



SimpleMail

APLICAȚIE E-MAIL

Proiect Inginerie Software

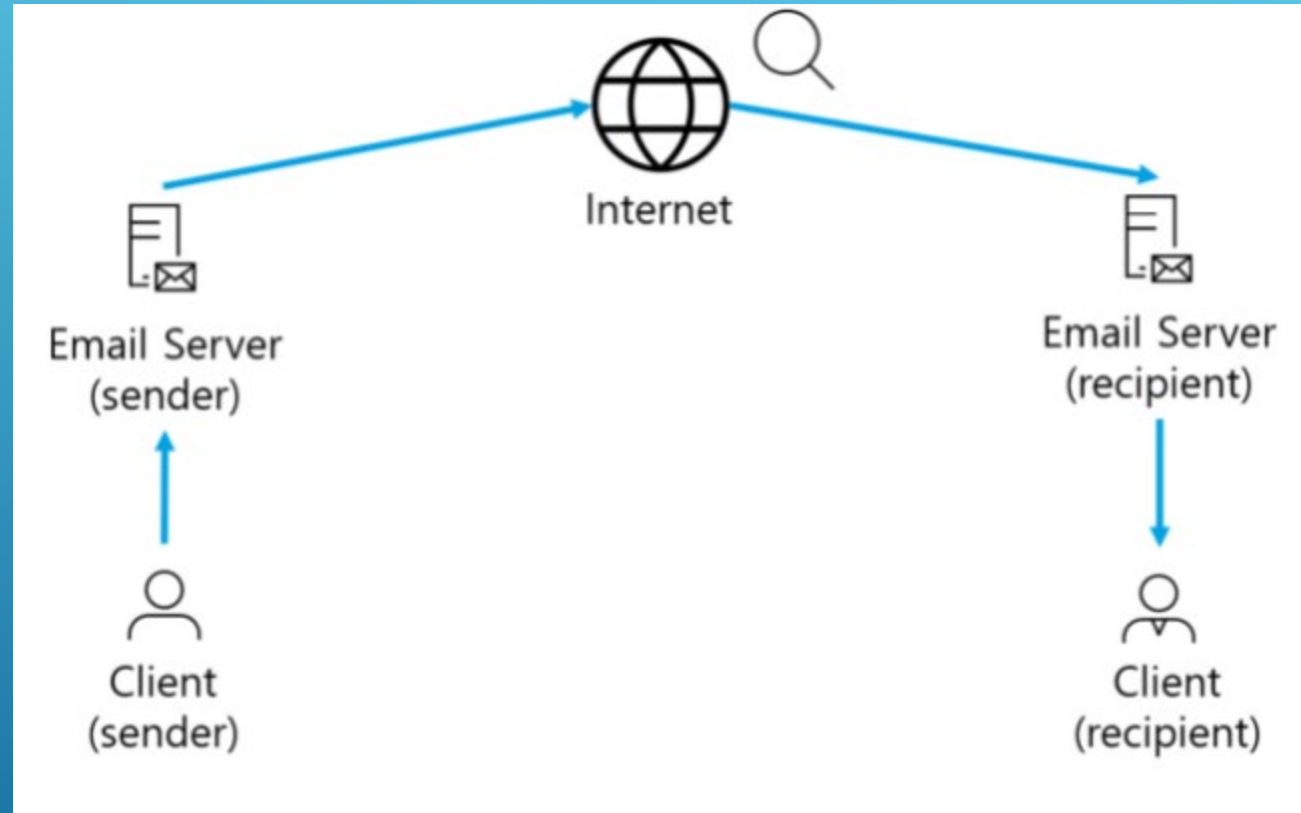
Studenti: Pucani Liviu Cătălin

Tanul Gabriel-Ștefan

Profesor coordonator: ing. Mitrea Delia-Alexandrina

- ❖ 1. Specificații și analiza sistemului
- ❖ 2. Precizarea limbajului de programare ales (plus motivație), a sistemului de operare sub care se face implementarea, a cerințelor hardware
- ❖ 3. Fundamente teoretice
- ❖ 4. Proiectarea aplicației
 - ❖ 4.1. Descrierea arhitecturii sistemului (plus diagrama)
 - ❖ 4.2. Identificarea funcționalităților aplicației (diagrama de cazuri de utilizare)
 - ❖ 4.3. Diagrama de clase
 - ❖ 4.4. Diagrame de interacțiune: activitate, secvență, colaborare
 - ❖ 4.5. Detalii de implementare: diagrama bazei de date, diagrama de clase cu attribute și metode, diagrama de module și componente
 - ❖ 4.6. Cazuri de testare
- ❖ 5. JUnit Testing
- ❖ 6. Manual de instalare și utilizare
- ❖ 7. Concluzii și dezvoltări ulterioare
- ❖ 8. Bibliografie

1. SPECIFICAȚII ȘI ANALIZA SISTEMULUI



2. PRECIZAREA LIMBAJULUI DE PROGRAMARE ALES, A SISTEMULUI DE OPERARE SUB CARE SE FACE IMPLEMENTAREA, A CERINȚELOR HARDWARE

- Limbaj de programare ales: **JAVA**
scris în **Intelij IDEA** în **Maven Project**
- Framework folosit pentru crearea celor 2
puncte de acces: **JAVA Spring**
- Bază de date scrisă în **MongoDB**
- Limbaje scripturale pentru manipularea
datelor și a părții front-end: **Thymeleaf**,
Javascript, **Bootstrap**
- Sistem de operare: **Windows 10**

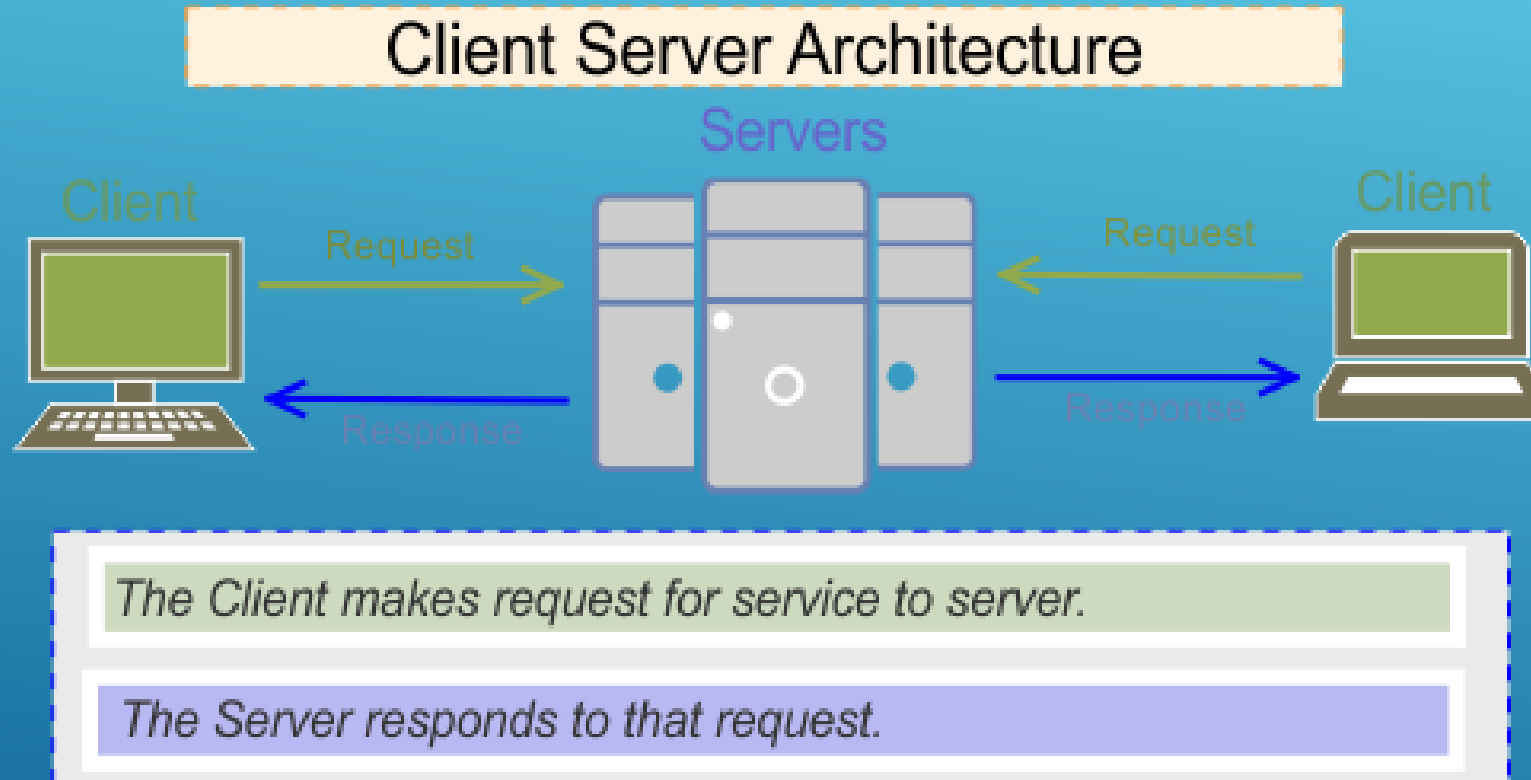


3. FUNDAMENTE TEORETICE

- Arhitectura client/server este o arhitectură de rețea bidirecțională în care fiecare calculator sau proces din rețea este un client sau un server.
- Calculatorul client este acela care interacționează cu un utilizator și care în majoritatea tipurilor de arhitecturi client/server are două sarcini: logica prezentării (interfața cu utilizatorul) și logica aplicației (afacerii).
- Calculatorul server poate fi orice calculator de birou puternic, super-servere specializate al căror rol într-o rețea este acela de a furniza servicii și resurse utilizatorilor. Funcția de bază a acestuia este responsabilitatea administrării accesului la baza de date.

