

**ConnectUs**

Document de proiectare arhitecturală

# **Cuprins**

## **1.Introducere**

- **1.1 Scopul sistemului**
- **1.2 Definiții, acronime**
- **1.3 Documente referințe**

## **2. Obiective de proiectare**

## **3. Arhitectura propusă**

- **3.1 Prezentare generala a arhitecturii sistemului**
- **3.2 Decompozitia în subsisteme și responsabilitățile fiecărui subsistem**
- **3.3 Distribuția subsistemelor pe platforme hardware/software**
- **3.4 Managementul datelor persistente**
- **3.5 Controlul accesului utilizatorilor la sistem**
- **3.6 Condițiile limita**

## 1.Introducere

- **1.1 Scopul sistemului**

ConnectUs e o rețea socială care conectează oamenii. Scopul produsului este să dea informații utilizatorilor despre evenimente și persoane pe care le cunosc sau s-ar putea să le cunoască. Utilizatorii ConnectUS pot să adauge prieteni, împartă video-uri pe care vor că prietenii lor să le vadă, încarce poze, comenteze la postările prietenilor, să comunice cu prietenii lor și să afle informații despre ce fac prietenii lor. Utilizatorii pot crea grupuri private unde participanții pot fi informați despre evenimente de către alți utilizatori ai grupului său de prieteni.

- **1.2 Definiții, acronime**

Backend - îndeplinește o sarcina specifica bazată pe datele obtinute de la frontend.

Frontend - este responsabil cu descărcarea datelor de la utilizator în diferite forme și transferarea acestora către backend.

React - o bibliotecă JavaScript open-source pentru construirea de interfețe de utilizator.

PostgreSQL - este un sistem open-source de baze de date relaționale.

Node.js - este un sistem open-source care execută cod JavaScript în afara unui browser web.

- **1.3 Documente referințe**

Software Requirement Document - ConnectUs

React - <https://reactjs.org/>

PostgreSQL - <https://www.postgresql.org/>

## 2. Obiective de proiectare

Programul va îndeplini următoarele cerințe funcționale principale: crearea unui cont, autentificare, căutarea de prieteni, completare profil, postare media/text, comentarea la postari, posibilitatea de like/dislike a postarilor. De asemenea, utilizatorul își va putea adăuga prieteni in lista sa de prieteni și să creeze grupuri cu aceștia pentru o interacțiune între utilizatori cât mai simpla. Utilizator va putea interacționa cu alți utilizatori din lista sa de prieteni prin mesageria instantanee.

Scopul acestui document de proiectare arhitecturala este de a prezenta elementele pe baza cărora o sa fie construită aplicația finală, conform documentului de specificație a cerințelor.

### 3. Arhitectura propusă

- **3.1 Prezentare generală a arhitecturii sistemului**

Arhitectura sistemului se folosește de modelul client-server, care distribuie sarcini între furnizorii de resurse sau servicii numite servere și solicitanții de servicii, numite clienți.

Clienții și serverele interacționează printr-un model de schimb de mesaje de tip request/response. Clientul trimite o cerere, iar serverul returnează un răspuns cu cererea clientului.

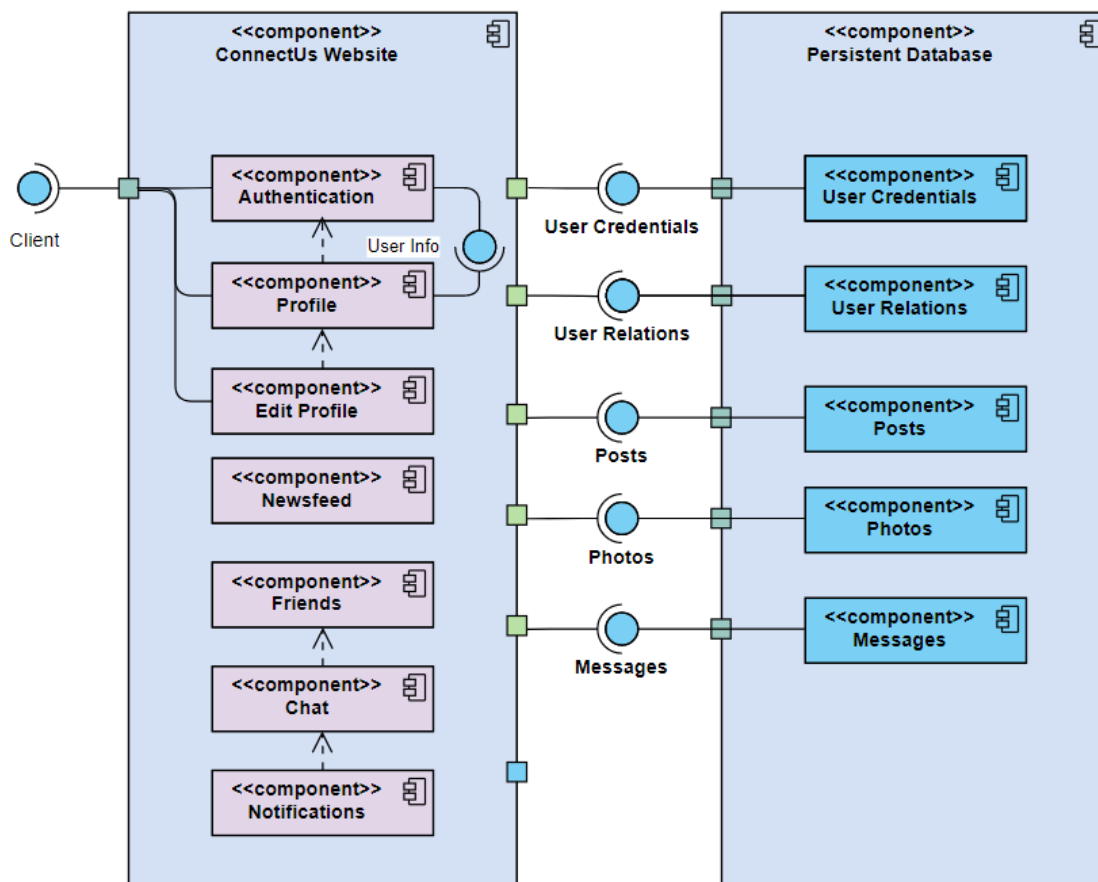
- **3.2 Decompozitia în subsisteme și responsabilitățile fiecărui subsistem**

