



Academia de Studii Economice din București
Facultatea de Cibernetică, Statistică și Informatică Economică
Specializarea Informatică Economică

PROIECT PROIECTAREA SISTEMELOR INFORMATICE

Aplicație web pentru organizarea activităților

Profesor coordonator

Cadru asociat drd. Bostan Andreea-Izabela

Student

Petrică Miruna-Alexandra

Grupa 1094, Seria E

CUPRINS

1. Prezentarea sistemului informatic.....	1
1.1. Descrierea generală a sistemului informatic.....	1
1.2. Specificarea cerințelor.....	1
1.2.1 Diagrame ale cazurilor de utilizare.....	3
1.2.2 Descrierea textuală a cazurilor de utilizare (2 cazuri).....	4
2. Analiza sistemului informatic.....	7
2.1 Diagrame de activitate (3 diagrame).....	7
2.2 Diagrama de clase.....	8
2.3 Diagrame de stare (2 diagrame).....	10
2.4 Diagrame de interacțiune (3 diagrame).....	11
2.5 Diagrame de procese și colaborare în BPMN (o diagramă procese, o diagramă colaborare).....	13
3. Proiectarea sistemului informatic.....	15
3.1 Diagrama de clase detaliată.....	15
3.2 Proiectarea bazei de date.....	15
3.3 Proiectarea interfețelor utilizator (4 exemple).....	17
3.4 Diagrama de componente.....	20
3.5 Diagrama de desfășurare.....	20
4. Implementarea sistemului informatic.....	21
4.1 Tehnologii utilizate în implementare.....	21
4.2 Prezentarea pe scurt a funcționalității sistemului.....	21
Bibliografie.....	23

1. Prezentarea sistemului informatic

1.1. Descrierea generală a sistemului informatic

Un sistem informatic reprezintă o colecție integrată de componente hardware, software, rețele, date și proceduri, care lucrează împreună pentru a colecta, prelucra, stoca și distribui informații într-o organizație sau într-un scop specific.

Sistemele informatice devin pe zi ce trece mai utile pentru societatea în care trăim, deoarece stilul de viață alert și multitudinea de sarcini zilnice ne împing adesea spre dorința de a ne simplifica traiul. Astfel, căutăm diferite metode și instrumente care ne pot ajuta să ducem rapid fiecare responsabilitate la un bun sfârșit.

Sistemul informatic în cauză reprezintă o aplicație web pentru organizarea activităților din cadrul unei firme. Acesta își propune să eficientizeze gestionarea activităților zilnice desfășurate în firmă și nu numai, prin oferirea posibilității de a le pune laolaltă într-un mediu comun partajat și de a le organiza cu ajutorul instrumentelor ce sunt puse la dispoziție.

Ideea în jurul căreia se construiește acest proiect este de a regăsi într-un singur loc toate detaliile referitoare la sarcinile ce trebuie desfășurate în cadrul firmei. Fiecare utilizator al aplicației, angajat al firmei respective, poate avea acces la activitățile sale și ale echipei, își poate organiza sarcinile în funcție de priorități, își poate seta timp limită pentru fiecare sarcină.

Sistemul informatic prezentat este bazat atât pe transparență, întrucât fiecare membru va putea vedea progresul întregii echipe și va putea asigna sarcini către ceilalți colegi, cât și pe simplitatea în ceea ce privește crearea de noi activități, datorită interfețelor prietenoase. Utilizatorii vor avea acces la activitățile firmei odată ce primesc codul unic de intrare în echipă de la administrator. Aplicația web descrisă contribuie, de asemenea, la o comunicare eficientă între angajații firmei și între departamentele acesteia.

Elementele prezentate anterior sunt câteva dintre funcționalitățile principale pe care le va implementa sistemul informatic ce va fi realizat, toate acestea având ca scop sporirea productivității angajaților firmei și eficientizarea proceselor de organizare și îndeplinire a sarcinilor de lucru.

1.2. Specificarea cerințelor

Într-un sistem informatic, prezentarea cerințelor specifice funcționalității acestuia este de o importanță ridicată, astfel încât acestea trebuie conturate cât mai clar, pentru stabilirea obiectivelor principale.

În acest capitol vor fi prezentate pe scurt principalele cerințe pe baza cărora va fi realizată aplicația web destinată organizării sarcinilor de lucru.

Cerințe:

- Autentificarea utilizatorilor în aplicație și autorizarea acestora, fiecare putând avea fie rolul de utilizator normal, fie rolul de administrator.
- Gestionarea sarcinilor prin posibilitatea de a adăuga, a modifica, a șterge sau a finaliza activitățile.
- Posibilitatea de a seta date limită pentru activități, a stabili prioritățile acestora, a seta stările activităților(de exemplu: activitate nouă, activitate în curs de realizare, activitate finalizată).
- Opțiunea de a vizualiza atât sarcinile proprii viitoare cât și lista cu toate sarcinile proprii finalizate, vizualizarea listelor de activități viitoare și anterioare pentru întreaga echipă.
- Utilizatorii pot sorta și filtra sarcinile în funcție de preferințe urmând diverse criterii, de exemplu, în funcție de prioritate sau de termenele limită.
- Fiecare modificare efectuată în aplicație asupra sarcinilor se va înregistra în baza de date, iar sistemul se va comporta corespunzător cu cerințele utilizatorilor.
- Administratorii vor putea vizualiza în pagina personală și codul specific echipei pe care o administrează. Codul va fi util pentru utilizatorii noi, deoarece cu ajutorul lui, administratorii le pot oferi acces în echipă.
- Administratorii pot adăuga utilizatori noi în echipă sau îi pot șterge pe cei existenți, pot restricționa accesul și vizibilitatea sarcinilor pentru anumiți utilizatori.
- Utilizatorii aplicației vor primi notificări și alerte cu privire la activitățile viitoare, în funcție de termenele limită și de nivelurile de prioritate ale sarcinilor. Fiecare sarcină nou creată va genera o notificare pentru persoana căreia i-a fost asignată, iar apropierea temporală de data limită specificată a unei activități va genera diferite alerte.
- Va fi facilitată colaborarea între angajați, întrucât aceștia vor putea partaja fișiere și informații, vor putea colabora pentru diverse sarcini comune, vor putea comunica eficient prin descrierea activităților în casetele special destinate

Astfel, soluția informatică va reprezenta o modalitate facilă de comunicare între angajații firmei, prin care totodată își pot stabili atribuțiile zilnice și le pot ordona după bunul plac. De asemenea, își propune să reprezinte o modalitate prietenoasă care va aduce laolaltă angajați cu numeroase funcții din firmă și îi va ajuta să colaboreze indiferent de experiența lor, având ca rezultat final creșterea productivității acestora și sporirea profitului firmei prin acest mod.

În continuare, după stabilirea cerințelor esențiale ale sistemului informatic, funcționalitățile acestuia vor fi reprezentate prin intermediul diagramelor cazurilor de utilizare. Diagramele vor fi realizate cu ajutorul Visual Paradigm Community Edition.