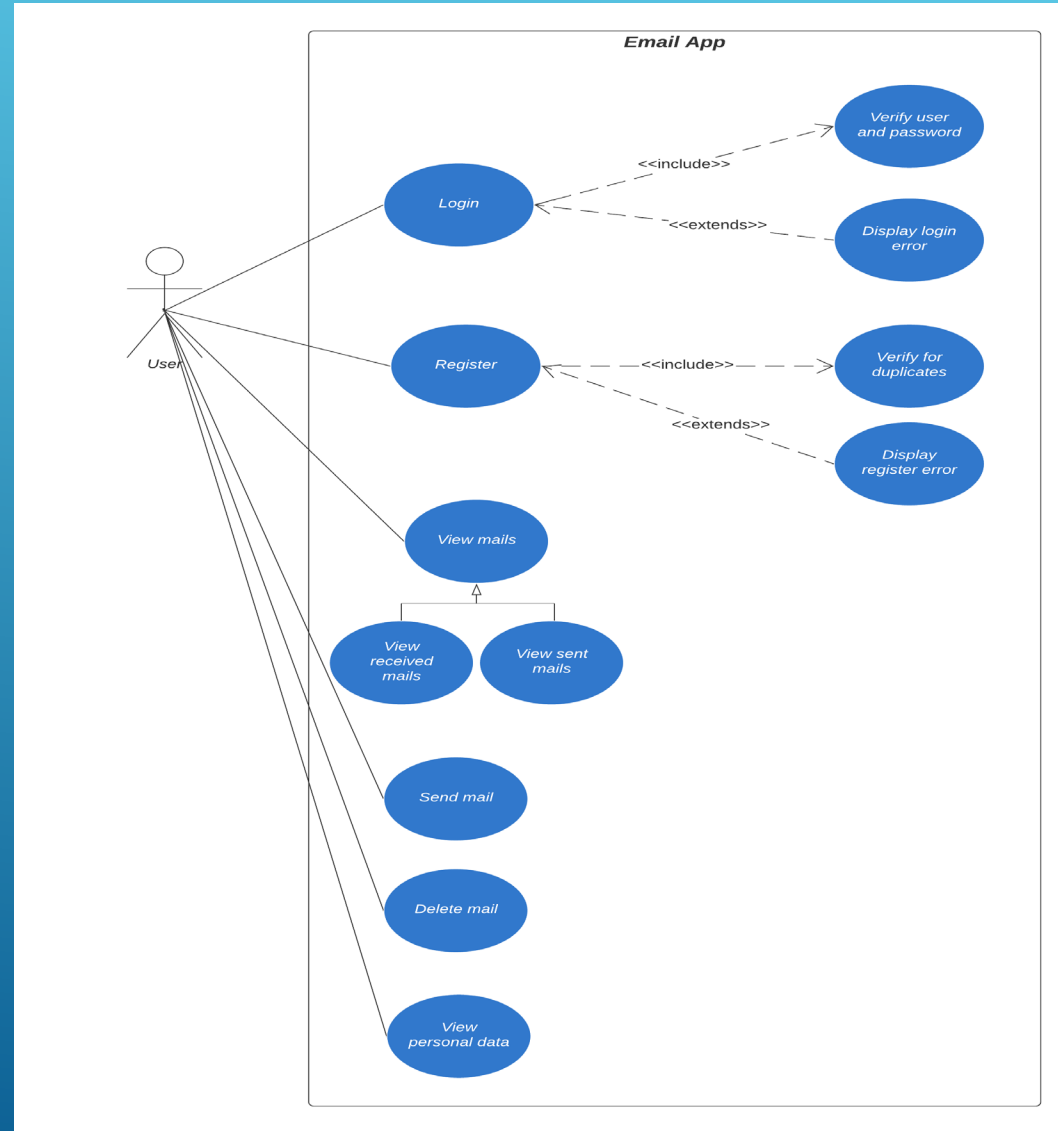
4. PROIECTAREA APLICAȚIEI

4.1. DESCRIEREA ARHITECTURII SISTEMULUI



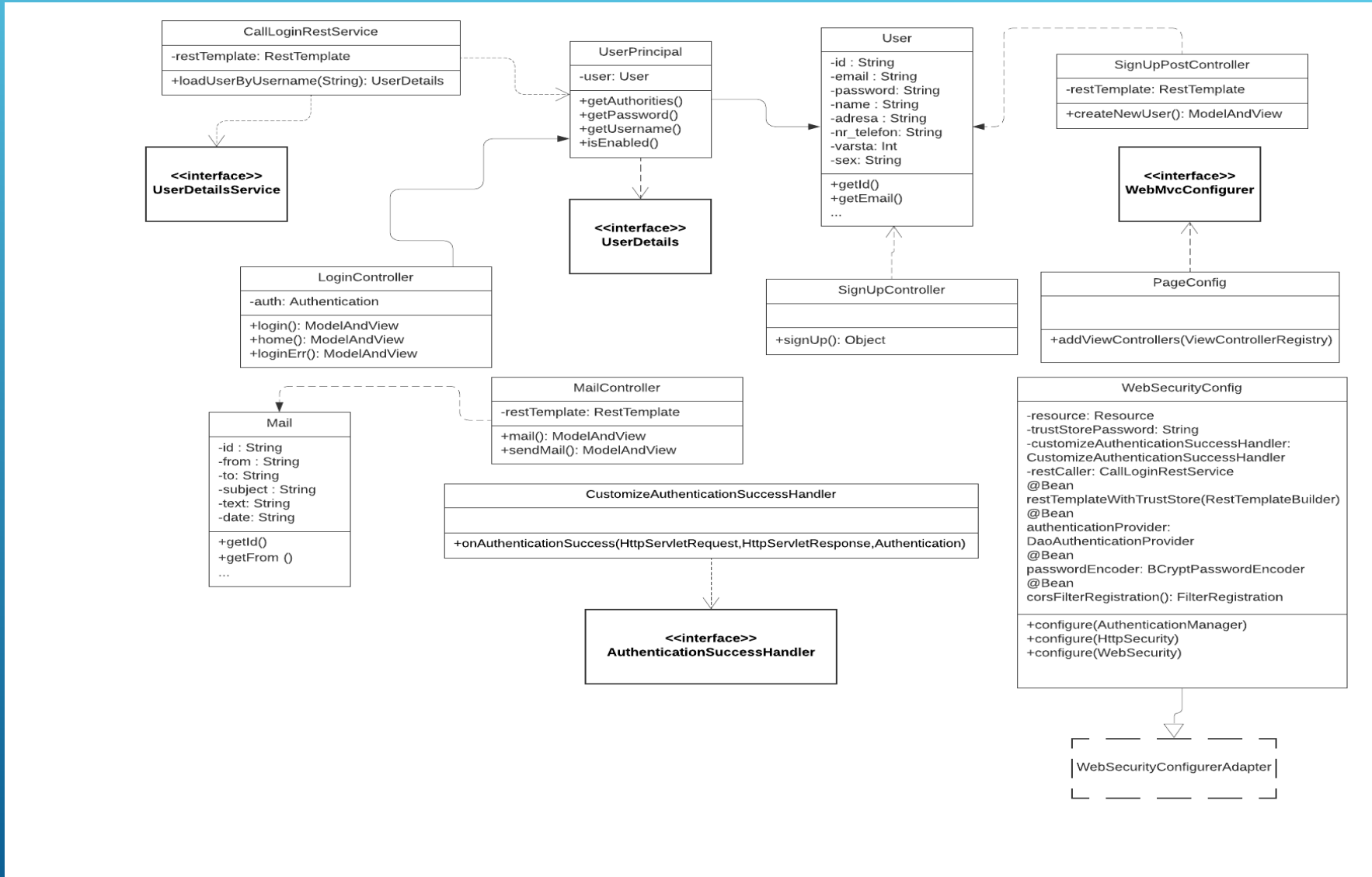
4. PROIECTAREA APLICAȚIEI

4.2.IDENTIFICAREA FUNCȚIONALITĂȚILOR APLICAȚIEI



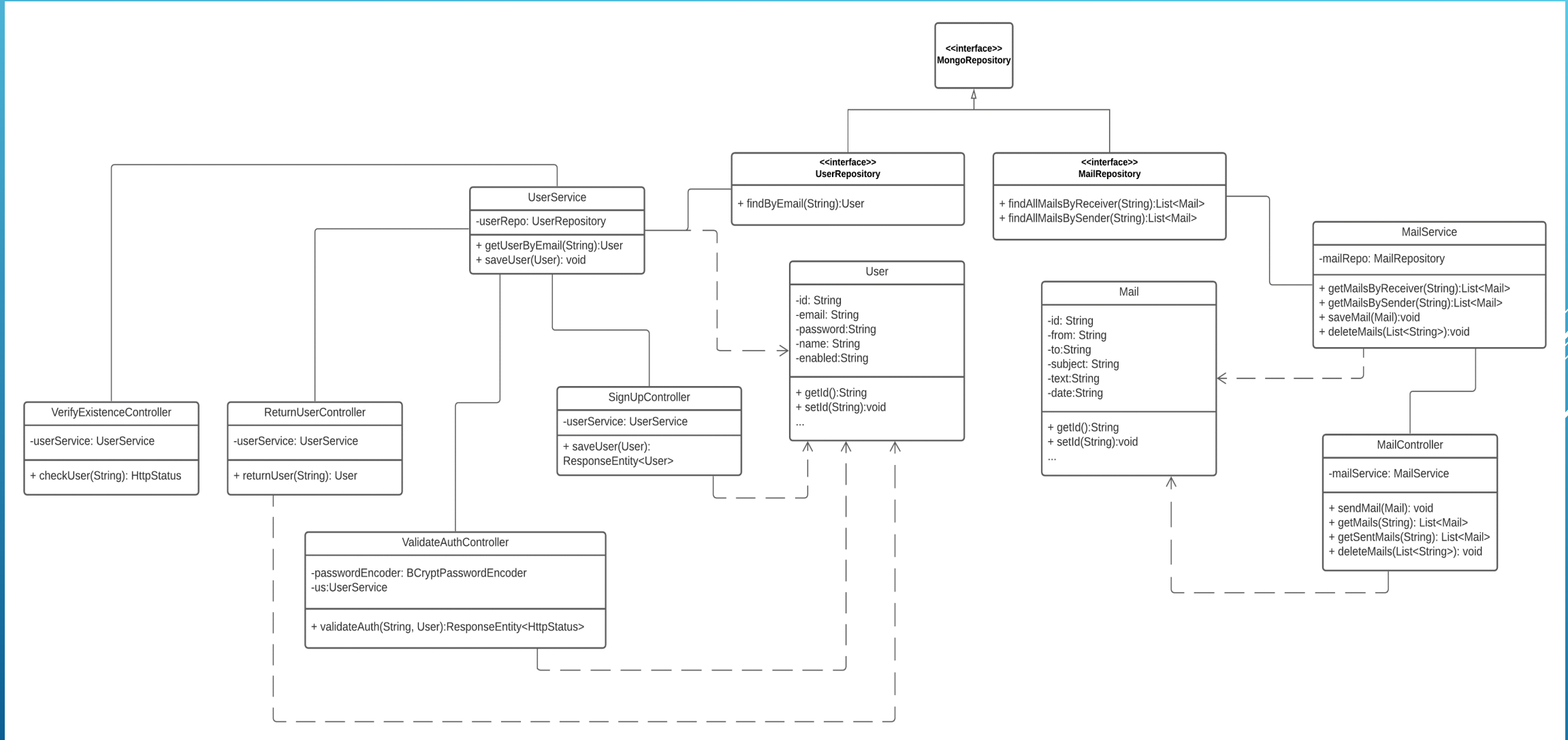
4. PROIECTAREA APLICAȚIEI

4.3. DIAGRAMA DE CLASE – CLIENT SIDE



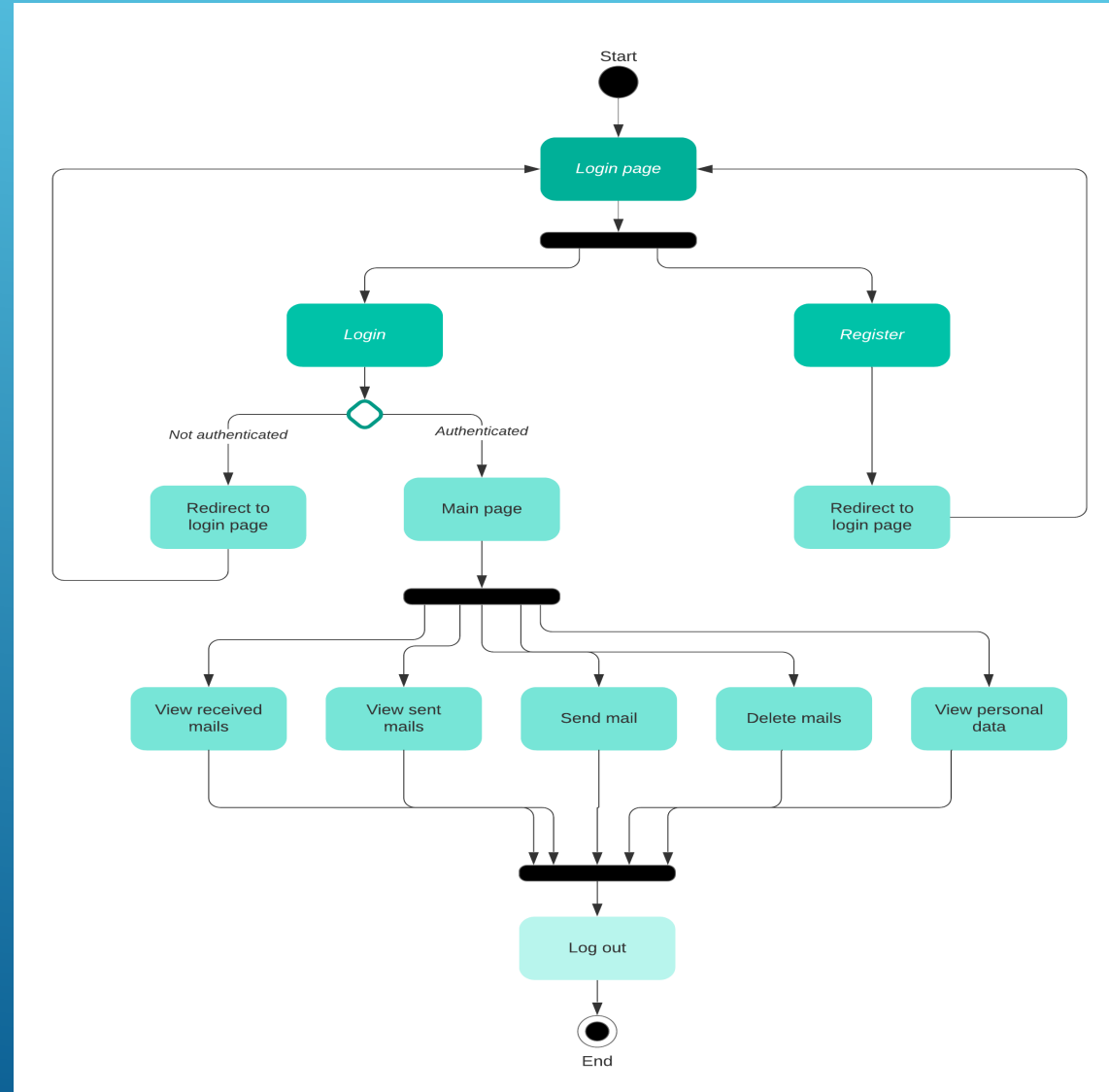
4. PROIECTAREA APLICAȚIEI

4.3. DIAGRAMA DE CLASE – SERVER SIDE



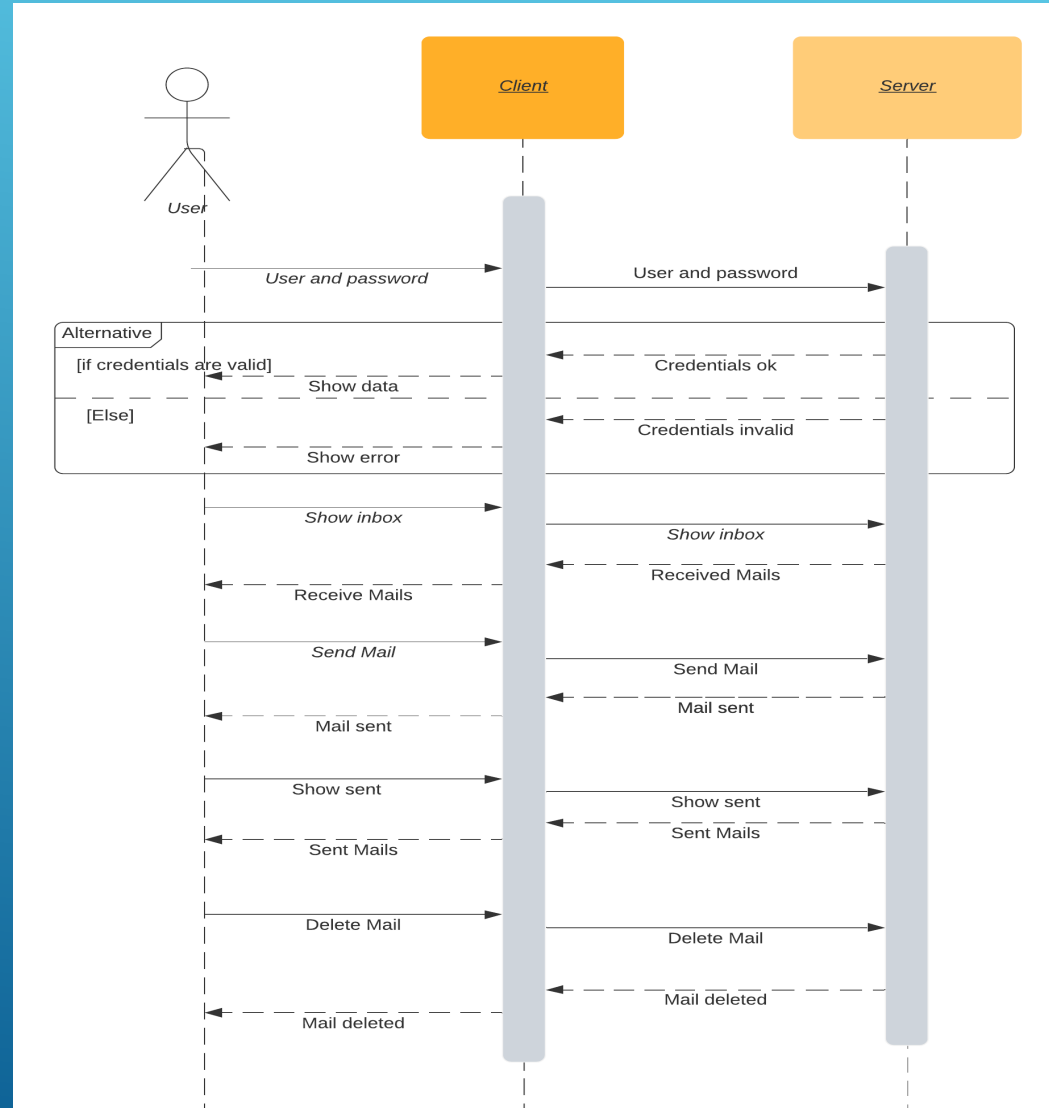
4. PROIECTAREA APLICAȚIEI

