National College of Ireland

BSCHSD 4 Computing

Software Development

Academic Year 2021/2022

Maciej Sobiecki

19457332

x19457332@student.ncirl.ie

Virtucard

Technical Report

Contents

Executive Summary

This report includes a review of the development of the VirtuCard website. The primary goal of which is raising awareness about the environmental impact of traditional celebration cards and providing individuals with an accessible and convenient alternative method. Through the course of this project, important features have been implemented, including a safe storage place for celebration cards, creating events, and a like and comment system. These enhancements have significantly improved the overall user experience of VirtuCard.

While significant progress has been made in achieving the project's objectives, it is important to acknowledge the limitations that were encountered. Challenges included the implemention of a scalable cloud-based database for the images and also, creating an online editor for the celebration cards. However, by implementing alternative approaches, the project mitigated these obstacles and delivered a functional web application that satisfied the initial expectations.

In summary, this web application has not only successfully raised awareness about the environmental impact of traditional greeting cards but also provided individuals with a user-friendly platform that encourages the use of sustainable practices. By reducing paper waste and promoting virtual exchanges, this website has taken a significant step towards a more environmentally conscious future.

Introduction

Background

The main motivation behind this project stems from the concern over the environmental impact of traditional celebration cards. Extensive research conducted during the preparation phase of the project has revealed astonishing statistics regarding the number of holiday cards mailed in the United States annually. It was discovered that approximately 1.3 billion holiday cards are sent each year [1].

This substantial volume of greeting cards has a significant environmental footprint. The research findings indicated that the production and distribution of these cards result in an ecological impact equivalent to charging 22 billion smartphones or generating the electricity needed for 22,000 homes in a single year [1]. Such statistics highlight the magnitude of the environmental consequences associated with traditional celebration cards.

In response to these findings, the decision was made to contribute to environmental conservation by developing a virtual celebration card web application. This innovative solution aims to provide an eco-friendly alternative to physical celebration cards. By transitioning from traditional paper-based cards to virtual cards, the project aims to mitigate the negative environmental effects caused by the production, transportation, and disposal of physical cards.

Aims

Virtucard web application will offers a platform where users can post virtual celebration cards for various occasions. The user will be able to create events which are showcased by a user uploaded image of a card followed by a description of the event. Other users of the website will be able to see the post made and interact with the post by leaving a like and a comment.

By leveraging digital technology, Virtucard web application eliminates the need for paper, printing, and physical transportation. This significantly reduces the carbon footprint associated with traditional cards. Additionally, the web application will promote sustainable practices by encouraging users to opt for electronic delivery methods instead of physical mail, further reducing energy consumption and waste generation.

The ultimate goal of this project is to raise awareness about the environmental impact of traditional greeting cards and provide individuals with an accessible and convenient alternative. By offering a user-friendly and visually appealing platform, the virtual card greeting web application aims to encourage widespread adoption of eco-friendly practices while still maintaining the heartfelt sentiment of sending greetings to loved ones.

Technology

To accomplish the goals that I have set out for this project I will utilize the following technologies:

As my project management tool, I have chosen Monday.com. Monday.com provides a platform for creating detailed project plans and managing tasks. It offers features such as task tacking, timelines, progress monitoring, and communication tools which will help me organise and streamlines my project.

Visual Studio Code will be the chosen code editor for the development process. It offers a range of features and extensions that enhance the coding experience, including syntax highlighting, integrated version control and debugging tools. VS code will be used throughout the entire development cycle, allowing for efficient coding and collaboration.

JavaScript will be used as the primary programming language for developing the backend of the web application. It will enable the creation of server-side logic and facilitate communication between the frontend and backend components. JavaScript's versatility and extensive libraries will be the perfect choice for the required functionalities of the application.

React.js will serve as the frontend framework for developing the user interface of Virtucard. It provides a component-based architecture that allows for the creation of reusable UI elements. React.js will be used to structure and develop the frontend components, manage the application state, and handle user interactions.

SASS will be used as the styling language for the front end of the web application. It extends the capabilities of CSS by providing features such as variables and nesting. SASS will be utilized to create a visually appealing and customised user interface for Virtucard.

Structure

In this technical report, I aim to provide comprehensive details regarding my website application. In Section 2.0, I will outline the essential requirements for the project, showcasing functional, data, user, environmental, and usability requirements. By showcasing the functional requirements, I will present a detailed use case scenario, allowing for a deeper understanding of the code. Additionally, I will describe the website's design and system architecture. To enhance comprehension, I will highlight the key functions employed in the code, sharing interesting code snippets as necessary. Furthermore, I will include screenshots of crucial pages, accompanied by explanations of their contents.

To ensure the robustness of the application, I will outline my testing plan and specifications used throughout the project, providing the results obtained from each test. Ultimately, I will conduct performance evaluations, assessing aspects such as scalability and correctness. In Section 3.0, I will conclude the project by presenting a comprehensive summary that highlights its advantages, disadvantages, strengths, and weaknesses.

Moving forward to Section 4.0, I will outline potential future directions the project could potentially take if additional resources and time were allocated. In Section 5.0, I will include references to the content discussed within this report. Finally, in Section 6.0, I will provide supplementary information such as reflective journals and the project proposal.

System

Requirements

Functional Requirements

This table provides a ranked order of functional requirements that have been implemented in this project.

Functional Require No.	Functional Requirement Description
FR 1	User registration
FR 2	User login
FR 3	User creating a post
FR 4	User creating an event
FR 5	User editing his details
FR 6	User post edit
FR 7	User viewing posts
FR 8	User viewing a post
FR 9	User viewing his events
FR 10	User viewing an event
FR 11	User deleting an event
FR 12	User deleting a post
FR 13	User Deleting his profile
FR 14	User Liking a post
FR 15	User Commenting on Post

Functional Requirement 1 – User Registration

Description & Priority

This use case shows the registrations steps of a new user. This use case is priority 1.

US-01

Scope

The scope of this use case is to allow the user to register into the page.

Description

This use case describes the procedures that the user takes to register web application.

Use Case Diagram
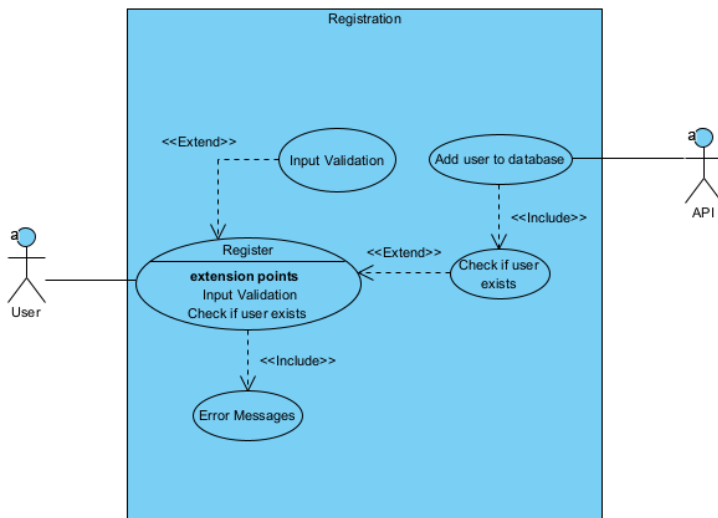
Flow Description

Precondition

N/A

Activation

The activation starts when the user clicks on the register button.

Main flow

1. User enters valid inputs into the form,
2. User is checked for invalid inputs,
3. API checks if the user already exists,
4. Adds user to the database.

Alternate flow

A1: User Exists

1. The API identifies the user that a user already exists,
2. The User receives an error message,
3. The use case continuous at position 1 of the main flow

A2: Invalid username

1. The user receives an error message,
2. The use case continuous at position 1 of the main flow

A3: Invalid email

1. The user receives an error message,
2. The use case continuous at position 1 of the main flow

A4: Invalid password

1. The user receives an error message,
2. The use case continuous at position 1 of the main flow

Exceptional flow
E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination
User is redirected to the login page.

Post condition
The API goes into wait state

Functional Requirement 2 – User Login

Description & Priority
This use case shows the login steps into the web application. This use case is priority 1.

US-02

Scope
The scope of this use case is to allow the user to login to the web application.

Description
This use case describes the procedures that the user takes to login to the web application.

Use Case Diagram
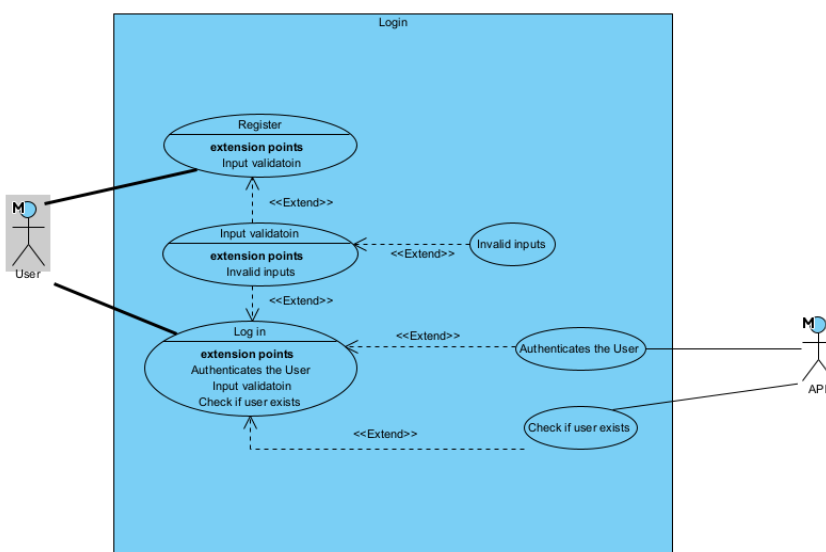
Precondition

User is registered to the website.

Activation

The activation starts when the user clicks on the login button in the navbar.

Main flow

1.   The system identifies the user is on the login page.
2.   User enters their valid inputs.
3.   System checks users' inputs
4.   API checks if the user exists in the database.
5.   API is authenticating the user.
6.   User is Logged in.

Alternate flow

A1: User Dose Not Exists

1.   The API identifies the user that a user dose not exists,
2.   The User receives an error message,
3.   The use case continuous at position 2 of the main flow

A2: Invalid username

1.   System identifies invalid username.
2.   The user receives an invalid username error message,
3.   The use case continuous at position 2 of the main flow

A3: Invalid email

1.   System identifies invalid email.
2.   The user receives an invalid email error message,
3.   The use case continuous at position 2 of the main flow

A4: Invalid password

1.   System identifies invalid password.
2.   The user receives an invalid password error message,
3.   The use case continuous at position 2of the main flow

Exceptional flow

E1: Lost connection

1.   The API lost connection to the user,
2.   The user receives an error message,
3.   The use case continuous at position 1 of the main flow

Termination

User is redirected home page.

Post condition

The API goes into wait state.

Functional Requirement 3 – User Creating a new post.

Description & Priority

This use case shows a user creating a new post. Priority 2

US-03

Scope

The scope of this use case is to show the user creating a new post.

Description

This use case describes the procedures that the user takes to create a new post.

Use Case Diagram
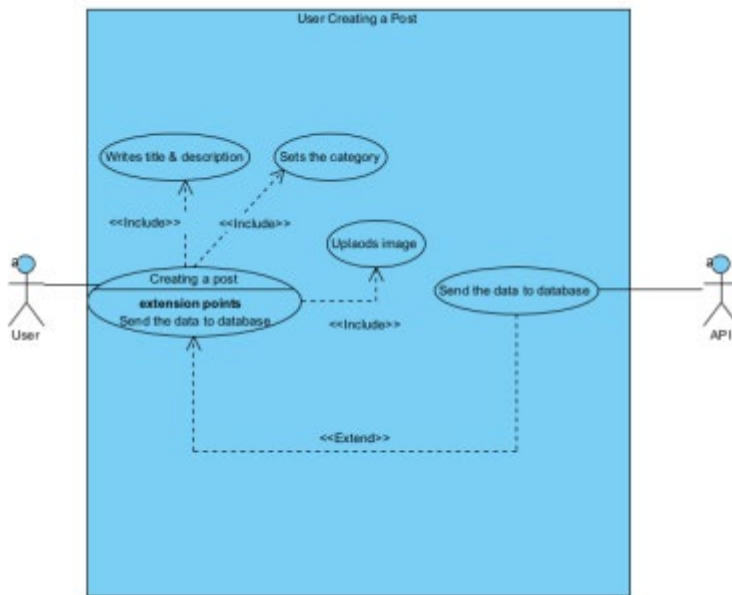
Flow Description

Precondition

- User is registered to the website.
- User is logged in and authenticated.

