

CT30A3204 Advanced Web Applications -

Luento-opetus 31.10.2022-5.3.2023

web project

Mandatory requirements

The backend uses Node.js, uses Express, and the language is JavaScript,

The database uses MongoDB,

Users have the option to register and log in, using tokens for authentication, only authenticated users can post,comment

Authenticated users can:

post new code snippet

Comment on an existing post

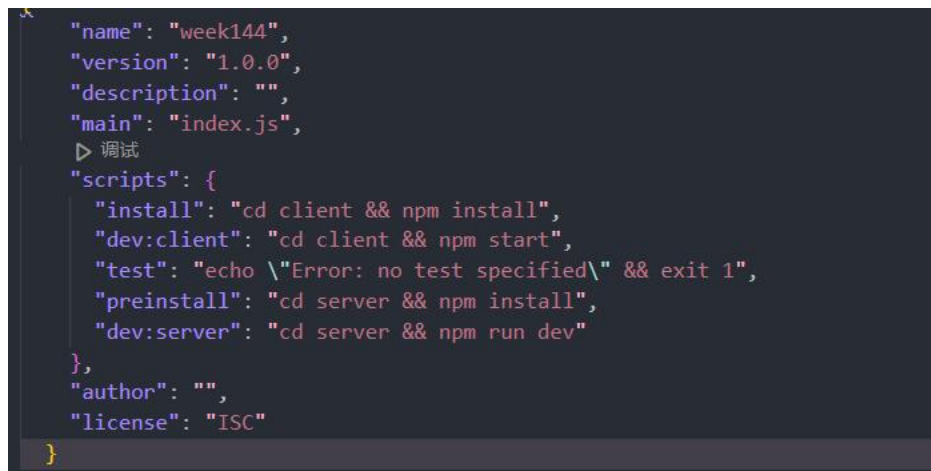
Unauthenticated users can see posts, comments (and vote counts)

There are some pages that list all the posts, each post is displayed in the form of a card, click the button to display comments

Responsive design is used, the react framework is used, and the Materialize library is used for rendering.

Installation Guide:

Project's package.json page has various instructions, you should run install, preinstall, and then run dev:server,dev:client



```
{
  "name": "week144",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "install": "cd client && npm install",
    "dev:client": "cd client && npm start",
    "test": "echo \"Error: no test specified\" && exit 1",
    "preinstall": "cd server && npm install",
    "dev:server": "cd server && npm run dev"
  },
  "author": "",
  "license": "ISC"
}
```

Both the front end and the server end need to run the program to proceed normally

User Manual:

First of all, the running program should open the local address localhost:3000, and the initial interface is login

MY BLOG

Login

Email *

Password *

LOGIN REGISTER

For the first use, you should register first, click the register button to enter the registration interface. After entering the registration interface, you can fill in your email and password, and you can also upload user pictures

MY BLOG

Register

Email *

Password *

Profile Picture: 选择文件 | 未选择文件

REGISTER

After registration, enter the login page, click the login button to log in successfully, and you will jump to the blog interface, or you can click the MY BLOG button
If you want to log out, please click the LOGOUT button

MY BLOG LOGOUT

Login

Email *

YufengPannn@qq.comnnnnnnnnnn

Password *

LOGIN REGISTER

Click MY BLOG, then enter the blog interface

[MY BLOG](#) [LOGOUT](#) [USER](#)

My Blog

POST

1

123
by 123

123

Votes: 1

UPVOTE DOWNVOTE

Last edited: 2023/3/4 14:07:30

SHOW COMMENTS EDIT POST DELETE

After entering the blog interface, you can write a blog, press the post button to upload, and at the same time, you can also realize the above functions

If you want to display comments, click the SHOW COMMENTS button, edit by pressing EDIT POST, and delete by pressing


DELETE, only the creator of the post can edit and delete, only logged in can post and comment, delete.