4.4. DIAGRAMA DE ACTIVITATE



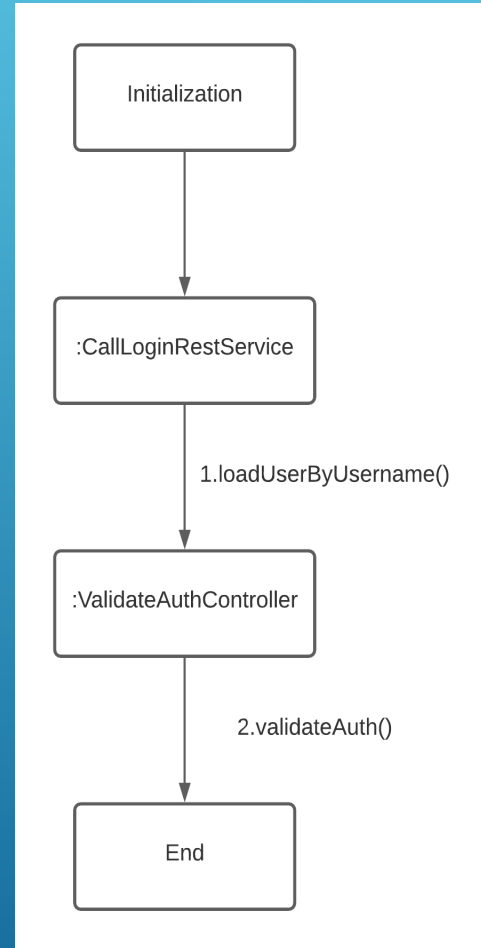
4. PROIECTAREA APLICAȚIEI

4.4. DIAGRAMA DE SECVENȚE



4. PROIECTAREA APLICAȚIEI

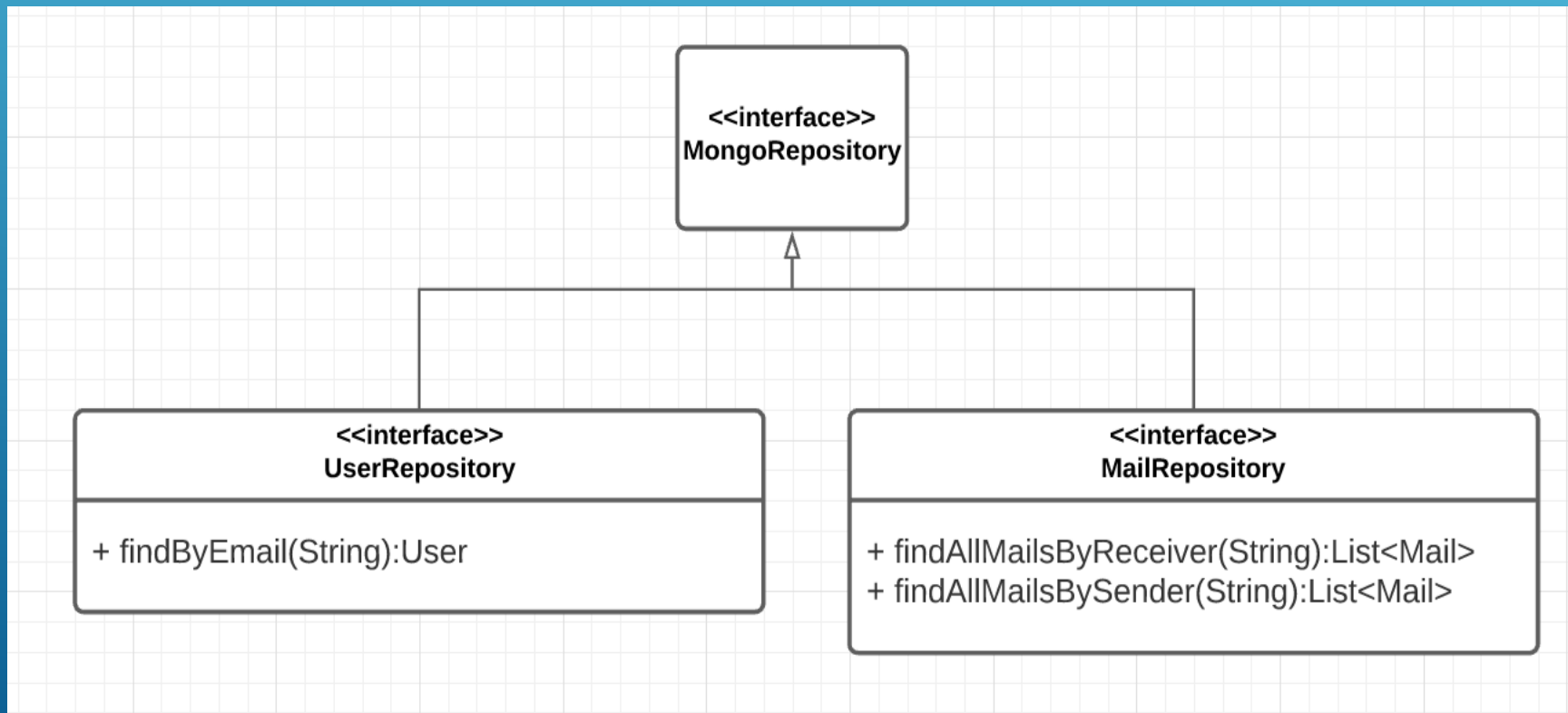
4.4. DIAGRAMA DE COLABORARE



4. PROIECTAREA APLICAȚIEI

4.5. DIAGRAMA DE CLASE – BAZĂ DE DATE

Baza de date se accesează folosind repository-ul `MongoRepository` predefinit care este moștenit de interfețele `UserRepository` și `MailRepository` manipulând datele în funcție de clasele arondate fiecărei interfețe.