Activation

The activation starts when the user clicks on the Post button in the navbar.

Main flow

1. The System checks the user is logged in.
2. User will enter valid inputs.
3. User successfully uploads an image.
4. User successfully selects the category.
5. User will click on the publish button.
6. System checks for validation.
7. The API will check the user for Authentication.
8. The API will send users post into the database.

Alternate flow

A1: User is not logged in.

1. System confirms the user is not logged in.
2. System displays error message.
3. The use case continues at position 1 of the main flow.

A2: User is not authenticated.

1. API confirms user is not authenticated.
2. The user receives an error message.
3. The use case continuous at position 1 of the main flow

Exceptional flow
E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination
User is redirected home page.

Post condition
The API goes into wait state.

Functional Requirement 4 – User Creating a new event.

Description & Priority
This use case shows a user creating a new event. Priority 2
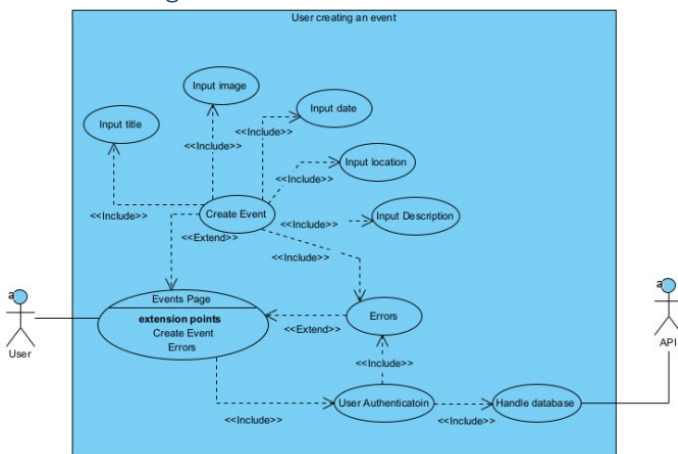
US-04

Scope
The scope of this use case is to show the user creating a new event.

Description
This use case describes the procedures that the user takes to create a new event.

Use Case Diagram

Flow Description

Precondition
- User is registered to the website.
- User is logged in and authenticated.

The activation starts when the user clicks on the "Your Events" button in the navbar.

Main flow
1. The System checks the user is logged in. (A1)
2. User will click on the create event button.
3. System will display a pop up form.
4. User will enter valid inputs. (A3)
5. User successfully uploads an image.
6. User successfully selects the date.
7. User will click on the Create Event button.
8. System checks for validation.
9. The API will check the user for Authentication. (A2)
10. The API will send users event into the database.

Alternate flow
A1: User is not logged in.

1. System confirms the user is not logged in.
2. System displays error message.
3. The use case continues at position 1 of the main flow.

A2: User is not authenticated.

1. API confirms user is not authenticated.
2. The user receives an error message.
3. The use case continuous at position 1 of the main flow

A3: User entered invalid input.

1. System denied invalid input.
2. The user receives an error message.
3. The use case continuous at position 1 of the main flow

Exceptional flow
E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination
- User clicks on the close button.
- New event window is closed.

Post condition
The API goes into wait state.

Functional Requirement 5 – User Editing their details.

Description & Priority

This use case shows a user updating their details. Priority 3

US-05

Scope

The scope of this use case is to show the user updating their details.

Description

This use case describes the procedures that the user takes update their details.

Use Case Diagram
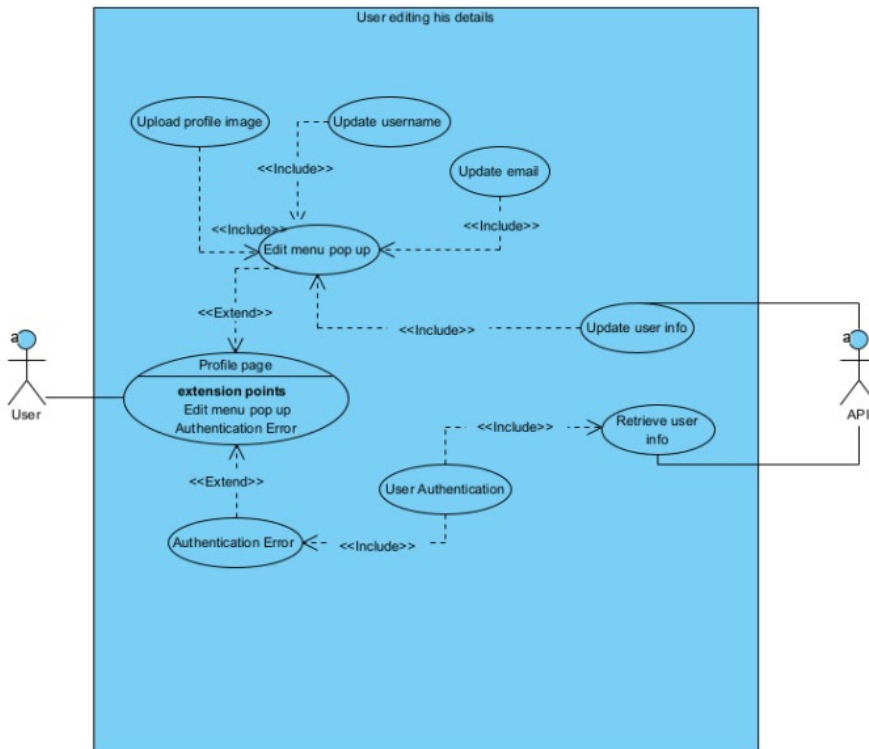
Flow Description

Precondition

- User is registered to the website.
- User is logged in and authenticated.

Activation

The activation starts when the user clicks on the "Edit Profile" button in profile page.

Main flow

1. The System checks the user is logged in. (A1)
2. The API will check the user for Authentication. (A2)
3. User will click on the update profile button.
4. System will display a pop-up form.
5. User will enter valid inputs. (A3)
6. User successfully uploads an image.
7. User will click on the Update button.
8. API will check existing users. (A4)
9. The API will send users event into the database.

A1: User is not logged in.

1. System confirms the user is not logged in.
2. System displays error message.
3. The use case continues at position 1 of the main flow.

A2: User is not authenticated.

1. API confirms user is not authenticated.
2. The user receives an error message.
3. The use case continuous at position 1 of the main flow

A3: User entered invalid input.

1. System denied invalid input.
2. The user receives an error message.
3. The use case continuous at position 5 of the main flow

A4: User Exists

1. The API identifies the user that a user already exists,
2. The User receives an error message,
3. The use case continuous at position 5 of the main flow

Exceptional flow

E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination

- User clicks on the close button.
- Update profile window is closed.

Post condition

The API goes into wait state.

Functional Requirement 6 – User updating a post.

Description & Priority

This use case shows a user updating their post. Priority 3

US-06

Scope

The scope of this use case is to show the user updating a post.

Description

This use case describes the procedures that the user takes to update a post.

Use Case Diagram
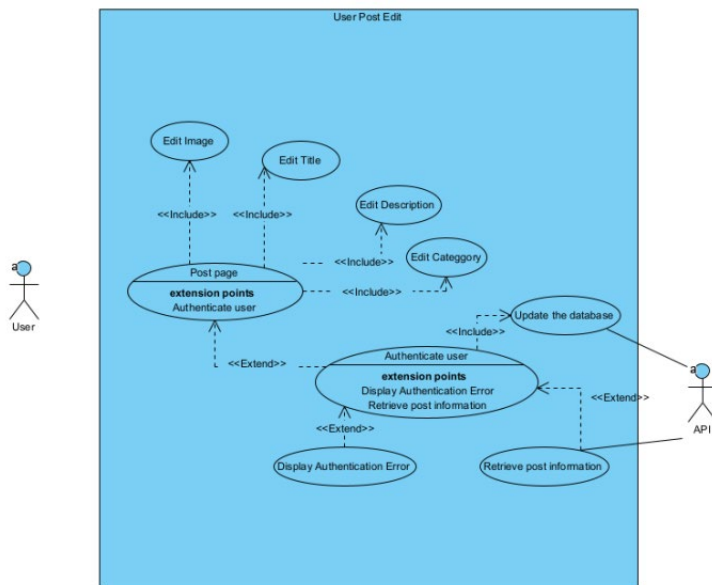
Flow Description

Precondition

- User is registered to the website.
- User is logged in and authenticated.
- User has made a post.

Activation

The activation starts when the user clicks on the edit icon in the single post page.

Main flow

1. The System checks the user is logged in. (A1)
2. The API will check the user for Authentication. (A2)
3. User will enter valid inputs.(A3)
4. User successfully uploads an image.
5. User successfully selects the category.
6. User will click on the publish button.
7. System checks for validation.
8. The API will check the user for Authentication. (A2)
9. The API will send users post into the database.

Alternate flow

A1: User is not logged in.

1. System confirms the user is not logged in.
2. System displays error message.
3. The use case continues at position 1 of the main flow.

A2: User is not authenticated.

1. API confirms user is not authenticated.
2. The user receives an error message.
3. The use case continuous at position 1 of the main flow

A3: User entered invalid input.

1. System confirms invalid input.
2. The user receives an error message.
3. The use case continuous at position 3 of the main flow

Exceptional flow

E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination
User is redirected home page.

Post condition
The API goes into wait state.

Functional Requirement 7 – User viewing posts.

Description & Priority
This use case shows a user viewing posts. Priority 4

US-07

Scope
The scope of this use case is to show the viewing posts.

Description
This use case describes the procedures that the user takes to view posts.

Use Case Diagram

Flow Description

Precondition

N/A
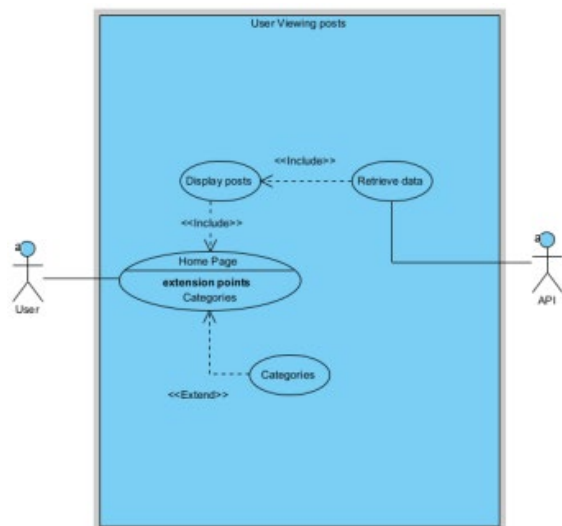
Activation

The activation starts when the user enters the home page.

Main flow

1. System checks if user is logged in.
2. User enters the home page.
3. API will receive posts.
4. User can view the posts.

Alternate flow

A1: No posts in the database.

1. System confirms invalid input.
2. The user receives an error message.
3. The use case continuous at position 3 of the main flow

Exceptional flow

E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination

User is redirected home page.

Post condition

The API goes into wait state.

Functional Requirement 8 – User Viewing a post.

Description & Priority

This use case shows a user viewing a post. Priority 4

US-08

Scope

The scope of this use case is to show viewing a post.

Description

This use case describes the procedures that the user takes to view a post.

