## Descriere Problema

Dezvoltați o aplicație client/server care poate fi utilizată pentru organizarea de conferințe științifice. Aplicația va avea 3 tipuri de utilizatori: participant conferință, organizator conferință și administrator.

Utilizatorii de tip participant pot efectua următoarele operații fără autentificare:

* Înscrierea la conferință (introducere date personale, inclusiv CV-ul și o fotografie);
* Vizualizarea programului conferinței pe secțiuni (la fiecare secțiune, pentru fiecare participant să fie vizibilă și o fotografie a acestuia, dar și accesarea CV-ului).

Utilizatorii de tip participant pot efectua următoarele operații după autentificare:

* Accesarea volumului conferinței.

Utilizatorii de tip organizator conferință pot efectua următoarele operații după autentificare:

* Acceptarea sau respingerea unui participant (notificare prin email);
* Operații CRUD în ceea ce privește persistența participanților și a lucrărilor prezentate;
* Generearea programului conferinței pe secțiuni;
* Filtrarea listei de participanți după secțiunea la care participă;
* Salvare liste filtrate cu informații despre lucrările prezentate în mai multe formate: csv, json, xml, doc;
* Vizualizarea unor statistici utilizând grafice (structură radială, structură inelară, de tip coloană, etc.).

Utilizatorii de tip administrator pot efectua următoarele operații după autentificare: ϖ Operații CRUD pentru informațiile legate de utilizatorii aplicației care necesită autentificare;

* Vizualizarea listei tuturor utilizatorilor care necesită autentificare;
* Filtrarea listei utilizatorilor care necesită autentificare după tip;
* Notificarea fiecărui utilizator care necesită autentificare prin cel puțin 2 variante (email, SMS, WhatsApp, Skype, etc.) la orice modificare a informațiilor de autentificare aferente acelui utilizator.

Interfața grafică a aplicației client va fi disponibilă în cel puțin 3 limbi de circulație internațională.

## Instrumente utilizate

* Framework Windows Presentation Foundation (WPF)
* Limbajul de programare C#
* Baza de date PostgreSQL

## Justificarea limbajului de programare ales

Pentru dezvoltarea proiectului de organizare a conferințelor științifice, am ales să folosim tehnologiile Windows Presentation Foundation (WPF) împreună cu limbajul de programare C#. WPF este un framework de dezvoltare de aplicații desktop pentru platforma Windows, iar C# este un limbaj de programare puternic și flexibil, care se integrează perfect cu WPF.

Prin utilizarea WPF, putem crea interfețe grafice moderne și atrăgătoare, care să ofere o experiență utilizator de înaltă calitate. WPF oferă un set bogat de controale și funcționalități pentru a construi interfețe complexe și interactive, precum și o arhitectură de legare de date flexibilă care facilitează manipularea și prezentarea datelor.

Prin intermediul limbajului C#, putem implementa logica de afaceri a aplicației într-un mod eficient și ușor de înțeles. C# oferă o sintaxă clară și concisă, suport pentru programare orientată pe obiecte și o gamă largă de funcționalități și biblioteci standard, ceea ce face să fie un limbaj potrivit pentru dezvoltarea aplicațiilor desktop.

Împreună, WPF și C# oferă un mediu de dezvoltare puternic și flexibil, care ne permite să construim o aplicație robustă, scalabilă și ușor de întreținut pentru organizarea conferințelor științifice. Această combinație de tehnologii ne oferă control deplin asupra aspectului și comportamentului aplicației noastre, și ne permite să răspundem eficient la cerințele și schimbările de business în timpul procesului de dezvoltare.

## Diagrama Usecase

Imaginea ilustrează o diagramă de cazuri de utilizare pentru un sistem de organizare a conferințelor științifice, evidențiind patru categorii principale de actori și interacțiunile lor cu sistemul. În centru, "LOGIN" reprezintă punctul de intrare în sistem, unde utilizatorii trebuie să se autentifice pentru a accesa diferite funcții.

Primul actor, denumit simplu „Actor”, se referă la participanții la conferință. Ei au posibilitatea de a vizualiza programul conferințelor pentru a se informa despre sesiuni și evenimente și pot de asemenea să se înscrie la conferințe, o acțiune ce presupune furnizarea de detalii personale și selecția evenimentelor de interes.

