Portfolio with log-in to edit

Emilia Fredriksson – frem22pu@student.ju.se

Web Development Fundamentals - Jönköping University – 2023

Contents

1) Introduction

In the past six years, my biggest dream has been to become a graphic designer. I have been studying in that direction for four years now and the number of projects I have created is huge. There are a lot of projects but no good way to store them, so everything is just a mess on my computer. Therefore, I am creating a portfolio website where I can store my projects that I, in the future, will share with employers. The website will be a way for me to show what I can to future employers or future clients who want my knowledge of graphic design and programming for different tasks.

This project offers me two big opportunities. The first one is to do what I love the most and design how I want my portfolio that represent me in the future will look like. The second one is the chance to practice my programming skills, while also showing for future employers and clients that I can develop websites.

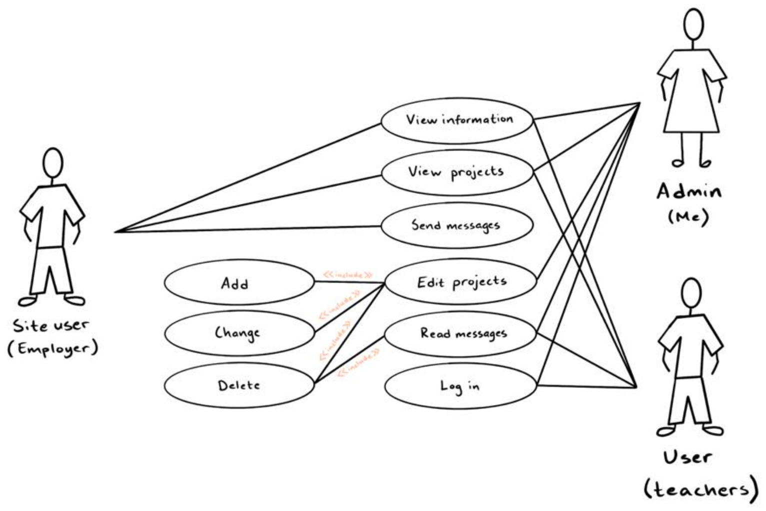


Figure 1.1.1 Case diagram

2) Method

2.1) Architecture

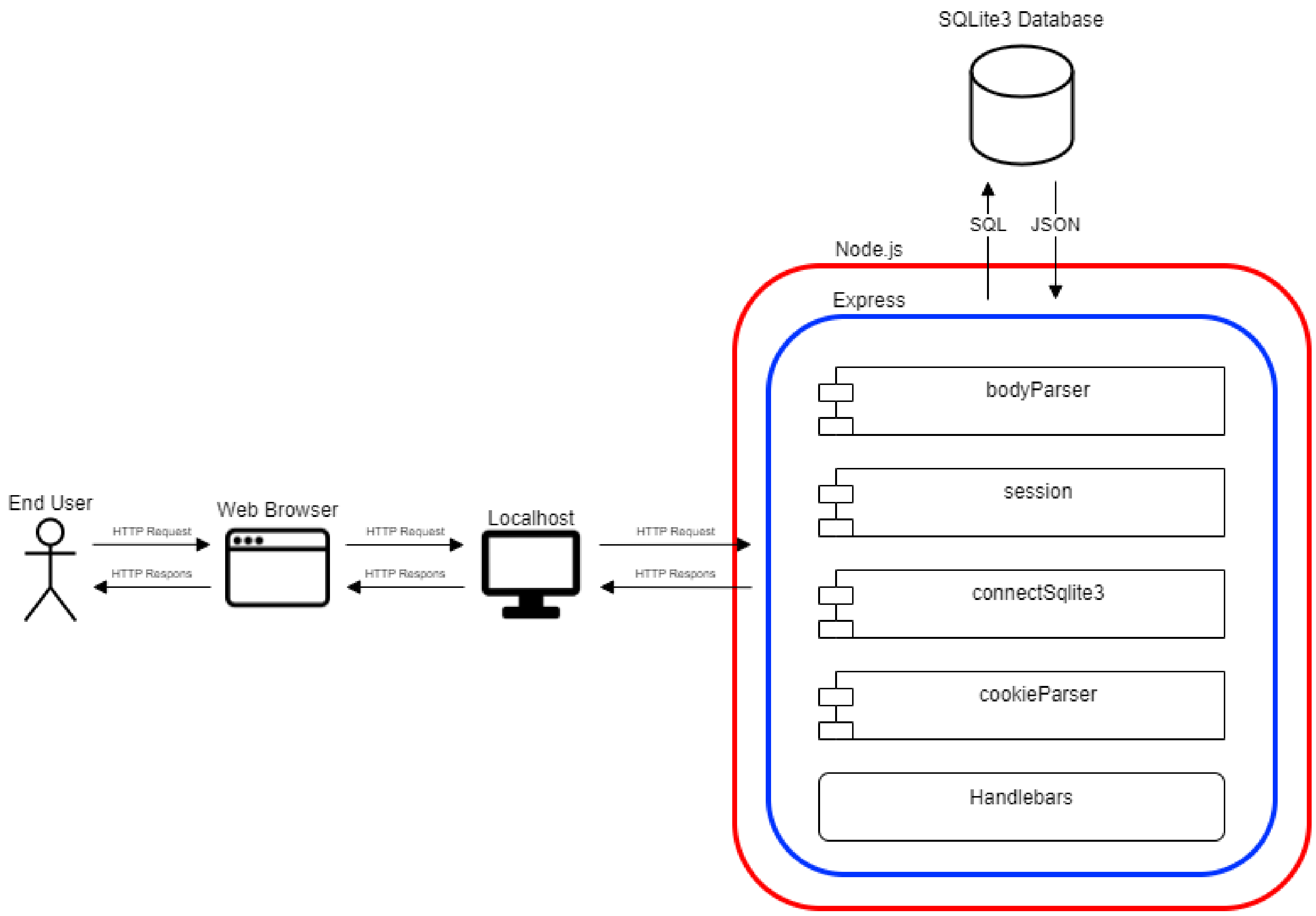


Figure 2.1.1. Components of my portfolio

2.2) Database

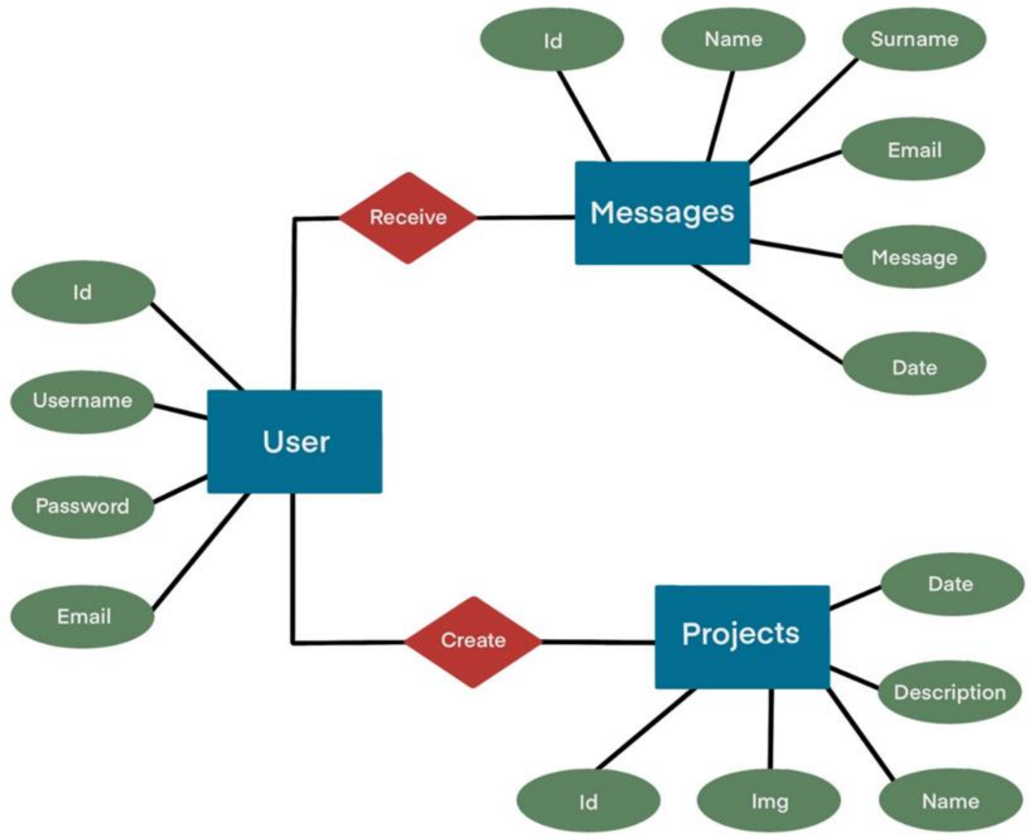
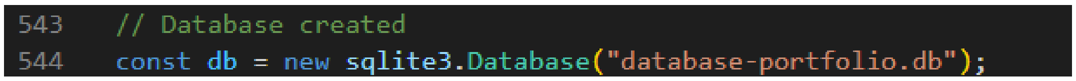


Figure 2.2.1. ER diagram

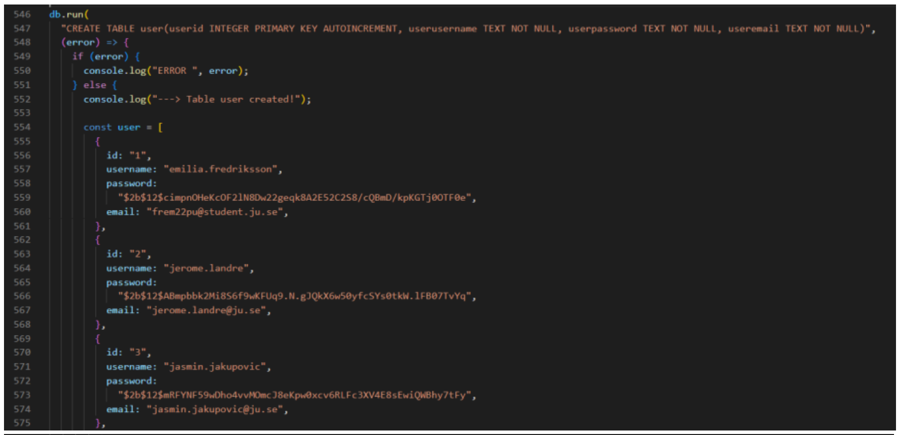
In this project, I have created a SQLite3 inside of my portfolio.js file. This file is stored inside a folder on my computer and in a GitHub directory. The database contains three tables. Those three tables are user, messages, and projects. They are used on different pages on the website but all of them are connected as you can see in figure 2.2.1.



```
// Database created
const db = new sqlite3.Database("database-portfolio.db");
```