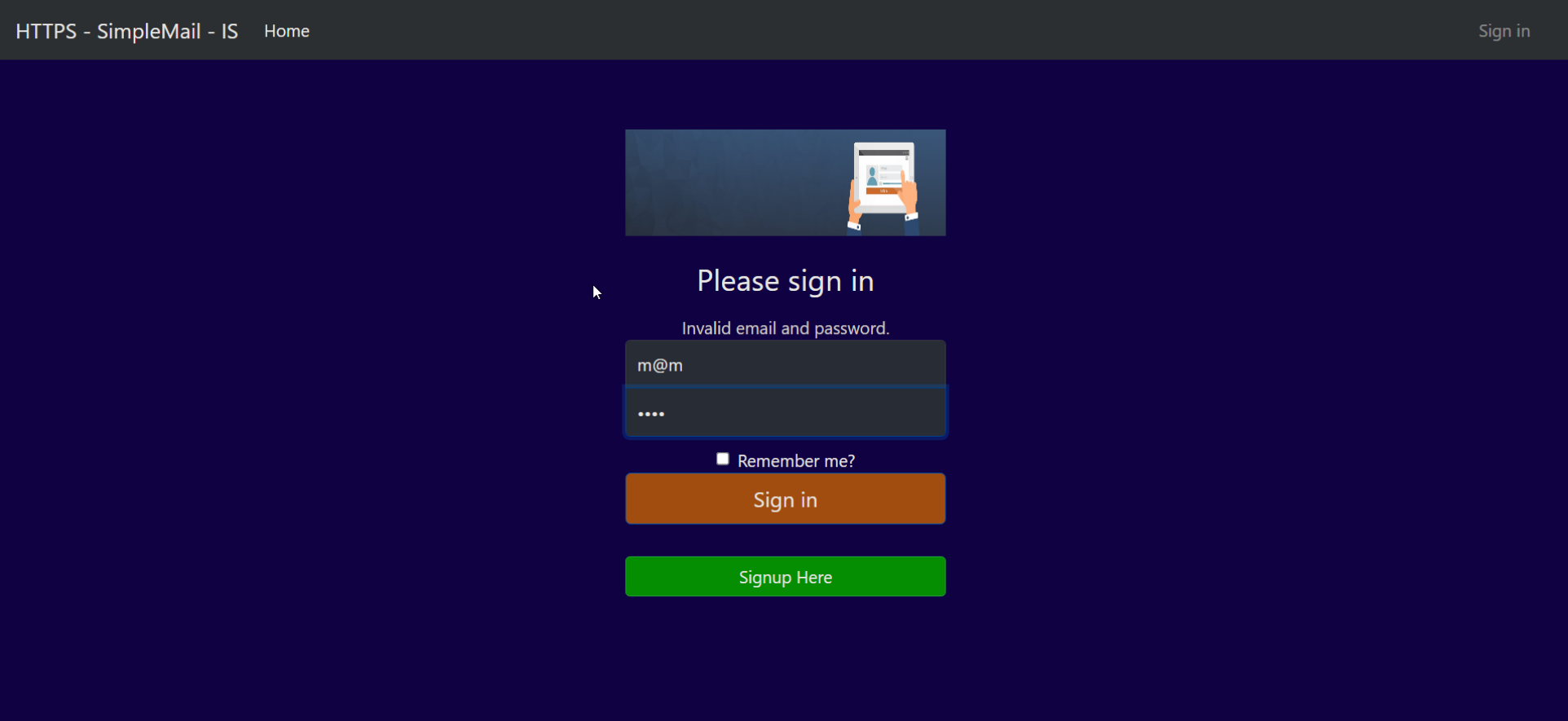


4. PROIECTAREA APLICAȚIEI

4.6. CAZURI DE TESTARE

Am pornit testarea de la partea de login, user-name și parolă incorecte apoi trimiterea și ștergerea unui email.

Afișarea unui mesaj corespunzător în cazul autentificării greșite

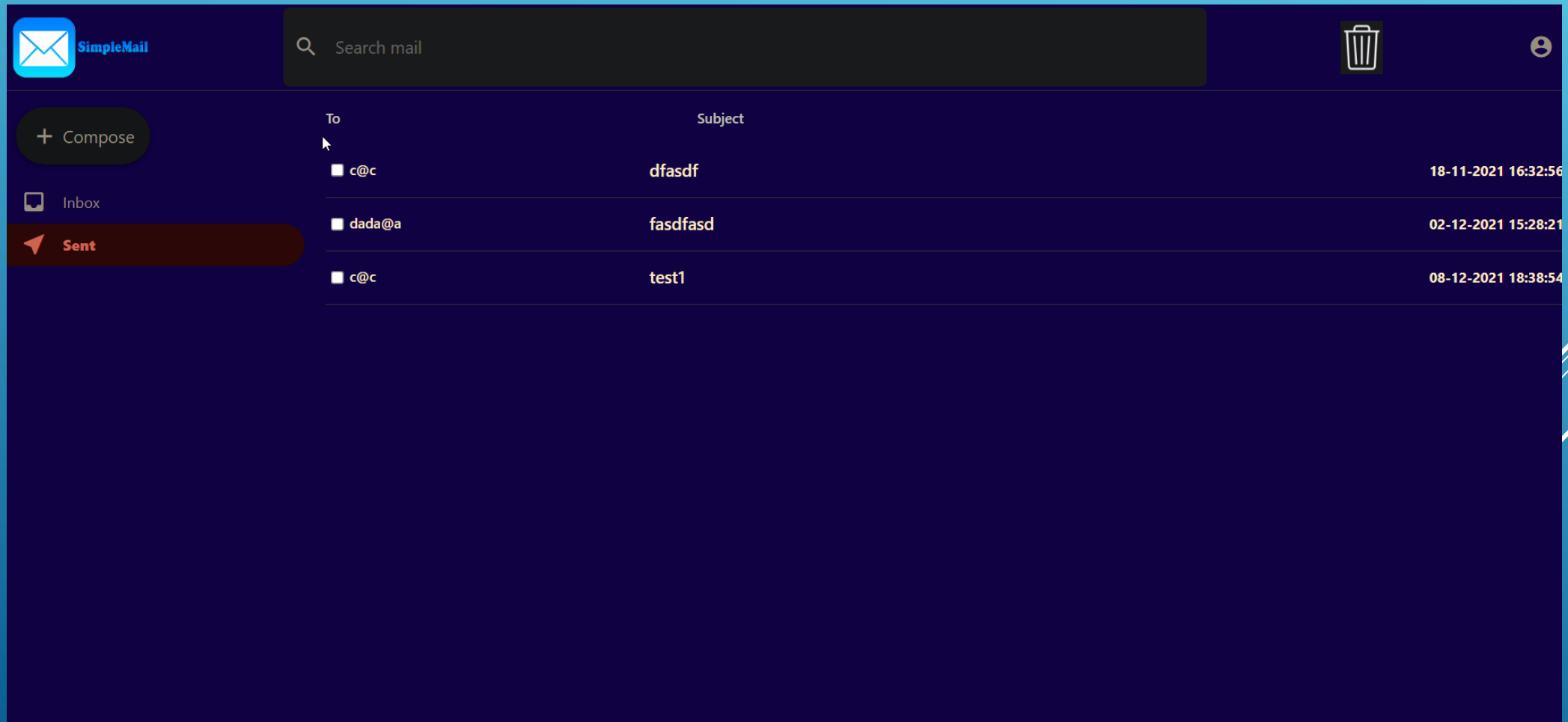


The screenshot displays a web browser window with the address bar showing "HTTPS - SimpleMail - IS" and a "Home" link. In the top right corner, there is a "Sign in" link. The main content area has a dark blue background. At the top center, there is a header image showing a person holding a tablet. Below this, the text "Please sign in" is displayed. Underneath, a message states "Invalid email and password." There are two input fields: the first contains "m@m" and the second contains four dots, indicating a password field. Below the password field is a checkbox labeled "Remember me?". At the bottom of the login section, there are two buttons: an orange "Sign in" button and a green "Signup Here" button.

4. PROIECTAREA APLICAȚIEI

4.6. CAZURI DE TESTARE

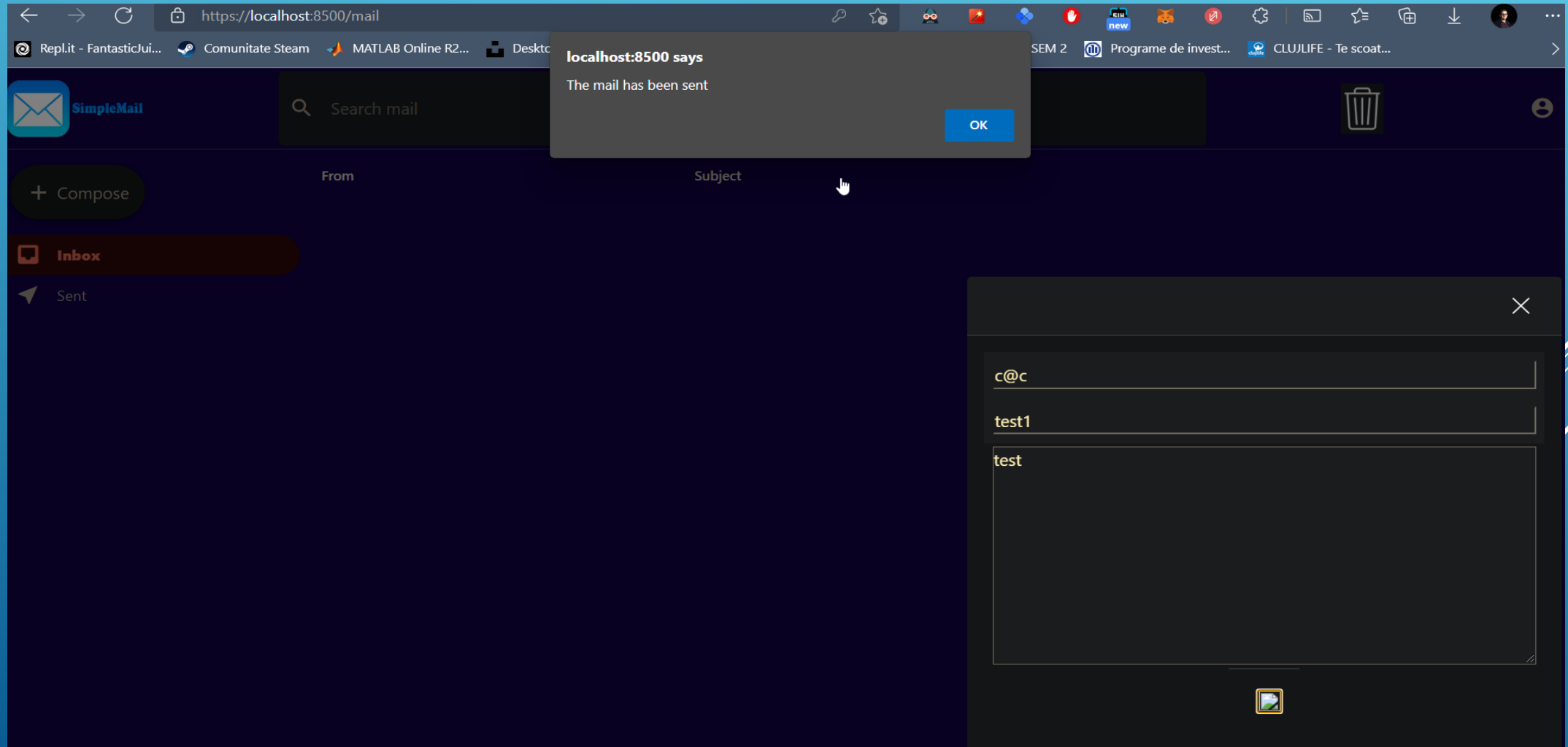
User autentificat, afișarea paginii principale unde este oferit accesul către căsuțele de “inbox” și “trimise”.