Use Case Diagram
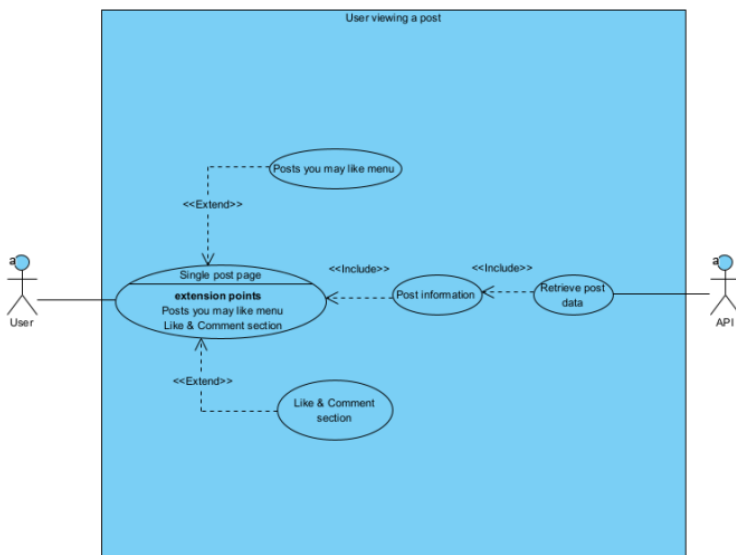
Flow Description

Precondition

- User is registered to the website.
- User is logged in and authenticated.

Activation

The activation starts when the user clicks on the Read More button under a specific post.

Main flow

1. The user clicks on the read more button.
2. The System checks the user is logged in. (A1)
3. The API will check the user for Authentication. (A2)
4. The API will transfer post data.
5. The API will transfer like data.
6. The API will transfer comment data.

Alternate flow

A1: User is not logged in.

1. System confirms the user is not logged in.
2. System displays error message.
3. The use case continues at position 1 of the main flow.

A2: User is not authenticated.

1. API confirms user is not authenticated.
2. The user receives an error message.
3. The use case continuous at position 1 of the main flow

Exceptional flow

E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination

User is changing the page.

Post condition

The API goes into wait state.

Functional Requirement 9 – User Viewing their events.

Description & Priority

This use case shows a user viewing their events. Priority 4

US-09

Scope

The scope of this use case is to show user viewing their events.

Description

This use case describes the procedures that the user takes to view their events.

Use Case Diagram
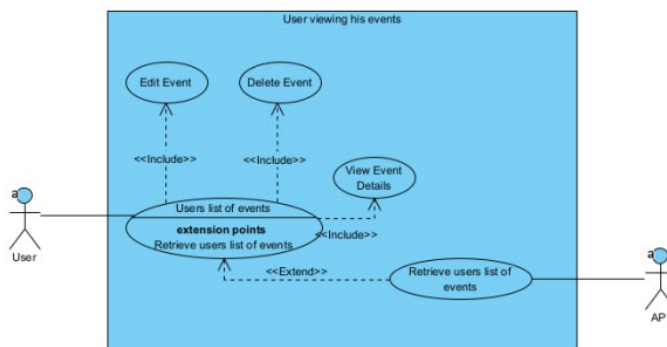
Flow Description

Precondition

- User is registered to the website.
- User is logged in and authenticated.

Activation

The activation starts when the user clicks on the "Your events" button in the navbar.

Main flow

1. The System checks the user is logged in. (A1)
2. The API will check the user for Authentication. (A2)
3. The API will transfer events data.
4. The User will be able to view his events.

Alternate flow

A1: User is not logged in.

1. System confirms the user is not logged in.
2. System displays error message.
3. The use case continues at position 1 of the main flow.

A2: User is not authenticated.

1. API confirms user is not authenticated.
2. The user receives an error message.
3. The use case continuous at position 1 of the main flow

Exceptional flow

E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination

User is changing the page.

Post condition

The API goes into wait state.

Functional Requirement 10 – User Viewing an event.

Description & Priority

This use case shows a user viewing an event. Priority 5

Scope

The scope of this use case is to show user viewing an event.

Description

This use case describes the procedures that the user takes to user viewing an event.

Use Case Diagram
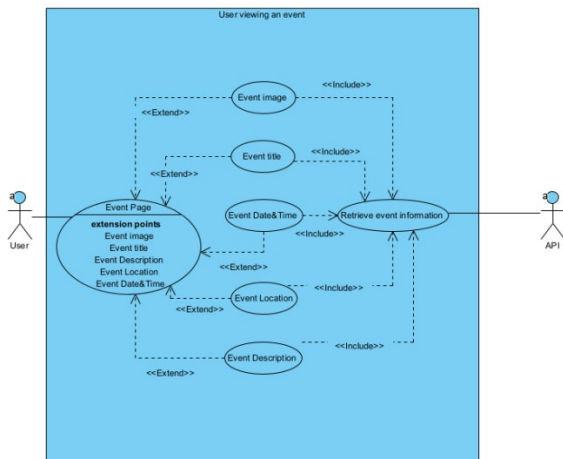
Flow Description

Precondition

User made an event.

Activation

The activation starts when the user clicks on the "Read More" button in the Events Page.

Main flow

1. User enters the event page.
2. The API will transfer events data.
3. The User will be able to view an event.

Alternate flow

A1: No event found.

1. API will confirm that no event has been found.
2. System displays error message.
3. The use case continues at position 1 of the main flow.

Exceptional flow

E1: Lost connection

1. The API lost connection to the user,
2. The user receives an error message,
3. The use case continuous at position 1 of the main flow

Termination

User is changing the page.

Post condition

The API goes into wait state.

Data Requirements

The development of VirtuCard has incorporated the following data requirements to ensure efficient management, security, and utilization of data:

User Data

User data is collected to establish an identification system for users when creating posts and writing comments. This includes user profiles, contact information, and preferences.

Card Content

Card content is collected to enable users to share their cards with others. This includes text and images.

Event Information

Event information is collected to facilitate the sharing of events among users. This includes event names, dates, locations, and all other relevant details.

User Requirements

Understanding the needs and expectations have been considered during the development of VirtuCard. User requirements focus on the functionalities and features that will enhance the user experience and engagement. By implementing these requirements, Virtucard will effectively meet the identified needs of the user. The following user requirements were included during the development process:

Card Posting and Interactions

Users are able to create and publish cards similar to a blog post. VirtuCard provides a user-friendly interface for composing and formatting card content. VirtuCard enables users to add image and text to their cards. Users are able to like and comment on the cards, promoting interaction and engagement.

Event Creation and Sharing

Users have the ability to create and share events on VirtuCard. This feature was implemented to enable users to create events with details such as the name, date, and location. A shareable link can be used to link or invite users. Event details are displayed clearly and attractively to improve event participation.

Profile Editing and Personalization

Users are able to edit and personalize their profiles on VirtuCard. VirtuCard offers a user-friendly profile management interface that allows user to update their information, including their profile picture.

Environmental Requirements

The environmental requirements for VirtuCard include the requirement of hardware, software, and network components for the application to operate effectively and securely. By ensuring these requirements are met, users can access and use VirtuCard optimally. The following environmental requirements should be considered:

Hardware

VirtuCard requires minimal hardware specifications for optimal performance. Users should have access to a computer or mobile device with a functional mouse and keyboard or touch screen. Any display device, such as a monitor or mobile screen, will enable users to fully experience VirtuCard's unique features.

Software

To access VirtuCard, users will need a web browser, which is a standard software available on most devices. VirtuCard is designed to be compatible with popular web browsers, such as Chrome, Firefox, and

Safari. For developers working on VirtuCard, a code editor like Visual Studio Code is recommended for efficient code editing and debugging.

Network

A reliable internet connection is essential for VirtuCard to function. Users should have access to a stable internet connection to ensure smooth browsing and interaction with the application. A broadband connection is preferable to minimize potential latency issues during card browsing and management.

Usability Requirements

Usability requirements are crucial for creating a successful web application that provide an enhanced user experience. These requirements ensure that users can effectively and efficiently accomplish their tasks while enjoying easy to navigate interfaces. Virtucard has considered the following usability requirements.

Learnability

Virtucard is designed to be intuitive and easy to learn for users of all levels of experience. Users should understand the features, functions, and navigations without extensive training or documentation.

Error Prevention and Handling

Virtucard proactively prevents errors and guides the user in correcting any mistakes. It uses clear and descriptive error messages and validation mechanisms to help users avoid errors.

Responsiveness

Virtucard responds quickly to user interactions, providing real- time feedback and reducing delays. Smooth page transitions and quick loading times contribute to a good user experience. Users should not face frustrating delays or unresponsive elements during their interactions with this application.


Design & Architecture

The VirtuCard system's architecture is divided into two folders: the API folder and the client folder.

The API folder houses the backend code responsible for handling communication between the database and the front end of the application. It serves as an interface that exposes the endpoint to the front end, allowing it to interact with the backend functionality. In this folder, one would find the routes and controllers. Controllers included all the functions that are required for the front end to run. The functions consist of SQL code which communicates to the database, authentication of a user, and the encryption of passwords. Some examples of the functions that can be found are login(), getPost(), addPost(). The Router folder includes files that create a router endpoint for which the front end can communicate to. For example, router.post("/register", register), This creates an endpoint for the frontend and assigns the function register from the controller file to it. Index.js is the main file of the backend. This file contains CORS option, handles uploaded files, and creates a storage for the files.

The client folder contains all the front-end related code, including page, component stylesheets and JavaScript files responsible for user interactions. It focuses on presenting the user interface and handling user input or actions, such as form submission, button click and navigation. Input validation is included in each form allowing for the appropriate data to be handled in the API. The client folder communicates with the backend API by making requests to the appropriate endpoints and receiving responses. This interaction enables the frontend to retrieve or send data to the backend for processing.

By separating the backend and frontend into distinct folders, a clear separation is maintained. This approach allows for easier maintenance and scalability, as changes in one part of the application is less likely to affect the other.

Database
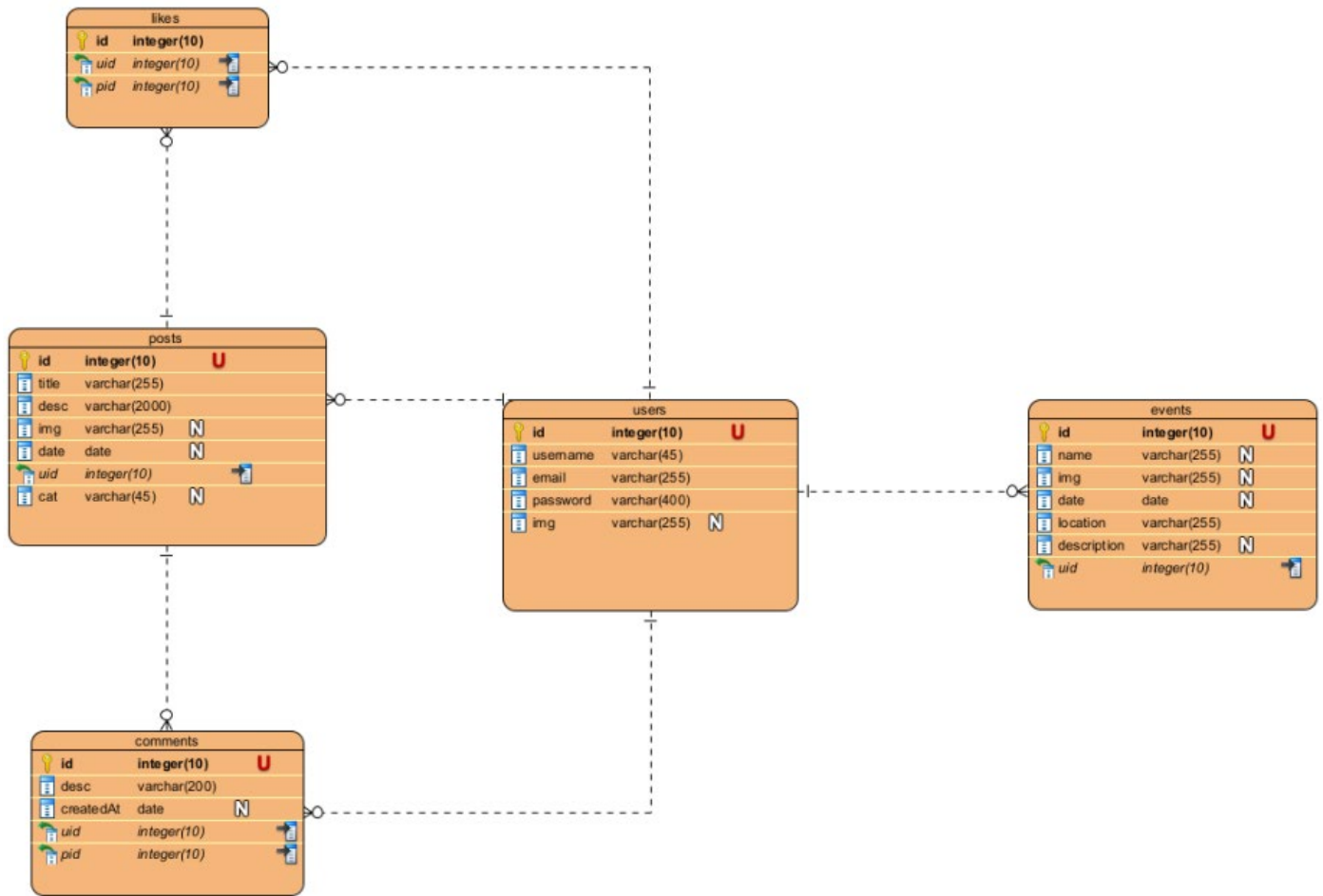
The above ERD diagram represents a visual structure to the database used by VirtuCard. It consists of the users, events, posts, comments, and likes tables. The users' tables have a one-to-many relationship with the posts, events, comments and likes tables through the uid foreign key. The posts table has a one-to-many relationship with the comments and likes table through the pid foreign key. This means that users only have access to the posts, evens, likes and comments that they own. Upon deletion of the user profile, a cascade function has been selected so that any data relevant to the user profile is destroyed upon deletion.