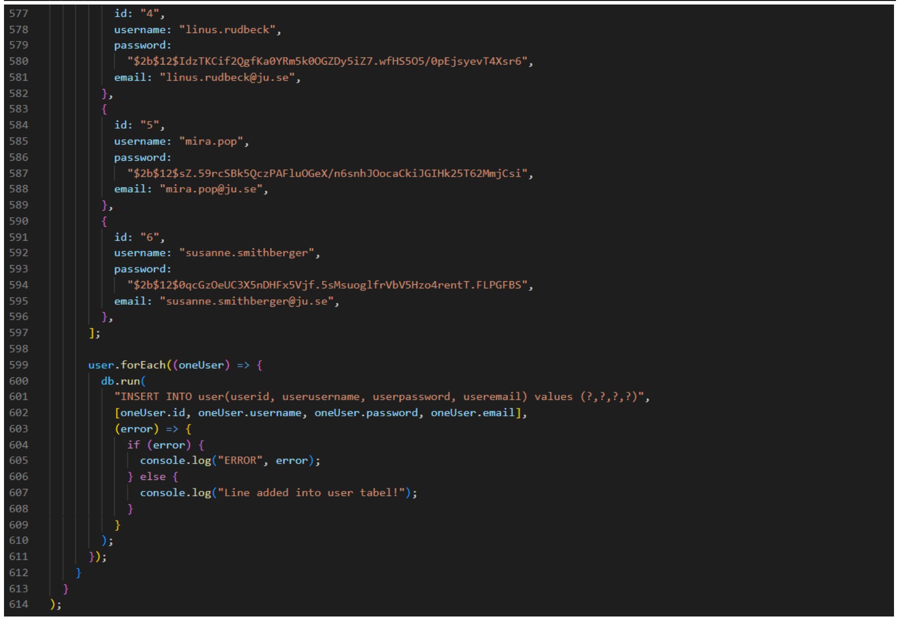
Figure 2.2.2. Database created inside portfolio.js

The first table is the user table. It is used for the login function and consists of the information about the people who can log in. It has four columns, and these are id, username, password, and email. The username and the password are the attributes that get checked when you log in to see if you exist in the database and if you are allowed to log in. The ID is used to identify if you are the admin or not, which then decides what you can do on the website. To make the login safe I have used hashed passwords in the database. My project is free from clear text passwords. The email doesn't get used as it is right now but an implement for the future could be that you could use either the username or the email to log in.



```
db.run(
  "CREATE TABLE user(userid INTEGER PRIMARY KEY AUTOINCREMENT, userusername TEXT NOT NULL, userpassword TEXT NOT NULL, useremail TEXT NOT NULL)",
  (error) => {
    if (error) {
      console.log("ERROR ", error);
    } else {
      console.log("---> Table user created!");

      const user = [
        {
          id: "1",
          username: "emilia.fredriksson",
          password:
            "$2b$12$cimpnOHeKcOF2lN8Dw22geqk8A2E52C2S8/cQBmD/kpKGTj0OTF0e",
          email: "frem22pu@student.ju.se",
        },
        {
          id: "2",
          username: "jerome.landre",
          password:
            "$2b$12$ABmpbbk2Mi8S6f9wKFUq9.N.gJQkX6w50yfcSYs0tkW.lFB07TvYq",
          email: "jerome.landre@ju.se",
        },
        {
          id: "3",
          username: "jasmin.jakupovic",
          password:
            "$2b$12$mRFYNF59wDho4vvMOmcJ8eKpw0xcv6RLFc3XV4E8sEwiQWBhy7tFy",
          email: "jasmin.jakupovic@ju.se",
        },
```

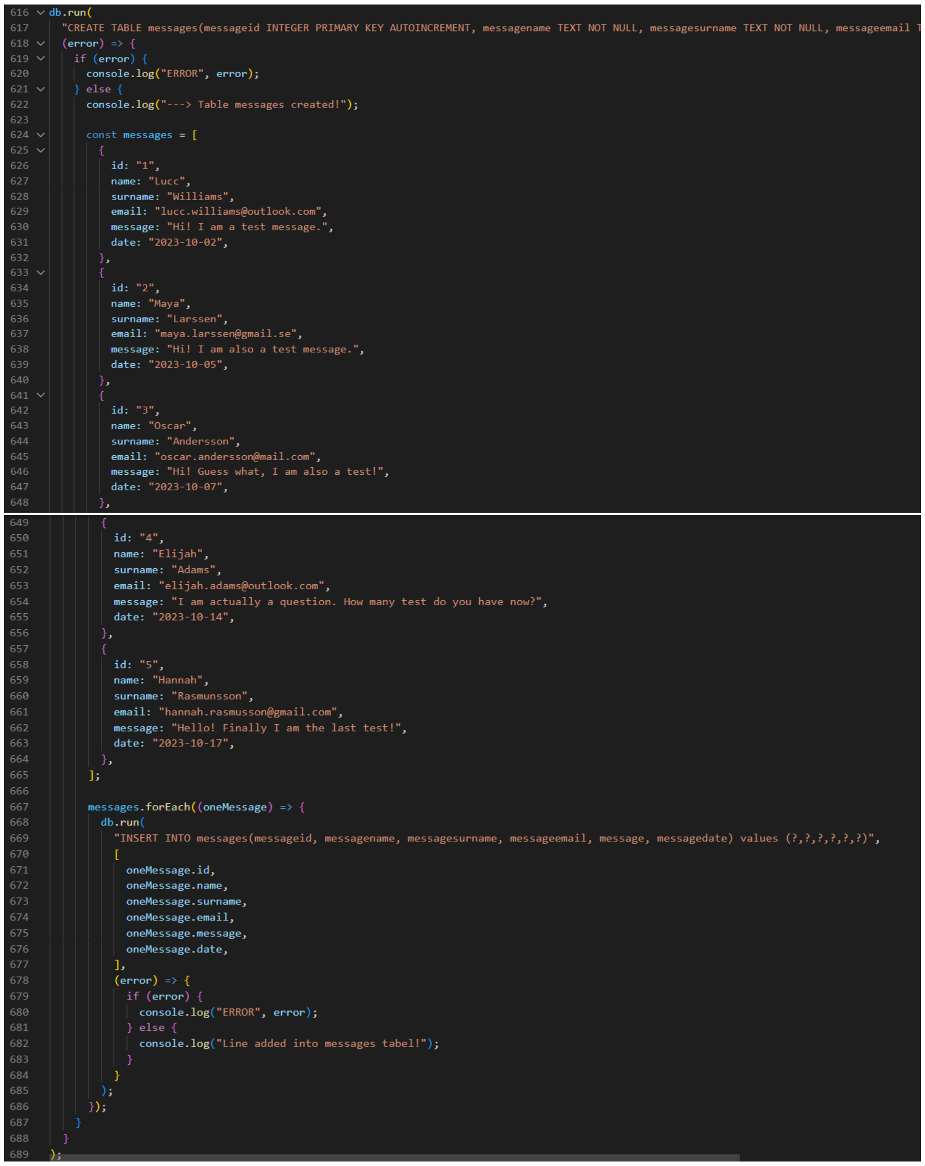


```
          id: "4",
          username: "linus.rudbeck",
          password:
            "$2b$12$IdzTKCif2QgfKa0YRm5k0OGZDy5iZ7.wfHS5O5/0pEjsyevT4Xsr6",
          email: "linus.rudbeck@ju.se",
        },
        {
          id: "5",
          username: "mira.pop",
          password:
            "$2b$12$sZ.59rcSBk5QczPAFluOGeX/n6snhJOocaCkiJGIHk25T62MmjCsi",
          email: "mira.pop@ju.se",
        },
        {
          id: "6",
          username: "susanne.smithberger",
          password:
            "$2b$12$0qcGzOeUC3X5nDHFx5Vjf.5sMsuoglfrVbV5Hzo4rentT.FLPGFBS",
          email: "susanne.smithberger@ju.se",
        },
      ];

      user.forEach((oneUser) => {
        db.run(
          "INSERT INTO user(userid, userusername, userpassword, useremail) values (?,?,?,?)",
          [oneUser.id, oneUser.username, oneUser.password, oneUser.email],
          (error) => {
            if (error) {
              console.log("ERROR", error);
            } else {
              console.log("Line added into user tabel!");
            }
          }
        );
      });
    }
  }
);
```

