



UE19CS204 – Web Technologies

Mini Project

Project Title - PAWsibilities

Section - A

Abhishek Aditya BS – PES1UG19CS019

Aniket Aayush – PES1UG19CS061

A Sai Chaithanya – PES1UG19CS002

GITHUB Link -

<https://github.com/Abhishek-Aditya-bs/PAWsibilities-WebTech-Project>

Abstract

The focus of our application is to facilitate the exchange of pets among pet lovers. It is a general-purpose application that can be used for donating, adopting and buying pets. It will help in connecting those who need a pet to those who want to get a responsible owner for their pet.

Our web app contains a landing page, a dashboard and pages to adopt, donate, and buy pets. It also contains the login and signup page to keep track of the user. The login page will allow the user to enter their credentials and they will be redirected to the dashboard. On dashboard, they can see their pets put for donation and can remove the same if the pet has already been adopted. The page also contains a feedback form where user can give their feedback regarding the website. The signup page contains a form to collect user details and create an account for him.

The page to donate a pet contains a form where user can put the details of the pet he wants to donate and the type of owner he wants for his pet. Upon submission of the form, the pet will be added to the adopt section where people can request to adopt the pet.

The page to adopt a pet contains the details of all the pets put for donation and user can contact the owner of the pet if he is interested in adopting him. Once the pet is adopted, the owner can remove it from the list.

The buy page contains the details of nearby pet shops present in the city along with their contact number, closing and opening time, etc.

All the pages in our web app are fully responsive and can be rendered easily on any screen size.

All the components used in the app are fully scalable.

Authorisation is present to validate the credentials on the login page and the details entered by the user gets stored in the database after hashing the password.

Technologies Used

1. Node.js/Express.js FrameWork for backend server side operations
2. MongoDB for DataBase Storage
3. React js , HTML 5 , CSS 3 , JSX, Bootstrap, JavaScript for frontend rendering

MERN Stack: MERN Stack is a Javascript Stack that is used for easier and faster deployment of full-stack web applications. MERN Stack comprises of 4 technologies namely: **MongoDB**, **Express**, **React** and **Node.js**. It is designed to make the development process smoother and easier. Each of these 4 powerful technologies provides an end-to-end framework for the developers to work in and each of these technologies play a big part in the development of web applications.

MODULES USED :

For Backend Server Side :

1. `bcryptjs` : To Hash the Password to be stored in the DataBase
2. `concurrently` : To run/Start both frontend and backend in the same Terminal.
3. `cors` : Allows restricted resources on a web page to be requested from another domain outside the domain from which the first resource was served.
4. `dotenv` : that loads environment variables from a `.env` file into `process.env`
5. `express` : Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.
6. `jsonwebtoken` : To implement JSON web tokens , Returns the JsonWebToken as string.
7. `mongoose` : Mongoose is an Object Data Modeling (ODM) library for MongoDB and Node.js.
8. `nodemon` : Automatically restarts the server if any changes are made .

For Frontend rendering :

1. `react-bootstrap` : Provides already built commonly used react Components
2. `bootstrap` : open-source CSS framework directed at responsive, mobile-first front-end web development.
3. `react-swipeable-views` : react component for auto swipeable views
4. `@material-ui/core` : A popular React UI framework
5. `@material-ui/icons` : Provides popular icons which can be used just by importing it
6. `react-router-dom` : client-side routing, allows us to build a single-page web application with navigation without the page refreshing as the user navigates
7. `axios` : Promise based HTTP client for the browser and node.js
8. `popper.js` : Popper.js is a positioning engine, its purpose is to calculate the position of an element to make it possible to position it near a given reference element.

Member Contributions

- Abhishek Aditya BS : Sign up page,login page and authentication system,Integration of Frontend and Backend and Database operations
- Aniket Aayush : Homepage,Dashboard,Buy page, Donate page,Adopt page
- A Sai Chaitanya : Backend routes (Handling HTTP requests from the frontend)