

Documentul de proiectare arhitecturala

Architectural Design Document

Structura generala a documentului

1. Introducere

1.1 Scopul documentului

Acest document reprezintă o privire în ansamblu a părții arhitecturale a sistemului, fiind văzută din mai multe aspecte: obiective de proiectare, decompoziția în subsisteme și distribuția acestora atât pe platforme hardware și software, cât și managementul datelor și accesul la sistem.

1.2 Scopul sistemului

InstantBFF este o aplicație ce oferă posibilitatea de a descoperi persoane cu pasiuni și interese comune. Sistemul se bazează pe o bază de date, unde sunt stocate informațiile despre fiecare utilizator în parte (date cu caracter personal, cât și preferințele legate de anumite aspecte). Acestea sunt folosite de către algoritmul de matching pentru a oferi unui utilizator persoanele cu gradul de potrivire cel mai mare.

1.3 Definiții, acronime

CSS - Cascading Style Sheets

HTML - HyperText MarkUp Language

hosting - When a hosting provider allocates space on a web server for a website to store its files

MVC - Model View Controller

MVVM - Model View ViewModel

MVP - Model View Presenter

DB - database

SDK - Software Development Kit

HTML - formă de marcare orientată către prezentarea documentelor text pe o singură pagină, utilizând un software de redare specializat, numit agent utilizator HTML, cel mai bun exemplu de astfel de software fiind browserul web.

1.4 Documente referințe

[Specificatia cerințelor LazyEdition](#)

[Architectural Design Document Example](#)

2. Obiective de proiectare

Descriu calități ale sistemului sau restricții manageriale, care trebuie urmărite în toate perioadele următoare ale dezvoltării.

Obiectivele proiectării se decid analizând cerințele nefuncționale.

- Exemple de obiective de proiectare:

- **Îndeplinirea cerințelor de performanță**: un anumit timp de răspuns al sistemului sau capacitatea sistemului de a executa un anumit număr de taskuri
- **Îndeplinirea cerințelor de calitate ale sistemului**: fiabilitatea, securitatea, siguranța în funcționare, ș.a.
- **Îndeplinirea cerințelor de operare cu sistemul**: comunicarea cu operatorii umani, ușurința de operare, ușurința de învățare a modului de operare, ș.a.
- **Îndeplinirea cerințelor de mentenanță** (adaptabilitatea, extensibilitatea, portabilitatea, ș.a)
- **Îndeplinirea cerințelor de planificare a proiectului**: termene, produse livrate, costuri.

- Obiectivele de proiectare pot fi prioritizate, atunci când nu toate pot fi îndeplinite, ținând cont de prioritizarea cerințelor nefuncționale.

Obiective de proiectare: Cum pot fi îndeplinite?

- ☐ Se stabilesc criteriile de proiectare care pot asigura îndeplinirea obiectivelor.
- ☐ Pentru fiecare obiectiv, se stabilesc:
 - Criterii primare (esențiale), care trebuie aplicate pentru atingerea obiectivului
 - Criterii secundare (de dorit), care dacă se aplică aduc un plus în realizarea obiectivului

Criterii de proiectare

- ❖ Principalele criterii de proiectare pot fi organizate în 5 grupuri:
 - Criterii de performanță
 - Criterii de încredere în sistem
 - Criterii de cost
 - Criterii de mentenanță
 - Criterii ale utilizatorului final

Criterii de performanță

- Asigurarea unui anumit timp de răspuns al sistemului la o cerere a utilizatorului
- Asigurarea unei anumite capacități de procesare (throughput): câte taskuri poate realiza sistemul într-o perioadă de timp specificată

Criterii de încredere (Dependability) - determină efortul care trebuie alocat în dezvoltare pentru a minimiza căderile sistemului și consecințele lor.

- **Cădere**: funcționare neconformă cu specificația.
- **Robustețea** (Robustness): capacitatea sistemului de a rezista (funcționa) la intrări neprevăzute.
- **Fiabilitatea** (Reliability): capacitatea de a funcționa conform comportamentului specificat.
- **Disponibilitatea** (Availability): procentul de timp în care sistemul poate fi folosit.
- **Toleranța la defecte** (Fault tolerance): capacitatea de a funcționa în condiții de cădere.
- **Securitatea** (Security): capacitatea de a rezista la atacuri cibernetice

Criterii de mentenanță - calități ale sistemului de care depinde ușurința de a fi modificat după livrare.

