Web Applications course project

Aleksi Järventausta

Link to Rahti:

http://little-blog-app-josnyt.rahtiapp.fi/

Link to Github:

https://github.com/AleksiJarventausta/webApplications

Read README to start the project.

Link to youtube:

https://youtu.be/fykPraS5mWw

A short description of the software, choice of libraries, and design choices.

This project is a little twitter clone where you can post blogs, comment others and search for other bloggers.

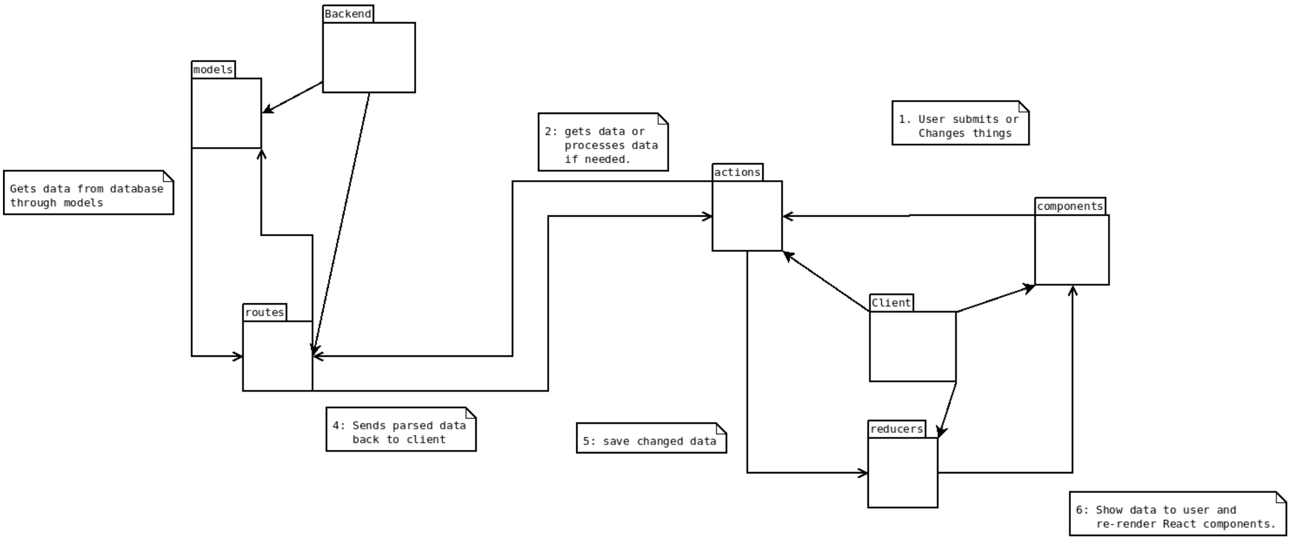
I chose to use Material-UI because of it's ease of use and it's documentation. I also chose to use React and redux because with those the software can be expanded without so big fear of overlapping and reusability.

Backend

for "security" I used jsonwebtoken, bcryptjs and passport libraries to learn something about authorization. Most of the things I learned from this project:

https://github.com/rishipr/teams

A figure of the program architecture, including: program components, where they are located, and how they communicate.



Gets data from database through models

2: gets data or processes data if needed.

1. User submits or Changes things

4: Sends parsed data back to client

5: save changed data

6: Show data to user and re-render React components.

Illustration 1: Project is divided to backend and client subprojects. Back end is express server and client is React app.

Points

Feature	Points
• Using React.js front-end • Using redux and an advanced React architecture	• 15 pts • +5 pts
• Running your application in a Docker container (or having the application Docker compatible) • Running your application in Rahti • Having more than one container (such as a separate database server or a load balancer)	• 5 pts • +5 pts • +3 pts
• Using a database, such as Mongo, Redis, or any SQL-compatible • Use an ORM and models in backend, such as the Mongoose (MongoDB) or Sequelize (SQL)	5 pts +3 pts
• Responsive front-end, such as through Materialize, Bootstrap, Semantic, or custom based on Grid • Advanced features, such as animations, effects, carousel, etc.	• 5 pts • +2 pts
User registration, authentication, and password storage	5 pts
Minimum implementation	30 pts
documentation	-10 pts
Users can comment blogs	2 pts
Search bar	2 pts
total	77