**ROMÂNIA**

**MINISTERUL APĂRĂRII NAȚIONALE**

**ACADEMIA TEHNICĂ MILITARĂ ,,FERDINAND I”**

**FACULTATEA DE SISTEME INFORMATICE ȘI SECURITATE CIBERNETICĂ**

**Specializare: Calculatoare și Sisteme Informatice pentru Apărare și Securitate Națională**



**SOLUŢIE DE SECURIZARE A SERVICIILOR INTEGRATE ŞI ANONIMIZAREA ACCESULUI LA INFRASTRUCTURI VIRTUALIZATE**

CONDUCĂTOR ȘTIINȚIFIC:

**Col. prof. univ. dr. ing. Mihai TOGAN**

ABSOLVENT:

**Student plt. Ionuţ-Alexandru PAVEL**

Conţine \_\_\_\_\_\_\_\_\_\_\_ file

Inventariat sub nr. \_\_\_\_\_\_

Poziţia din indicator: \_\_\_\_

Termen de păstrare: \_\_\_\_\_

**BUCUREŞTI**

**2022**

# ABSTRACT

Collaboration tools involve multiple types of software or services that allow people to work toghether. Collaboration tools can route work through a process, distribute pieces and tasks to involved parties, and help to coordinate activities. This paper presents the implementation of the web-based collaborative real-time application using ReactJs and NodeJs frameworks. The application provides a secure, controlled environment for real-time document review, coauthoring and redaction. Also, it allows the creation of private or group conversation beetween users and allows text message transfer. Documents can be saved in encrypted format using server storage and personal data is stored using MySql relational database. Sending messages is implemented using WebSocket computer communications protocol because of the need to keep a stable connection between users and to achieve real-time message transmission. Messages are persistent data, being stored in the database together with user data. The main purpose of the project is to document and develop an application that facilitates teamwork through technological solutions.

# REZUMAT

Cuprins

[ABSTRACT 5](#_Toc104454128)

[REZUMAT 6](#_Toc104454129)

[LISTĂ DE ABREVIERI 8](#_Toc104454130)

[TABELĂ FIGURI 9](#_Toc104454131)

[1. Introducere 11](#_Toc104454132)

[1.1 Prezentare generală 11](#_Toc104454133)

[1.2 Importanţa temei 11](#_Toc104454134)

[1.3 Utilitatea sistemului 12](#_Toc104454135)

[2 Stadiul actual 12](#_Toc104454136)

[2.1 Sisteme similare 12](#_Toc104454137)

[2.2 Dezvoltarea aplicaţiilor web 12](#_Toc104454138)

[2.3 Tehnologii folosite în dezvoltarea aplicaţiilor de tip colaborativ?? 12](#_Toc104454139)

[3 API-uri folosite 12](#_Toc104454140)

[3.1 React 12](#_Toc104454141)

[3.2 NodeJS 12](#_Toc104454142)

[3.3 MySQL 12](#_Toc104454143)

[3.4 Socket.Io 12](#_Toc104454144)

[3.5 Axios 12](#_Toc104454145)

[4 Structura proiectului 13](#_Toc104454146)

[4.1 Cerinţele sistemului 13](#_Toc104454147)

[4.1.1 Cerinţe funcţionale 13](#_Toc104454148)

[4.1.2 Cerinţe non-funcţionale 13](#_Toc104454149)

[4.2 Arhitectura sistemului 13](#_Toc104454150)

[4.2.1 Modulele funcţionale ale sistemului 13](#_Toc104454151)

[4.2.2 Arhitectura aplicaţiei client 13](#_Toc104454152)

[4.2.3 Arhitectura aplicaţiei server 13](#_Toc104454153)

[4.2.4 Persistenţa datelor 13](#_Toc104454154)

[5 Implementarea software 13](#_Toc104454155)

[5.1 Aplicaţia client 13](#_Toc104454156)

[5.2 Aplicaţia web server 13](#_Toc104454157)

[5.2.1 Răspunderea request-urilor 13](#_Toc104454158)

[5.3 Implementarea bazei de date 13](#_Toc104454159)

[5.3.1 Diagramă 13](#_Toc104454160)

[5.4 Metode de securizare 13](#_Toc104454161)

[5.4.1 Securizarea canalului de comunicaţie 13](#_Toc104454162)

[5.4.2 Criptarea fişierelor stocate 13](#_Toc104454163)

[5.4.3 Protecţie împotriva atacurilor CSRF 13](#_Toc104454164)

[5.4.4 Controlul accesului prin firewall 13](#_Toc104454165)

[5.4.5 Alertare în cazul încercărilor neautorizate 13](#_Toc104454166)

[6 Utilizarea aplicaţiei 14](#_Toc104454167)

[6.1 Actorii sistemului 14](#_Toc104454168)

[6.2 Cazurile de utilizare 14](#_Toc104454169)

[6.3 Testarea sistemului 14](#_Toc104454170)

[6.4 Raport de testare 14](#_Toc104454171)

[6.5 Diagrame UML 14](#_Toc104454172)

[6.5.1 Diagrama cazurilor de utilizare 14](#_Toc104454173)

[6.5.2 Diagrama de activităţi 14](#_Toc104454174)

[7 Concluzii 14](#_Toc104454175)

[7.1 Analizarea rezultatelor obţinute 14](#_Toc104454176)

[7.2 Dezvoltări ulterioare 14](#_Toc104454177)

[8 Bibliografie 14](#_Toc104454178)

[9 Anexe 14](#_Toc104454179)

# LISTĂ DE ABREVIERI

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# TABELĂ FIGURI

# 1. Introducere

## Prezentare generală

Proiectul îşi propune implementarea unei soluţii software de tip aplicaţie web cu scopul de a asigura activitatea colaborativă în regim de mobilitate şi în mod securizat pentru membrii prezenţi într-o organizaţie.