4. PROIECTAREA APLICAȚIEI

4.6. CAZURI DE TESTARE

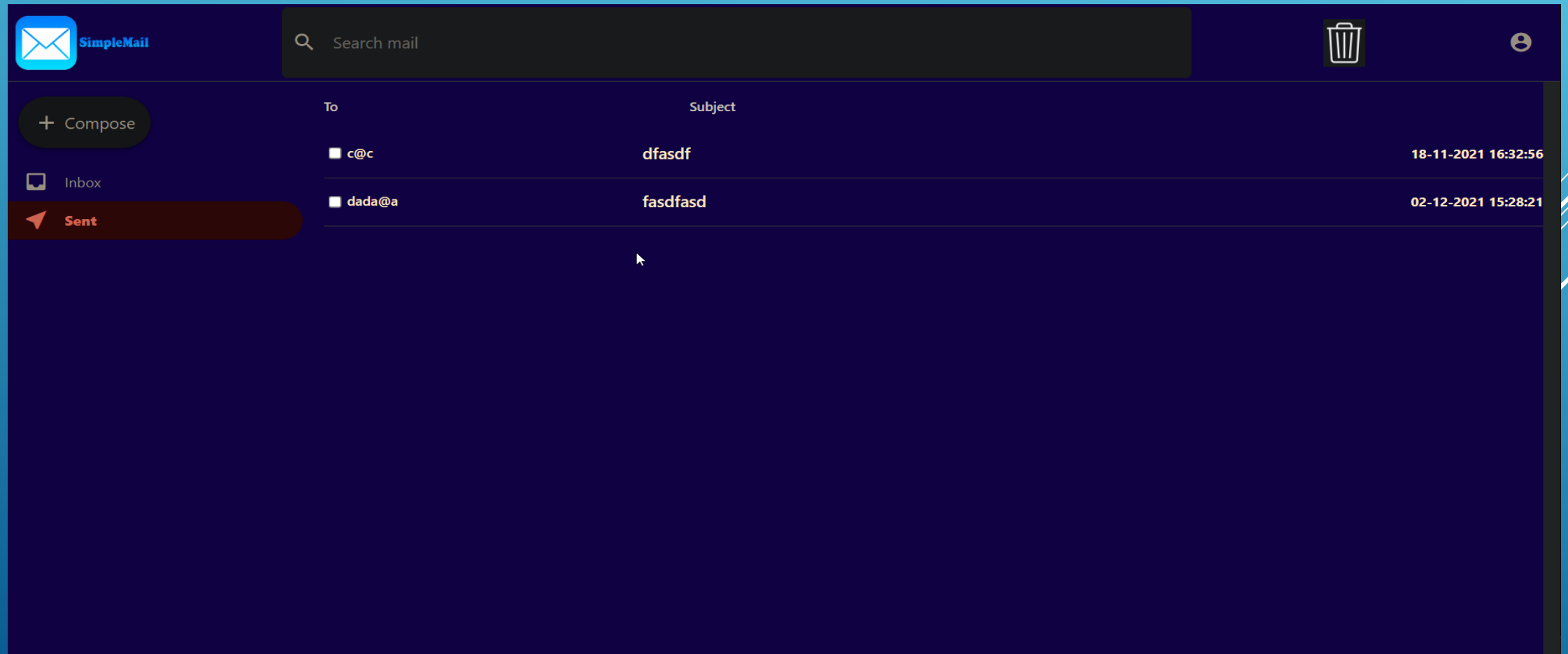
Trimiterea unui e-mail cu succes.



4. PROIECTAREA APLICAȚIEI

4.6. CAZURI DE TESTARE

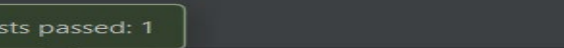
ștergerea e-mailului trimis anterior (test1). A fost selectat din checkbox și apoi șters folosind butonul coșului de gunoi.



5. JUNIT TESTING

```
class ClientTest {
    @Autowired
    private RestTemplate restTemplate;

    /**
     * CHECK IF USER "c2@c" contains an email from "m@m"
     */
    @Test
    void verifyMailExistence() {
        Mail mail=new Mail();
        mail.setFrom("m@m");
        mail.setTo("c2@c");
        ArrayList<HashMap<String,String>> mails = restTemplate.getForObject("https://localhost:8082/getMails/c2@c",ArrayList.class);
        Iterator<HashMap<String,String>> it=mails.iterator();
        String from = null,to=null;
        while(it.hasNext())
        {
            HashMap<String,String> hm=it.next();
            from=hm.get("from");
            to = hm.get("to");
            if(from.equals(mail.getFrom()))
            {
                to.equals(mail.getTo());
                break;
            }
        }
        assertEquals(mail.getFrom(),from);
        assertEquals(mail.getTo(),to);
    }
}
```



Test Results

Test Name	Duration	Status
ClientTest	765 ms	1
verifyMailExistance()	765 ms	1

Tests passed: 1

5. JUNIT TESTING

```
@Autowired
private ReturnUserController returnUserController;

@Autowired
private ValidateAuthController validateAuthController;

@Autowired
private VerifyExistenceController verifyExistenceController;

/**
 * This test verifies if the api link localhost/{email} returns an HTTP status code as "ACCEPTED"
 * if an user tries to register with an email
 * that doesn't exist and "IM_USED" if the email already exists in the database.
 */
@Test
void verifyExistenceTest() {
    HttpStatus httpStatus=verifyExistenceController.checkUser( email: "m@m");
    HttpStatus httpStatus1=verifyExistenceController.checkUser( email: "m@mmm");

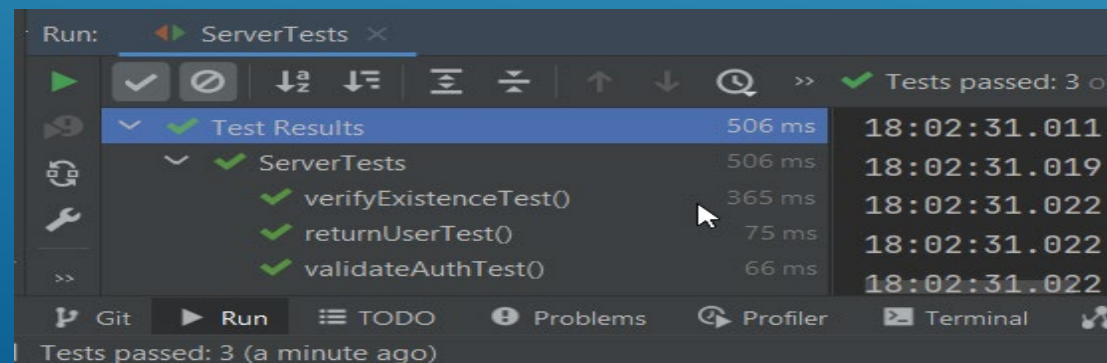
    assertEquals(httpStatus,HttpStatus.IM_USED);
    assertEquals(httpStatus1,HttpStatus.ACCEPTED);
}
```

5. JUNIT TESTING

```
@Test
void returnUserTest() {
    BCryptPasswordEncoder bCryptPasswordEncoder=new BCryptPasswordEncoder();
    User u=new User();
    u.setEmail("m@m");
    u.setPassword("pass");


    User uTest=returnUserController.returnUser( email: "m@m");
    assertEquals(uTest.getEmail(),u.getEmail());
    assertTrue(bCryptPasswordEncoder.matches(u.getPassword(),uTest.getPassword()));
}

/**
 * This test verifies if the api link localhost/checkuser/{email} return an "ACCEPTED" Http status code that will be used for
 * validating the authentication of the client
 */
@Test
void validateAuthTest() {
    User u2=new User();
    u2.setEmail("m@m");
    u2.setPassword("pass");
    ResponseEntity<HttpStatus> uTest=validateAuthController.validateAuth( email: "m@m",u2);
    assertEquals(uTest.getStatusCode(),HttpStatus.ACCEPTED);
}
}
```



6. MANUAL DE INSTALARE SI UTILIZARE

6.1 INREGISTRAREA UNUI CONT



Signup Here

Email

Password

Full-name

Address

Telephone number

Age

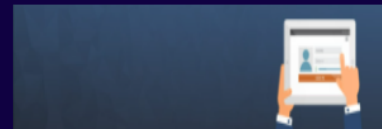
Gender

Sign Up

Sign In

6. MANUAL DE INSTALARE SI UTILIZARE

6.2 AUTENTIFICARE



Please sign in

proiect_IS@simplemail.com

....|

☐ Remember me?