1.2.1 Diagrame ale cazurilor de utilizare

- Diagrama generală a întregului sistem

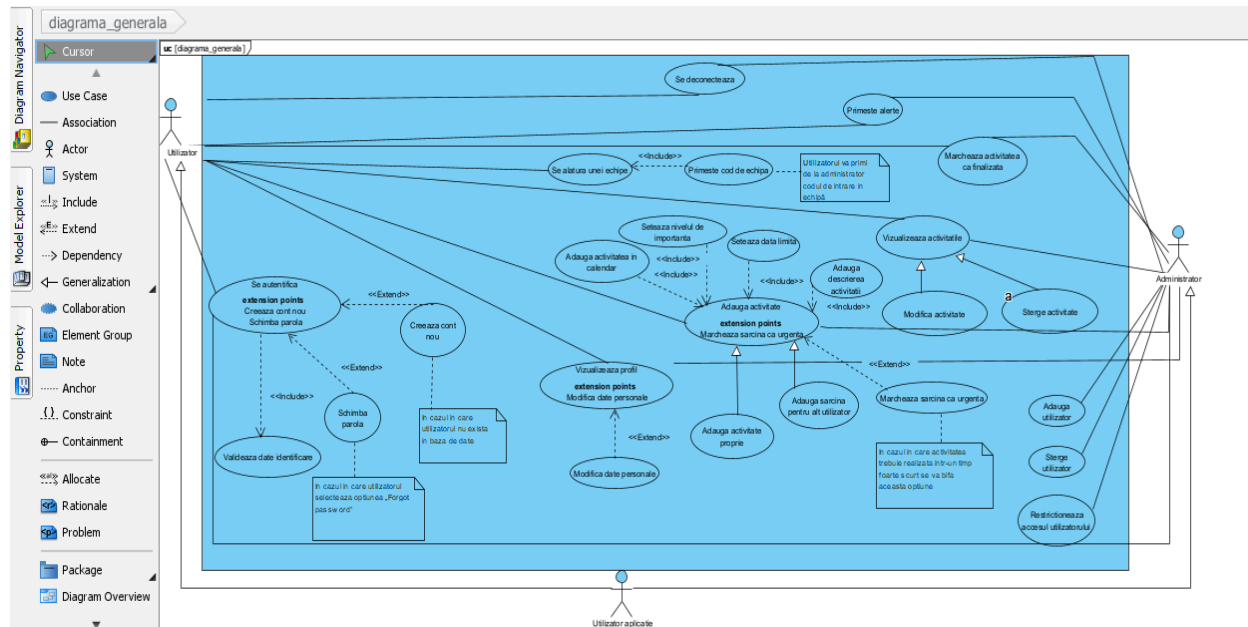


Figura 1.1. Diagrama generală a întregului sistem

Această diagramă prezintă principalele funcționalități ale aplicației web pentru organizarea activităților, evidențiind actorii și modurile în care aceștia pot interacționa cu sistemul. Sunt prezenți doi actori, ambii generalizând un utilizator standard al aplicației: utilizatorul principal, care este angajatul și are dreptul să adauge activități în liste și administratorul, care, pe lângă atribuțiile unui simplu utilizator, are dreptul de a adăuga sau șterge alți utilizatori.

- Diagrama generală a sistemului de organizare destinat utilizatorului

- Descrierea textuală a cazului general de utilizare

Element al cazului de utilizare	Descriere
Cod	CU01
Stare	Schiță
Scop	Eficientizarea organizării activităților dintr-o firmă
Nume	Gestionarea sarcinilor angajaților unei firme
Actor principal	Utilizator aplicație (Utilizator, Administrator)
Descriere	Folosirea aplicației pentru a gestiona activități reale din cadrul unei firme, pentru organizarea eficientă a acestora într-un mod plăcut vizual, posibilitatea de a asigna sarcini atât individual cât și altor colegi din echipă, accesarea unui calendar care conține toate activitățile personale viitoare
Precondiții	Utilizatorul trebuie să își creeze cont și să se autentifice în aplicație. Un utilizator poate accesa mediul de lucru din aplicație specific firmei odată ce este angajat în firmă și a primit un cod de intrare în echipă de la administrator.
Postcondiții	Sarcinile sunt vizualizate, gestionate și actualizate conform acțiunilor utilizatorului. Orice modificări sau actualizări sunt salvate și reflectate în sistem.
Declanșator	Un utilizator accesează aplicația web pentru a se autentifica și a-și organiza activitățile zilnice
Flux de bază	Utilizatorul își creează cont, se autentifică în aplicație, vizualizează task-urile, adaugă sarcini noi, primește notificări pentru activități viitoare, bifează finalizarea sarcinilor.
Fluxuri alternative	Utilizatorul modifică o sarcină existentă, își poate vizualiza propriul profil din aplicație, poate schimba parola și gestiona detaliile contului
Relații	Se extinde prin cazul de utilizare pentru autentificare.
Frecvența utilizării	La fiecare utilizare a aplicației
Reguli de afaceri	Administratorul poate adăuga sau șterge alți utilizatori

- Diagrama textuală a cazului de organizare a activităților unui utilizator

Element al cazului de utilizare	Descriere
Cod	CU02
Stare	Schiță
Scop	Crearea și detalierea unei noi sarcini de lucru pentru a fi preluată și completată de persoana căreia îi este asignată, vizualizarea și completarea activităților viitoare proprii.
Nume	Organizarea proprie a activităților în aplicație
Actor principal	Utilizator
Descriere	Cazul descrie modul în care un utilizator interacționează cu aplicația web, prin crearea și gestionarea sarcinilor. Acesta poate vizualiza, adăuga, actualiza sau șterge sarcini, precum și să modifice gradul de prioritate sau termenele limită pentru task-uri.
Precondiții	Utilizatorul trebuie să își creeze cont, să se autentifice în aplicație și să primească un cod care permite intrarea în echipă de la administrator.
Postcondiții	Starea sarcinilor este actualizată de către sistem conform acțiunilor efectuate de utilizator.
Declanșator	Utilizatorul accesează aplicația pentru a-și gestiona activitățile și face parte dintr-o echipă.
Flux de bază	Utilizatorul își vizualizează sarcinile, poate adăuga sarcini noi, modifică activitățile existente, editează detaliile unei activități, marchează o sarcină ca finalizată.
Fluxuri alternative	Utilizatorul șterge o activitate din listă. Utilizatorul prelungește data limită pentru o activitate. Utilizatorul reasignează o activitate unei alte persoane decât cea setată inițial.
Relații	-
Frecvența utilizării	La fiecare utilizare a aplicației.
Reguli de afaceri	La adăugarea sau modificarea unei sarcini, sistemul verifică dacă termenul limită specificat este în viitor

2. Analiza sistemului informatic

2.1 Diagrame de activitate (3 diagrame)

- Diagrama de activitate pentru descrierea procesului de autentificare în aplicație

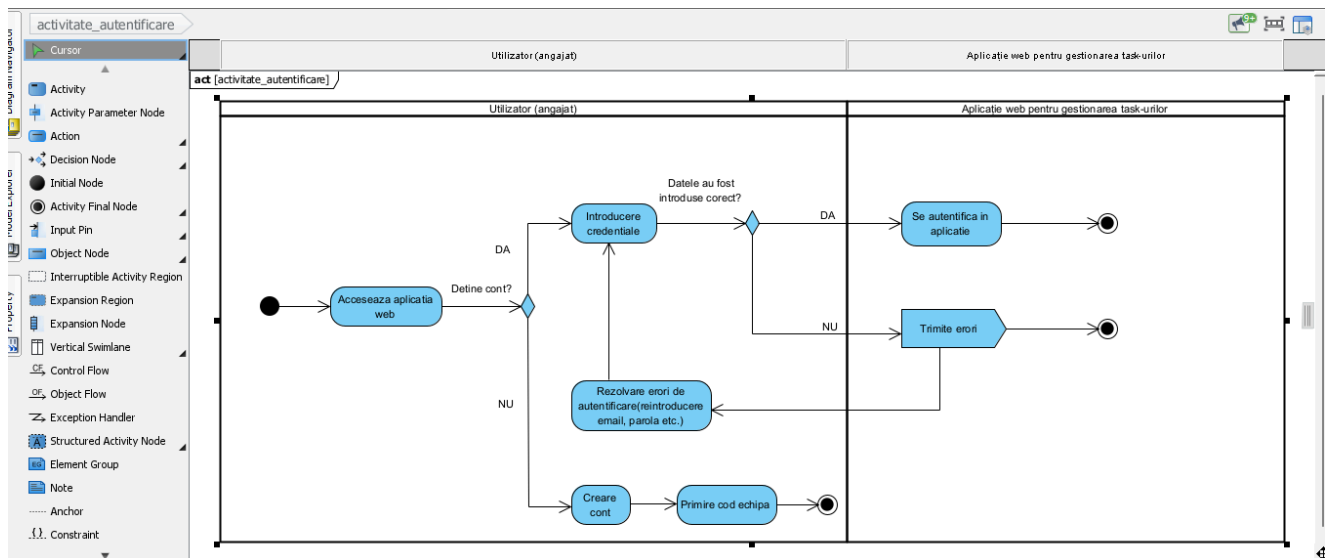


Figura 1.4. Diagrama de activitate pentru descrierea procesului de autentificare în aplicație

- Diagrama de activitate pentru descrierea activității de adăugare a unei sarcini