Soluţia propusă va permite interacţiunea dintre utilizatori folosind mediul online în următoarele scenarii:

1. Inter-schimbarea de mesaje, documente sau fişiere multimedia;
2. Stocarea de fişiere la distanţă şi accesul lor din aplicaţie;
3. Editarea de text real-time în mod colaborativ;
4. Realizarea de apeluri video la nivel de grup.

Pentru îndeplinirea cerinţelor tehnice enumerate mai sus am decis să recurg la implementarea unei arhitecturi web de tip Client-Server. Astfel, soluţia software propusă este împărţită în două părţi mari componente: aplicaţia server şi aplicaţia client. Pentru implementarea aplicaţiei server am folosit Node.js, platformă de dezvoltare open source pentru executarea codului JavaScript. Avantajele acestui mediu de rulare sunt capabilitatea de a gestiona un număr foarte mare de conexiuni simultane cu un debit ridicat de date, ceea ce echivalează cu o scalabilitate ridicată. De asemenea, un alt avantaj al acestei platforme este că permite comunicaţia în timp real prin intermediul protocolului WebSocket, fiind foarte folositor în aplicaţiile de tip mesagerie sau transfer de date. Pentru implementarea aplicaţiei client am folosit ReactJs, bibliotecă de JavaScript destinată dezvoltării de interfeţe web. Librăria ReactJs se remarcă prin simplitatea folosirii şi dezvoltarea componentelor modulare care pot fi reutilizare. Pentru persistenţa datelor am ales să folosesc o bază de date de tip MySql, datorită nevoii de a reţine relaţiile de interacţiune dintre utilizatori sub forma unei scheme bine definite. Posibil adăugarea şi unei baze de date nerelaţionale pentru salvarea mesajelor (analizare MongoDB).

## Importanţa temei

Progresul tehnologic şi dorinţa micşorării timpul de muncă împreună cu nevoia oameniilor de a menţine comunicaţia la distanţă a dus la dezvoltarea de software colaborativ care să satisfacă aceste cerinţe într-o manieră uşor de gestionat pentru utilizator. Astfel software-ul colaborativ este dezvoltat cu scopul de a ajuta persoanele care lucrează la o sarcină comună să îşi atingă obiectivele.

Organizaţiile guvernamentale lucrează cu date clasificate şi există riscul ca informaţia să fie vulnerabilă dacă sunt folosite soluţii software externe, chiar dacă acestea sunt puse la dispoziţie de către dezvoltatori consacraţi. În acelaşi timp există riscul ca o aplicaţie destinată publicului să colecteze date cu privire la utilizator, precum locaţia acestuia, adresa IP sau activitatea acestuia la nivelul dispozitivului ăe care este folosită. Astfel, apare necesitatea unei soluţii dezvoltate şi administrate de organizaţia în cauză, pentru a reduce riscul colectării şi interceptării datelor.

## Utilitatea sistemului

Soluţia propusă are scopul de a asigura activitatea colaborativă la nivelul unei organizaţii şi de a satisface reglementările de securitate prin prelucrarea şi stocarea sigură a informaţiilor, fără a folosi alte servicii software externe. Comunicaţia dintre utilizatori se realizează folosind mesaje de tip text criptate, schimbul de fişiere sau apeluri video de tip conferinţă. Un alt obiectiv este ca aplicaţia să fie uşor de folosit şi de a putea fi folosită în condiţii de mobilitate fără necesitatea instalării de programe software pe resursele hardware utilizate.

# Stadiul actual

## Sisteme similare

## Dezvoltarea aplicaţiilor web

## Tehnologii folosite în dezvoltarea aplicaţiilor de tip colaborativ??

# API-uri folosite

## React

## NodeJS

## MySQL

## Socket.Io

## Axios

# Structura proiectului

## Cerinţele sistemului

### Cerinţe funcţionale

### Cerinţe non-funcţionale

## Arhitectura sistemului

### Modulele funcţionale ale sistemului

### Arhitectura aplicaţiei client

#### Diagrame clase

### Arhitectura aplicaţiei server

#### Diagrame Clase

### Persistenţa datelor

#### Diagrama Baza de date

# Implementarea software

## Aplicaţia client

## Aplicaţia web server

### Răspunderea request-urilor

## Implementarea bazei de date

### Diagramă

## Metode de securizare

### Securizarea canalului de comunicaţie

### Criptarea fişierelor stocate

### Protecţie împotriva atacurilor CSRF

### Controlul accesului prin firewall

### Alertare în cazul încercărilor neautorizate

# Utilizarea aplicaţiei

## Actorii sistemului

## Cazurile de utilizare

## Testarea sistemului

## Raport de testare

## Diagrame UML

### Diagrama cazurilor de utilizare

### Diagrama de activităţi

# Concluzii

## Analizarea rezultatelor obţinute

## Dezvoltări ulterioare

# Bibliografie

# Anexe