Următorul actor este „Utilizatorul”, care reprezintă persoanele înregistrate și autentificate în sistem. Responsabilitatea principală a acestui utilizator este accesarea volumelor de conferință, ce pot include diverse materiale și publicații asociate evenimentului.

Organizatorul joacă un rol cheie în gestionarea și coordonarea aspectelor administrative ale conferinței.Ei dețin responsabilitatea de a administra baza de date a conferinței, executând operațiuni CRUD (Create, Read, Update, Delete) pentru gestionarea participanților și a lucrărilor prezentate. Acest lucru asigură că toate informațiile relevante sunt actualizate și accesibile. Unul dintre rolurile cheie este capacitatea de a accepta sau respinge participanții la conferință, cu posibilitatea de a trimite notificări prin email, ceea ce facilitează o comunicare eficientă și promptă. Pe lângă acestea, organizatorii au capacitatea de a genera programul conferinței, organizându-l pe secțiuni pentru o navigare mai ușoară și o mai bună organizare, filtrarea listei de participanți pe secțiuni este, de asemenea, o funcție disponibilă. În cele din urmă, organizatorii pot salva și exporta liste filtrate, împreună cu informații despre lucrările prezentate, într-o varietate de formate de fișiere cum ar fi CSV, JSON, XML și DOC.

A diagram of a company

Description automatically generated

Figura 1 :Diagrama Usecase

În sfârșit, există rolul de Administrator, care are responsabilități extinse în cadrul sistemului. Administratorul supraveghează conturile de utilizatori, gestionând activități precum crearea, modificarea și ștergerea acestora, și are posibilitatea de a vizualiza liste complete de utilizatori și detalii asociate.

## Diagrama UML

### Server

A screenshot of a computer program

Description automatically generated

A diagram with text on it

Description automatically generatedA pink chart with black text

Description automatically generatedA screenshot of a computer program

Description automatically generated

Diagrama prezintă o reprezentare simplificată a unei arhitecturi orientate pe servicii (SOA). SOA este un model de proiectare software care promovează utilizarea serviciilor slab cuplate pentru construirea aplicațiilor flexibile și scalabile.

Componentele cheie ale diagramei:

* DTO (Obiect de Transfer de Date) DTO-urile sunt folosite pentru transferul datelor între servicii. Ele sunt de obicei obiecte Java simple (POJO-uri) care încapsulează datele necesare de un serviciu.
* Repositoare:Repozitoriile oferă o modalitate de stocare și recuperare a datelor. Ele pot fi implementate folosind diverse tehnologii, cum ar fi bazele de date, sistemele de fișiere sau cache-urile de memorie.
* Servicii:Serviciile sunt componentele de bază ale unei SOA. Acestea sunt unități auto-conținute de funcționalitate care furnizează capacități de afaceri specifice. Serviciile comunică între ele folosind protocoale standardizate, cum ar fi SOAP sau REST.

### Client

A screenshot of a computer

Description automatically generated

A close-up of a computer screen

Description automatically generated

Există șase clase: ConferenceDTO, ParticipantDTO, UserDTO, PresentationDTO, PackeageDTO și Builder. Fiecare clasă are atribute și metode.

Atributele sunt proprietățile unei clase, precum numele, emailul sau ID-ul. De exemplu, clasa ConferenceDTO are un atribut numit "titlu" care este un șir de caractere.

Metodele sunt funcțiile pe care o clasă le poate realiza. De exemplu, clasa ConferenceDTO are o metodă numită "ToString" care convertește clasa într-un șir de caractere.

ConferenceDTO: Această clasă reprezintă un eveniment de conferință. Are atribute precum titlu, locație, dată și o listă de prezentări.

ParticipantDTO: Această clasă reprezintă un participant la o conferință. Are atribute precum nume, email și număr de telefon.

UserDTO: Această clasă reprezintă un utilizator al sistemului. Are atribute precum nume, email, parolă și tip de utilizator.

PresentationDTO: Această clasă reprezintă o prezentare dată la o conferință. Are atribute precum titlu, dată, oră și speaker (posibil reprezentat de ParticipantDTO).

Builder: Această clasă este clasă utilitară care ajută la crearea instanțelor celorlalte clase. Are metode pentru setarea atributelor fiecărei clase.

