

Project Document

CT30A3203 Web Applications

Technology choises	3
Installation guide	3
User manual	3
Additional features	4

Technology choises

Backend uses Node.js with Express and MongoDB. JWT is used for authentication. All server functions are in /routes/api.js file so router is also used. Three Mongoose models are used in server (User, Snippet and Comment). Database runs in port 27017 and is named as "waprojectdb".

Frontend is created with React. Scaling application in mobile and in desktop browser is done with materialize. Code in snippets is printed in Highlight component and that is imported from react-highlight.js in addition with 'Rainbow' style from highlightjs.org. To display time in correct format, moment is used.

I chose these solutions since I don't know well enough any other options and React because it is fairly simple and somewhat works like object oriented programming.

Client part uses port 3000 and server part port 4000.

Installation guide

Download zip-file of application from GitHub and unzip it in some folder. After that run "npm i" in project-folder and in both client- and server-folder (both are found in project folder). After node modules are installed run "npm run dev" -command in project folder to start both application.

User manual

When you start the application you'll be at the home page. All snippets in database will be listed here. At the top of the page there is links to home, register and login page. After you register you'll be asked to preceed to login page (NOTE: only first registered user becomes an admin so first user in the database is an admin user). After you login you'll be redirected back to the home page. Also login link at the top is changed to logout button. Pressing this button logouts the user and changes logout button back to login link.

At the home page logged in user can post new code snippets. After submitting new post page is reloaded and title of this post is added to the page. This title is also link to snippet's own page. If there are more than 10 posts they are divided in to different pages and navigation bar with page numbers appears. At the snippet page there is snippet's title and below are description and code snippet itself. On the left side there is number of votes, number of total votes and vote up and down buttons (latter ones only if user is logged in). On the right side there are edit button (only if user posted this snippet or if user is an admin) and delete button (only if user is an admin). Below code snippet is timestamp of last time of edit and comment section. If post is edited area for that is creted below

code snippet. Submitting changes reloads the page. Deleting post will also delete all comments related to it.

All comments related to the snippet are displayed here and below them is option to comment the post (if user is logged in). On sides of every comment there are same options is with code snippet (votes on the left and edit and delete on the right). Comments also have timestamp in same manner as with snippets. If user edits the comment textarea and submit button are created below of this comment.

Application scales in desktop browsers and mobile. Also many buttons (mostly submit, edit and delete) will reload the page when pressed.

Additional features

Feature	Points
Users can edit their own posts and comments	4
React as frontside framework	5
Admin account that can edit all posts and comments and delete them	3
Vote up or down posts and comments	3
Use of highlight library highlight.js	2
Pager function when there are more than 10 posts available	2
Posts and comments have last edited timestamp	2

Total point from additional features is 21 and adding it to 25 from basic features stacks up to 41 points.