## Descriere Problema

Dezvoltați o aplicație 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:

* Operații CRUD pentru informațiile legate de utilizatorii aplicației care necesită autentificare;
* Vizualizarea listei tuturor utilizatorilor care necesită autentificare și filtrarea acesteia după tipul utilizatorilor.

Interfața grafică a aplicației 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

O imagine care conține text, diagramă, captură de ecran, linie

Descriere generată automat

Figura 1 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.

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

* Model

O imagine care conține text, captură de ecran, număr, software

Descriere generată automat

Figura 2 Diagrama Model

Diagrama UML descrie arhitectura unui sistem Repository pentru gestionarea unui model de organizare a conferințelor. Clasa centrală este Repository, care conține metodele de bază pentru operațiile cu baza de date, cum ar fi deschiderea și închiderea conexiunilor, și executarea interogărilor. Aceasta servește drept clasă de bază pentru celelalte repository-uri specifice, care moștenesc aceste metode. Există mai multe repository-uri specifice în sistem. PrezentareRepository administrează datele legate de prezentări în sistem. Are metode specifice pentru crearea, obținerea, actualizarea și ștergerea prezentărilor. Aceasta interacționează cu clasa Prezentare, care stochează detaliile unei prezentări, cum ar fi id, Title, Author, Description, Date, Time, Section și ConferenceID. UtilizatorRepository se ocupă cu manipularea datelor utilizatorilor. Include metode pentru crearea, obținerea și actualizarea utilizatorilor. Clasa User reflectă un utilizator individual cu detalii precum id, name, email, password, UserType, phone și alte informații personale.ConferintaRepository administrează datele despre conferințe. Conține funcționalități pentru crearea, obținerea, actualizarea și ștergerea conferințelor. Clasa Conference reține informații despre o conferință individuală, inclusiv id, Title, Location și Date.ParticipantRepository este responsabil pentru gestionarea participanților la conferințe. Include metode pentru obținerea informațiilor despre participanți, adăugarea și actualizarea acestora. Clasa Participant extinde clasa User, adăugând câmpuri suplimentare specifice participanților, cum ar fi id și informațiile despre prezentările la care participă.StatisticsRepository conține metode pentru obținerea diverselor statistici legate de participanți și conferințe, cum ar fi numărul de participanți per conferință sau per secțiune și media de participanți per conferință.Participant\_PrezentareRepository administrează relația dintre participanți și prezentări. Conține metode pentru crearea, obținerea, actualizareași ștergerea acestor relații.

În diagrama UML, mai sunt prezente următoarele clase și interfețe. Subject și Observer sunt utilizate pentru a implementa un design pattern de tip Observer, unde Subject deține o listă de observatori și notifică aceștia despre schimbări. IOObserver este o interfață ce definește metoda Updated pentru actualizarea observatorilor.

Aceste componente interacționează pentru a forma un sistem robust de gestionare a conferințelor, oferind funcționalități complete pentru manipularea datelor utilizatorilor, prezentărilor, conferințelor și participanților, precum și pentru obținerea de statistici relevante. Sistemul este bine structurat, permițând extinderea și întreținerea eficientă prin separarea responsabilităților în diferite repository-uri și utilizarea unor pattern-uri de design recunoscute.

* View

O imagine care conține text, captură de ecran, document, Font

Descriere generată automat

Figura 3 Diagrama View

Diagrama UML prezentată descrie arhitectura interfeței utilizator (View) pentru un sistem de organizare a conferințelor.AdminGUI gestionează interfața pentru administratori și conține diverse butoane și elemente pentru manipularea datelor utilizatorilor și a prezentărilor. Printre acestea se numără CreateUserButton, UpdateUserButton, DeleteUserButton, SearchButton, LogOutButton, RefreshButton, și ExportPDFButton. De asemenea, include controale pentru afișarea și manipularea datelor, cum ar fi UserTable, NameTextBox, EmailTextBox, PasswordTextBox, UserTypeComboBox, și TelephoneTextBox. Aceasta oferă funcționalități pentru crearea, actualizarea, ștergerea și căutarea utilizatorilor, precum și pentru afișarea și exportarea datelor în format PDF.

ChartView este responsabilă pentru afișarea diverselor tipuri de grafice. Include metode pentru afișarea diagramelor inelare (ShowRingChart), diagramelor liniare (ShowLineChartDialog), prezentărilor per autor (ShowPresentationsByAuthorChart), prezentărilor per zi (ShowPresentationsPerDayChart), participanților per secțiune (ShowParticipantsBySectionChart), și participanților per conferință (ShowParticipantsByConferenceChart).

HomeGUI reprezintă interfața principală a sistemului și conține butoane pentru navigarea între diferitele secțiuni ale aplicației. Acestea includ LoginButton, AdminButton, OrganizerButton, și UserButton. Aceasta oferă o metodă centralizată pentru accesarea diferitelor funcționalități ale aplicației.