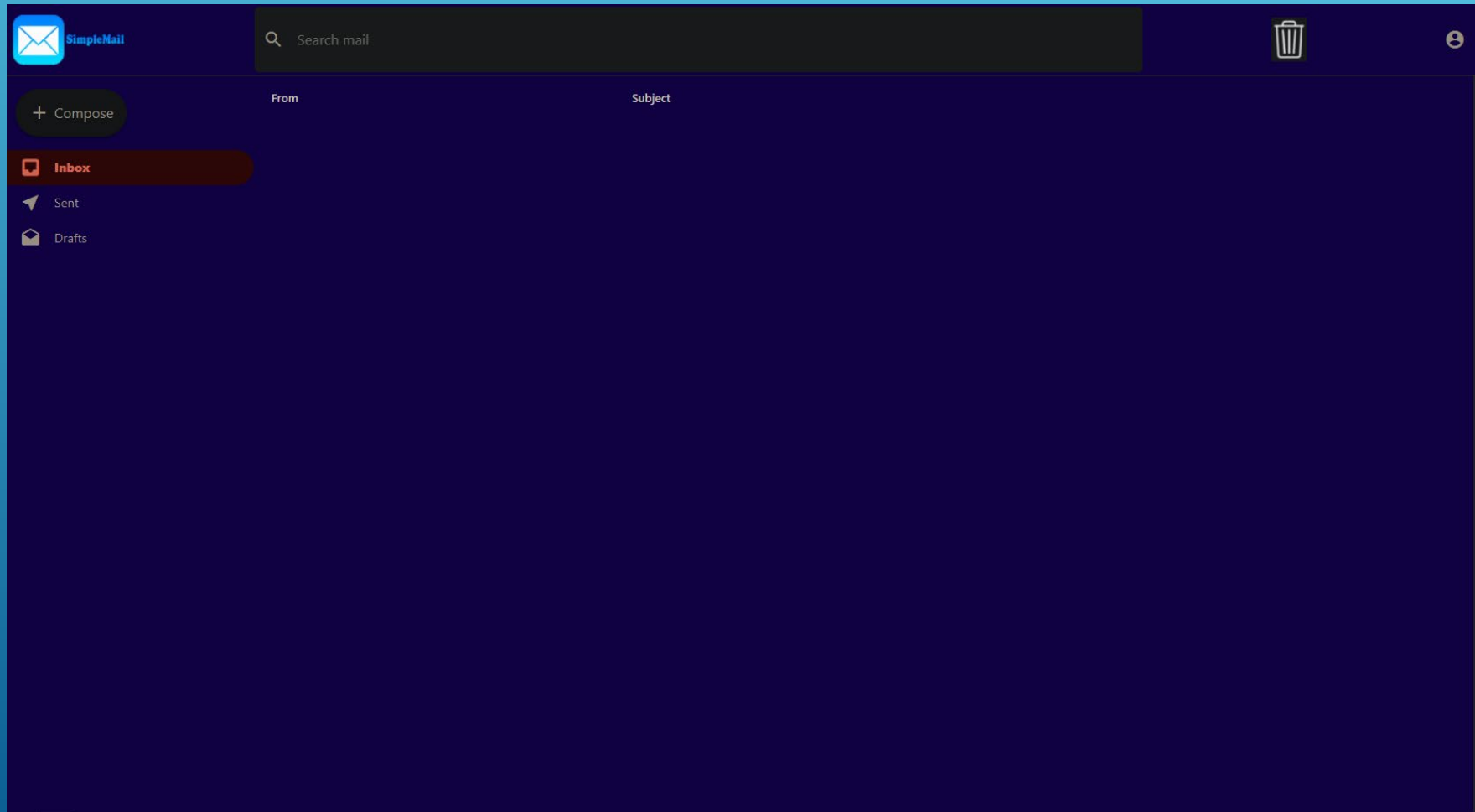
Sign in

Signup Here

6. MANUAL DE INSTALARE SI UTILIZARE

6.3 PAGINA PRINCIPALA

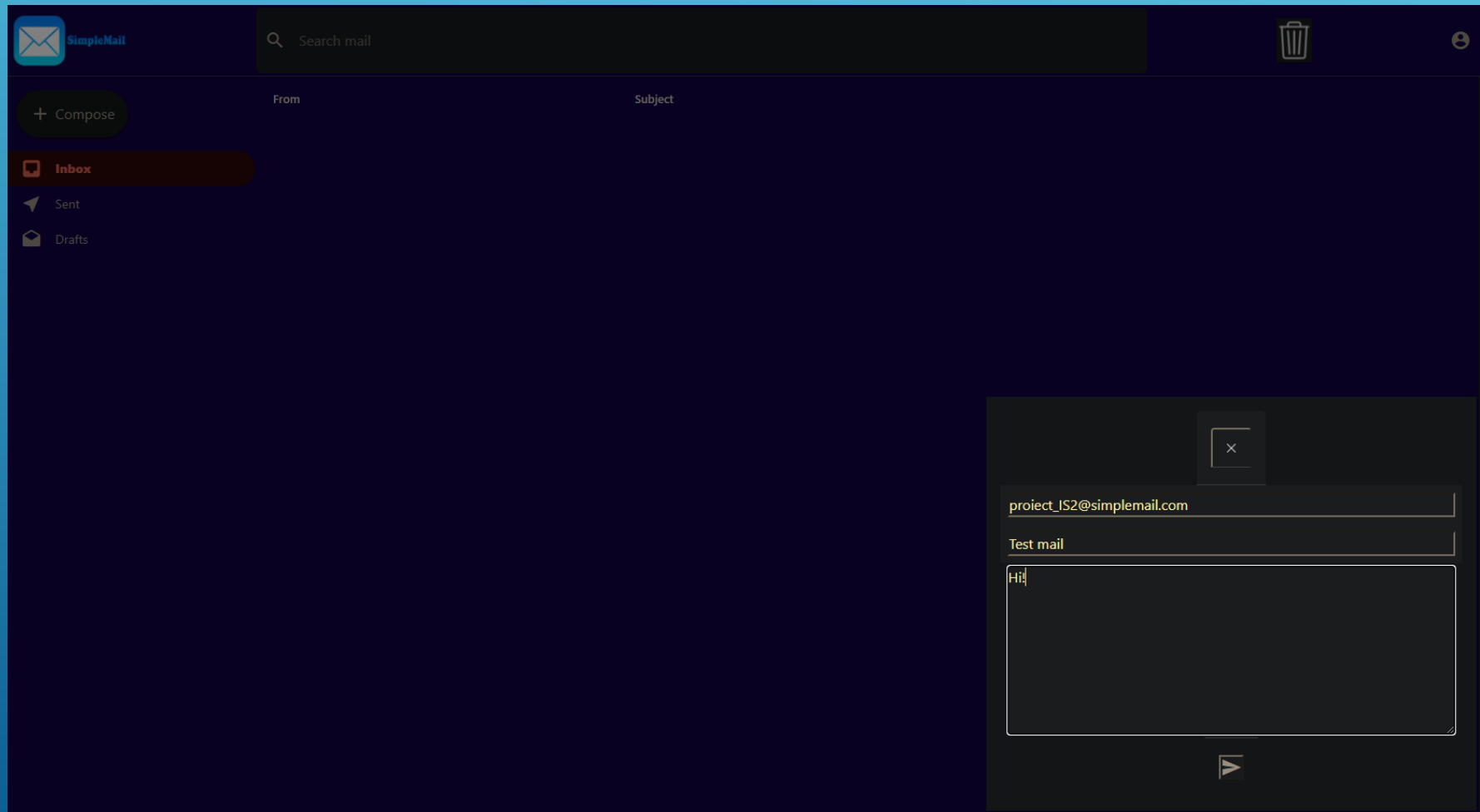
Pagina principala din care utilizatorul isi poate vizualiza si compune mail-urile



6. MANUAL DE INSTALARE SI UTILIZARE

6.4 COMPUNEREA UNUI MAIL

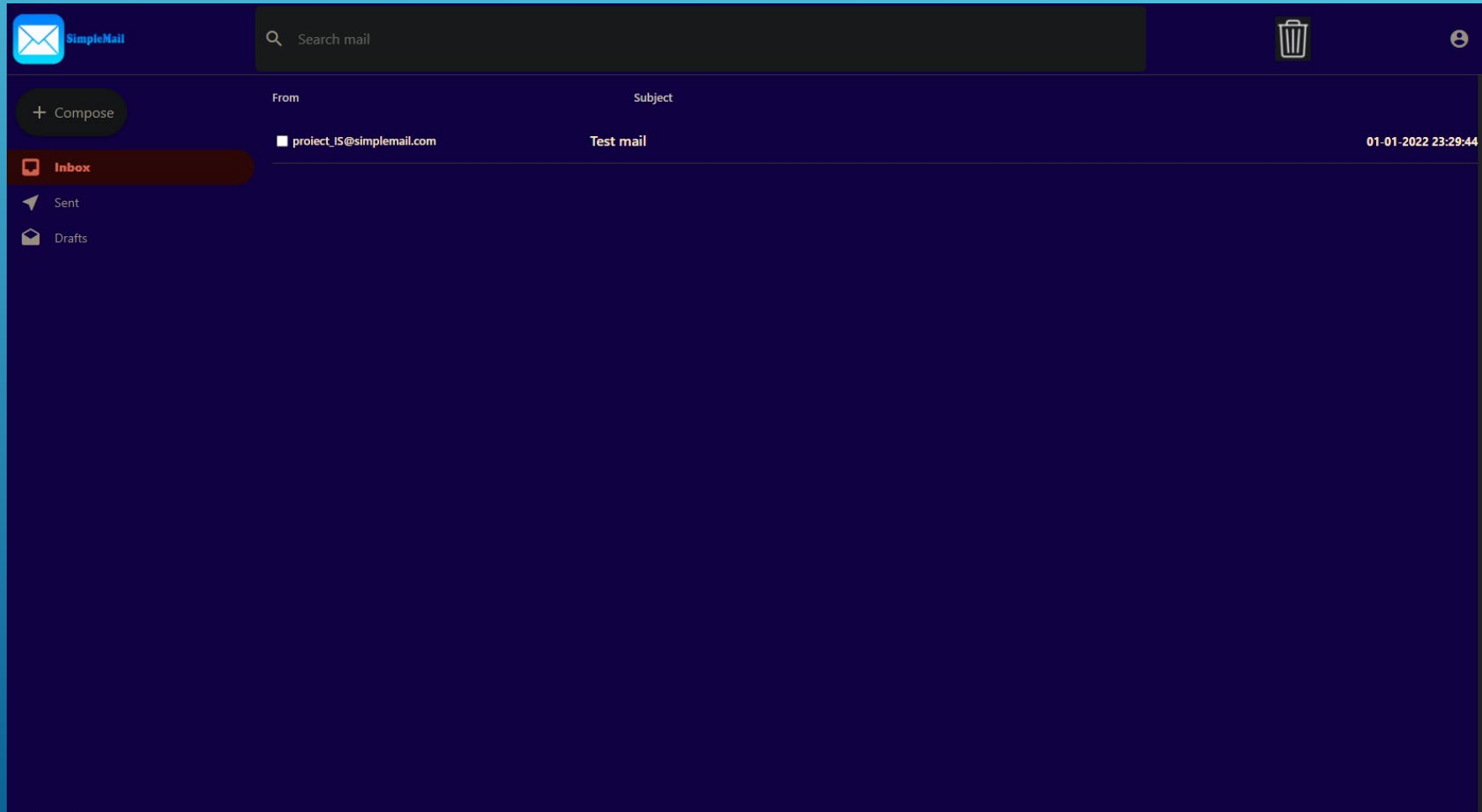
Pentru compunerea unui email se apasa butonul “compose” care va deschide o fereastra pop-up in care se poate compune un mail



6. MANUAL DE INSTALARE SI UTILIZARE

6.5 VIZUALIZAREA MAILURILOR PRIMITE

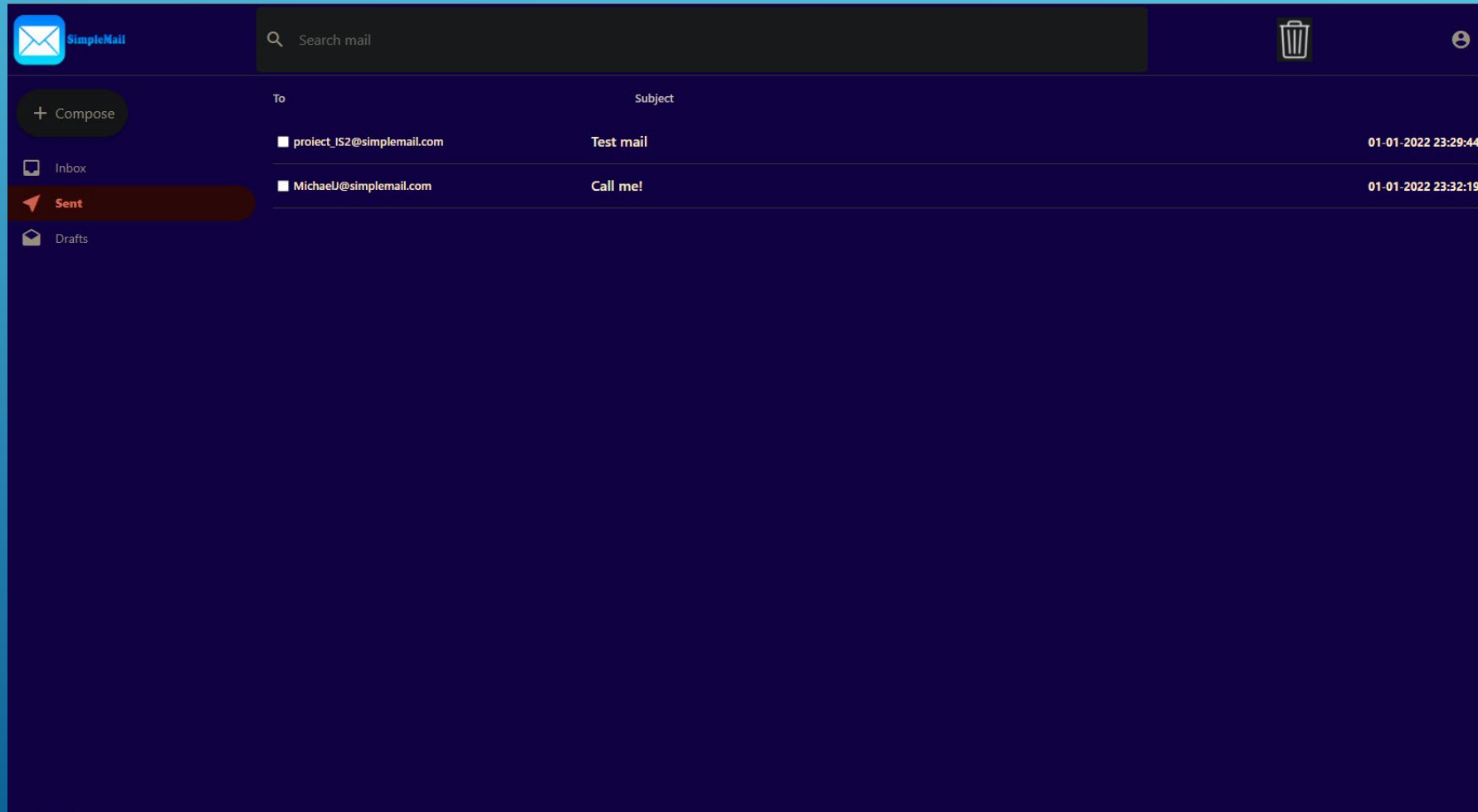
Mailurile trimise pot fi vazute accesand categoria "Inbox"