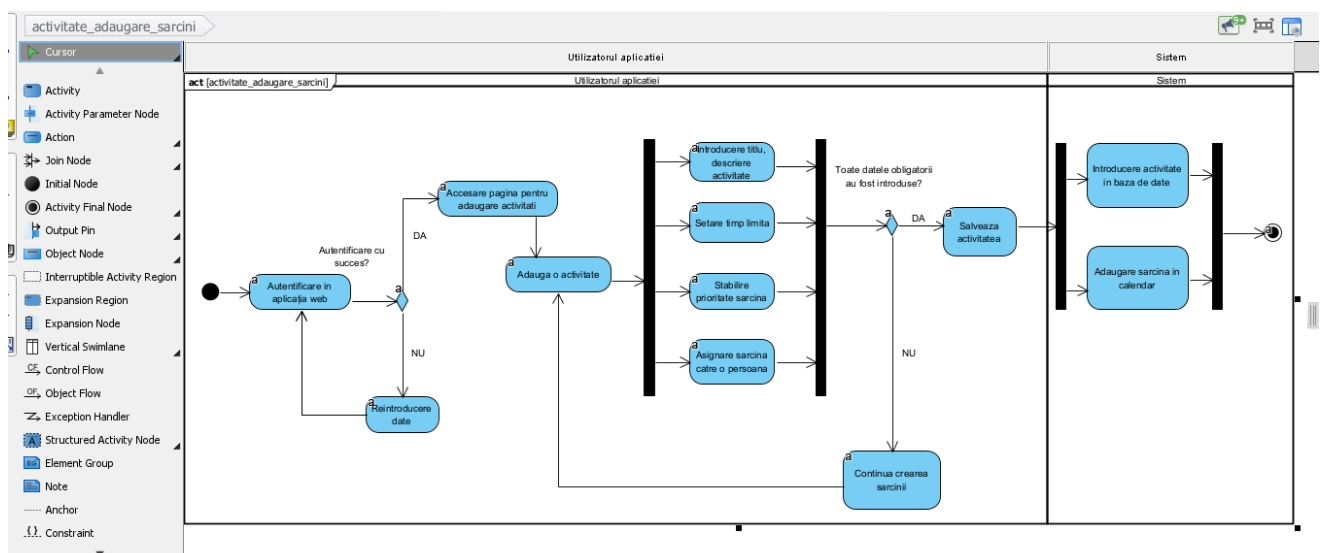


Figura 1.5. Diagrama de activitate pentru descrierea activității de adăugare a unei sarcini

- Diagrama de activitate pentru descrierea procesului de vizualizare și modificare a datelor personale în aplicație

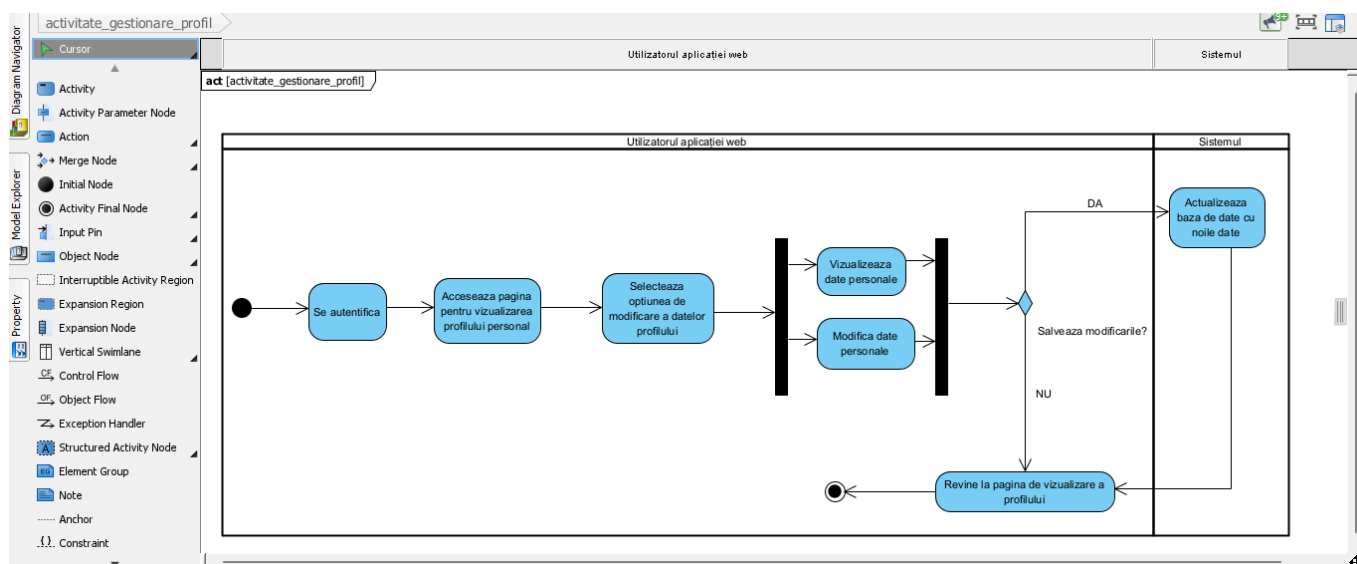


Figura 1.6. Diagrama de activitate pentru vizualizarea și modificarea a datelor personale ale utilizatorului

2.2 Diagrama de clase

Diagrama de clase este un tip de diagramă utilizată în realizarea sistemelor informatice pentru a ilustra structura statică a acestora. Diagrama oferă o perspectivă vizuală asupra claselor din cadrul sistemului, precum și a atributelor specifice și a relațiilor dintre acestea. Principalele componente ale diagramelor de clase sunt clasele, attributele lor, metodele din cadrul claselor, relațiile dintre entități și multiplicitățile acestora.

Aplicația conține 8 clase sau entități, după cum urmează, cu următoarele caracteristici:

Utilizator -clasa care reprezintă utilizatorul principal al aplicației, orice angajat al firmei care are cont în aplicație, acces la toate meniurile și poate planifica viitoarele activități.

Administrator -reprezintă utilizatorul care, pe lângă atribuțiile normale pe care le au alți utilizatori, acesta poate să șteargă sau să adauge noi utilizatori în echipă și poate oferi accesul în echipă pentru noii utilizatori

Activitate -clasa care conține toate attributele referitoare la o activitate planificată și înregistrată în sistem, precum titlul acesteia, descrierea integrală, data limită până la care poate fi efectuată, gradul de prioritate, persoana căreia i-a fost asignată.

Notificare -conține toate informațiile despre notificările pe care utilizatorii le primesc cu privire la date limită viitoare sau sarcini noi, are attribute precum ID-ul destinatarului și titlul notificării

Echipa -este clasa care conține attribute referitoare la lista utilizatorilor care se află într-o anumită echipă și numărul acestora

ManagerActivitati -reprezintă o interfață care va fi implementată atât de clasa Utilizator, cât și de clasa Administrator. Aceasta va conține toate metodele care se pot aplica pentru crearea

unei noi activități în sistem: adăugarea unei activități, ștergerea unei activități, modificarea unei activități, setarea timpului limită, setarea priorității sarcinii.

RaportActivitate -creată în special pentru a urmări câteva aspecte utile după finalizarea activității, precum dacă aceasta a depășit timpul limită acordat și care a fost durata sa totală de realizare.

IstoricSarcina -clasă care stochează informații cu privire la parcursul unei activități, istoricul utilizatorilor cărora le-a fost asignată aceasta, modificarea în timp descrierii sau a titlului acesteia și momentele de timp la care au fost făcute toate aceste modificări.

În diagrama de clase se poate observa relația de implementare a interfeței ManagerActivitati de către cele două clase, Utilizator și Administrator, care îi vor prelua metodele specifice. De asemenea, în cadrul diagramei am folosit atât asocieri simple între clase, cât și relații de agregare și compoziție. O relație de agregare se află între clasa Echipa și clasa Utilizator, întrucât utilizatorul poate exista și fără a face parte dintr-o anumită echipă. Astfel că, dacă am șterge echipa, utilizatorii ar fi în continuare înregistrați în aplicația web. Relația de compoziție se află atât între Activitate și RaportActivitate, cât și între Activitate și IstoricActivitate. Acest lucru se datorează faptului că, prin definiție, o relație de compunere este relația care evidențiază că entitățile care alcătuiesc clasa principală nu există independent de aceasta. Astfel, istoricul unei activități nu ar mai putea să existe în cazul în care activitatea respectivă nu există, și totodată nici raportul cu privire la aceasta.

În cadrul unei singure echipe pot exista unul sau mai mulți administratori ai acesteia, fapt evidențiat prin multiplicitatea reprezentată în asocierea simplă dintre clasele Administrator și Echipa.

Între clasa Administrator și clasa Utilizator există, de asemenea, o asociere simplă, iar relația de multiplicitate evidențiază faptul că unul sau mai mulți administratori ai unei echipe pot șterge sau adăuga unul, mai mulți utilizatori ai unei echipe, sau niciunul.

Multiplicitatea specificată în cadrul asocierii simple dintre Utilizator și Notificare arată faptul că un singur utilizator poate primi oricâte notificări de la sistem, asemenea și în cazul administratorului și al notificărilor sale.

Asocierea simplă dintre clasele Utilizator și Activitate și multiplicitatea acesteia nuanțează faptul că un singur utilizator poate crea și gestiona oricât de multe activități, sau niciuna.