Implementation

The key and sole algorithm that I have used in this project is the bycryp package. Bycrypt is a password hashing function that encrypts the user's password upon submitting the registration form [2]. The input to the bcrypt function is the password string which has a maximum string value of 72 bytes, a numeric cost and a 16-byt salt value which in my instance is 10. The salt usually is a randomly selected value by the user. Bcrypt function uses these inputs to create a 24-byte hash. Bcrypt uses the hash algorithm identifier of $2a$.

```javascript
export const register = (req, res) => {

    // Check if user already exists in the database
    const q = "SELECT * FROM users WHERE email = ? OR username = ?"

    db.query(q, [req.body.email, req.body.username], (err, data) => {
        if (err) return res.json(err)
        if (data.length > 0) return res.status(409).json("User already exists!")

        // Hash the password and insert the user's information into the database
        const salt = bcrypt.genSaltSync(10);
        const hash = bcrypt.hashSync(req.body.password, salt)

        const q = "INSERT INTO users(`username`,`email`,`password`) VALUES (?)"
        const values = [
            req.body.username,
            req.body.email,
            hash,
        ]

        db.query(q, [values], (err, data) => {
            if (err) return res.json(err)
            return res.status(200).json("User has been created.")
        })

    })

})
```

In the provided code snippet we can se the register function in the backend API routes file. Initially this function checks if the user is already registeted by checking if the query can select the matching usernam or emai from the users table. If it matches and the data has a length of more than 0 it returns an error that the user already exsits. If the data is not found in the users table, the salt variable is genereate with the value of 10. Then the hashed password is genereted with the encryption of the user provided password the the salt value. The entered values by the user is inserted into the database with the hash password instead of the actual passoword. Return response is made that the user is created. This is the output of such encryption

"$2a$10$3q.rBmiFAKhbPXdTQDnv0elinB/ZwtsdcOU2ekGp33HynV.mqX13y"

API

Main file in the API folder is the index.js. This file sets up an express server that handles requests to various API endpoints releated to user authentication, posts, events, comments an likes. It uses middleware packages such as expess.json() to parse incoming JSON data, cookie-parser to parse cookies and cors to enable Cross-Origin Resource Sharing. It also uses multer middleware to handle file uploads for posts, profiles and events.

Set up of the express server

```javascript
// Handle requests to API endpoints for user authentication, posts, events, comments, and likes
app.use("/api/user", userRoutes)
app.use("/api/auth", authRoutes)
app.use("/api/posts", postRoutes)
app.use("/api/events", eventRoutes)
app.use("/api/comments", commentsRoutes)
app.use("/api/likes", likesRoutes)
```

This is done to all three types of upload available on virtucard being events image, post image, and user profile image.

```javascript
// Configure storage options for file uploads
const postStorage = multer.diskStorage({
    destination: function (req, file, cb) {
        cb(null, '../client/public/upload/posts')
    },
    filename: function (req, file, cb) {
        cb(null, Date.now() + file.originalname)
    }
})
```

```javascript
// Create instances of the multer middleware for file uploads
const postUpload = multer({
    storage: postStorage,
    limits: {
        fileSize: 1024 * 1024 * 5, // Limit file size to 5MB
        files: 1 // Limit number of files to 1
    }
})
```

```javascript
// Handle file uploads for posts, profiles, and events
app.post('/api/postUpload', postUpload.single('file'), function (req, res) {
    const file = req.file
    res.status(200).json(file.filename)
})
```

As shown in the code above its creating the endpoint for the indevidual Routes. This framework is crucial to the operation of this weeb application.

Client

In the front end of virtucar the main file is called App.js This file containes react code that sets up a client-side router using react-router-dom library. It defines serveral routes for differe pages of the application. It defines a layout component which includes a navbar, a footer, and a contenent area.

```
  return (
    <QueryClientProvider client={queryClient}>
      <Navbar />
      <Outlet />
      <Footer />
    </QueryClientProvider>
  )
}

// Creating the router
const router = createBrowserRouter([
  {
    path: "/",
    element: <Layout />,
    children: [
      {
        path: "/",
        element: <Home />
      },
      {
        path: "/post/:id",
        element: <Single />
      },
      {
        path: "/write",
        element: <Write />
      },
      {
        path: "/profile",
        element: <Profile />
      },
      {
        path: "/events",
        element: <Events />
      },
      {
        path: "/events/:id",
        element: <Event />
      },
```
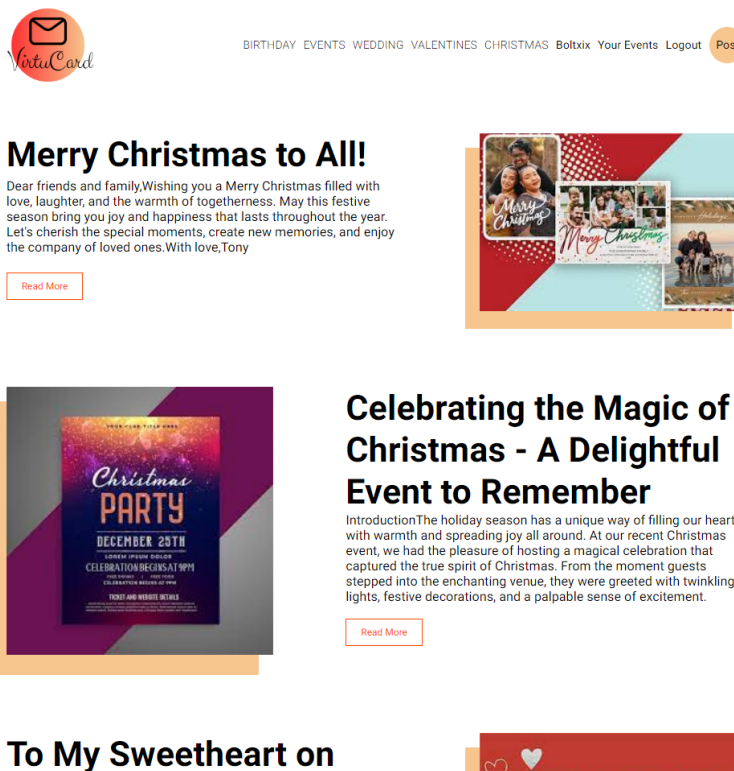
Home Page

Represented above is the main page to the application, at the top of the page the is the Navbar component which includes the Logo of VirtuCard, acting as redirect to home page button. Next to the logo, different categories can be found which act as a search bar navigating users to the post categories they want to see. If the user is logged, their username is displayed in the navbar component. It is used as a redirect to the profile page where the user can see their profile information and the actions to edit and delete the user. "Your events "buttons are a functional button that redirects the user to the events page. In the events page the user can find all the events the user has created functions such as create new event and delete the event are present. The logout buttons are used to logout the user from the application, it sets the local storage to null and securely removes the cookies from the session. The Post button is a link to the core functionality of the web application, it directs the user to a page that presents a form. The form is filled out by the user and is displayed here in the home page. On the home page a list of all posts is displayed, the title and a brief description can be found. Each post contains a read more button which redirects the user to the single post page.
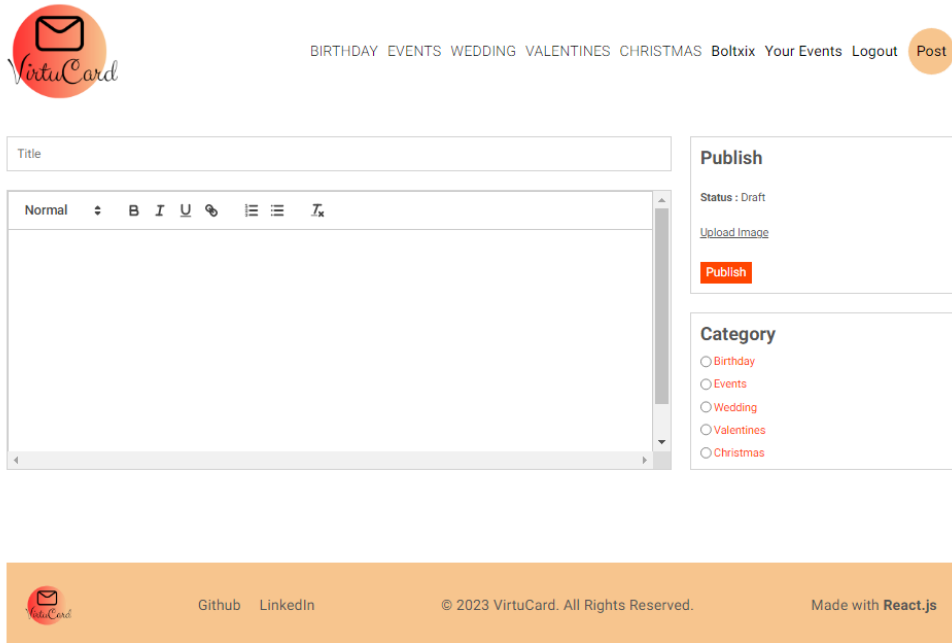
Write Page

Screenshot above showcases the write page. In this page the user can enter information about the post they want to create. It requires details such as the Title of the post and the description of the post. The category must be selected by the user to help others with navigating the home page. The user must accompany this form with an upload of a single jpeg or .png image. Input validation is done on this page, and it is presented to the user under the publish button. Once valid information is entered the user can publish the post, the data is sent to the backend API which stores the post in the database.

In the screenshot above is the GUI for the user's events, in this page user can create a new event which is done through prompting the "create event" button. When pressed the button opens a pop-up window where the user can submit a form which requires information such as title, description, location, date, and image. Every user has their own personal events page. The user can click on the details button which opens the event page.

As mentioned earlier, when the user clicks on the read more button under a specific post, they are navigated to a single post page. In this page the user is able to view the contents of the post. The user can interact with the post by liking the post and creating a comment. If the user is the owner of the post the user can edit the post and delete the post. Other posts that are in the same category as the selected post will be displayed in the menu component labelled as "Other posts you may like"

Testing

Private testing was conducted at the development of each page, functionality & component. Testing that has been reported was done at the end of each month which tested the functionalities and its styling as well as its backend requests. This provided assurance that the new additions did not break the previous code. I have written out a large number of test cases which tested every page and component of the front-end application. Upon finding any defects, it is identified with its own id number Once the fix is implemented the id of the defect is included in the commit message to the GitHub repository. The test for the failed test case is done again to ensure the fix has resolved the issue.