Figure 2.2.3. Table User in code

The second table is the message table. It consists of six columns which are id, name, surname, email, message, and date. This table is used on the contact page, or the message page as it is called if you are logged in. On the contact page, the fields you fill in to send a message are connected to the database in a way that when the message is sent it will get stored in the database. On the message page, the messages table is displayed as messages. It shows all the information stored in the messages table so that only people who are logged in can see the messages.



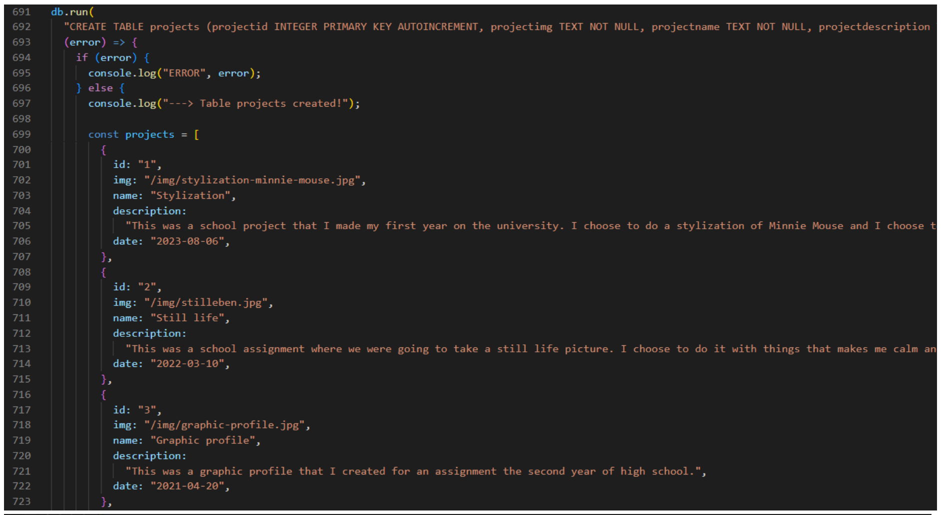
```
db.run(
  "CREATE TABLE messages(messageid INTEGER PRIMARY KEY AUTOINCREMENT, messagename TEXT NOT NULL, messagesurname TEXT NOT NULL, messageemail T
  (error) => {
    if (error) {
      console.log("ERROR", error);
    } else {
      console.log("---> Table messages created!");

      const messages = [
        {
          id: "1",
          name: "Lucc",
          surname: "Williams",
          email: "lucc.williams@outlook.com",
          message: "Hi! I am a test message.",
          date: "2023-10-02",
        },
        {
          id: "2",
          name: "Maya",
          surname: "Larssen",
          email: "maya.larssen@gmail.se",
          message: "Hi! I am also a test message.",
          date: "2023-10-05",
        },
        {
          id: "3",
          name: "Oscar",
          surname: "Andersson",
          email: "oscar.andersson@mail.com",
          message: "Hi! Guess what, I am also a test!",
          date: "2023-10-07",
        },
        {
          id: "4",
          name: "Elijah",
          surname: "Adams",
          email: "elijah.adams@outlook.com",
          message: "I am actually a question. How many test do you have now?",
          date: "2023-10-14",
        },
        {
          id: "5",
          name: "Hannah",
          surname: "Rasmunsson",
          email: "hannah.rasmusson@gmail.com",
          message: "Hello! Finally I am the last test!",
          date: "2023-10-17",
        },
      ];

      messages.forEach((oneMessage) => {
        db.run(
          "INSERT INTO messages(messageid, messagename, messagesurname, messageemail, message, messagedate) values (?,?,?,?,?,?)",
          [
            oneMessage.id,
            oneMessage.name,
            oneMessage.surname,
            oneMessage.email,
            oneMessage.message,
            oneMessage.date,
          ],
          (error) => {
            if (error) {
              console.log("ERROR", error);
            } else {
              console.log("Line added into messages tabel!");
            }
          }
        );
      });
    }
  }
);
```