- **Extensibilitatea:** ușurința de a adăuga funcționalități sistemului. Ușurința de modificare a sistemului
- **Portabilitatea:** ușurința de a porta sistemul pe diferite platforme
- **Claritatea codului** (Readability): ușurința de a înțelege sistemul prin citirea codului sursă
- **Trasabilitatea cerințelor:** ușurința de a face corespondența între codul sursă și cerințe specifice.
- **Extensibilitatea**, ușurința de modificare, adaptabilitatea – se urmăresc la proiectarea arhitecturală a sistemului.

Criterii ale utilizatorului final

- **Utilitatea:** cât de mult îl ajută sistemul pe utilizator în viața de zi cu zi.
- **Ușurința de utilizare** (Usability): cât de ușor este pentru utilizator să folosească aplicația/website-ul

Criterii de cost: asigurarea respectării costurilor stabilite prin contractul cu clientul

- **Costul dezvoltării:** costul dezvoltării sistemului inițial
- **Costul actualizării** (upgrade) sistemului: costul modificării sistemului existent pentru a obține sistemul actualizat, asigurându-se compatibilitatea cu sistemul înlocuit (backward compatibility)
- **Costul mentenanței:** costul necesar pentru repararea defectelor sistemului și efectuarea de îmbunătățiri
- **Costul administrării:** costul necesar administrării sistemului

3. Arhitectura propusă

3.1 Prezentare generală a arhitecturii sistemul

Avem ca scop separarea interfeței cu utilizatorul de datele și logica aplicației.

Drept stil arhitectural am ales: MVC(**Model-View-Controller**).

Clientul este reprezentat de utilizatorii aplicației, cât și de administratorul aplicației și interacțiunea dintre aceștia. Ei pot realiza următoarele acțiuni, după cum urmează:

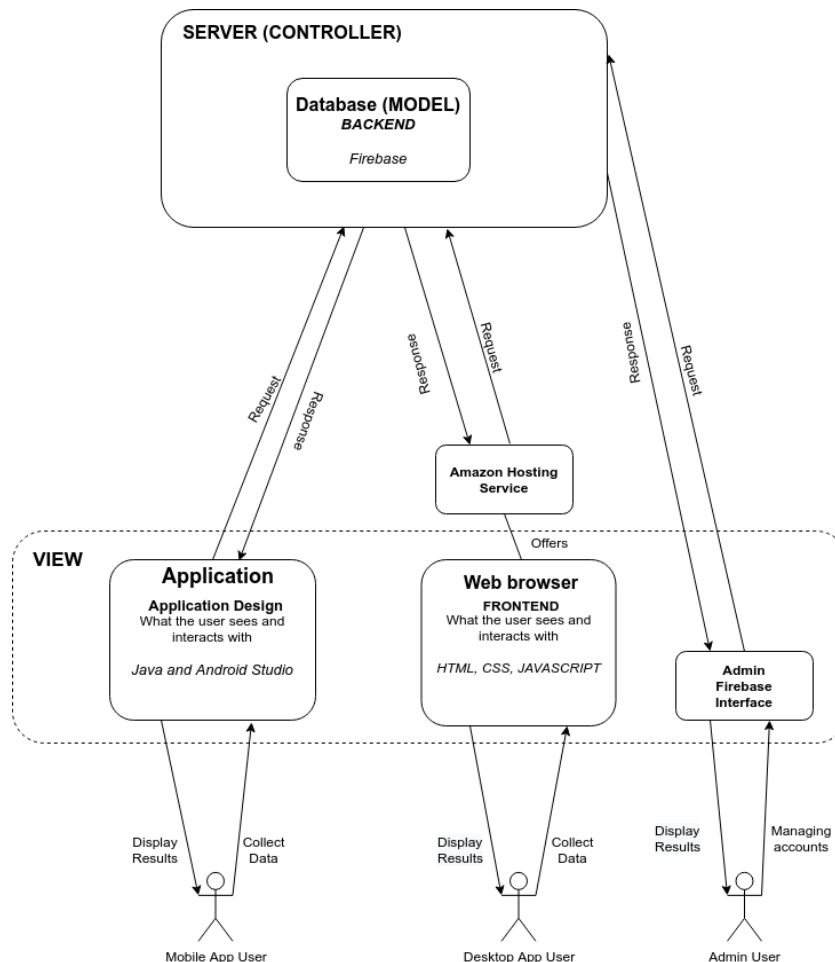
Utilizatorul aplicației:

- își va crea un cont
- își va face un profil în care va menționa: nume, o poza, vârsta, sexul și paginile de socializare
- va da un test de personalitate pentru a putea să își găsească alți utilizatori cu interese comune
- dacă ambii utilizatori sunt de acord se vor afișa paginile de socializare.

Administratorul aplicației:

- se va asigura că aplicația merge bine
- poate să șteargă utilizatori care pun în pericol siguranța aplicației și a utilizatorilor acestora.

Din punct de vedere al funcționalității, ne propunem ca aplicația să poată susține un număr erabil de utilizatori. Astfel, pentru partea de backend vom folosi un database, iar pentru partea tend JavaScript și HTML/CSS/React.



O arhitectura MVC este alcătuită din 3 tipuri de subsisteme:

Subsistemul **Model** are următoarele responsabilități:

- reprezentare a datelor specifice aplicației
- operațiile de acces la date
- operațiile de actualizare a datelor, conform interacțiunii cu utilizatorul și logicii aplicației

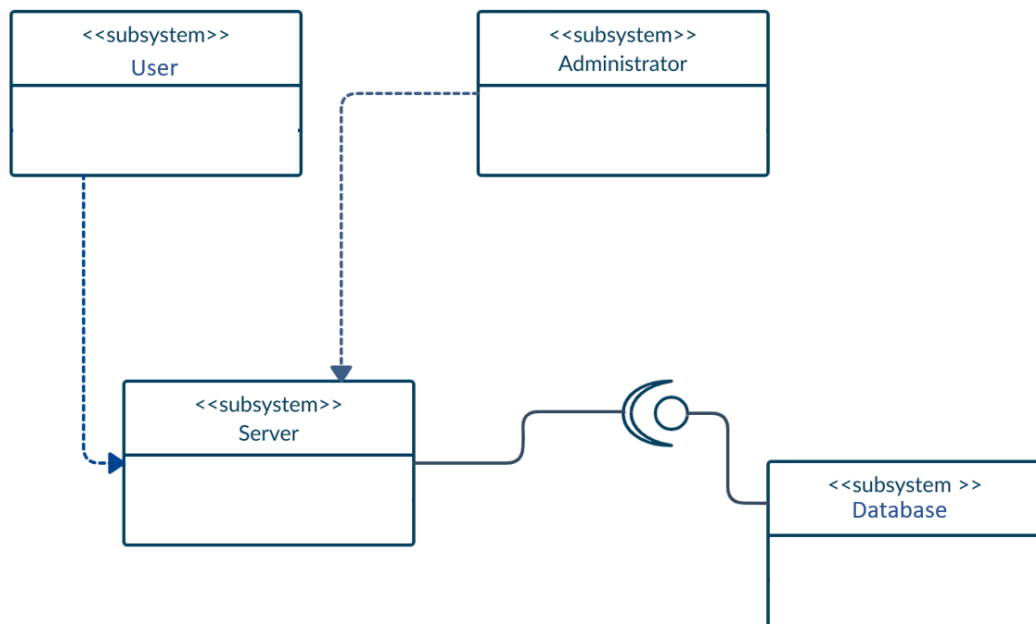
Subsistemul View:

- ecran prin care utilizatorul interacționează cu sistemul
- transmite Controller-ului mesaje eveniment generate de interacțiunea cu utilizatorul
- actualizează prezentarea datelor aplicației în fereastră, la orice modificare a lor

Subsistemul Controller:

- procesează evenimentele generate de interacțiunea cu utilizatorul
- transmite mesaje la Model pentru accesarea sau modificarea datelor în conformitate cu interacțiunea utilizatorului
- transmite mesaje către View pentru a actualiza design-ul prezentat pe ecran

3.2 Decompoziția în subsisteme și responsabilitățile fiecărui subsistem



În figura de mai sus se descrie relațiile dintre subsistemele principale ale aplicației. Atât subsistemul **User**, cât și cel **Administrator** depind de subsistemul **Server** întrucât folosesc funcțiile implementate de acesta. Subsistemul server se conectează la interfața pusă la dispoziție de subsistemul **Database**.

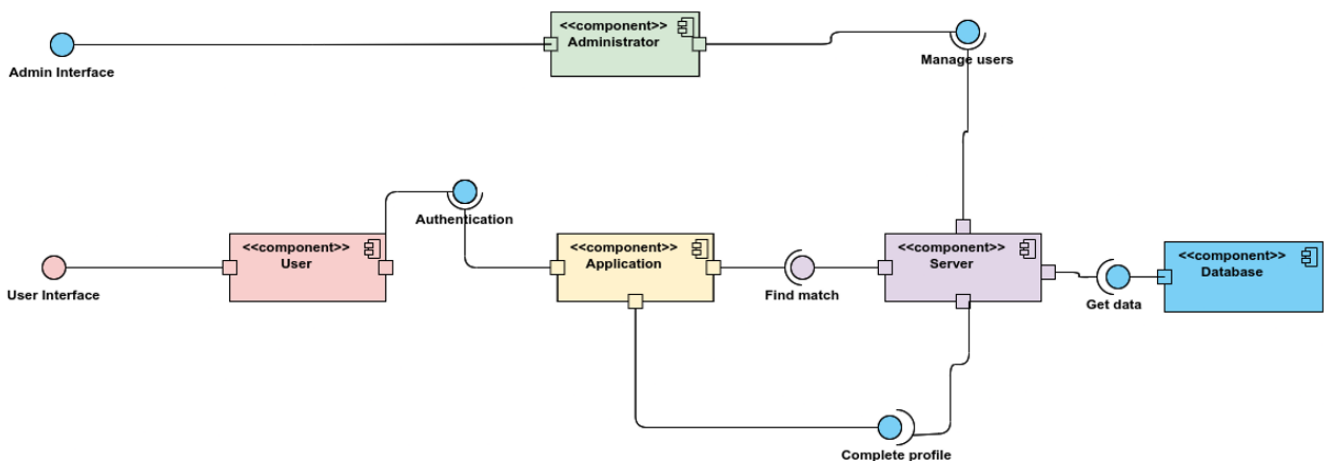


Diagrama de componente prezintă modul de relaționare prin care se realizează legătura între utilizator/administrator în momentul interacțiunii cu componenta aplicație, respectiv server.