Testing report

Testing Report for VirtuCard

Introduction

This report gives a thorough description of the testing done during the creation of VirtuCard, a web-based application made to give users a digital platform for sharing and discussing celebration cards. The main goal of the testing was to make sure that the application was reliable, usable, and functional. The findings of these tests are compiled in the report, which also identifies any significant problems or flaws that were found.

Overall, in this my testing:

Number of tests:	PASS / FAIL
95	PASS
6	FAIL

Testcase 01 - Navigation Bar

User Story - 01

Title: Navbar is displayed	Priority: 1	
User Story: As a user who has not registered yet I want to see the navigation bar displayed And I want to see the logo displayed And I want to see the categories "Birthday, Events, Wedding, Valentines, Christmas" Displayed And I want to see the "Login" button displayed And I want to see the "Post" button displayed		
Acceptance Criteria: Given user has not been registered yet When the user is on the home page Then the navigation bar is displayed And the logo displayed And the categories "Birthday, Events, Wedding, Valentines, Christmas" Displayed And the "Login" button displayed And the "Post" button displayed		

- Logo is displayed as expected
- Categories are displayed as expected
- Login button is displayed as expected
- Post button is displayed as expected

User Story - 02

Title: Nab Bar Logged out Display	Priority: 1	
User Story: As a user who has not registered yet I want to see my username in the navigation bar And see the logout button So that I can access the user profile page And logout from my account		
Acceptance Criteria: Given user has not been registered yet When the user is on the home page Then the user cannot see the username And the user cannot see the logout button		

Results:

- User cannot see his username as expected
- User cannot see the logout button as expected

User Story - 03

Title: Navigation Bar – Post button – Logged out	Priority: 1	
User Story: As a user who has not registered yet I want to be able to access the post page through clicking the "Post" button So That I can create a post		
Acceptance Criteria: Given user has not been registered yet When the user is on the home page And Clicks on the button page Then the user is navigated to the write page		

And a message is displayed for the user to log in

Results:

Test 1:

- User has been navigated to the Post page with no error
  - Defect has been raised, D-01

Test:2

- User has been navigated to the post page and an Error message has been displayed for the user to login as expected
  - Defect D-01 fixed

User Story - 04

Title: Navbar Logged in Display	Priority: 1	
User Story: As a user who has already registered And is logged in I want to see my username in the navigation bar And see the Your Event button And see the login button change to logout button So that I can access the user profile page And see my events And logout from my account User used to log in: Maciek		

Acceptance Criteria:
Given user has already registered And user is logged in When the user is on the home page Then the user can see the username And the user can see Your Events button And the user can see the login button change to the logout button

Results:

- Username is displayed as expected
- Your Events is displayed as expected
- Login button has changed to the Logout button as expected

User Story - 05

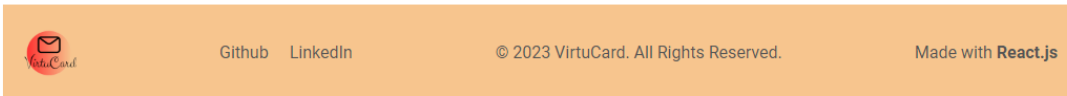| Title: Navbar – Username button functionality	Priority: 2	
User Story: As a user who has already registered And is logged in I want to see my username in the navigation bar So that I can access the user profile page User used to log in: Maciek		
Acceptance Criteria: Given user has already registered And user is logged in When the user clicks on his username Then the user is navigated to the user profile page		

BIRTHDAY EVENTS WEDDING VALENTINES CHRISTMAS Maciek Your Events Logout Post

Maciek
sobiecki.maciek@gmail.com

Edit Profile Delete Profile

Github LinkedIn © 2023 VirtuCard. All Rights Reserved. Made with **React.js**

- User is navigated to the user page as expected

User Story - 06

Title: Navbar – Your Events button functionality	Priority: 2	
User Story: As a user who has already registered And is logged in I want to see "Your Events" button So that I can access my events User used to log in: Maciek		
Acceptance Criteria: Given user has already registered And user is logged in		

When the user clicks on the "Your Events" button Then the user is navigated to the user's events page		

Results:

- User is navigated to the user's event page

User Story - 07

Title: Navbar – Logout Button Functionality	Priority: 1	
User Story: As a user who has already registered And is logged in I want to see the Log Out button So that I can log out of my profile User used to log in: Maciek		
Acceptance Criteria: Given user has already registered And user is logged in When the user clicks on logout button Then the user is navigated to the main page And Your Events button is no longer displayed And Users username is no longer displayed And the "logout" button is changed to login		

- User is navigated to the main page as expected
- "Your Events" button is no longer displayed as expected
- "Username" is no longer displayed as expected
- The logout button is changed to login as expected

User Story - 08

Title: Navigation Bar – Post button – Logged In	Priority: 1	
User Story: As a user who has registered And is logged in I want to be able to access the post page through clicking the "Post" button So That I can create a post		
Acceptance Criteria: Given user has registered And is logged in When the user is on the home page And Clicks on the post button Then the user is navigated to the write page		

Results:

- User has been navigated to the write page as expected

User Story - 09

Title: Navigation Bar – Categories	Priority: 1	
User Story: As a user who is not logged in I want to be able to access the different categories of cards So That I can view the cards that a sorted by category		
Acceptance Criteria: Given user is not logged in When the user clicks on the different post categories Then the user is navigated to the desired category and posts are displayed only according to that category		

Results:

Valentines

Valentines

- Clicked on Wedding to go to the wedding category

VirtuCard BIRTHDAY EVENTS WEDDING VALENTINES CHRISTMAS Maciek Your Events Logout Post

Wedding

- Wedding category is displayed as expected

User Story - 01

Title: Footer - Display	Priority: 3	
User Story: As a user who is on the home page I want to see the footer displayed according to the design specification So That I can view the footer And see the logo displayed And see the GitHub text And see the LinkedIn text And see the copyright reserved And see the Made with react.js text		
Acceptance Criteria: Given user is on the home page When the user is on the bottom of the page Then the user can see the footer displayed And see the logo displayed And see the GitHub text And see the LinkedIn text And see the copyright reserved And see the Made with react.js text		
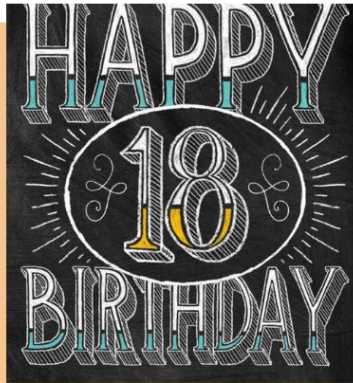
- User can see the footer displayed as expected
- User can see the logo displayed as expected
- User can see the GitHub text as expected
- User can see the LinkedIn text as expected
- User can see the copyright reserved text displayed as expected
- User can see the made with React.js as expected

User Story - 02

Title: Footer – Display Functionality	Priority: 5	
User Story: As a user who is on the home page And is viewing the footer component I want to click on the GitHub And I want to click on the LinkedIn Text So That I can navigate to the Virtucard GitHub reposatory And I can navigate to the LinkedIn profile of Maciej Sobiecki		
Acceptance Criteria: Given user is on the home page And is viewing the footer component When the user clicks on the GitHub Text And the user clicks on the LinkedIn Text Then the user is navigated to the Virtucard GitHub page		

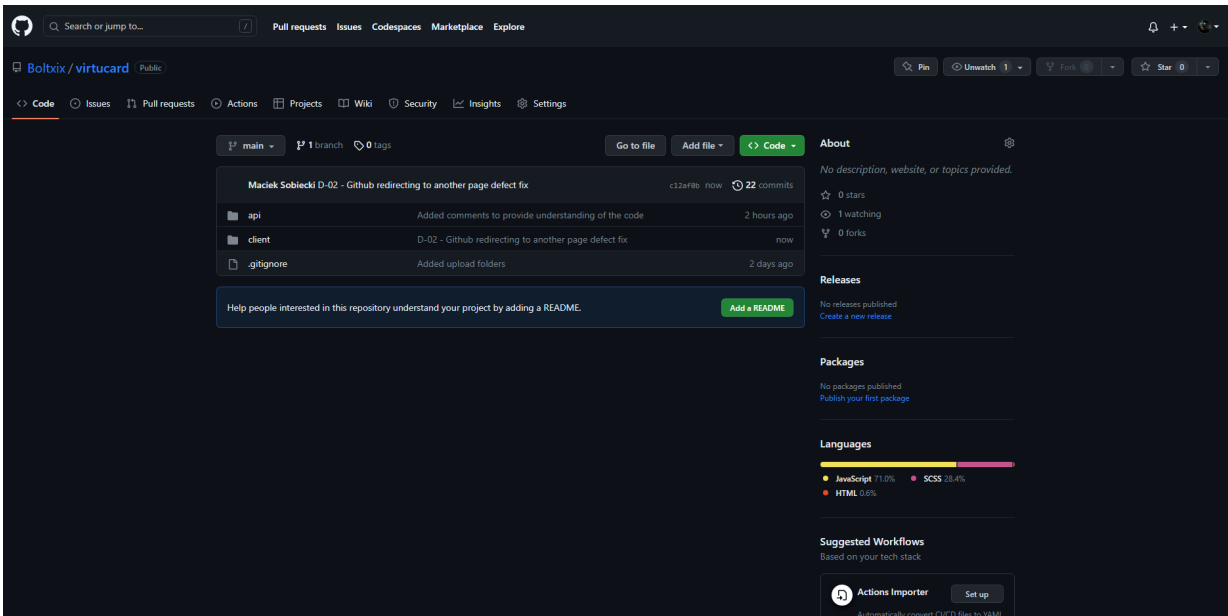| And the user is navigated to the LinkedIn profile of Maciej Sobiecki |
| --- |

Results:

Test 1:
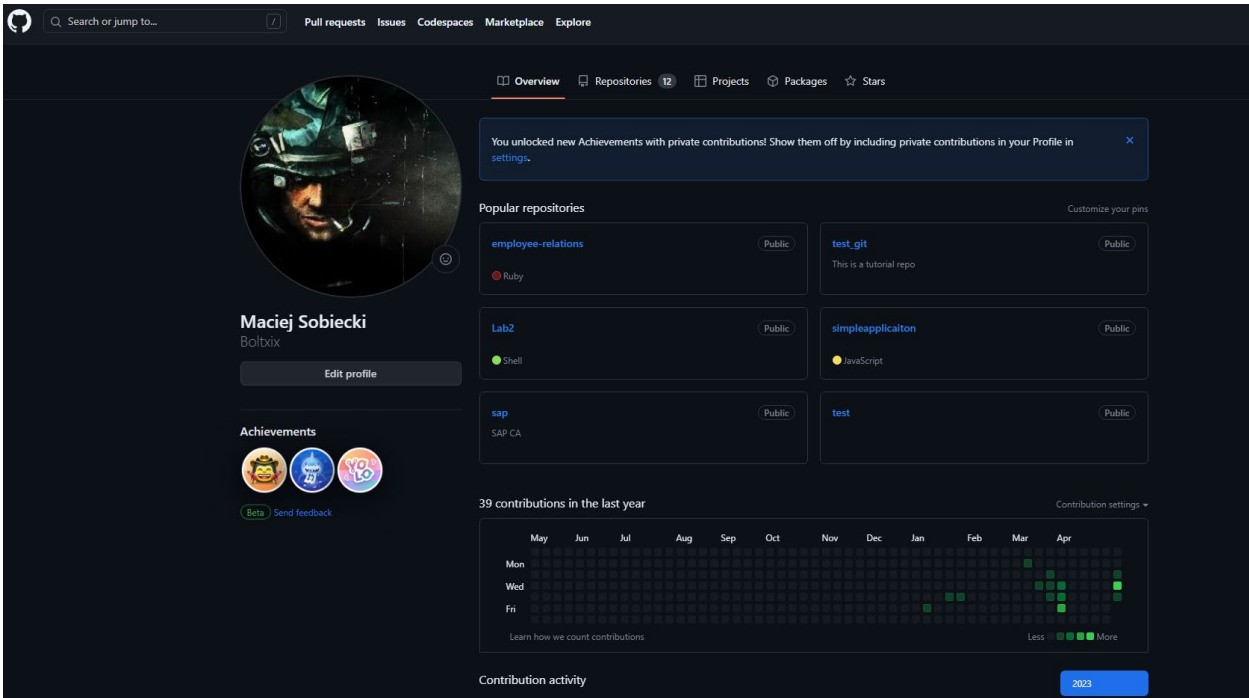
- User has been incorrectly navigated to the Maciej Sobiecki GitHub page rather than the virtucard repository
  - Defect raised – D-02