A screenshot of a computer

Description automatically generated

Diagrama afișează dreptunghiuri care reprezintă clasele, care sunt planuri pentru crearea obiectelor. Aceste clase includ: ParticipantService: Această clasă probabil gestionează operațiunile legate de participanți, posibil participanți la conferință sau prezentatori. UserService: Această clasă probabil se ocupă de conturile de utilizator și gestionarea lor în cadrul sistemului. PresentationService: Această clasă probabil gestionează operațiunile legate de prezentări, cum ar fi crearea, actualizarea și ștergerea acestora. Atribute: În interiorul fiecărui dreptunghi de clasă, există cutii de text mai mici care reprezintă atributele. Acestea sunt proprietăți ale clasei care definesc caracteristicile sale. De exemplu, clasa ParticipantDTO ar putea avea atribute precum "nume" și "email" pentru a stoca informațiile participanților. Metode: De asemenea, în interiorul dreptunghiurilor de clasă, există cutii de text cu un două puncte (:) care separă numele metodei și parametrii săi. Metodele sunt funcțiile pe care clasa le poate realiza. De exemplu, clasa UserService ar putea avea o metodă numită "GetUserByEmailAndPassword" care primește ca parametri emailul și parola și probabil returnează un obiect de utilizator. Relații: Linii conectează clasele, semnificând relațiile dintre ele. Iată diferitele tipuri de relații arătate în diagramă: Asociație: O linie solidă cu o etichetă opțională la fiecare capăt indică o asociație. O asociație înseamnă că două clase au o legătură de un fel. De exemplu, există o asociație între ParticipantService și ParticipantDTO, probabil arătând că ParticipantService lucrează cu obiecte ParticipantDTO. Asociație direcționată: O asociație direcționată este indicată printr-o linie solidă cu o săgeată la unul dintre capete. Săgeata indică clasa care are o relație mai puternică în asociație. În imagine, există o asociație direcționată de la UserService la GetUserByEmailAndPassword, indicând că UserService folosește metoda GetUserByEmailAndPassword. Dependență: O linie punctată cu o săgeată la clasa dependentă semnifică o dependență. Acest lucru înseamnă că o clasă depinde de alta pentru a funcționa. Aici, există o dependență de la ParticipantService la PresentationService, sugerând că ParticipantService ar putea depinde de PresentationService pentru unele funcționalități. În ansamblu, această diagramă de clase oferă o privire asupra modului în care diferite părți ale sistemului interacționează pentru a gestiona participanții, utilizatorii și prezentările.

A screenshot of a computer

Description automatically generated

ClientHost: Această clasă reprezintă partea de client a aplicației. Are metode pentru conectarea la server, solicitarea serviciilor și, posibil, salvarea prezentărilor.

UserService: Această clasă probabil se ocupă de conturile de utilizator și gestionarea lor în cadrul sistemului. Are metode pentru crearea, actualizarea și ștergerea utilizatorilor, precum și pentru obținerea utilizatorilor după email și parolă.

ParticipantService: Această clasă probabil se ocupă de operațiunile legate de participanți, posibil participanți la conferință sau prezentatori. Are metode pentru crearea, actualizarea, ștergerea participanților și pentru obținerea lor pe baza diferitelor criterii.

PresentationService: Această clasă probabil se ocupă de operațiunile legate de prezentări, cum ar fi crearea, actualizarea, ștergerea și obținerea prezentărilor.

FileWriter: Această clasă probabil se ocupă de scrierea prezentărilor într-un fișier într-un format specific. Are metode pentru salvarea prezentărilor pe baza unei strategii (de exemplu, DocxSaveStrategy, JsonSaveStrategy, XmlSaveStrategy).

SaveStrategy: Aceasta este o clasă abstractă (indicată de numele italicizat) care definește interfața pentru salvarea prezentărilor. Are o subclasă pentru fiecare format de fișier (docx, json, xml) care implementează logica reală de salvare.

PackageDTO: Această clasă probabil reprezintă un pachet de date care este trimis între client și server. Are atribute precum mesaj, date și builder.