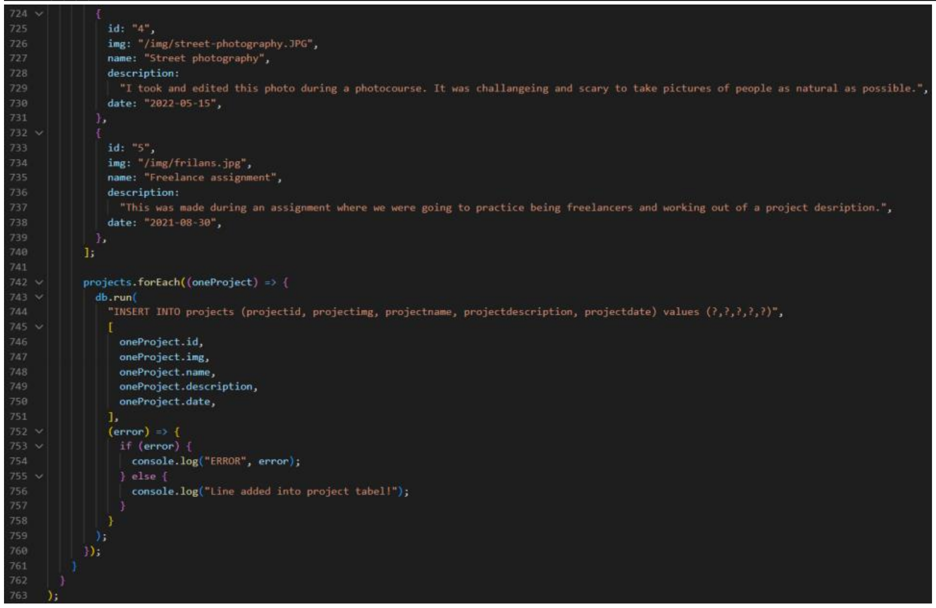
Table 2.2.4. Table Messages in code

The third but most important table is the projects table. This table is used on the project page and the separate project pages. If you are an admin you can add, delete, and modify things inside this table. The projects table has five columns which are id, img, name, description, and date. All this information is used when displaying the projects for everyone looking at the website.



```
db.run(
  "CREATE TABLE projects (projectid INTEGER PRIMARY KEY AUTOINCREMENT, projectimg TEXT NOT NULL, projectname TEXT NOT NULL, projectdescription
  (error) => {
    if (error) {
      console.log("ERROR", error);
    } else {
      console.log("---> Table projects created!");

      const projects = [
        {
          id: "1",
          img: "/img/stylization-minnie-mouse.jpg",
          name: "Stylization",
          description:
            "This was a school project that I made my first year on the university. I choose to do a stylization of Minnie Mouse and I choose t
          date: "2023-08-06",
        },
        {
          id: "2",
          img: "/img/stilleben.jpg",
          name: "Still life",
          description:
            "This was a school assignment where we were going to take a still life picture. I choose to do it with things that makes me calm an
          date: "2022-03-10",
        },
        {
          id: "3",
          img: "/img/graphic-profile.jpg",
          name: "Graphic profile",
          description:
            "This was a graphic profile that I created for an assignment the second year of high school.",
          date: "2021-04-20",
        },
```