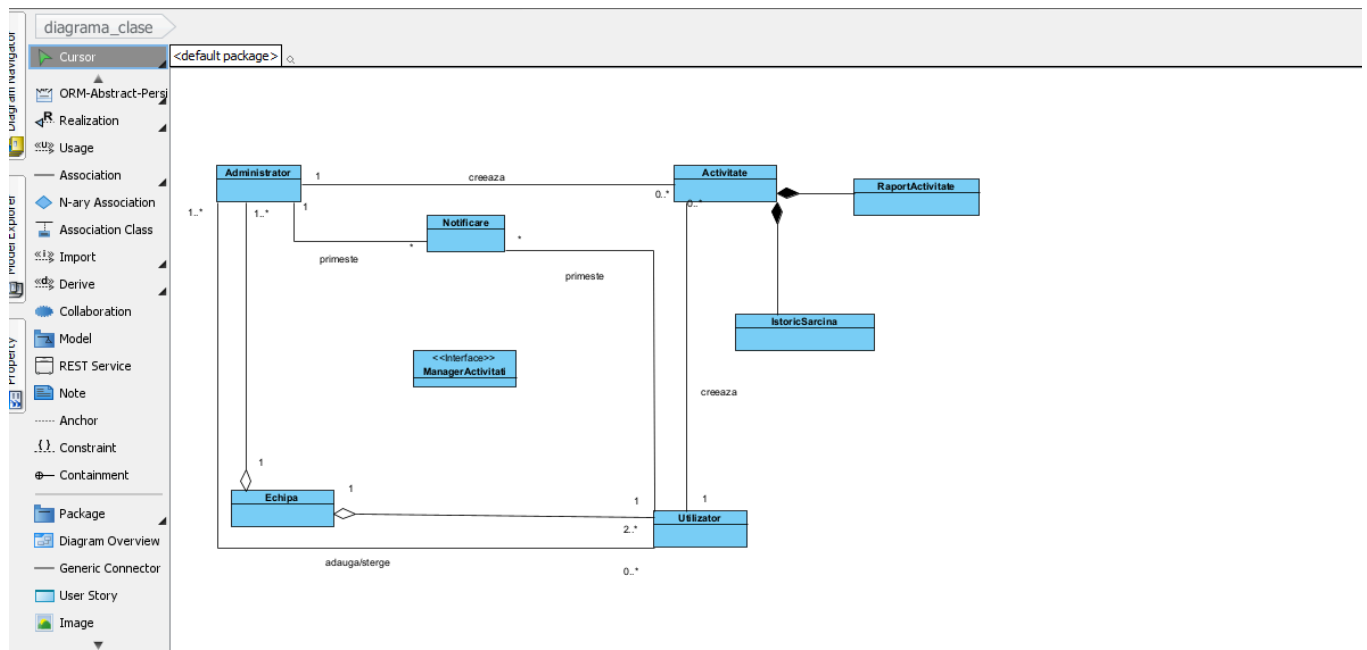


Figura 1.7. Diagrama de clase

În continuare, diagrama prezentată va fi utilizată pentru a construi diagrama detaliată de clase, care va cuprinde toate atributele și metodele claselor, având scopul final al generării întregii baze de date a proiectului.

2.3 Diagrame de stare (2 diagrame)

Diagramele de mașini cu stări modelează starea dinamică a unui obiect specific. O stare reprezintă o perioadă sau o situație din existența unui obiect care satisface în acel moment anumite condiții, efectuează anumite activități sau așteaptă anumite evenimente. Evenimentele reprezintă tranziția unui obiect dintr-o stare în alta. Cu alte cuvinte, o diagramă de stare este o reprezentare grafică a stărilor și tranzițiilor pe care un obiect le poate avea în cadrul unui sistem.

- 1) Diagrama de stare pentru descrierea tratării notificărilor de către utilizator

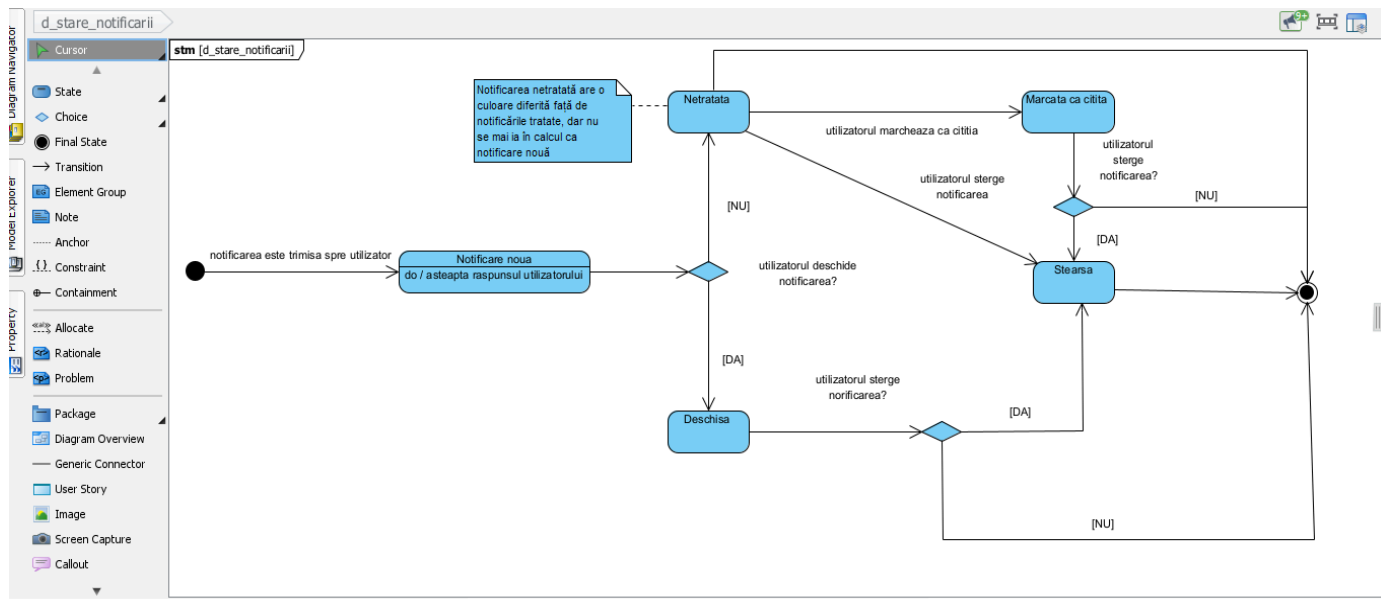


Figura 1.8. Diagrama de stare pentru descrierea tratării notificărilor

2) Diagrama de stare pentru descrierea ciclului de viață al activităților

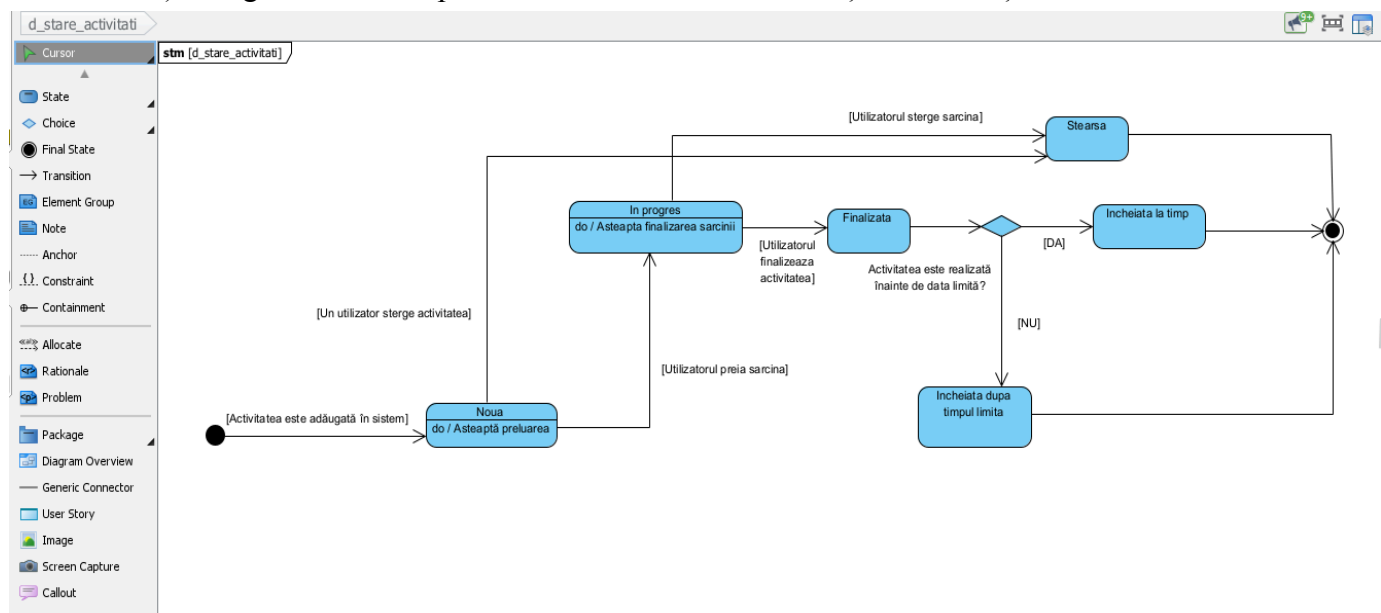


Figura 1.9. Diagrama de stare pentru descrierea ciclului de viață al activităților

2.4 Diagrame de interacțiune (3 diagrame)

Diagramele de interacțiune modelează aspectele dinamice ale sistemului. Ele sunt alcătuite dintr-un set de obiecte și din relațiile dintre ele, incluzând și mesajele pe care obiectele le transmit de la unul la altul. Cu alte cuvinte, diagrama de interacțiune este o reprezentare vizuală a interacțiunilor dintre diferitele entități în cadrul unui sistem informatic, concentrându-se în special pe comunicarea și schimbul de mesaje între acestea.