Click on the User interface to enter the information page of the registrant

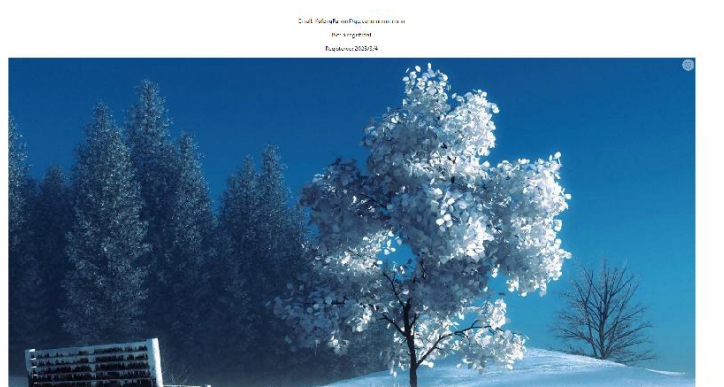
Email: YufengPannn@qq.comnnnnnnnnnn

Bio: a registrant

Registered: 2023/3/4

 User profile

If you upload a picture, you can see the picture on this page



Next is **the implementation of detailed functions**

First list the functions I implemented

Basic features (as stated in the previous chapter) with well written documentation	25
Users can edit their own comments/posts	4
Utilization of a frontside framework, such as React, but you can also use Angular, Vue or some other	5
Use some highlight library for the code snippets, for example https://highlightjs.org/	2
Admin account with rights to edit all the post and comments and delete content (if a post is removed, all its comments should be removed too)	3
Test software for accessibility;	3
Vote (up or down) posts and comments (only one vote per user)	3
User profiles can have images which are show next to posts/comments	3
User can click username and see user profile page where name, register date, (user picture) and user bio is listed	2
Last edited timestamp is stored and shown with posts/comments	2

The target score is 52 points. Some parts may be done according to my inappropriate understanding. I hope the teacher will understand

What follows is an introduction to each function:

Users can edit their own comments/posts

You can change the post and comment after clicking EDIT POST

Click save to save the changed content.

The screenshot displays a web application interface. At the top, there is a red circle with the number '1'. Below it, a post titled '123123 by 123' is shown. The post has a 'Title *' field containing '123123' and a 'Content *' field containing '21312'. Below the content field are two red buttons: 'SAVE' and 'CANCEL'. Below the post, there is a 'Comment *' field and an 'Author *' field. A red 'COMMENT' button is positioned below these fields. Below the comment field, there is a list of comments. The first comment is by 'qwer' with a score of 1. It has 'UPVOTE' and 'DOWNVOTE' buttons. Below the comment, it says 'Last edited: 2023/3/4 14:15:02'. The second comment is by 'd'.

Utilization of a frontside framework, such as React, but you can also use Angular, Vue or some other

Using the react framework

Use some highlight library for the code snippets, for example <https://highlightjs.org/>

Using the highlight library



Admin account with rights to edit all the post and comments and delete content (if a post is removed, all its comments should be removed too)

The administrator account is Administrator@gmail.com

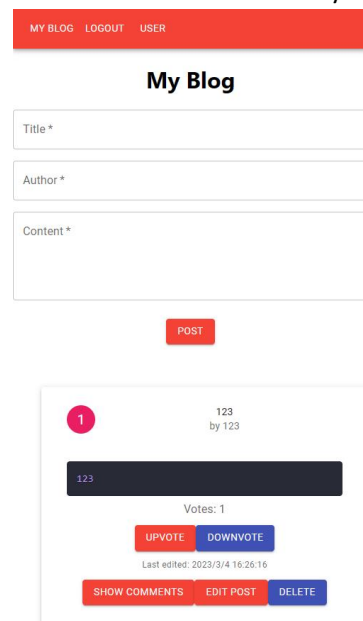
password is !Abc123321123

can edit and delete anything.