OrganizerGUI gestionează interfața pentru organizatorii de conferințe. Aceasta include butoane pentru crearea, actualizarea, ștergerea și căutarea conferințelor, cum ar fi CreateConferenceButton, UpdateConferenceButton, DeleteConferenceButton, și SearchConferenceButton. De asemenea, conține un tabel pentru afișarea conferințelor (ConferenceTable).

LoginGUI se ocupă de interfața de autentificare a utilizatorilor și include controale pentru introducerea datelor de autentificare, cum ar fi EmailTextBox, PasswordTextBox, și butonul de autentificare LoginButton.

UtilizatorGUI administrează interfața pentru utilizatori și include controale pentru afișarea și manipularea datelor despre conferințe, cum ar fi ConferenceTable, BackButton, și GetTable.

* Controller

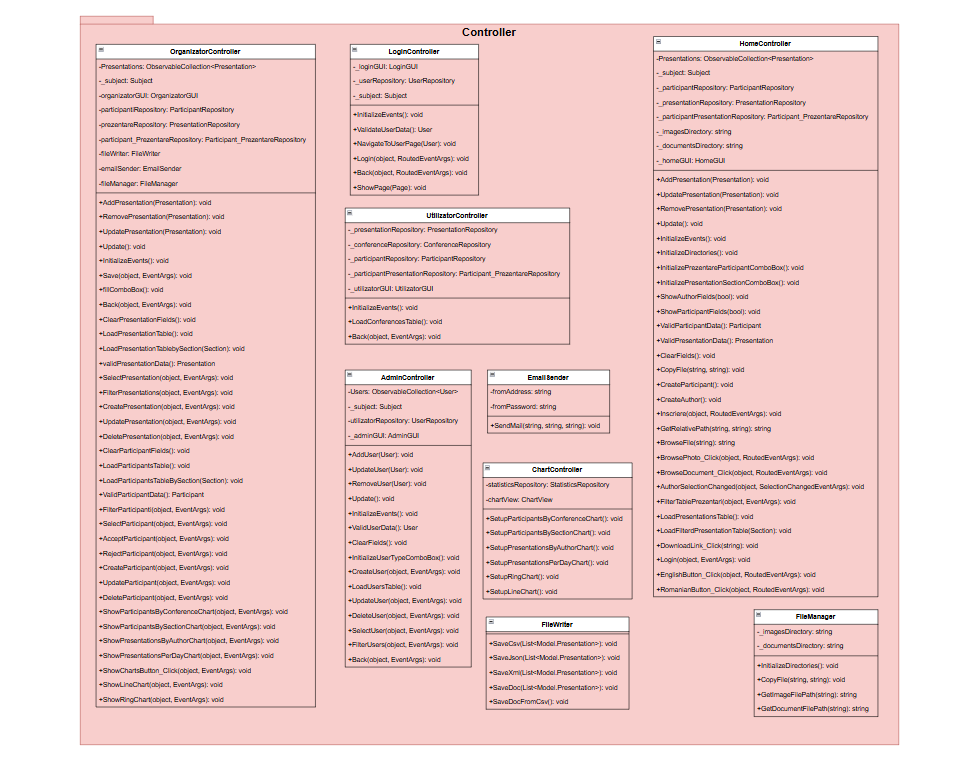


Figura 4 Diagrama Controller

OrganizerController gestionează operațiunile legate de organizatorii de conferințe. Acesta include metode pentru adăugarea, actualizarea și ștergerea prezentărilor (AddPresentation, Update, RemovePresentation), încărcarea prezentărilor și participanților (LoadPresentationTable, LoadParticipantsTableBySection), precum și funcționalități pentru gestionarea evenimentelor (SelectPresentation, SaveEvent). OrganizerController utilizează diverse repository-uri, inclusiv ParticipantRepository, PresentationRepository, și Participant\_PrezentareRepository, și se bazează pe FileManager pentru gestionarea fișierelor și pe EmailSender pentru trimiterea email-urilor.

LoginController se ocupă de autentificarea utilizatorilor. Include metode pentru inițializarea controlerului (Initialize), navigarea către paginile utilizatorilor (NavigateToUserPage), și pentru afișarea paginii de autentificare (ShowPage). Acesta utilizează UserRepository și LoginGUI pentru gestionarea autentificării și interfeței.

UtilizatorController gestionează operațiunile legate de utilizatori. Acesta include metode pentru încărcarea conferințelor și participanților (LoadConferencesTable, LoadParticipantsTable), precum și pentru gestionarea evenimentelor (Back). Utilizează repository-uri precum PresentationRepository, ConferenceRepository, ParticipantRepository, și Participant\_PrezentareRepository, și interacționează cu UtilizatorGUI.