Test 2:

User has been correctly navigated to the virtucard repository as expected

User has been correctly navigated to the Maciej Sobiecki LinkedIn page as expected

Testcase 03 – Home page

User Story - 01

Title: Home page Display	Priority: 1	
User Story: As a user who has not registered yet I want to see the posts displayed according to the design specification So that I see the title of the post And I see the image of the post And I see the description of the post And I want to see the read more button		
Acceptance Criteria: Given user has not been registered yet When the user is on the home page Then the user can see the posts And the title of the post And the image of the post And the description of the pos And the read more button		

- User can see the post as expected
- User can see the title of the post as expected
- User can see the image of the post as expected
- User can see the description of the post as expected
- User can see the read more button as expected

User Story - 02

Title: Home page – Read more Button – Not logged in	Priority: 1	
User Story: As a user who is not logged in I want to be able to click on the read more button So that I can view more details about the selected post		
Acceptance Criteria: Given user is not logged in When the user clicks on the read more button Then the user is navigated to the selected post page And an error is displayed to view the page		

Results:

Test 1:

- Error is displayed when trying to go to the posts details when clicking on the read more button

Test 2:

BIRTHDAY EVENTS WEDDING VALENTINES CHRISTMAS   Login   Post

Please login to view more details about the post!

Github   LinkedIn   © 2023 VirtuCard. All Rights Reserved.   Made with **React.js**
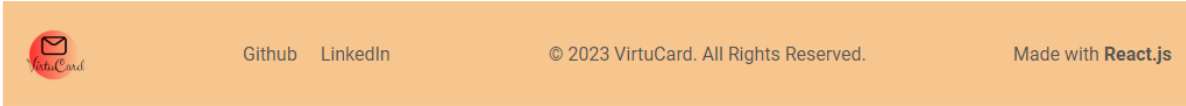
- Error is displayed as expected when trying to view a post
    - o   Defect D-03 fixed

User Story - 03

Title: Home page – Read more Button – Logged in	Priority: 1	
User Story: As a user who is logged in I want to be able to click on the read more button So that I can view more details about the selected post		
Acceptance Criteria: Given user is logged in When the user clicks on the read more button Then the user is navigated to the selected post page And can view more details		

Test 1:

Valentines

- Logged in user can read more details about the post as expected

Testcase 03 – Single Page
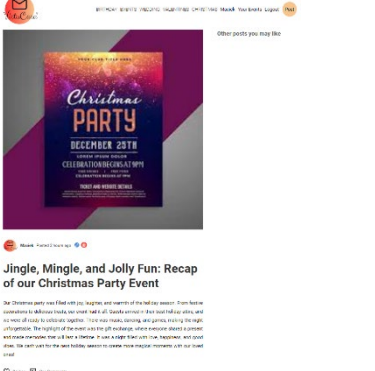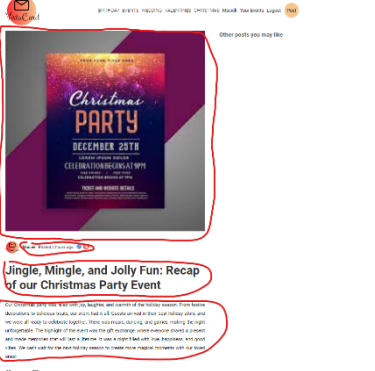
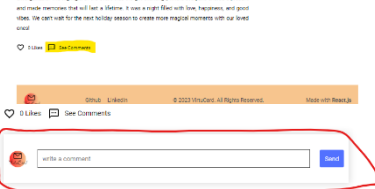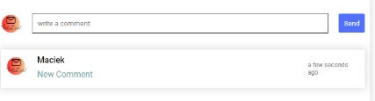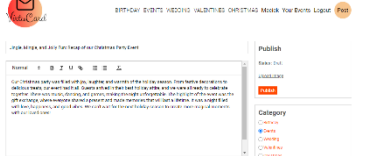| Test Case Name: Single Post Page Test |
| --- |
| Test Case ID: VC-SP-001 |
| Priority: High |
| Test Scenario:
To verify that the Single Post page is displaying the correct post details and functionality |
| Preconditions:
• The user is logged in.
• The user has created a post.
• The post has been published. |
| Test Steps:
1. Navigate to the Single Post page.
2. Verify that the post image, title, description, author, date, likes, and comments are displayed correctly.
3. Verify that the like button is displayed and can be clicked to like/unlike the post.
4. Verify that the comment section is hidden by default and can be toggled by clicking the "See Comments" button.
5. Verify that the comments section displays the correct comments for the post.
6. Verify that the edit and delete buttons are displayed if the user is the author of the post.
7. Verify that clicking the edit button navigates to the edit page with the post data pre-filled.
8. Verify that clicking the delete button prompts the user to confirm the deletion and deletes the post if confirmed.
9. Verify that an error message is displayed if there is no current user. |
| Expected Result: |

All test steps should pass, and the Single Post page should display the correct post details and functionality.
User Story: As a registered user, I want to view a single post with its details and functionality, so that I can like/unlike the post, view comments, and interact with the post if I am the author. I also want to be able to navigate to the edit page to edit the post and delete the post if necessary. If I am not logged in, I want to see an error message asking me to log in.

Results:

Test Step:	Test:	PASS OR FAIL
1.		PASS
2.		PASS
3.		PASS
4.		PASS
5.		PASS
6.		PASS
7.		PASS

8.		PASS
9.		PASS

Testcase 04 – Write Page

Test Case Name: Write Page Test
Test Case ID: VC-WP-001
Priority: High
Test Scenario: To verify that the Write page is functioning correctly and allows users to create or edit a blog post.
Preconditions: • The user is logged in.
Test Steps: 1. Navigate to the Write page.   2. Verify that the page displays an input field for the post title, a rich text editor for the post content, and radio buttons for selecting the post category.   3. Verify that the page displays a button to upload an image file and a button to publish the post.   4. Enter a title for the post in the input field.   5. Enter some content for the post in the rich text editor.   6. Select a category for the post using the radio buttons.   7. Click the button to upload an image file and select an image file from your computer.   8. Click the button to publish the post.   9. Verify that the post is published and displayed on the home page.
Expected Result: All test steps should pass, and the Single Post page should display the correct post details and functionality.
User Story: As a registered user, I want to view a single post with its details and functionality, so that I can like/unlike the post, view comments, and interact with the post if I am the author. I also want to be able to navigate to the edit page to edit the post and delete the post if necessary. If I am not logged in, I want to see an error message asking me to log in.

Results:

Test Step:	Test:	PASS OR FAIL
1.		PASS

2.			PASS
3.			PASS
4.	Title		PASS
5.	Normal B I U ⊟ ⊟ 𝐼ₓ Description		PASS
6.	Category ● Birthday ○ Events		PASS
7.			PASS
8.	Upload Image Publish		PASS
9.	Title Description		PASS

Testcase 05 – Login Page

Test Case Name: Login page test
Test Case ID: VC-LP-001
Priority: High
Test Scenario: To ensure that the Login page functions correctly
Preconditions: • User must be registered
Test Steps: 1. Load the Login page. 2. Verify that the page displays the correct picture, a description of the platform, and a link to the registration page. 3. Verify that the login form has fields for username and password, and a "Login" button. 4. Leave the username and password fields blank and click the "Login" button. 5. Verify that error messages appear below the username and password fields indicating that they are required. 6. Enter an invalid username (e.g., a string of spaces) and a valid password and click the "Login" button.

7. Verify that an error message appears below the username field indicating that it is invalid.
8. Enter a valid username and an invalid password (e.g., too short) and click the "Login" button.
9. Verify that an error message appears below the password field indicating that it is invalid.
10. Enter a valid username and password and click the "Login" button.
11. Verify that the login function is called, and the user is navigated to the home page.

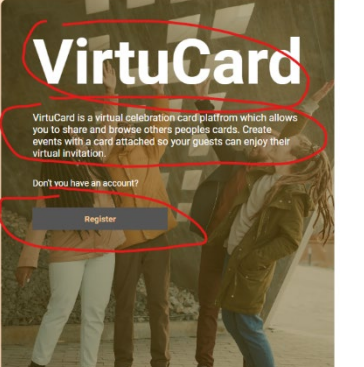|	
Expected Result: All test steps should pass, and the Login page should successfully direct the user to the main home page	
User Story: As a registered user of VirtuCard, I want to be able to log in to my account, So that I can access my saved cards and create new ones.	

Results:

Test Step:	Test:	PASS OR FAIL
1.		PASS
2.		PASS
3.		PASS

4.	**Login** username _____ password _____ Login	PASS
5.	**Login** username _____ ==Username is required== password _____ ==Password is required== Login	PASS
6.	**Login** Username is required .... Password is required Login	PASS
7.	**rd** hich allows Create enjoy their **Login** _____ .... _____ User Not Found Login	FAIL **DEFECT – D-05 Raised**
7. DEFECT FIX	**Login** _____ ==Username can only contain letters, numbers, and symbols== .... _____ Login	PASS **DEFECT – D-05 FIXED**

8.		PASS
9.	Password must be at least 6 characters long	PASS
10.		PASS
11.		PASS

Testcase 06 – Register Page

Test Case Name: Register page test
Test Case ID: VC-RP-001
Priority: High
Test Scenario: To ensure that the Register page functions correctly
Preconditions: • User must be logged out
Test Steps: 1. Open the VirtuCard registration page   2. Verify that the page displays the correct picture, a description of the platform, and a link to the login page.   3. Verify that the page displays a form with input fields for username, email, and password, and a submit button.   4. Leave the username, email and password fields blank and click the register button

5. Verify that error messages appear below the username, email and password fields indicating that they are required.
6. Enter an invalid username (e.g., a string of spaces), an invalid email (e.g., test.com), and an invalid password (e.g., 1234) and click the "Login" button.
7. Verify that correct error messages appear below the input fields.
8. Enter a valid username, email, and password in the input fields.
9. Click the submit button.
10. Verify that the user is redirected to the login page.
11. Enter the registered username and password in the login form.
12. Click the login button.
13. Verify that the user is redirected to the VirtuCard dashboard.
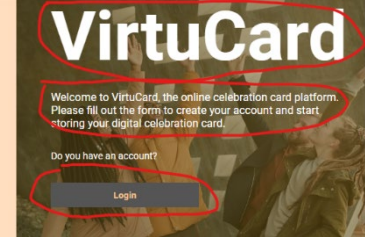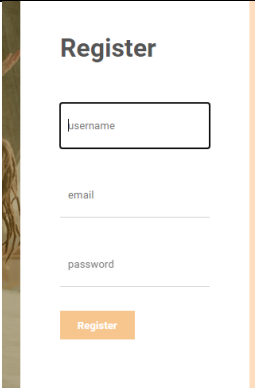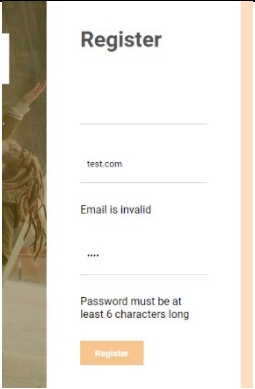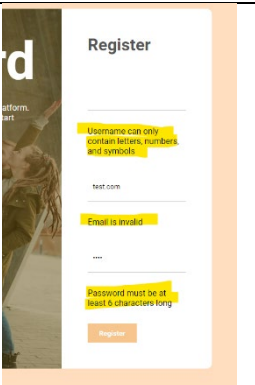
Expected Result:
All test steps should pass, and the Register page should successfully register the user to the database and direct the user to the login page
User Story:
As a new user, I want to be able to register for an account on VirtuCard so that I can start storing my digital celebration cards.