6. MANUAL DE INSTALARE SI UTILIZARE

6.6 VIZUALIZAREA MAILURILOR TRIMISE

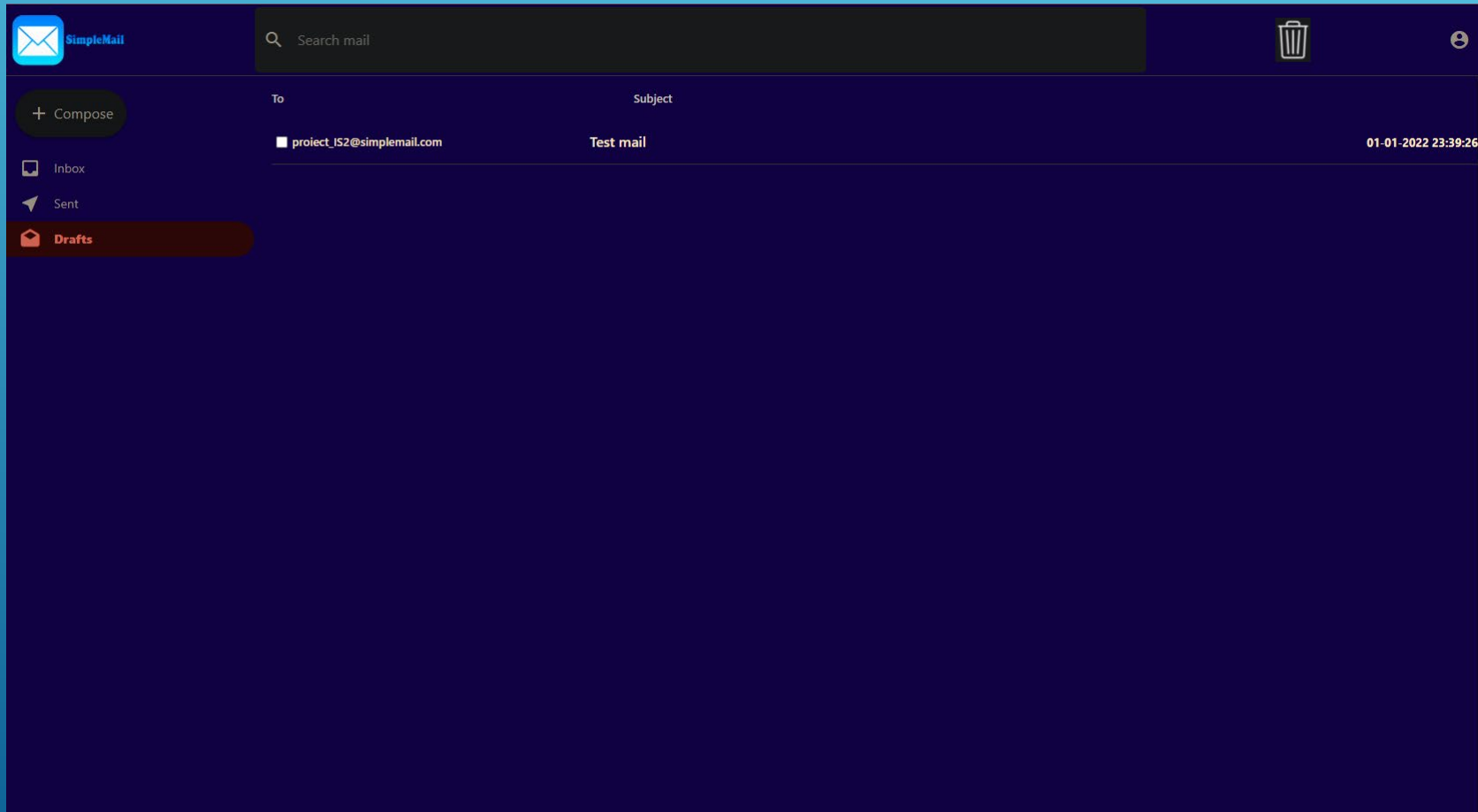
Mailurile trimise pot fi vazute accesand categoria "Sent"



6. MANUAL DE INSTALARE SI UTILIZARE

6.7 SALVAREA UNUI MAIL CA DRAFT

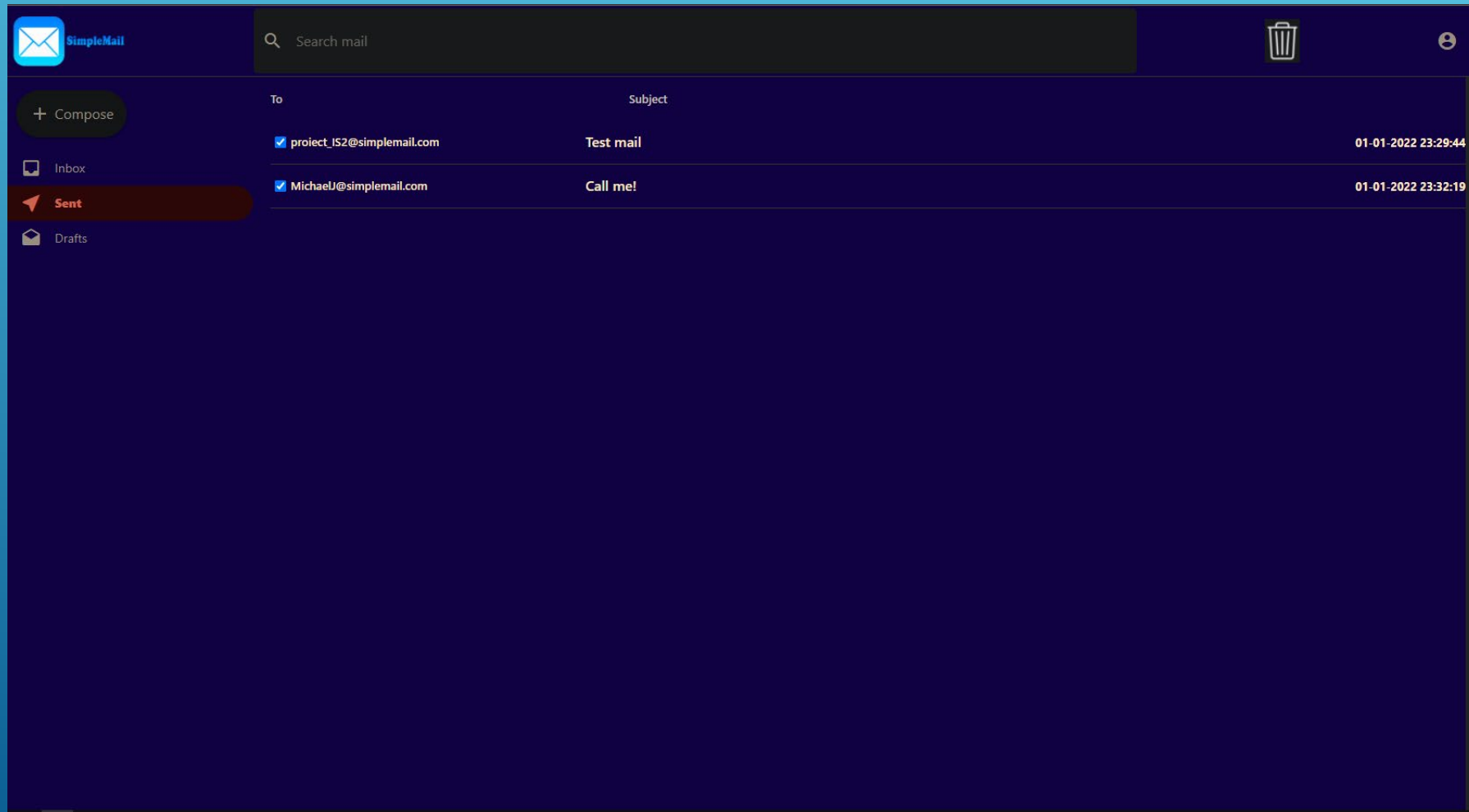
Un mail inceput dar care nu a fost trimis se va salva in categoria drafts. Acesta poate fi editat si trimis ulterior



6. MANUAL DE INSTALARE SI UTILIZARE

6.8 STERGEREA A UNUIA/MAI MULTOR MAILURI

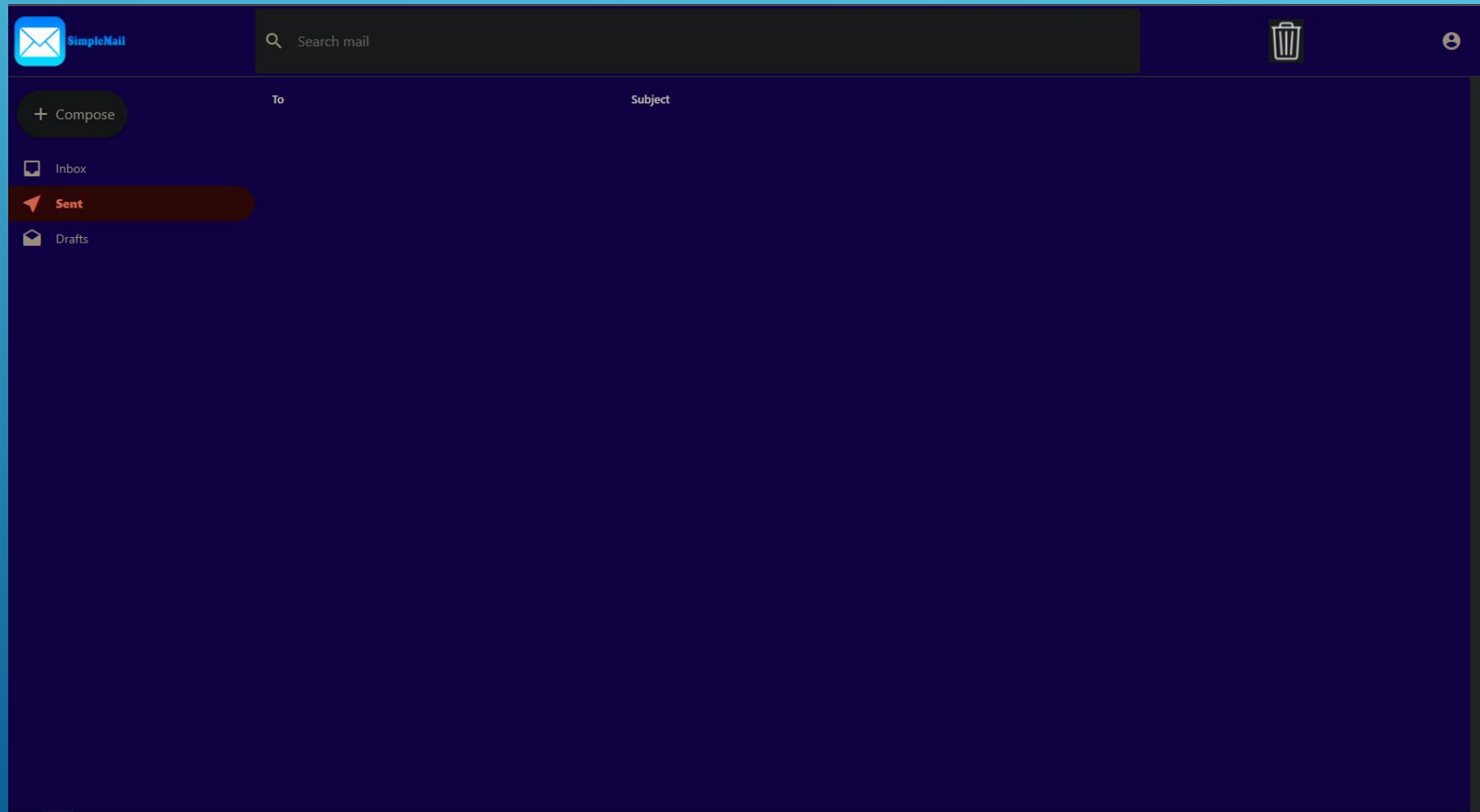
Se selecteaza mailuri care doresc sa fie sterse, iar apoi se apasa pe butonul de stergere