Există două tipuri principale de diagrame de interacțiune: diagrama de secvență și diagrama

de comunicare. Cele două diagrame sunt echivalente din punct de vedere semantic și se pot transforma una din alta.

- Diagrama de secvență care ilustrează procesul de autentificare în aplicație

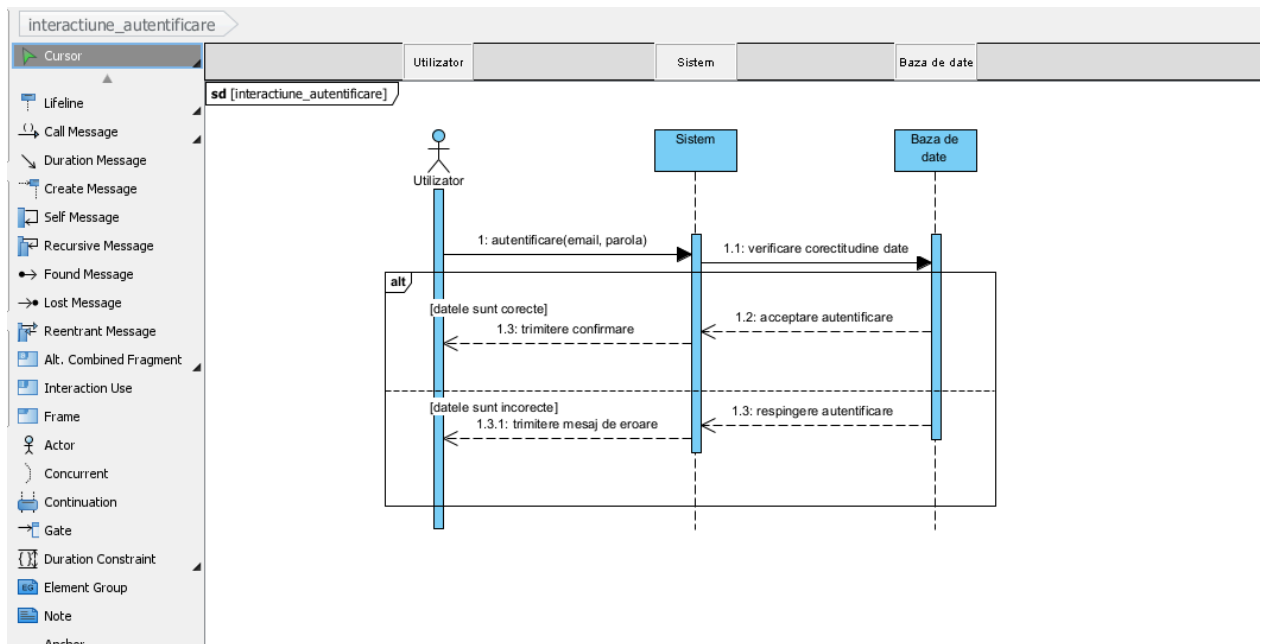


Figura 1.9. Diagrama de secvență care ilustrează procesul de autentificare în aplicație

- Diagrama de secvență pentru crearea și bifarea unei activități

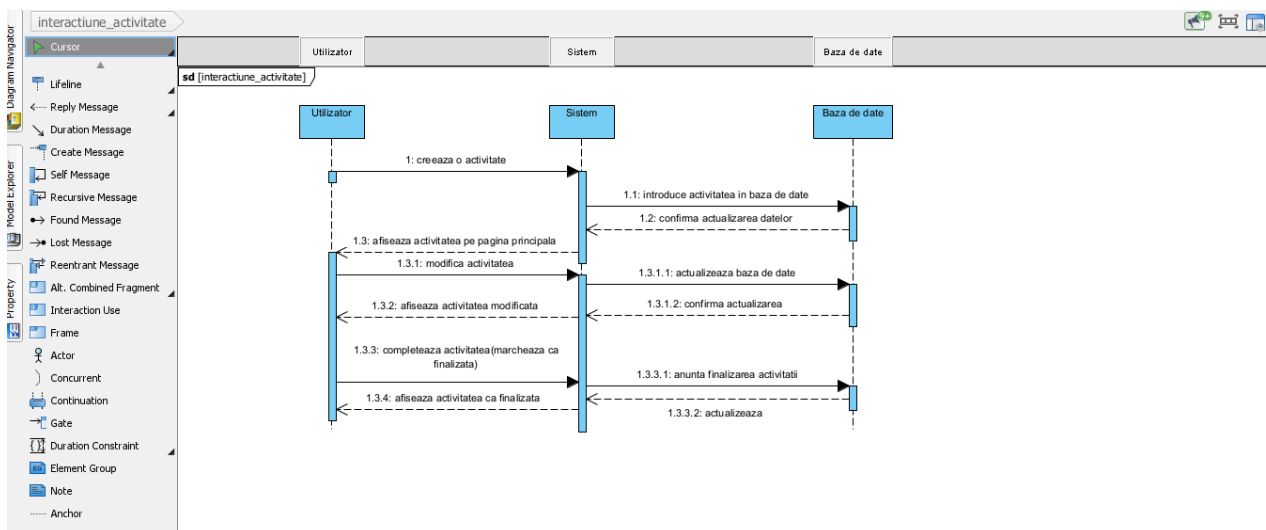


Figura 2.1. Diagrama de secvență pentru crearea și bifarea unei activități

- Diagrama de comunicare care evidențiază procesul de ștergere din echipă de către administrator a unui utilizator

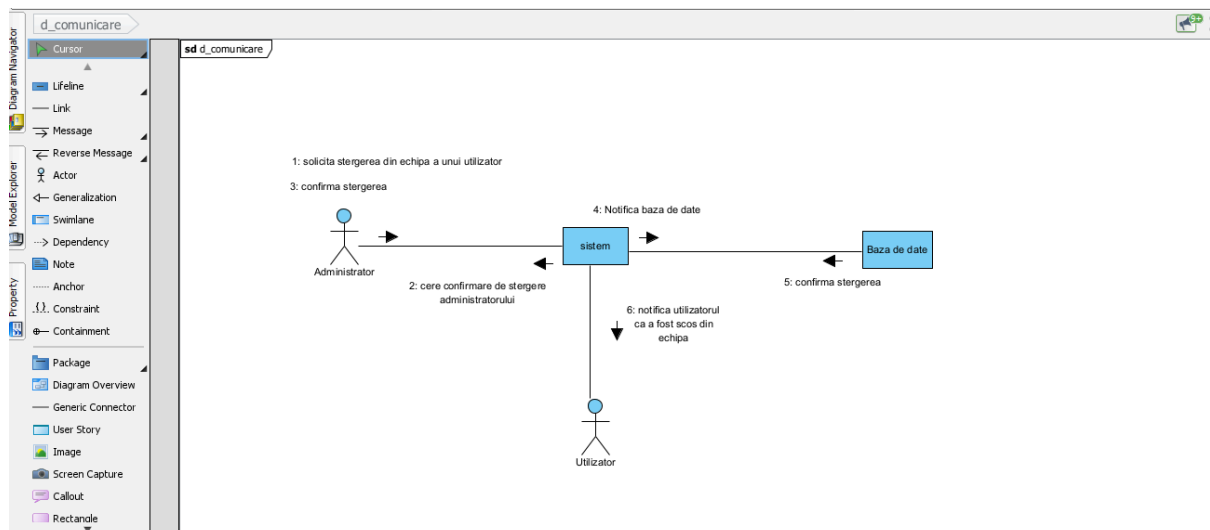


Figura 2.2. Diagrama de comunicare care evidențiază procesul de ștergere din echipă de către administrator a unui utilizator

2.5 Diagrame de procese și colaborare în BPMN (o diagramă procese, o diagramă colaborare)

Diagramele de proces permit optimizarea proceselor din lumea reală, prin simularea lor. Elementele care stau la baza creării unei astfel de diagrame sunt evenimentele, activitățile, fluxurile de frecvență și simbolurile pentru reprezentarea datelor. În cele ce urmează, am realizat o diagramă de procese și o diagramă de colaborare, folosind Visual Paradigm Enterprise Edition.