Results:

Test Step:	Test:	PASS OR FAIL
1.		PASS
2.		PASS
3.		PASS

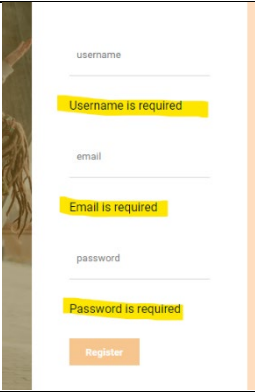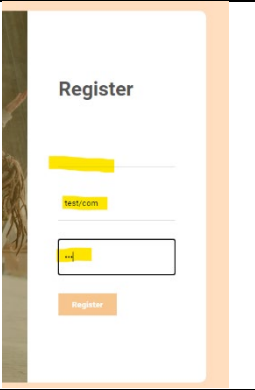|			
4.	Register username email password Register		PASS
5.	username Username is required email Email is required password Password is required Register		PASS
6.	Register test/com ··· Register		PASS
7.	Register test.com Email is invalid ···· Password must be at least 6 characters long Register		FAIL **DEFECT – D-06 Raised**
7. DEFECT FIX	Register Username can only contain letters, numbers, and symbols test.com Email is invalid ···· Password must be at least 6 characters long Register		PASS **DEFECT – D-06 FIXED**

8. TestUser TestUser@gmail.com 123456		PASS
10.		PASS
11. TestUser 123456		PASS
12.		PASS
13.		PASS

Testcase 07 – Profile Page

Test Case Name: Profile page test
Test Case ID: VC-PP-001
Priority: High
Test Scenario: To ensure that the Profile page functions correctly

Preconditions:
    • The user is logged in
    • The user is on the profile page

Test Steps:
  1. Verify that the profile page displays the user's profile picture, username, and email.
  2. Verify that the "Edit Profile" button opens a form where the user can update their profile information.
  3. Enter a valid username and email but don't upload a profile picture. Click update.
  4. Verify the message has popped up "User updated successfully"
  5. Close the window and verify that the user information has changed to the one entered
  6. Open the edit profile window again and enter the current user username and email
  7. Verify the message "User already exists"
  8. Enter an invalid username and invalid email. Click update profile

9. Verify the error message for the email and username is displayed under each input.
10. Verify that the "Delete Profile" button prompts the user to confirm the deletion and deletes the user's profile if confirmed.
11. Verify that the user is logged out of the website after their profile is deleted.
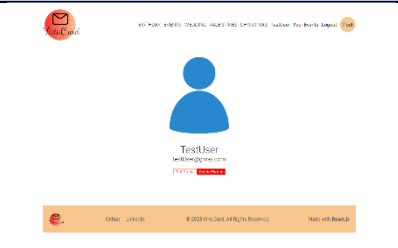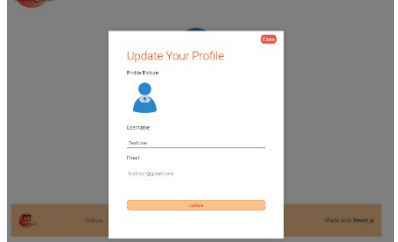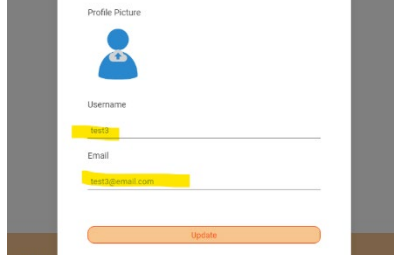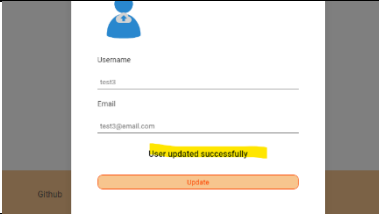12. Verify that an error message is displayed if the user is not logged in.
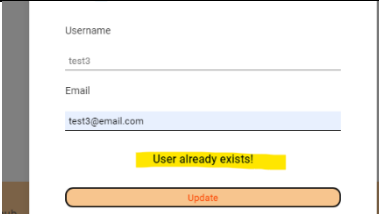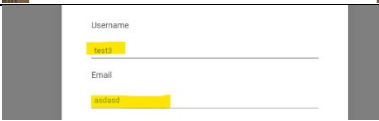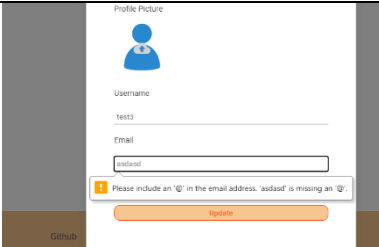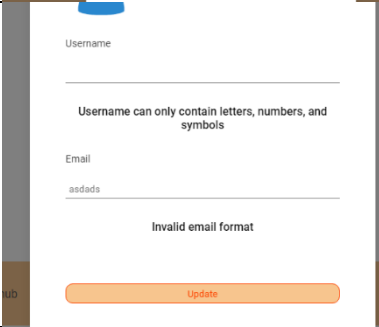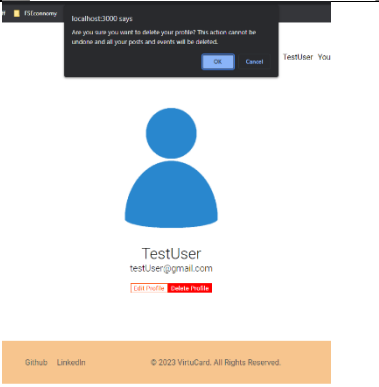
Expected Result:
- When the user navigates to the profile page, their profile picture, username, and email should be displayed.
- When the user clicks the "Edit Profile" button, a form should open where they can update their username, email, and profile picture. After submitting the form, the user's profile information should be updated, and the form should close.
- When the user clicks the "Delete Profile" button, they should be prompted to confirm the deletion. If they confirm, their profile should be deleted, and they should be logged out of the website.
- After the user's profile is deleted, they should be redirected to the home page and should not be able to access any pages that require authentication.
- If the user is not logged in, an error message should be displayed prompting them to log in to view their profile.

User Story:
As a registered user, I want to be able to view and manage my profile information on the website. When I navigate to the profile page, I should see my profile picture, username, and email displayed. I should also be able to edit my profile information by clicking the "Edit Profile" button, which opens a form where I can update my username, email, and profile picture. If I click the "Delete Profile" button, I should be prompted to confirm the deletion, and if I confirm, my profile should be deleted, and I should be logged out of the website. If I am not logged in, I should see an error message prompting me to log in to view my profile.

Results:
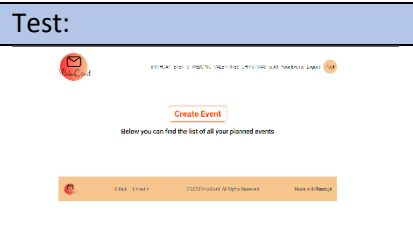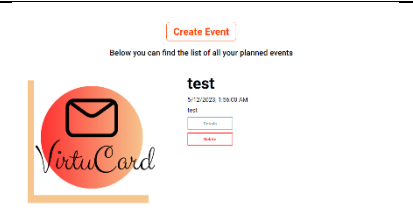
Test Step:	Test:	PASS OR FAIL
1.		PASS
2.		PASS
3.		PASS
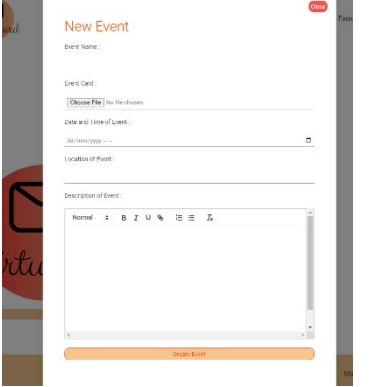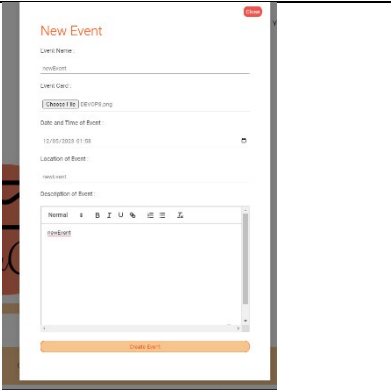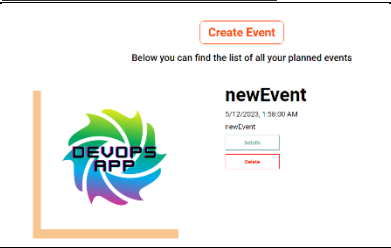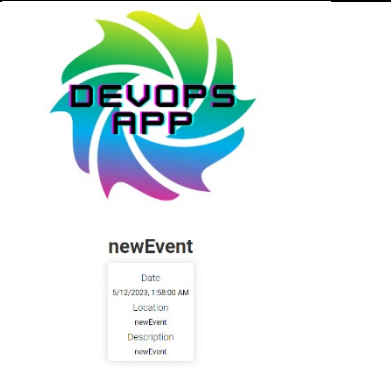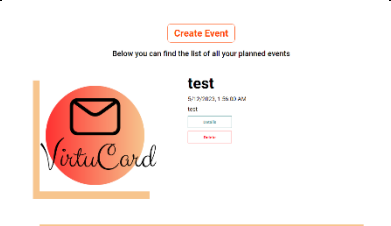
4.		PASS
5.		PASS
6.		PASS
7.		PASS
8.		FAIL DEFECT – D-07 RAISED
8. DEFECT FIX		PASS DEFECT – D-07 FIXED
9.		PASS
10.		PASS
11.		PASS

Test Case Name: Events page test
Test Case ID: VC-EP-001
Priority: High
Test Scenario: To ensure that the Events page functions correctly
Preconditions: - The user is logged in   - User must have at least one event created   - Use must have acess to two accounts
Test Steps: 1. Open the Events page. 2. Verify that the page displays a "Create Event" button and a list of existing events. 3. Click the "Create Event" button and verify that a form to create a new event is displayed. 4. Fill out the form with valid data and submit it. 5. Verify that the new event is added to the list of events. 6. Click the "Details" button for an existing event and verify that the event details page is displayed. 7. Click the "Delete" button for an existing event and verify that a confirmation dialog is displayed. 8. Click "OK" on the confirmation dialog and verify that the event is removed from the list of events. 9. Log out and verify that an error message is displayed instead of the list of events. 10. Log in as a different user and verify that only that user's events are displayed.
Expected Result: - The Events page should display a "Create Event" button and a list of existing events.   - Clicking the "Create Event" button should display a form to create a new event.   - Submitting the form with valid data should add the new event to the list of events.   - Clicking the "Details" button for an existing event should navigate to the event details page.   - Clicking the "Delete" button for an existing event should display a confirmation dialog.   - Clicking "OK" on the confirmation dialog should remove the event from the list of events.   - Logging out should display an error message instead of the list of events.   - Logging in as a different user should display only that user's events.
User Story: As a registered user, I want to be able to view a list of all my planned events, create new events, and delete existing events.