6. MANUAL DE INSTALARE SI UTILIZARE

6.8 STERGEREA A UNUIA/MAI MULTOR MAILURI

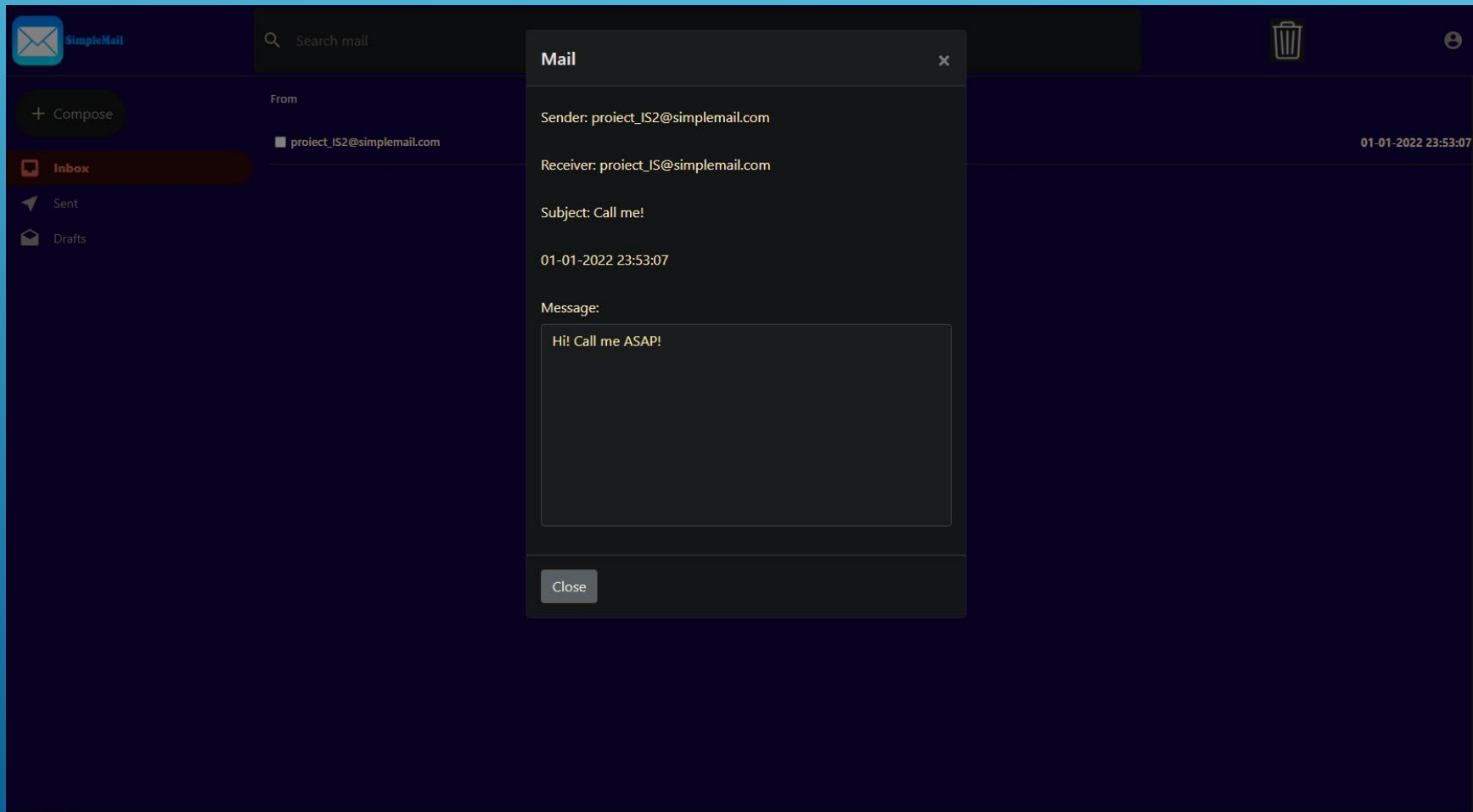
Se selecteaza mailuri care doresc sa fie sterse, iar apoi se apasa pe butonul de stergere



6. MANUAL DE INSTALARE SI UTILIZARE

6.9 VIZUALIZARE INTREGULUI MAIL

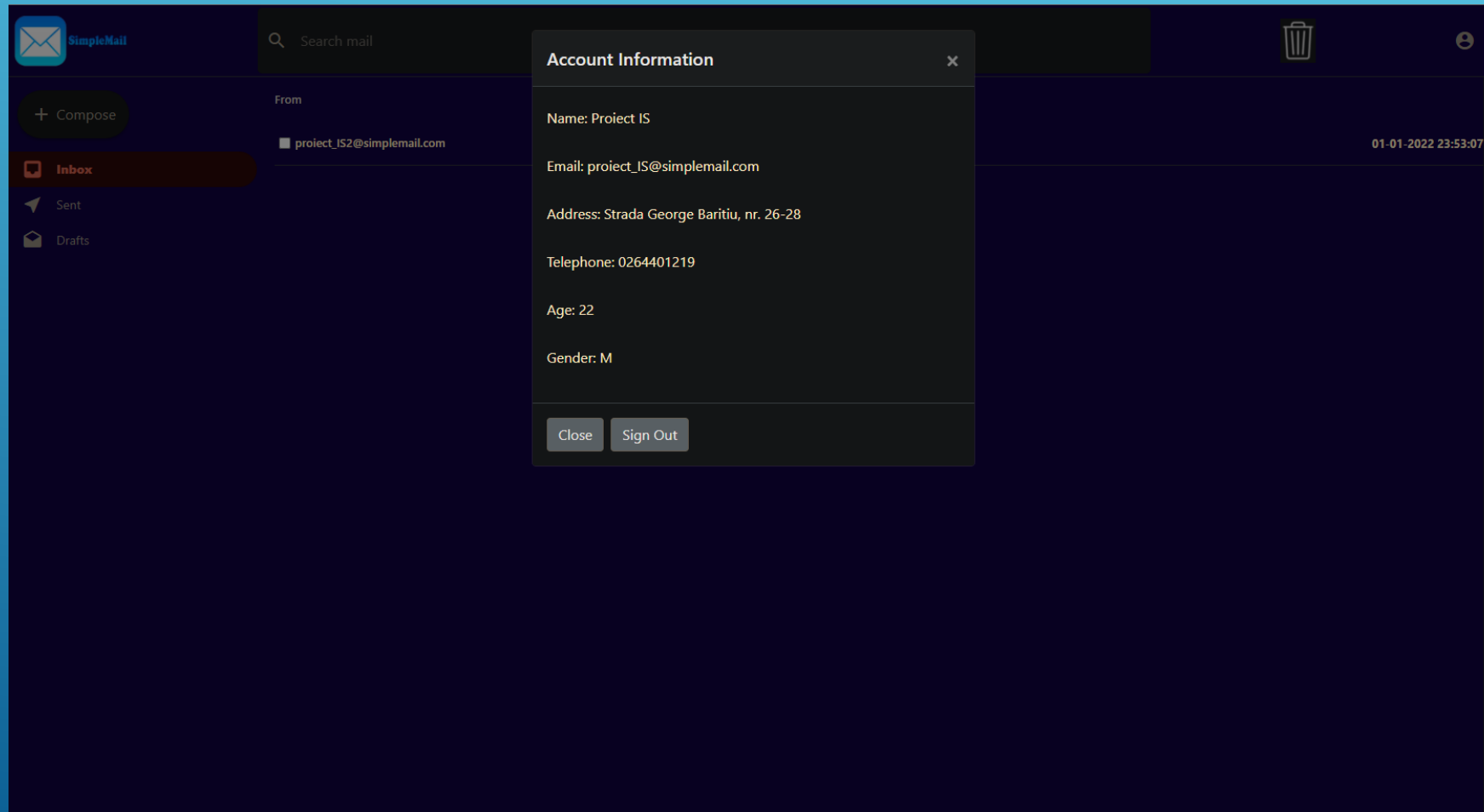
Se selecteaza mailul din categoria “inbox” sau “sent” si va apareea o fereastra pop-up care contine toate detaliile mail-ului.




6. MANUAL DE INSTALARE SI UTILIZARE

6.10 VIZUALIZARE DATELOR PERSONALE

Se apasa pe butonul din coltul dreapta sus “person” care va afisa datele utilizatorului. Totodata, in acest meniu se afla si butonul “Sign Out” care va deloga utilizatorul



7.1 DEZVOLTARI ULTERIOARE

- Introducerea a mai multor categorii de mailuri
 - Adaugarea atasamentelor la un email
 - Posibilitatea de a schimba fontul la compunerea unui email
 - Trimiterea programata a unui email
 - Adaugare functie CC
- 
- Several thin, parallel white lines of varying lengths and slopes are positioned in the bottom right corner of the slide, creating a modern, abstract graphic element.

7.2 CONCLUZII

In urma dezvoltarii acestei aplicatii web, am invatat cum sa lucram cu framework-ul Java Spring atat pe partea de back-end de la crearea end-point-urilor si a serviciilor asociate cu bazele de date pana la crearea unor certificate SSL pentru a putea folosi protocolul https, cat si pe partea de front-end de la formarea unor template-uri cu ajutorul Thymeleaf pana la manipularea elementelor din pagina web cu javascript.



8. BIBLIOGRAFIE

- <https://www.thymeleaf.org/documentation.html>
- <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>
- <https://spring.io/guides/gs/rest-service-cors/>
- <https://www.baeldung.com/spring-boot-https-self-signed-certificate>
- <https://www.baeldung.com/spring-security-registration-password-encoding-bcrypt>
- <https://www.mongodb.com/compatibility/spring-boot>
- <https://www.w3schools.com/js/default.asp>
- <https://www.w3schools.com/jquery/default.asp>
- <https://getbootstrap.com/docs/4.0/components/modal/>