Diagrama de colaborare evidențiază o interacțiune sincronizată a două sau mai multe procese.

- Diagrama de proces pentru activitatea utilizatorului de adăugare a unei sarcini

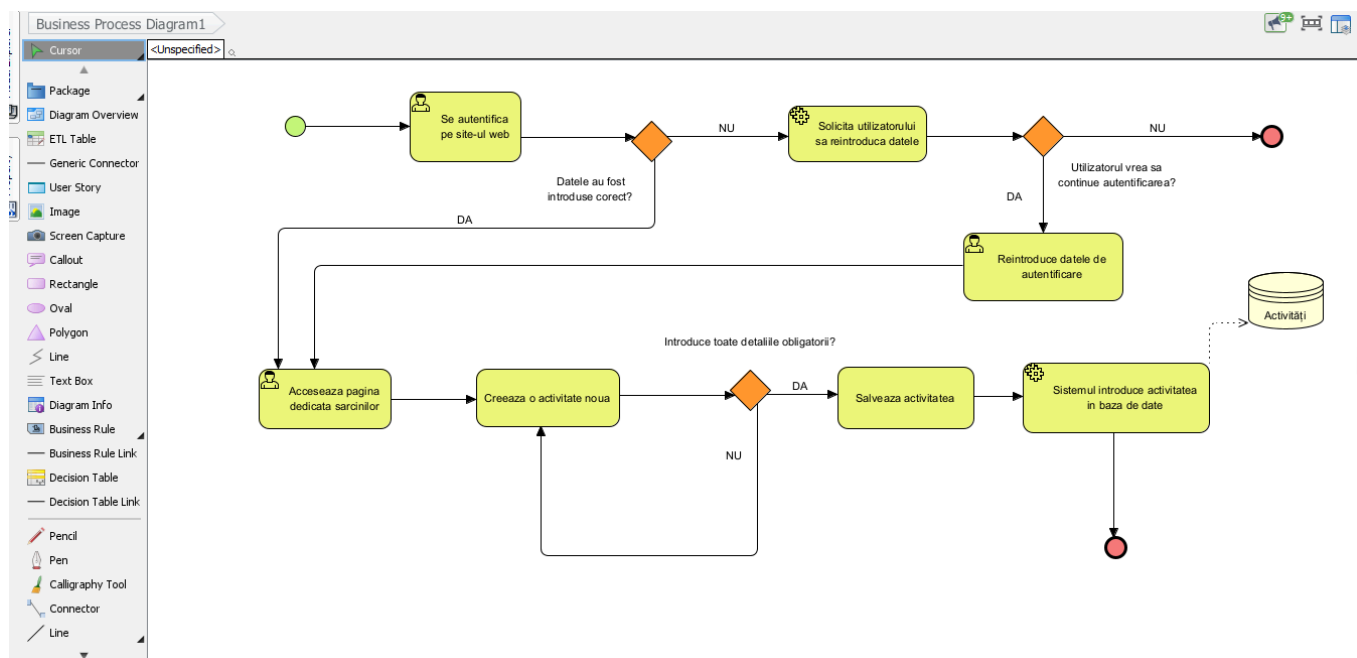


Figura 2.3. Diagrama de proces pentru activitatea utilizatorului de adăugare a unei sarcini

➤ Diagrama de colaborare care evidențiază procesul de modificare a sarcinilor

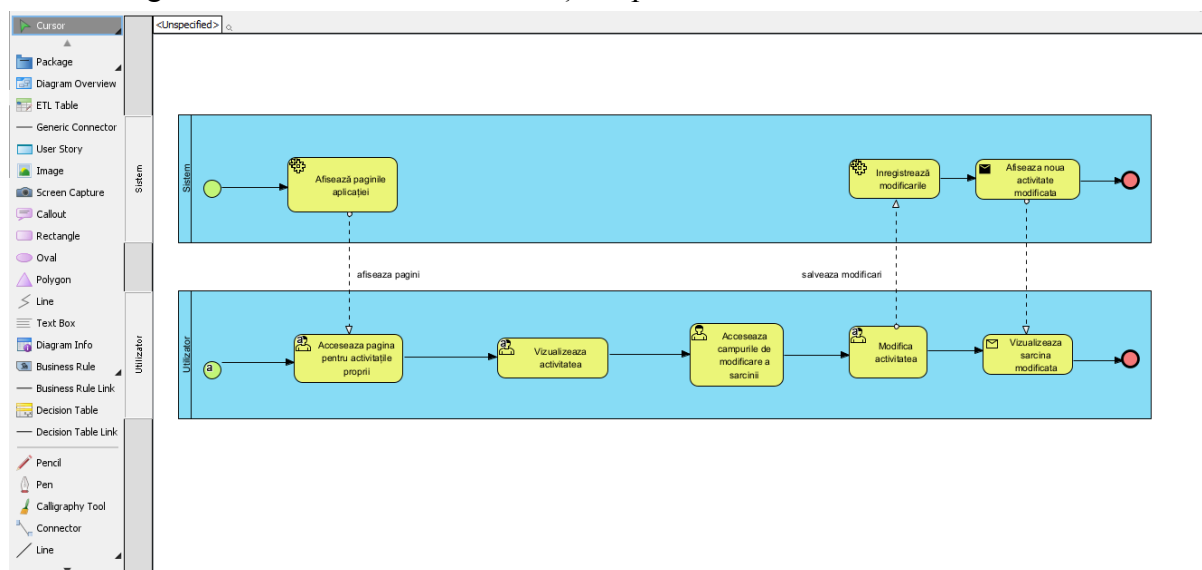


Figura 2.4. Diagrama de colaborare pentru procesul de modificare a sarcinilor

3. Proiectarea sistemului informatic

3.1 Diagrama de clase detaliată

Diagrama de clase detaliată se constituie din clasele prezentate anterior, împreună cu atributele și metodele din cadrul acestora. Aceasta oferă în acest mod o imagine complexă și detaliată a comportamentelor claselor și în concordanță cu celelalte clase.

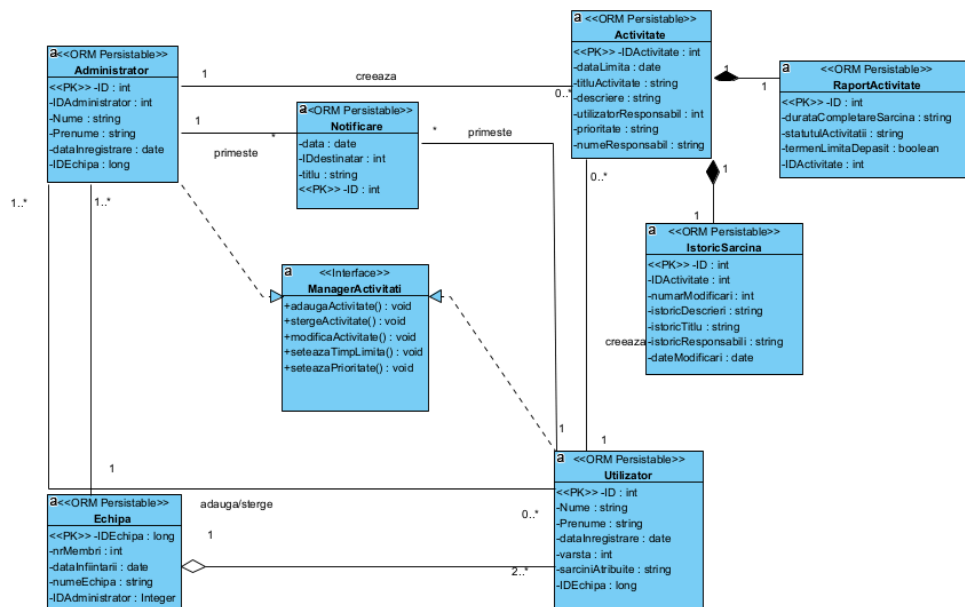


Figura 2.5. Diagrama detaliată a claselor

3.2 Proiectarea bazei de date

Proiectarea bazei de date presupune mai multe etape, precum analiza cerințelor textuale, modelarea conceptuală, stabilirea entităților și a atributelor, a cheilor primare și secundare, toate acestea fiind urmate mai apoi de proiectarea fizică a bazei de date specifice proiectului.