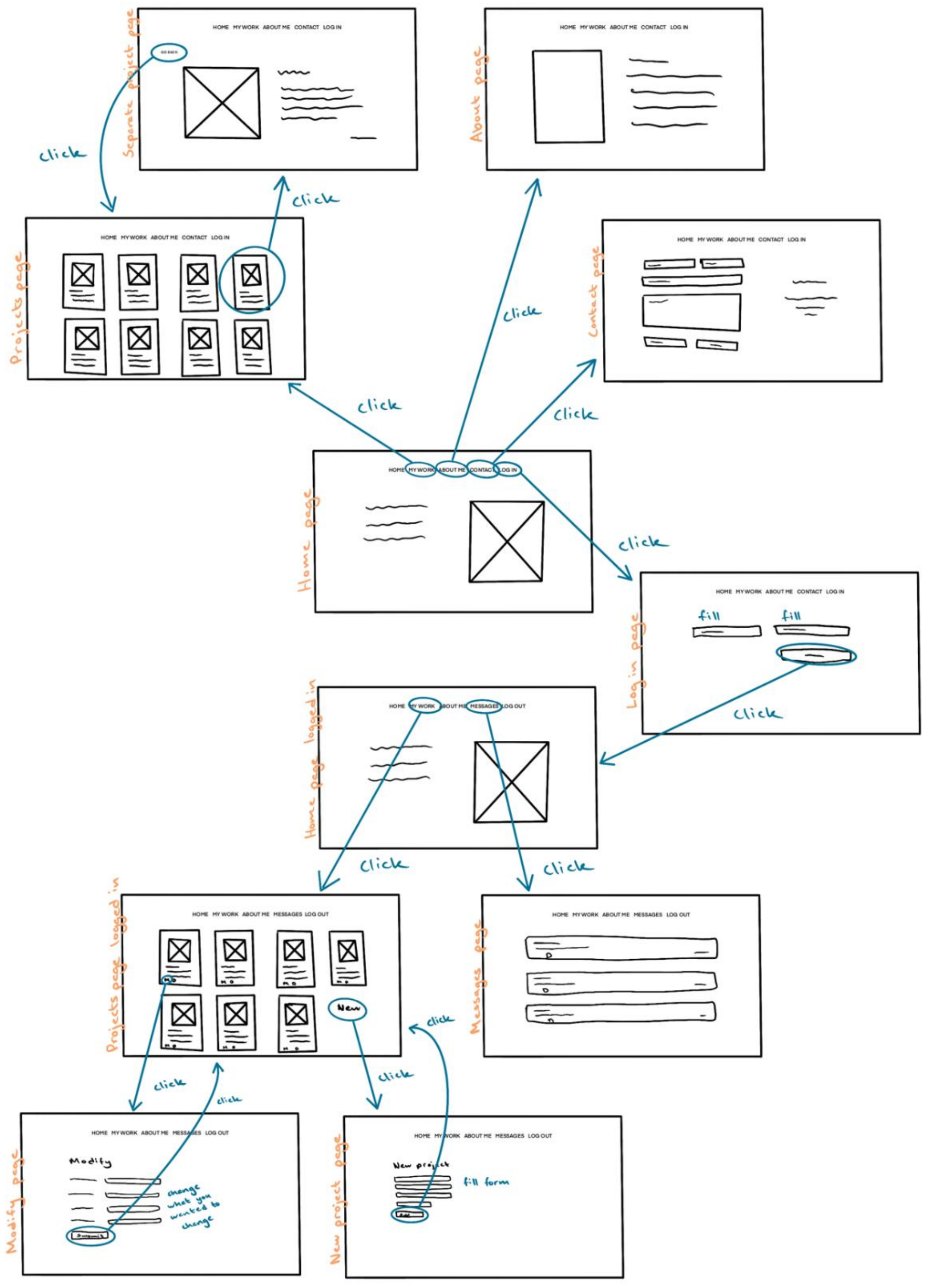


```
        {
          id: "4",
          img: "/img/street-photography.JPG",
          name: "Street photography",
          description:
            "I took and edited this photo during a photocourse. It was challangeing and scary to take pictures of people as natural as possible.",
          date: "2022-05-15",
        },
        {
          id: "5",
          img: "/img/frilans.jpg",
          name: "Freelance assignment",
          description:
            "This was made during an assignment where we were going to practice being freelancers and working out of a project desription.",
          date: "2021-08-30",
        },
      ];

      projects.forEach((oneProject) => {
        db.run(
          "INSERT INTO projects (projectid, projectimg, projectname, projectdescription, projectdate) values (?,?,?,?,?)",
          [
            oneProject.id,
            oneProject.img,
            oneProject.name,
            oneProject.description,
            oneProject.date,
          ],
          (error) => {
            if (error) {
              console.log("ERROR", error);
            } else {
              console.log("Line added into project tabel!");
            }
          }
        );
      });
    }
  }
);
```

Figure 2.2.5. Table Projects in code

2.3) Graphical User Interface



2.4) Web Application

2.4.1) Languages

I have used HTML to structure the elements on the web pages. HTML stands for Hyper Text Markup Language and is the standard markup used for creating web pages (w3school, n.d.). With the use of tags, you tell the browser how the web age should be displayed, and in what order elements should show.

The second language I used where CSS. CSS stands for Cascading Style Sheets and is used to style the HTML elements (Rudbeck, 2023). I have used an external stylesheet, that I called my-styles, which is linked in my main.handlebars.

The third language I used was JavaScript. JavaScript is used to add interactivity and dynamic content to web pages (Rudbeck, 2023). It is used for both front-end and back-end development. In this project, I have run it on the server side with node.js.

2.4.2) Framework

Express is a framework I have used in this project. It is a framework for node.js that makes it possible to simplify the code used in node.js (Landré, 2023). I used Express because it is flexible and makes it easier to create web applications.

2.4.3) Desing pattern

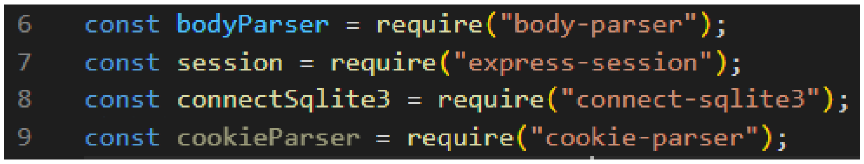
I used the design pattern MVC, Model-View-Controller. The model is the data that is usually stored in a database (Landré, 2023), which mine is. The view is the part that generates the HTML code and here I have used handlebars which is a page template generator. You pass the context, the data, to the template you made, and handlebars will generate the HTML code. Then last but not least the controller is what receives incoming HTTP requests, fetches the model, asks for the generated code for the data, and sends back the HTTP response with the HTML.

2.4.4) Security

I have used a basic authentication for my website. I am using a simple username and password to log in to my website. To make it more secure I implemented hashed passwords in my database instead of clear text passwords. I have used bcrypt to do this. Hashed passwords give a defense against getting your password exposed if the database where they are stored gets compromised.