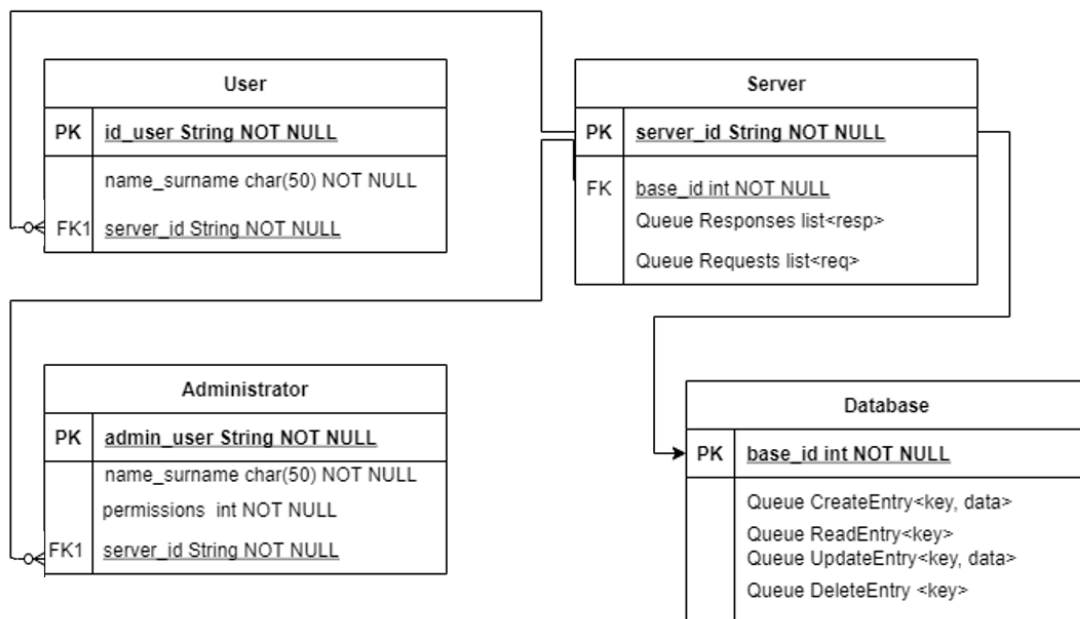
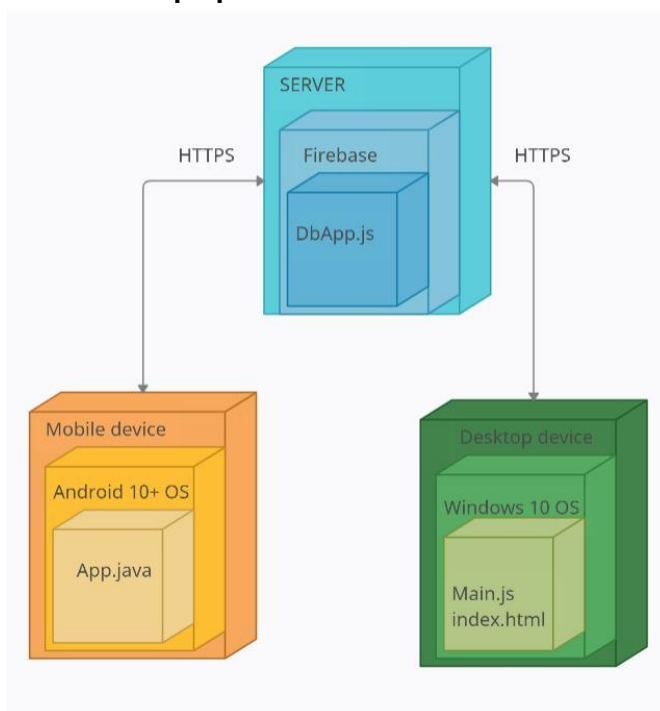


Diagrama ERD are rolul de a prezenta modul în care interacționează datele în sistemul propus.

3.3 Distribuția subsistemelor pe platforme hardware/software



Subsistemul **Model** si **Controller** se va afla pe o platforma hardware, pe serverele celor de la Google deoarece folosim Google Firebase.

Subsistemul **View** este unul software, care se afla pe dispozitivul fiecărui utilizator.

3.4 Managementul datelor persistente

Aplicația noastră va avea date care nu se distrug atunci când aplicația mobilă este închisă sau atunci când website-ul a fost închis.

Datele fiecărui utilizator pot fi regăsite și actualizate în cursul mai multor executări.

Datele vor fi disponibile pe fiecare dispozitiv pe care se autentifică utilizatorul respectiv. Pentru a realiza această persistentă a datelor, vom folosi o bază de date deoarece este:

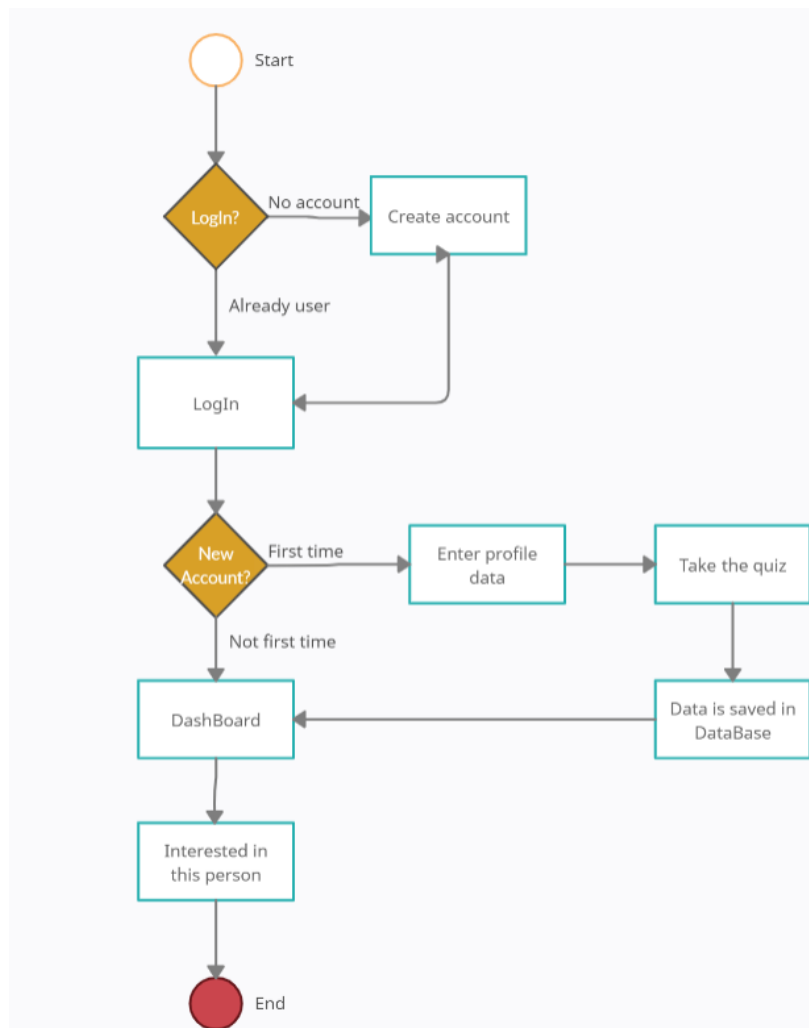
- flexibilă
- scalabilă
- portabilă
- suportă scrieri/citiri concurente

3.5 Controlul accesului utilizatorului la sistem

Actor	Date salvate local	Baza de date	Codul sursă
Utilizator	Da	Nu	Nu
Administrator	Da	Da	Da

3.6 Fluxul global al controlului

Fluxul global al controlului este structurat în mai multe etape de transformare a datelor de intrare, pentru a produce anumite rezultate.



3.7 Condițiile limită

Condițiile limită sunt pornirea/oprirea sistemului, tratarea condițiilor speciale și se pot adăuga subsisteme.

Cazurile limită pot fi cauzate de:

- un defect software
- o greșeală de operare a web/mobile app
- o întrerupere a sistemului electric, unde se află serverul
- atacuri la securitatea sistemului

Anumite excepții pot fi tolerate de sistem și incluse în proiectarea unor componente.

Pentru fiecare tip de cădere prevăzută se scrie un caz de utilizare care extinde unul dintre cazurile de utilizare descrise în faza de extragere a cerințelor.

- Pentru a rezolva defectul software, vom rula diverse Unit Testing.
- Pentru a evita greșeală de operare, ne vom asigura că UX ul este foarte intuitiv.
- Pentru situația în care serverul pică, ne vom asigura că avem un backup pe un alt server.
- Securitatea sistemului va fi realizată prin implementarea certificatului SSL.