Pentru crearea bazei de date a proiectului, au fost urmați următorii pași:

1) Identificarea entităților din cadrul bazei de date

După cum am reprezentat și în diagrama de clase detaliată, entitățile din sistemul informatic vor fi: Utilizator, Administrator, Echipa, Activitate, Notificare, IstoricSarcina, RaportActivitate.

2) Identificarea relațiilor dintre entități

În acest pas se vor stabili asocierile dintre entități și cardinalitatea acestora (1:1, 1:M, M:M). Astfel, în baza de date se pot face următoarele asocieri:

Un administrator poate crea una, mai multe activități sau niciuna, iar o singură activitate poate fi creată de un singur administrator. Astfel încât, între cele două entități există o relație de tipul one-to-many. Similar, între entitățile Utilizator și Activitate există același tip de relație, one-to-many.

O echipă poate avea un singur administrator, iar un administrator poate administra una sau mai multe echipe. Astfel, se regăsește o relație de tip one-to-many între cele două entități.

Un utilizator poate primi oricât de multe notificări, iar o singură notificare poate fi primită de un singur utilizator. Relația dintre cele două entități este una de tipul one-to-many, similar cazului dintre entitățile Administrator și Notificare.

O echipă poate avea cel puțin doi utilizatori, iar un utilizator poate face parte din mai multe echipe simultan. Astfel, între cele două există o relație de tipul many-to-many.

O activitate poate avea un singur istoric al său, iar istoricul aparține unei singure activități. Similar, în urma unei activități va reieși un singur raport de tipul specificat, iar raportul va fi specific unei singure activități. Astfel, în ambele cazuri se recunoaște o relație între entități de tipul one-to-one.

3) Stabilirea atributelor fiecărei entități

Pentru acest pas, se va detalia fiecare entitate în parte, prin prezentarea atributelor acesteia. Atributele entităților dintr-o bază de date sunt folosite pentru a defini și detalia caracteristicile sau informațiile asociate cu acele entități. Ele reprezintă coloanele din tabelele bazei de date și sunt fundamentale pentru a descrie și organiza datele în mod coerent. Am stabilit totodată atributele cu rolul de cheie primară și secundară ale entităților. Cheia primară este atributul sau gruparea de attribute care identifică în mod unic fiecare înregistrare dintr-o tabelă. Cheia secundară este atributul sau gruparea de attribute dintr-o entitate, care reprezintă cheia primară a unei alte entități.

-Utilizator

Atribute: *IDAdministrator* (cheie primară) (întreg), *Nume*(șir de caractere), *Prenume*(șir de caractere), *dataInregistrare* (dată calendaristică), *sarciniAtribuite* (șir de caractere), *varsta* (întreg), *IDEchipa* (cheie externă)(întreg)

Administrator

Atribute: *IDAdministrator* (cheie primară) (întreg), *Nume* (șir de caractere), *Prenume* (șir de caractere), *dataInregistrare* (dată calendaristică), *IDEchipa* (long)

Activitate

Atribute: *IDActivitate*(cheie primară)(întreg), *dataLimita* (dată calendaristică), *titluActivitate*(șir de caractere), *descriere*(șir de caractere), *utilizatorResponsabil*(întreg)(cheie secundară), *prioritate*(șir de caractere), *numeResponsabil*(șir de caractere)

Echipa

Atribute: *IDEchipa*(cheie primară)(long), *IDAdministrator*(cheie secundară), *nrMembri*(int), *dataInfiintarii*(dată calendaristică), *numeEchipa*(șir de caractere)

Notificare

Atribute: *ID*(cheie primară)(întreg), *data*(dată calendaristică), *IDdestinatar* (cheie secundară) (întreg), *titlu*(șir de caractere)

IstoricSarcina

1) Pagina de autentificare

Planningly

https://planningly.com

Welcome to Planningly!

Let's plan today!

Email:

Password:

☐ Keep me logged in

No account? [Create Account](#)

Forgot password? [Change password](#)

Figura 2.7. Interfața pentru pagina de autentificare

2) Pagina pentru vizualizarea sarcinilor proprii

A Web Page

https://planningly.com

Planningly

Your tasks Finished tasks Team tasks [Create new task](#)

Do an important thing [See more](#) 7 Jan 2024 HIGH In progress

Do another important thing [See more](#) 10 Jan 2024 MEDIUM New

<< < 1 of 3 > >>

You have 21 tasks!

Figura 2.8. Interfața pentru vizualizarea sarcinilor proprii

3) Afișarea notificărilor în pagină

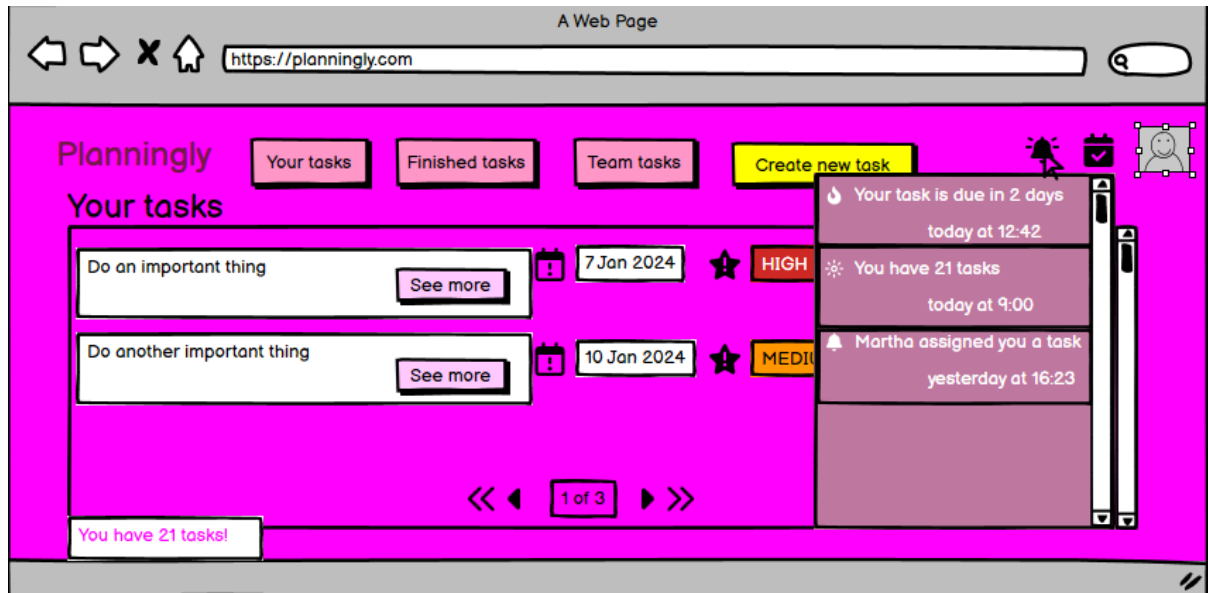


Figura 2.9. Afișarea notificărilor în pagină

4) Pagina de adăugare a unei noi sarcini

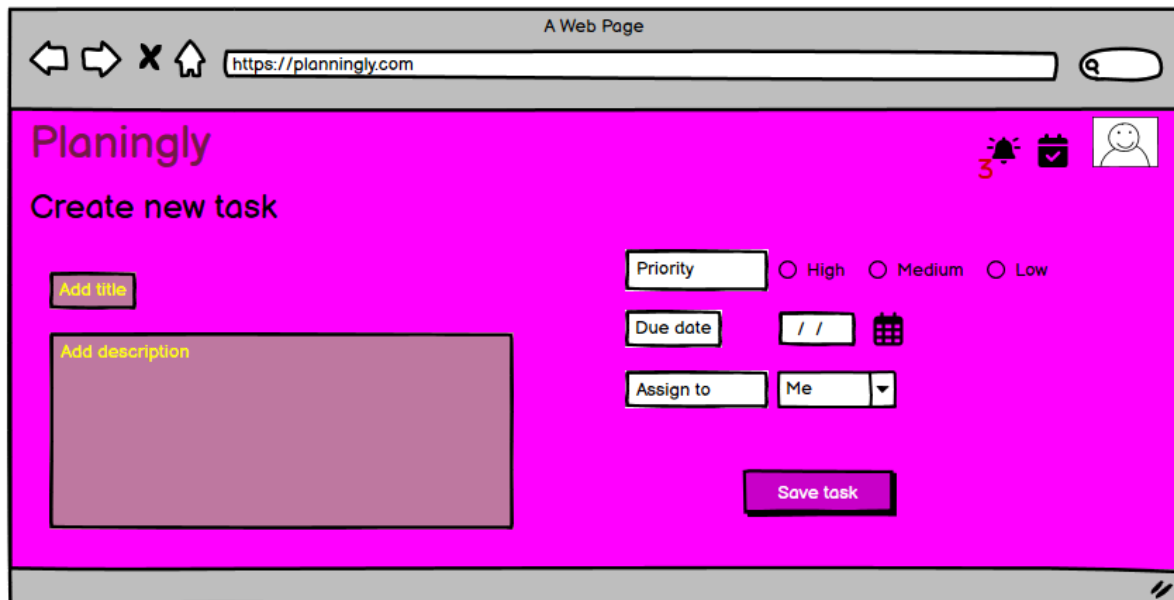


Figura 3.1. Interfața pentru pagina de adăugare a unei sarcini noi

Proiectarea interfețelor pentru utilizator a fost realizată folosind Balsamiq Wireframes.