Test software for accessibility;

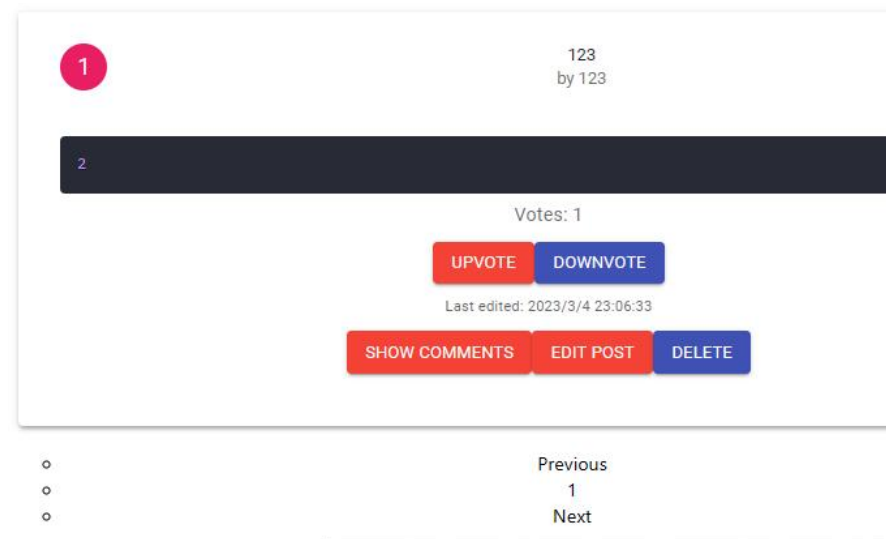
Tested on the andriod virtual machine, you can input something, and it can change with the change of the screen size,

Screen readers work with my application

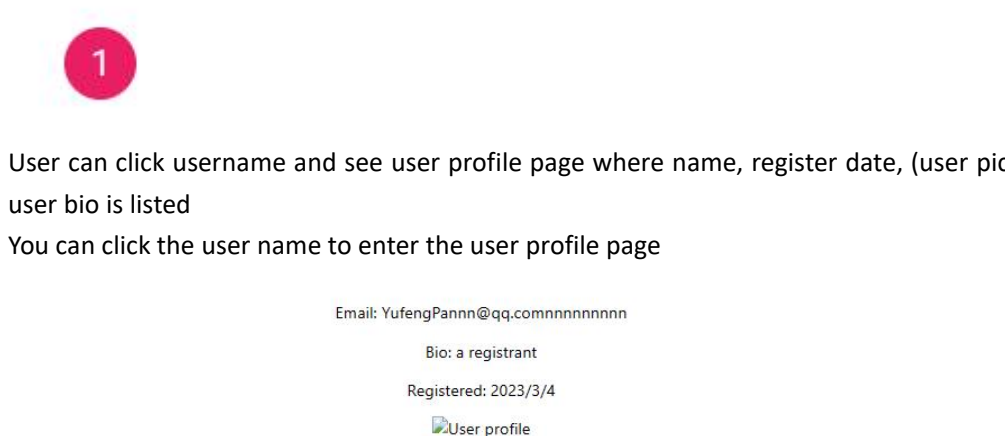


Vote (up or down) posts and comments (only one vote per user)

Can vote, each person can only cast one vote, vote for or against, comments can also vote.



User profiles can have images which are show next to posts/comments
There will be a picture displayed next to the post



User can click username and see user profile page where name, register date, (user picture) and user bio is listed
You can click the user name to enter the user profile page

Last edited timestamp is stored and shown with posts/comments
The time of the last edit exists, in posts and comments.



The above is all my introduction

Re-list the features:

First list the functions I implemented

Basic features (as stated in the previous chapter) with well written documentation	25
Users can edit their own comments/posts	4
Utilization of a frontside framework, such as React, but you can also use Angular, Vue or some other	5
Use some highlight library for the code snippets, for example https://highlightjs.org/	2
Admin account with rights to edit all the post and comments and delete content (if a post is removed, all its comments should be removed too)	3
Test software for accessibility;	3
Vote (up or down) posts and comments (only one vote per user)	3
User profiles can have images which are show next to posts/comments	3
User can click username and see user profile page where name, register date, (user picture) and user bio is listed	2
Last edited timestamp is stored and shown with posts/comments	2
The target score is 52 points.	