Planificarea implementării proiectului

ID	Nume	Asignat	Sprint	Estimare	Mențiuni
1.	Instalare tool-uri necesare pt Android	Nicolae, Olivia	1	1h	Instalare Android Studio, Android SDK
2.	Instalare tool-uri necesare pt Web	Diana, Emilia	1	1h	Instalare tool-uri pt rulare server local, hosting etc
3.	Inițializare Google Firebase	Nicolae, Olivia, Diana, Emilia	1	2h	Inițializarea Google Firebase ca database comun pt website și mobile app
4.	Conectare Google Firebase [Mobile App]	Nicolae, Olivia	1	4h	Realizarea conexiunii dintre aplicația mobilă și database
5.	Conectare Google Firebase [Web App]	Diana, Emilia	1	4h	Realizarea conexiunii dintre website și database
6.	Login/Register/Auth [Mobile App]	Nicolae, Olivia	1	6h	Vom adăuga opțiunea de register cu Email si password și vom salva în database datele
7.	Login/Register/Auth [Web App]	Diana, Emilia	1	6h	Vom adăuga opțiunea de register cu Email si password și vom salva în database datele
8.	Funcționalitatea de a adăuga interese [Mobile App]	Nicolae, Olivia	2	4h	Userul va putea selecta interese multiple Interesele vor fi salvate în google Firebase;
9.	Funcționalitatea de a adăuga interese [Web App]	Diana, Emilia	2	6h	Userul va putea selecta interese multiple Interesele vor fi salvate în google Firebase;
10.	Opțiunea de a adăuga tipul de personalitate [Mobile App]	Nicolae, Olivia	2	2h	Userul va putea sa introduca tipul sau de personalitate. Dacă nu își cunoaște tipul personalității, va fi redirectat catre: https://www.16personalities.com/ro

11.	Opțiunea de a adăuga tipul de personalitate [Web App]	Diana, Emilia	2	2h	Userul va putea sa introduca tipul sau de personalitate. Dacă nu își cunoaște tipul personalității, va fi redirectat catre: https://www.16personalities.com/ro
12.	Opțiunea de a adăuga o fotografie de profil, descriere a profilului, locație [Mobile App]	Nicolae, Olivia	2	6h	Toate datele vor fi salvate în database
13.	Opțiunea de a adăuga o fotografie de profil, descriere a profilului, locația [Web App]	Diana, Emilia	2	6h	Toate datele vor fi salvate în database
14.	Administrare cont (Reset password, modificare poza de profil, modificare descriere) [Mobile App]	Nicolae, Olivia	2	6h	Modificarea datelor din database
15.	Administrare cont (Reset password, modificare poza de profil, modificare descriere) [Web App]	Diana, Emilia	2	6h	Modificarea datelor din database
16.	Implementare algoritm de matching [Mobile App]	Nicolae, Olivia	3	4h	Documentare despre un algoritm avansat de matching și implementarea lui
17.	Implementare algoritm de match [Web App]	Diana, Emilia	3	4h	Documentare despre un algoritm avansat de matching și implementarea lui
18.	Opțiunea de a vedea persoane cu interese similare și procentajul de compatibilitate [Mobile App]	Nicolae, Olivia	3	4h	Se va afișa o listă cu persoanele cu interese similare și un procent de compatibilitate
19.	Opțiunea de a vedea persoane cu interese similare și procentajul	Diana, Emilia	3	4h	Se va afișa o listă cu persoanele cu interese similare și un procent de

	de compatibilitate [Web App]				compatibilitate
20.	Adaugare link catre profilul de Instagram, Facebook, Snapchat [Mobile App]	Nicolae, Olivia	3	1h	Poti adauga link către profilul de pe alte rețele sociale
21.	Adaugare link catre profilul de Instagram, Facebook, Snapchat [Web App]	Diana, Emilia	3	1h	Poti adauga link către profilul de pe alte rețele sociale
22.	Un buton de a da "Interested" la o persoana din lista de persoane compatibile [Mobile app]	Nicolae, Olivia	3	1h	Vom salva în database persoanele interesate de o alta persoana.
23.	Un buton de a da "Interested" la o persoana din lista de persoane compatibile [Web app]	Diana, Emilia	3	1h	Vom salva în database persoanele interesate de o alta persoana.
24.	Id-ul de insta devine vizibil după ce ambele persoane au dat Interested [Mobile App]	Nicolae, Olivia	3	1h	Dacă ambele persoane sunt interesate, atunci se va face schimb automat de id-uri de instagram/snap/Facebook
25.	Id-ul de insta devine vizibil după ce ambele persoane au dat Interested [Web App]	Diana, Emilia	3	1h	Dacă ambele persoane sunt interesate, atunci se va face schimb automat de id-uri de instagram/snap/Facebook