Sistemul este împărțit în următoarele subsisteme:

- Server
- Client
- Baza de date

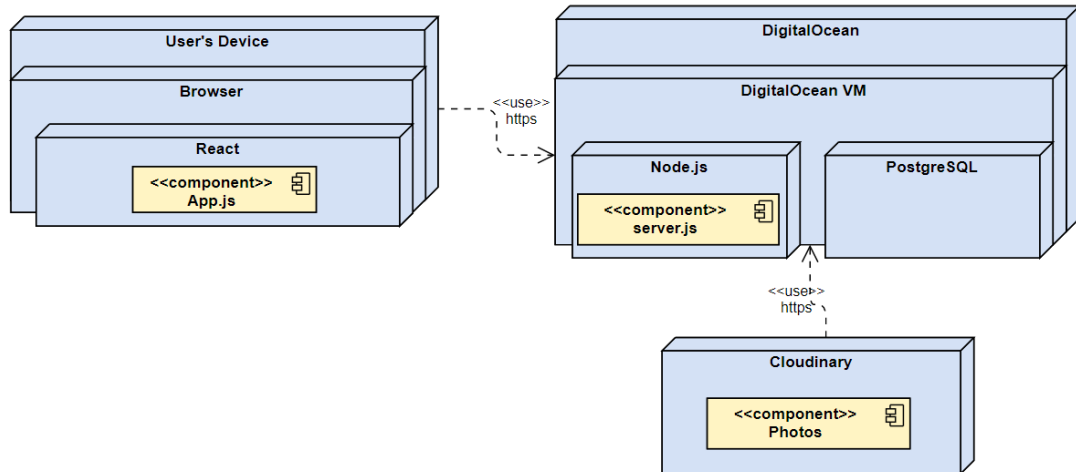
Subsistemele rezultate depind unul de celălalt, iar fiecare are diferite responsabilități definite astfel:

- Clienții apelează servicii oferite de server.
- Serverul are rolul de a răspunde la cererile făcute de client prin interactionarea cu baza de date. Acesta trebuie sa extraga informatii din baza de date în funcție de cererile clientului sau chiar sa adauge informații noi, cum ar fi în cazul unui cont nou, sau să șteargă din baza de date anumite informații. De asemenea, acesta conecteaza utilizatorii care folosesc mesageria instantanee.
- Baza de date stochează datele.



### • 3.3 Distribuția subsistemelor pe platforme hardware/software

Acest sistem este conceput cu scopul de a oferi functionalitati minimele cu cerințe de sistem minime. În consecință distribuirea sistemului pe platforme nu necesita un hardware performant, subsistemele solicitand o configuratie minima atat hardware cat și software. Utilizatorul trebuie să aibă acces la internet și să aibă acces la un dispozitiv cu un browser, precum Android, iOS, Windows, Linux, etc.