2.4.5) Middleware's

I have used four middleware's in my code. The first one is the bodyParser which converts data from an HTTP request to a format that the web application can understand. The second one is the session which allows you to store and manage session data, for instance, user state. The third one is connectSqlite3 which allows you to store session data into a database. The fourth one is the cookieParser which is used to parse cookies attached to incoming HTTP requests.



```
const bodyParser = require("body-parser");
const session = require("express-session");
const connectSqlite3 = require("connect-sqlite3");
const cookieParser = require("cookie-parser");
```

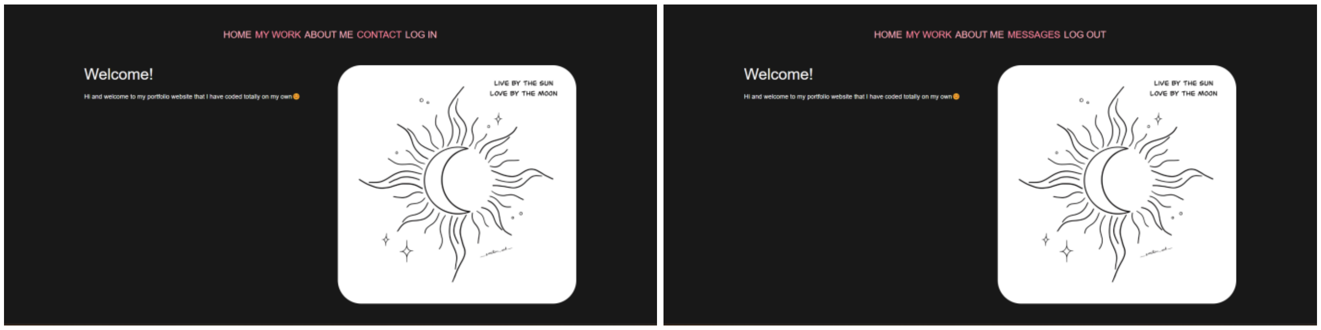
3) Results

After a lot of work, I have a done website. The functionalities I wanted to have is working, could maybe be a bit more stylish than they are but that is a thing I will keep working on. The website has a login for an admin (username: emilia.fredriksson password: WebDev) that can change things and also some non-admin users (username: jerome.landre password: WebDevJerome) that can log in but not change anything.

The website is running on port 2003.

Overall, I am happy with the result. I have used methods that I never heard about before and I managed to do this, not-very beautiful but, working website. Implementation of the functionality for the future could be to add CRUD operations on all the information so that the admin could change information on every page.

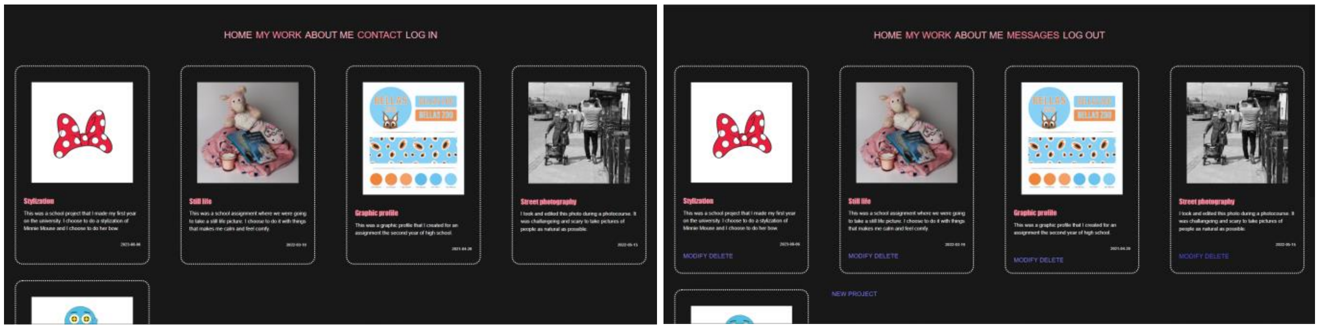
Another thing could be to add some sort of filtering for the different projects. Last but not least add so that you can choose to log in with either your username or your email.


HOME MY WORK ABOUT ME CONTACT LOG IN
LIVE BY THE SUN
LOVE BY THE MOON
HOME MY WORK ABOUT ME MESSAGES LOG OUT
LIVE BY THE SUN
LOVE BY THE MOON

Welcome!

Hi and welcome to my portfolio website that I have coded totally on my own😊

Figure 3.1 Result of the home page first as a site user, second as admin


HOME MY WORK ABOUT ME CONTACT LOG IN
HOME MY WORK ABOUT ME MESSAGES LOG OUT
MODIFY DELETE
NEW PROJECT

Stylization

This was a school project that I made my first year on the university. I choose to do a stylization of Minnie Mouse and I choose to do her bow.

2023-06-06

Still life

This was a school assignment where we were going to take a still life picture. I choose to do it with things that makes me calm and feel comfy.

2022-03-19

Graphic profile

This was a graphic profile that I created for an assignment the second year of high school.

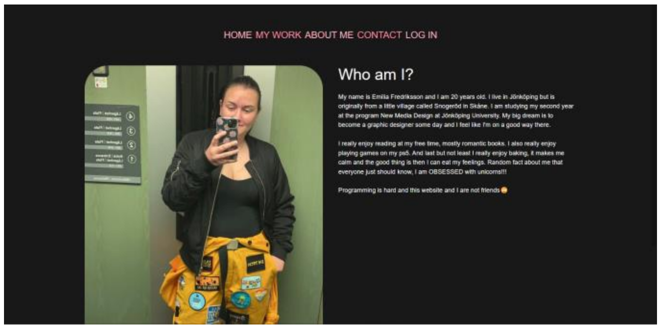
2021-04-26

Street photography

I took and edited this photo during a photocourse. It was challenging and scary to take pictures of people as natural as possible.