AdminController este responsabil pentru gestionarea operațiunilor administrative. Include metode pentru adăugarea, actualizarea și ștergerea utilizatorilor (AddUser, UpdateUser, RemoveUser), precum și pentru gestionarea evenimentelor (SelectUser, ClearFields, FilterUsers). Acesta utilizează UserRepository și AdminGUI pentru a interacționa cu modelul și interfața.

HomeController gestionează logica principală a aplicației și navigarea între diferitele secțiuni ale acesteia. Include metode pentru gestionarea prezentărilor (AddPresentation, UpdatePresentation, RemovePresentation), inițializarea diferitelor componente (InitializeEvents, InitializePresenterComboBox), și pentru gestionarea documentelor și fișierelor (BrowseDocument, Download). Utilizează repository-uri pentru prezentări, participanți și participanți-prezentări, și interacționează cu HomeGUI.

ChartController gestionează afișarea graficelor statistice. Include metode pentru configurarea diferitelor tipuri de grafice (SetupParticipantsByConferenceChart, SetupPresentationsByDayChart) și pentru afișarea acestora (ShowRingChart, ShowLineChartDialog). Utilizează StatisticsRepository și interacționează cu ChartView.

EmailSender și FileWriter sunt componente auxiliare utilizate de alte controllere pentru trimiterea email-urilor și gestionarea fișierelor. EmailSender include metode pentru trimiterea email-urilor (SendMail), iar FileWriter include metode pentru salvarea și deschiderea documentelor (SaveCSV, SaveDOCX, SavePDF, SaveDOCFromCSV).FileManager este responsabil pentru gestionarea fișierelor și directoarelor din aplicație. Include metode pentru crearea directoarelor (InitializeDirectories), copierea fișierelor (CopyFile), și obținerea drumurilor către documente și imagini (GetDocumentsFullPath, GetImageFullPath).

* DiagramaUML

O imagine care conține text, captură de ecran, Paralel, diagramă

Descriere generată automat

## Descriere aplicație

În etapa inițială, participanții sunt întâmpinați de un formular de înscriere amplasat convenabil pe pagina principală. Acest formular este conceput pentru a fi cât mai simplu și accesibil, solicitând doar informații esențiale precum numele, e-mailul, numărul de telefon și o scurtă prezentare. Scopul este de a încuraja înscrierile prin minimizarea barierelor administrative, permițând în același timp organizatorilor să colecteze date relevante pentru configurarea și personalizarea experienței fiecărui participant.

După completarea formularului de înscriere, utilizatorii sunt direcționați către o pagină de login, unde își pot accesa contul introducând adresa de e-mail și parola. Această etapă de autentificare este esențială pentru a asigura securitatea datelor și pentru a oferi acces la funcții specifice, bazate pe tipul de utilizator: participant, organizator sau administrator. Depinzând de acest statut, utilizatorii sunt redirecționați către pagini personalizate, care le afișează informații și opțiuni relevante rolului lor în cadrul aplicației.

În cazul în care apar erori la autentificare, sistemul este programat să afișeze mesaje de eroare intuitive, care să îi ghideze pe utilizatori în rezolvarea problemelor întâmpinate. Aceasta asigură o experiență de utilizare fluidă și reduce frustrările care ar putea apărea în procesul de login.

Pe lângă facilitățile oferite participanților, aplicația pune un accent deosebit și pe panoul de administrare, oferind administratorilor un set complet de instrumente pentru gestionarea utilizatorilor și evenimentelor. În cadrul acestui panou, administratorii pot adăuga sau actualiza înregistrările utilizatorilor, având la dispoziție câmpuri pentru introducerea datelor esențiale precum numele, e-mailul, parola, tipul de utilizator și numărul de telefon. Aceasta permite o administrare eficientă și rapidă a conturilor, asigurând totodată o organizare corespunzătoare a evenimentelor.

Secțiunea de administrare a conferințelor oferă organizatorilor un set robust de instrumente pentru gestionarea eficientă a participanților și a prezentărilor. Organizatorii pot adăuga noi participanți și prezentări, utilizând câmpuri dedicate pentru introducerea detaliilor precum nume, email, telefon și identificatorul prezentării asociate. Funcțiile avansate permit organizatorilor nu doar să adauge, dar și să accepte sau să respingă participanții, precum și să actualizeze sau să elimine înregistrările existente, facilitând un control detaliat asupra cine va participa la eveniment.

Interfața include funcționalități de filtrare care permit organizatorilor să afișeze prezentările și participanții în funcție de secțiunea specifică a conferinței, facilitând organizarea tematică și îmbunătățind experiența participanților. În plus, există opțiuni pentru salvarea și exportul detaliilor prezentărilor în diferite formate de fișier precum CSV, JSON, XML și DOC, ceea ce le permite organizatorilor să distribuie programul conferinței și să colaboreze eficient cu alte părți interesate.