Dependență: O linie punctată cu o săgeată la clasa dependentă semnifică o dependență. Acest lucru înseamnă că o clasă depinde de alta pentru a funcționa. Aici, există o săgeată de dependență de la FileWriter la SaveStrategy, indicând că FileWriter depinde de interfața SaveStrategy pentru a salva prezentările. Există, de asemenea, o dependență de la fiecare subclasă a SaveStrategy (DocxSaveStrategy, JsonSaveStrategy, XmlSaveStrategy) la FileWriter, arătând că acestea depind de FileWriter pentru a efectua scrierea efectivă a fișierului.

A diagram of a company

Description automatically generated with medium confidence

HomePresenter, LoginPresenter, UtilizatorPresenter, AdminPresenter, OrganizatorPresenter: Aceste clase probabil reprezintă diferiți prezentatori pentru interfața utilizatorului. Fiecare prezentator ar putea fi responsabil pentru o anumită vedere sau funcționalitate în sistem. De exemplu, HomePresenter ar putea gestiona ecranul de start, LoginPresenter ar putea gestiona funcționalitatea de conectare, și așa mai departe.UserService: Această clasă probabil se ocupă de conturile de utilizator și de gestionarea lor în cadrul sistemului. Are metode pentru crearea, actualizarea și ștergerea utilizatorilor, precum și pentru obținerea utilizatorilor după email și parolă.UserManagement Facade: Această clasă acționează ca o fațadă, oferind o interfață simplificată către sistemul de gestionare a utilizatorilor. Ascunde complexitatea sistemului prin încapsularea funcționalităților oferite de UserService și potențial alte clase.

A screenshot of a computer program

Description automatically generated

## Diagrame de Secvente

A white sheet of paper with black text

Description automatically generated

A screenshot of a diagram

Description automatically generatedA white paper with black text

Description automatically generatedA white sheet of paper with black text

Description automatically generatedA white sheet of paper with black text

Description automatically generatedA white paper with black text

Description automatically generatedA white sheet of paper with black text

Description automatically generated

## Descriere aplicație

Această platformă de gestionare a conferințelor oferă o experiență cuprinzătoare și fluentă atât pentru participanți, cât și pentru organizatori și administratori. La nivelul utilizatorilor, procesul de înscriere este simplificat la maxim, reducând barierele administrative și facilitând implicarea în evenimente. Formularul de înscriere solicită doar informațiile esențiale, precum numele, e-mailul, numărul de telefon și o scurtă prezentare, astfel încât să nu impună o sarcină greoaie participanților.

După completarea formularului de înscriere, utilizatorii sunt ghidați către o pagină de autentificare, unde pot accesa conturile lor utilizând adresa de e-mail și parola. Această etapă de autentificare este crucială pentru asigurarea securității datelor și pentru personalizarea experienței fiecărui utilizator în funcție de rolul lor în cadrul conferinței: participanți, organizatori sau administratori. Redirecționarea către pagini personalizate după autentificare îmbunătățește și mai mult experiența utilizatorului, oferindu-le acces rapid la informațiile și opțiunile relevante.

În cazul în care apar erori în timpul procesului de autentificare, sistemul este programat să afișeze mesaje de eroare clare și intuitive, care să ghideze utilizatorii în rezolvarea problemelor. Aceasta este o practică excelentă pentru menținerea unei experiențe de utilizare fără probleme și pentru reducerea frustrărilor.

Platforma pune de asemenea un accent deosebit pe panoul de administrare, oferind administratorilor instrumente puternice pentru gestionarea eficientă a utilizatorilor și a evenimentelor. Aceste instrumente permit adăugarea, actualizarea și gestionarea conturilor utilizatorilor, asigurând în același timp o organizare corespunzătoare a conferințelor.

Secțiunea de administrare a conferințelor oferă organizatorilor un set robust de instrumente pentru gestionarea eficientă a participanților și a prezentărilor. Funcțiile avansate permit organizatorilor să gestioneze participanții, să actualizeze sau să elimine înregistrările și să filtreze prezentările și participanții în funcție de secțiunea specifică a conferinței.

În plus, platforma permite salvarea și exportul detaliilor prezentărilor în diferite formate de fișier, precum CSV, JSON, XML și DOC, facilitând distribuirea programului conferinței și colaborarea eficientă cu alte părți interesate. Aceste funcționalități avansate completează experiența conferințelor, contribuind la organizarea eficientă și la îmbunătățirea experienței participanților și a organizatorilor.