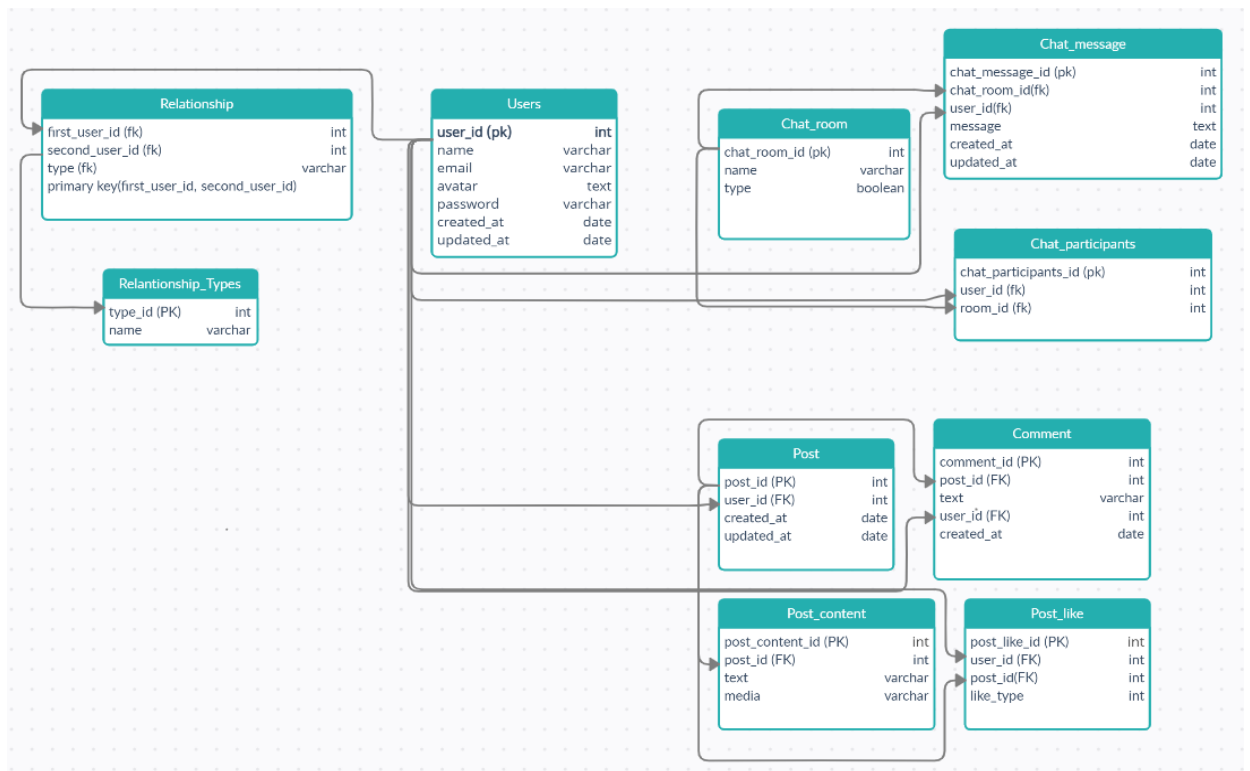


### • 3.4 Managementul datelor persistente

Datele persistente ale sistemului sunt reprezentate de datele despre utilizatori, informații personale sau relații de prietenie, și postările sau comentariile făcute de aceștia. În acest scop, backend-ul va fi realizat folosind PostgreSQL și Node.js, în timp ce pentru frontend se va folosi React. Proiectarea bazei de date urmărește următoarele aspecte:

- stocarea datelor într-un mod cat mai bine structurat pentru a evita schimbarea masiva a bazei de date in viitor.
- volum de date transferat minimal la fiecare cerere, astfel campurile sunt structurate în așa fel incat sa fie trimise mereu doar informațiile absolut necesare

Astfel, schema conceptuala a bazei de date este urmatoarea:



### ● 3.5 Controlul accesului utilizatorilor la sistem

Se consideră că utilizatorii vor fi orice persoana (indiferent de vârstă, sex, religie sau rasă) care știe să folosească un browser de internet, în cazul folosirii pe calculator, sau a unei aplicații mobile în situația utilizării de pe un telefon mobil. Aceștia nu sunt nevoiți să aibă cunoștințe avansate în domeniul calculatoarelor.

Utilizatorii au dreptul de a crea un cont, de a se loga și autentifica, de a căuta și adăuga alți utilizatori în lista de prieteni și de a transmite mesaje acestora. De asemenea, utilizatorii pot să își modifice propriul profil și să creeze postări de tip media, text. Aceștia pot să interacționeze cu postările altor utilizatori cu comentarii și like / dislike. Datele sensibile sunt reținute într-un mod criptat în baza de date pentru a preveni daunele provocate de un eventual atac.

### ● 3.6 Condițiile limita

Condițiile limită se referă la acordul dintre producător și utilizator referitoare la confidențialitatea datelor personale ale utilizatorului. Datele utilizatorului nu vor fi folosite în alte scopuri în afara de utilizarea platformei. Pentru a putea folosi ConnectUs, utilizatorul trebuie acceptat și să își asume respectarea condițiilor prezentate la crearea contului.