2022-05-15

Figure 3.2 Result of projects page first as site user, second as admin

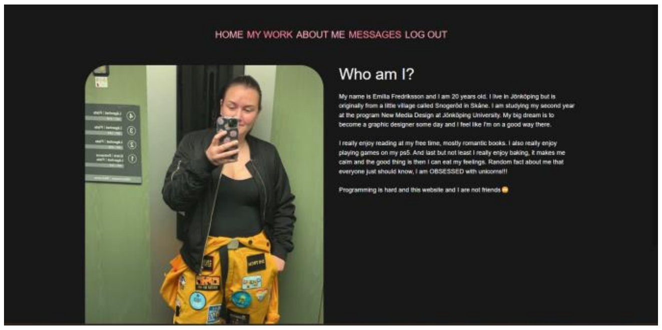


Who am I?

My name is Emilia Fredriksson and I am 20 years old. I live in Jönköping but is originally from a little village called Snogeröd in Skåne. I am studying my second year at the program New Media Design at Jönköping University. My big dream is to become a graphic designer some day and I feel like I'm on a good way there.

I really enjoy reading at my free time, mostly romantic books. I also really enjoy playing games on my ps5. And last but not least I really enjoy baking, it makes me calm and the good thing is then I can eat my feelings. Random fact about me that everyone just should know, I am OBSESSED with unicorns!!!

Programming is hard and this website and I are not friends😬



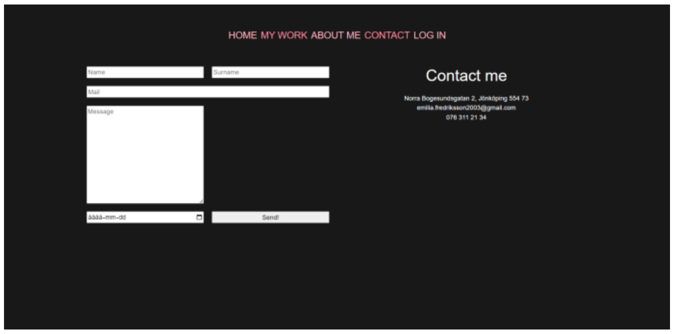
Who am I?

My name is Emilia Fredriksson and I am 20 years old. I live in Jönköping but is originally from a little village called Snogeröd in Skåne. I am studying my second year at the program New Media Design at Jönköping University. My big dream is to become a graphic designer some day and I feel like I'm on a good way there.

I really enjoy reading at my free time, mostly romantic books. I also really enjoy playing games on my ps5. And last but not least I really enjoy baking, it makes me calm and the good thing is then I can eat my feelings. Random fact about me that everyone just should know, I am OBSESSED with unicorns!!!

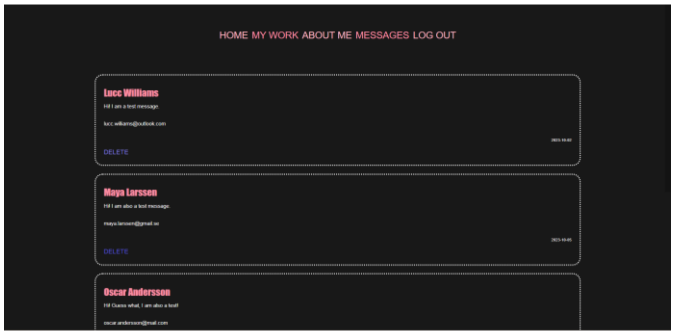
Programming is hard and this website and I are not friends😬

Figure 3.3 Result of About page first as site user, second as admin



Contact me

Norra Bogesundsgatan 2, Jönköping 554 73
emilia.fredriksson2003@gmail.com
076 311 21 34



Lucc Williams
Hi! I am a test message.
lucc.williams@outlook.com
2023-10-02
DELETE

Maya Larssen
Hi! I am also a test message.
maya.larssen@gmail.se
2023-10-05
DELETE

Oscar Andersson
Hi! Guess what, I am also a test!
oscar.andersson@mail.com

Figure 3.4 Result of the contact page and result of the message page

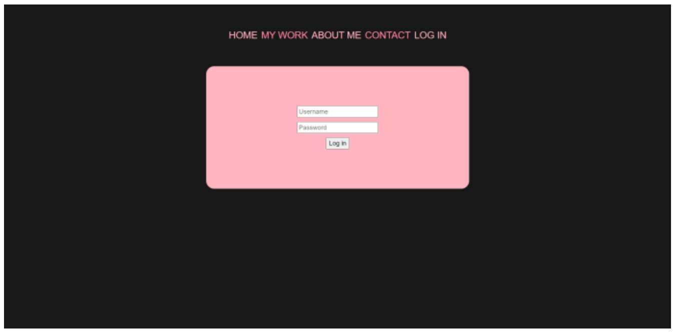


Figure 3.5 Result of login page

4) Discussion

This has been a very challenging, but also fun project. It has contributed to new knowledge about web development and how to go from static websites to more dynamic websites. I have used methods I never used before and with those methods created a portfolio for myself that is much more practical than the one I had before.

It has been interesting to learn more about servers and how to use JavaScript from the server side instead of the client side. It has also been interesting to get knowledge about and use databases to store information, which also made it easier to display projects without needing to add every project, one by one, in code.

I have learned to use CRUD which made adding, modifying, and deleting projects from my portfolio very easy. I have also learned a lot about security and how to implement that only one or a couple of people can change things on the website.

Back-end development is a really important part of websites and I have gotten a way better understanding of that by doing this project. In combination with the front-end development that I had knowledge about before this project, I could now do very dynamic and interesting websites.

5) References

Landré, J. (2023). *Express.* Canvas. https://canvas.ju.se

Landré, J. (2023). *MVC & Handlebars.* Canvas. https://canvas.ju.se

Rudbeck, L. (2023). *Introduction to HTML.* Canvas. https://canvas.ju.se

Rudbeck, L. (2023). *Introduction to JavaScript.* Canvas. https://canvas.ju.se

w3school (n.d.). *HTML introduction.* https://www.w3schools.com/html/html_intro.asp