

CT30A3204 Advanced Web Applications

Course project

Toni Koskinen toni.koskinen@student.lut.fi

Introduction

In this project, I created a web application where an authenticated user can create a profile and browse profiles of other users. Users can like or dislike profiles and start a chat if they have liked each other's profiles. In summary, this application is a simple Tinder clone.

Technology choices

The backend of the application has been implemented with Node.js using Express.js framework. The frontend has been implemented utilizing React framework, in addition Materialize CSS framework has been used to provide responsive design for the application. All the user data such as profiles and chats are stored in MongoDB. Authentication and authorization are done using session-based authorization with Passport middleware.

Installation guide

Clone the repository of this project to your device with command: **git clone**

https://github.com/ToniKoskinen/AWP_Project. Then navigate to root folder of this project and run **npm install**, to install dependencies. To run this project with your device, run **npm run dev**, this command will start running both client- and server-side of the application. Finally, open <http://localhost:3000/> in your browser.

User manual

An unauthenticated user can either register or login to webpage and these activities can be seen in the navigation bar. After a successful registration, user can navigate to login and fill the login information to see all the activities that require authentication.

After successful authentication, user can see his/her profile with description and registration date in the main page, the user can also edit description of the profile. All the matches of the user are shown below the navigation bar and by clicking a name of a match, user can start a chat with this match. Using the navigation bar when authenticated, user can either navigate to his/her own profile (by clicking his/her name), browse other profiles or logout.

When browsing profiles of other users, the user can either like or dislike the profile. After the user has seen all the profiles of other users, he/she gets option to browse profiles again. If the user gets a match with other user, the application gives an option to start a chat and adds this user to the matches list.

In the chat activity, the user can exchange messages with other user and by clicking the name of the other user, the user can see their profile including description and registration date.

Features

Feature	Points
Basic features and documentation	25
Utilization of frontside framework with React	5
If match is being found the UI gives option to start chat immediately. Implemented using Materialize toasts.	2

Time stamp of each message is stored and shown within chat.	2
The shown options in the navigation bar are based on if a user is signed in or not.	2
User can click username and see user profile page where name, register date and user bio is listed.	2
New possible messages in a chat are fetched automatically, so user doesn't have to do it manually.	2
<p>Registration information must be valid:</p> <ul style="list-style-type: none"> - Email is not taken. - Strong password: Minimum length 8, at least one lowercase letter, at least one uppercase letter, at least one number and at least one symbol. <p>In addition, all required fields must be filled to send registration information</p>	2
Total	42