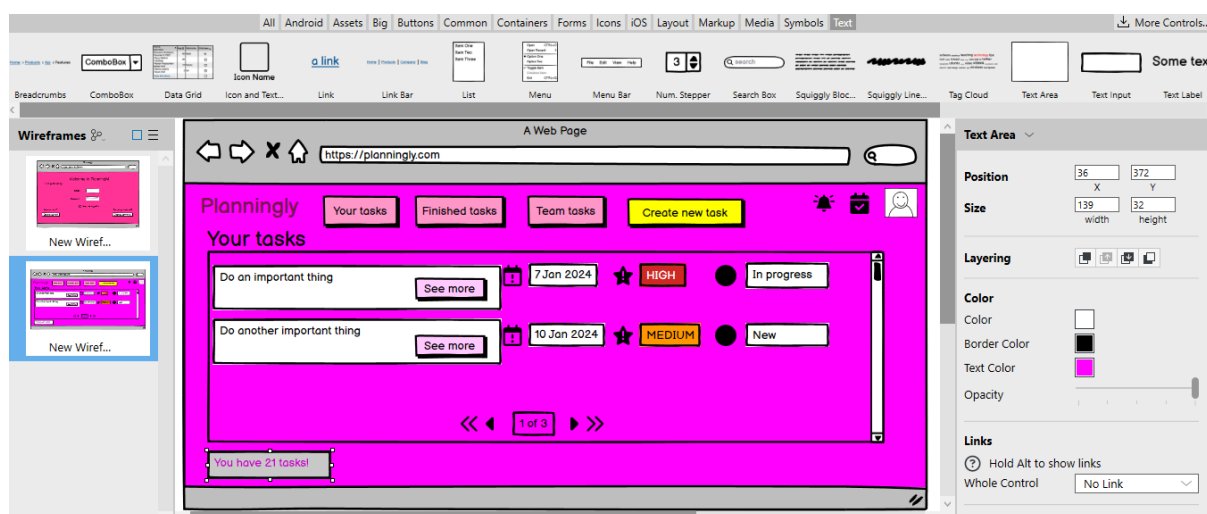


Figura 3.2. Utilizarea Balsamiq Wireframes

3.4 Diagrama de componente

Diagrama de componente reprezintă dependențele existente între diverse componente care alcătuiesc sistemului informatic proiectat. Aceste dependențe pot fi statice, cazul în care au loc în etapele de compilare sau de link-editare, sau dinamice, care au loc la momentul execuției. Așadar, oferă o perspectivă amplă asupra structurii fizice a unui sistem și a interacțiunii acestuia cu alte componente sau părți ale sistemului.

Această diagramă modelează arhitectura de ansamblu, identificând componentele sistemului și interfețele acestuia.

3.5 Diagrama de desfășurare

O diagramă de desfășurare reprezintă relațiile dintre componentele hardware utilizate în infrastructura fizică a sistemului informatic.

4. Implementarea sistemului informatic

4.1 Tehnologii utilizate în implementare

Referitor la tehnologiile utilizate în implementare, pentru sistemul informatic voi folosi limbajele de programare HTML și CSS pentru partea de front end, pentru structură și stilizare, alături de framework-urile React și Bootstrap pentru un aspect vizual cât mai plăcut pentru utilizatori. În ceea ce privește partea de back-end pentru implementarea funcționalităților aplicației, voi folosi limbajul de programare JavaScript în mediul de execuție Node.js și framework-ul Express.js pentru interactivitate și dezvoltarea serverului. Pentru stocarea și gestionarea datelor introduse în sistem voi folosi baza de date MySQL.

4.2 Prezentarea pe scurt a funcționalității sistemului

După cum am prezentat în paginile anterioare, sistemul informatic ce urmează a fi proiectat are ca principal scop eficientizarea organizării sarcinilor zilnice ale angajaților unei firme și facilitarea comunicării dintre aceștia, indiferent de funcția pe care o ocupă sau de departamentul din care fac parte. Totodată, sistemul urmărește redarea transparenței în firmă, astfel încât un utilizator poate vedea activitățile viitoare ale altor utilizatori și își pot programa reciproc diferite de sarcini de lucru.

Astfel, pentru a îndeplini acest scop final descris, sistemul va dispune de numeroase funcționalități care vor permite efectuarea acțiunilor utilizatorilor. În același timp, aplicația web va avea o interfață prietenoasă, intuitivă și ușor de înțeles, astfel încât să nu îi pună și mai mult în dificultate pe utilizatorii sistemului, ci să îi ajute să își organizeze și să își finalizeze sarcinile într-un timp cât mai scurt.

Funcționalitatea de bază de care va dispune sistemul este posibilitatea de a adăuga și gestiona sarcinile, prin accesarea paginii special dedicate task-urilor și utilizarea butoanelor de creare și a casetelor de descriere a sarcinilor. Apoi, vor avea loc scenariile de utilizare reprezentate anterior prin diagrame.

Utilizatorii vor putea urmări progresul sarcinilor și îl vor putea seta în moduri diferite, de la „sarcină nouă” la „sarcină în progres” și ulterior „sarcină finalizată”. Setarea termenelor limită va fi utilă pentru a stabili rapiditatea cu care vor trebui realizate unele sarcini. Pentru a seta termenele limită, utilizatorilor li se va deschide o fereastră calendar unde vor putea alege ziua și ora limită pentru fiecare activitate. Dacă nu o vor seta manual, aceasta va fi setată implicit cu o dată limită de 3 zile. După cum am precizat, utilizatorii își vor putea atribui reciproc sarcini de lucru împreună cu datele limită pentru acestea.

În scopul personalizării interfețelor corespunzătoare paginilor pe care le vor accesa utilizatorii, aceștia vor avea posibilitatea de a schimba tematica de culori și fonturile vizibile în contul personal, fapt care va duce la creșterea entuziasmului acestora pentru utilizarea aplicației și în acest fel creșterea productivității personale.

Partajarea informațiilor între angajați va fi facilitată prin existența butoanelor alăturate casetelor de descriere pentru activități, prin intermediul cărora vor putea atașa alte documente utile în realizarea activității respective.

În ceea ce privește notificările, utilizatorii vor primi astfel de alerte atunci când li se asignează noi sarcini sau în cazurile în care se apropie termenul limită pentru o anumită activitate sau au depășit deja data limită a unei activități.

Stocarea tuturor datelor introduce în sistem, referitoare la utilizatori, la datele personale ale acestora, la echipa din care fac parte, la toate activitățile introduse, la istoricul sarcinilor și la notificările pe care el le primesc, vor fi stocate într-o bază de date și se va asigura totodată securitatea datelor.

Așadar, sistemul informatic creat pentru organizarea activităților dintr-o firmă va aduce numeroase beneficii pentru utilizatorii săi, angajații, care își vor spori productivitatea și motivația pentru sarcini. Vizualizarea clară a programului de lucru va contribui la respectarea termenelor limită, iar monitorizarea progresului referitor la fiecare activitate va ajuta la evaluarea performanțelor individuale și ale echipei.

Bibliografie

- <https://www.visual-paradigm.com/tutorials/how-to-draw-state-machine-diagram-in-uml/>
- <https://balsamiq.com/tutorials/articles/firstwireframe/>
- <https://www.visual-paradigm.com/tutorials/how-to-draw-bpmn-diagram/>
- <https://circle.visual-paradigm.com/diagram-examples/>
- <https://blog.hubspot.com/marketing/business-requirement-document>
- materialele puse dispoziție pe platforma online.ase.ro