizatorii de tip secretariat pot sa adauge utilizatori de tip student sa studieze in cadrul mai multor specializari simultan;

## **4.5 Functia avizier**

### **4.2.1 Descriere si prioritate**

Functia aceasta permite vizualizarea stirilor de interes pentru studenti, precum modalitatile de evaluare si modificarea orarelor.

Aprobarea si stergerea acestor anunturi se face de catre secretariat.

Prioritatea acestei functii este scazuta.

### **4.2.2 Stimul/Secvente de raspuns**

Input:

- Anunturi legate de modalitati de evaluare;
- Anunturi legate de modificarea orarelor;

Output:

- Afisare pe pagina avizier;

### **4.2.3 Cerinte functionale**

Pagina va permite vizualizarea anunturilor. In momentul in care s-a adaugat un anunt, modificat sau sters, se va afisa in log.

REQ-1: Utilizatorii de tip profesor pot trimite propuneri de modalitati de evaluare;

REQ-2: Utilizatorii de tip student pot vizualiza modalitatile de evaluare.

## **4.6 Functia istoricul documentelor**

### **4.6.1 Descriere si prioritate**

Functia permite generarea si listarea de documente din sesiuni anterioare.

### **4.6.2 Stimul/Secvente de raspuns**

Input:

- Orare;
- Situatie scolara;

Output:

- Generarea documentelor;

### **4.6.3 Cerinte functionale**

Secretariatul va putea accesa datele anterioare si va avea posibilitatea salvarii lor, fie ca si date dintr-o baza de date, fie ca raport.

REQ-1: Utilizatorii de tip secretariat pot salva foile matricole in mod offline, in formatul bazei de date;

REQ-2: Utilizatorii de tip secretariat pot salva foile matricole in mod offline, in format PDF;

REQ-3: Utilizatorii de tip secretariat pot salva orarele precedente in mod offline, in formatul bazei de date

REQ-4: Utilizatorii de tip secretariat pot salva orarele precedente in mod offline, in format PDF

## **5. Alte Cerinte Nefunctionale**

### **5.1 Cerinte de performanta**

- Timpul de acces server asteptat: 1 secunda
- Timpul de primire raspuns in urma unei interogari: 1 secunda
- Timpul maxim de acces server: 3 secunda
- Timp de asteptare pentru adaugare date in baza de date cu succes: 1 secunda

### **5.2 Cerinte de siguranta**

Nu se aplica.

### **5.3 Cerinte de securitate**

- REQ-1: CNP-ul utilizatorului va fi salvata criptat, prin intermediul framework-ului Django;
- REQ-2: Parola va fi salvata criptat, prin intermediul framework-ului Django.
- REQ-3: Parola este unica.
- REQ-4: Parola va fi transmisa utilizatorului (identificat prin CNP) odata cu adaugarea ei in baza de date.

### **5.4 Atributele calitatii software**

#### **1. Caracterul # nu va fi folosita in cadrul unui comentariu**

Python nu suporta imbricarea comentariilor, desi exista compilatoare care suporta ca si o extensie a limbajului. Un comentariu incepe cu # si continua pana la sfarsitul liniei. Orice # care apare in cadrul comentariu este o incalcare ale acestei reguli. De exemplu:

#ceva comentariu #abatere de la regula

#### **2. Toate bibliotecile utilizate in codul sursa trebuie sa fie scrise in conformitate cu prevederile acestui document si trebuie sa fi fost supuse la validari corespunzatoare.**

Aceasta regula se refera la orice biblioteci utilizate in codul sursa, care pot include biblioteci standard furnizate cu compilatorul sau alte biblioteci preluate prin pip.

#### **3. Toate variabilele automate trebuie sa le fi atribuit o valoare inainte de a fi utilizate**

Intentia acestei reguli este ca toate variabilele sa fi fost scrise inainte de a fi citite. Acest lucru nu necesita neaparat initializarea la declarare.

#### **4. Variabilele intr-o functie nu vor folosi acelasi nume ca variabilele din exteriorul functiei, astfel ascunzand acea variabila.**

Exact ca si in alte limbaje, variabilele locale ar trebui sa aiba un nume diferit de variabilele globale.

#### **5. Functiile trebuie denumite sugestiv, iar scopul lor trebuie sa fie unic.**

Scopul unei functii trebuie sa fie unicat, iar numele ei trebuie sa fie sugestiv pentru o usoara intelegere a codului.

#### **6. Un obiect sau o functie va fi declarata doar intr-un singur fisier**

Nu vom declara obiecte cu acelasi nume in multiple fisiere.

#### **7. Pentru rularea corecta a functiilor, indentarea trebuie respectata**

In Python, indentarea ofera apartenenta la o functie. Nerespectarea indentarii duce la cod scris gresit.

#### **8. Nu trebuie să se utilizeze comportamentul nedefinit sau nespecificat.**

Python, spre exemplu, nu defineste valoarea variabilelor neinitializate, deci comportamentul acestora este nedefinit.



Inercarea modificării unui dicționar în timpul iterării prin el duce la un comportament nespecificat.

9. **Mai multe compilatoare și/sau limbaje trebuie utilizate doar dacă există un standard de interfață definit comun pentru codul obiect, la care limbajele/compilatoarele/asamblorii se conformează.**

Un exemplu de standard de interfață definit comun pentru codul obiect în Python este ABI (Application Binary Interface), care permite interoperabilitatea între limbajul Python și alte limbaje de programare, cum ar fi C, C++ sau Fortran.

10. **În Python, stilul recomandat pentru comentarii este să se folosească simbolul "#" pentru comentarii pe o singură linie.**

## **5.5 Reguli de afacere**

Nu se aplica.

## **6. Alte Cerinte**

Nu se aplica.

### **Anexa A: Glosar**

Nu se aplica.

### **Anexa B: Modele de analiza**

Nu se aplica.

### **Anexa C: Lista de stabilit**

Nu se aplica.