Results:

Test Step:	Test:	PASS OR FAIL
1.		PASS
2.		PASS
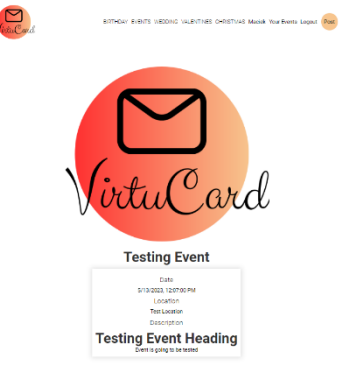
3.		PASS
4.		PASS
5.		PASS
6.		PASS
7.		PASS
8.		PASS
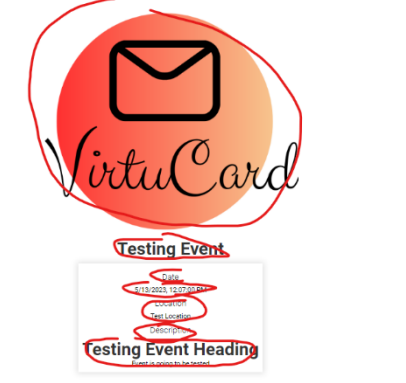
9.		PASS
10.		PASS

Testcase 09 – Event Page

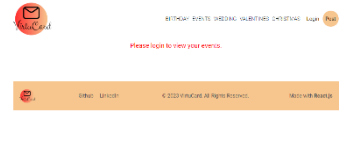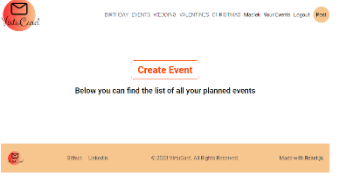| Test Case Name: Event page test |
|---|
| Test Case ID: VC-EtP-001 |
| Priority: High |
| Test Scenario:
 To ensure that the Event page functions correctly |
| Preconditions:
 • The user is logged in
 • The user has created an event |
| Test Steps:
 1. Navigate to the event page.
 2. Verify that the event image, name, date, location, and description are displayed.
 3. Verify that the event image is displayed correctly.
 4. Verify that the event name is displayed correctly.
 5. Verify that the event date is displayed correctly.
 6. Verify that the event location is displayed correctly.
 7. Verify that the event description is displayed correctly. |
| Expected Result:
 • The event page should render correctly with all the event information displayed accurately. |
| User Story:
 As a user, I want to view the details of a specific event so that I can learn more about it and decide whether to attend it or not. When I navigate to the event page, I expect to see the event image, name, date, location, and description displayed accurately. |

Results:

Test Step:	Test:	PASS OR FAIL
1.		PASS

2.		PASS
3.		PASS
4.	Testing Event	PASS
5.	Date 5/13/2023, 12:07:00 PM	PASS
6.	Location Test Location	PASS
7.	Description **Testing Event Heading** Event is going to be tested	PASS

Evaluation

To assess the performance of the web page, several performance evaluations were conducted. The objective was to measure the page's loading speed and resource utilization. Primary tool to measure these performances was google lighthouse. The projects expectations were to score the minimum average of 80%. The core pages have been all been analysed when multiple data entries were present at the page. Here are the results being scored out of 100%:

Page	Performance	Accessibility	Best Practices	SEO
Home	74	95	100	90
Single	91	97	92	100
User Profile	99	93	100	100
Events	95	95	100	100
Event	80	97	92	100

AVERAGES:	87.8	95.4	96.8	98

Based on the results above it can be said that the test results were successful. It has surpassed the initial expectations and scored relatively high in all categories.

Conclusions

In this report, we have reviewed the development of the web application with the primary goal of raising awareness about the environmental impact of traditional greeting cards and provide individuals with an accessible and convenient alternative. As we conclude our development, it is essential to reflect on the progress made and the impact this project has on addressing these objectives.

Through the course of this project, several crucial features have been implemented, including a safe storage place for celebration cards, the virtual exchange of celebration cards and virtual invitations. These enhancements have significantly improved the overall user experience and efficient of the web application.

While we have made significant progress in achieving the projects objectives, it is important to acknowledge the limitations that were encountered. Two of the challenges were implementing a scalable cloud-based database for the images and creating an online editor for the celebration cards. However, by implementing alternative approaches the project mitigated these obstacles and delivered a functional web application that satisfies the initial expectations.

As this project concludes, it is clear that the web application has immense potential for future expansion. The foundation that has been built for future development and innovations in areas such as virtual cards. The knowledge gained from this project will serve as a steppingstone for future development in the field.

In summary, this web application has not only successfully raised awareness about the environmental impact of traditional greeting cards but also provided individuals with a user-friendly platform that encourages the adoption of sustainable practices. By reducing paper waste and promoting virtual exchanges, we have taken a significant step towards a more environmentally conscious future.

Further Development or Research

With additional time and resources, there are several promising ideas that this project could take for further development and research. Firstly, Virtucard could expand to integrate with popular social media platforms, this would enhance its reach and user engagement. Users could share their virtual celebration cards and invitations on platforms such as Facebook, Instagram or Twitter further discussing the topic of sustainability and encouraging more individuals to peruse digital alternatives. Another idea could be providing an environmental impact assessments of the web application digital approach providing statistics on the amount of paper waste save, carbon emissions reduce, and other environmental benefits achieved through the use of the web application. With the incorporation of elements such as virtual badges, achievements or even a point system, it could encourage users to create and send virtual cards regularly providing a sense of accomplishment and engagement.

By exploring this direction for further development and research, this project has the potential to create an even more robust and impactful web application, solidifying its position as a sustainable alternative to traditional celebration cards and further contributing to a greener future.

References

[1] Fuller, K. (2021) 'The Environmental Impact of Greeting Cards—And What to Send Instead', *Brightly*, 4 November. Available at: https://brightly.eco/blog/environmental-impact-of-greeting-cards (Accessed: 14 May 2023).

[2] Pravos, N. and Mazieres, D. (2023) *Bcrypt*, *Wikipedia*. Available at: https://en.wikipedia.org/wiki/Bcrypt (Accessed: 14 May 2023).

[3] Node.js web application framework (no date) Express. Available at: https://expressjs.com/ (Accessed: 14 May 2023).

National College of Ireland

Project Proposal

Virtucard Web Application

12th May 2023

Computing Project

Software Development

Academic Year 2022/2023

Maciej Sobiecki

19457332

X19457332@student.ncirl.ie

Contents

Objectives

In this project, I set out to create an application for managing celebration and invitation cards. This application will provide a service in which people will have a space to store their most memorable celebration, greetings, and invitation cards. User will be able to register and login to upload photos of their cards in order to keep them digitally protected and safeguarded. Users will be equipped to keep their account information up to date and also to delete it whenever they would like. Other users will be able to like and comment under posts. This application will also provide an easy way to invite a large group of people to your event by being able to set up an event with an invitation image, date, location, and a description. This will allow users keep up to date regarding upcoming events.

Background

I choose to undertake this project for many different reasons. Firstly, I have personally always had a hard time storing cards that I have received. They would always either end up stored in a random, already cluttered cupboard or be thrown out in the rubbish. Storing them digitally in one space will allow me to come back to these cards and share my memories with others. The second reason is the fact that most of my family still live in Poland, I have always had a hard time sending birthday cards overseas through the post. With Virtucard, I would easily be able to send a birthday card through a post. Finally, most important of all I would like to contribute to promote the reduction in the use of paper. There are countless events that occur during one person's life, now if you are to multiple that number by billions, you can see how much paper is wasted on everyday greetings cards. With VirtuCard, I am hoping to save a great amount of paper usage in the world which has previously been spent on the production of celebration and invitation cards.

To meet the objectives set out in Section 1.0, I will conduct thorough research on different competitors I can find in the current market to ensure all the features mentioned in the project pitch and objectives set out in Section 1.0 will stand out. I will create a secure login registration system that uses web tokens and cookies authentication system for the user to securely create an account, and create, edit, and delete posts. To allow the user to like and comment under a post I will create an easy to use like and comment feature with which a user will be able to comment under a specific post and like/unlike a post. To ensure that the celebration and invitation cards are stored safely, I will include a cookies authentication system so that the user only have access to manage their own cards. To allow the user to efficiently manage their events, I will create an efficient event organiser feature which will allow the user to add an image and details to an event. Users will be able to share the link to the invitees, therefore they can access the information for the event.

State of the Art

Currently, there is a lack of greetings card management applications in the market. There is one that has a fairly similar concept to my project which is Paper Less Post. Some similar applications I came across have an option to print and send cards to users for a physical copy.

VirtuCard would stand out and be innovative by allowing the user to upload a picture of already received cards in person to safe keep. There is currently no application that contains this feature. Paper Less Post,

which is the most similar application that exists, allows to only store the cards that you have picked out or generated from their website.

Technical Approach

My approach to the development off the application will be carefully considered after a lot of research into the viability of the application. After thorough research of the competitors and the market, I will analyse and identify the requirements for my application.

In terms of structure, together with my project supervisor, I will conduct monthly or weekly Scrum meetings to identify and plan out tasks, milestones, and durations of tasks and deadlines using Monday.com. Monday.com is a cloud-based platform that allows users to create their own applications and project management software.

I will use Monday.com to keep track of all the implementation to the application and to also provide myself and my project supervisor with a detailed look at my ongoing progression. In Monday.com, I will also be able to provide information on the current task that I am doing at the current time. I will be able to show that either I am finished the task or am having difficulties or that I haven't started it yet.

I will also consider using ghantt charts to provide more information regarding my time management.

Technical Details

The JavaScript language will play a crucial role in the application's development. Its primary purpose will be to serve as the groundwork for the application, handling all the backend functionalities. By using JavaScript, I will be able to establish communication between the frontend and the database. This means that any requests made by the user on the frontend, will be transmitted to the database through JavaScript.

In terms of security, the backend will implement various features to ensure the application's protection. One of these features is Cross-Origin Resource Sharing (CORS), which controls access to resources from different domains. It helps prevent unauthorized access to sensitive data by enforcing restrictions on which domains can interact with the backend.

To enhance security even further, the backend will also utilize JSON Web Tokens (JWT) and cookies. JSON Web Tokens allow for secure authentication and authorization by providing digitally signed tokens that can be validated and trusted. Cookies, on the other hand, are small pieces of data stored on the user's browser to maintain session information and enhance user experience.

Moving on to the frontend, the application will be developed using React.js, an open-source JavaScript framework and library. React.js enables me to organize and structure my code in a more manageable way. It facilitates the development process by allowing me to break down the application into reusable components. These components can be utilized across different pages, enhancing code reusability, and reducing redundancy.

For styling the website, I will employ SASS (Syntactically Awesome Style Sheets), which is an advanced variant of CSS (Cascading Style Sheets). While CSS utilizes a plain text syntax, SASS introduces a more structured syntax with additional features. This allows for more efficient and maintainable styling code, enabling you to customize the visual appearance of the application more easily.

As for the primary database, I will be using MySQL. MySQL is a widely used relational database management system that offers simplicity in development and easy deployment to hosting websites. Platforms like Heroku provide a free tier for database deployment, making it a suitable choice for hosting

my application's database. With MySQL, I will have the ability to view all the entries in the database directly from my development PC, providing convenient access for debugging and development purposes.

Special Resources Required

As the project progresses, several resources may be required to enhance the functionality and user experience of the web application. These resources include:

Product Design: For future development, a product designer can be hired to design cards that the user may choose from. This resource would be utilized when a new functionality can be implemented for users to use pre-made cards for events. This will enhance the customization option available to users, making the application more versatile and user-friendly.

UX Designer. Hiring a UX designer can greatly improve the visual appearance and artistic aspect of the website. A UX designer would focus on creating a pleasing user experience. Their expertise in design and user behaviour, can contribute to a more engaging and enjoyable website for users.

Specialized Database Plans: To accommodate high upload and user traffic on the website, it may be necessary to acquire special price plans for the database. These plans may offer increased storage capacity, faster processing speeds and enhanced scalability. Opting for such plans ensures a smooth and efficient performance of the application, especially during periods of high traffic.

By utilizing these special resources, Virtucard can benefit from better design, more functionality and ability to handle high user traffic. These additions would contribute to a more professional, visually appealing, and user-friendly experience.

To keep myself and my supervisor up to date, we have agreed upon creating a Monday.com page. As previously mentioned, Monday.com allows me to track each step of my project, including development time and report writing. The following information that I have provided below will give a general idea of how the project will be planned out.
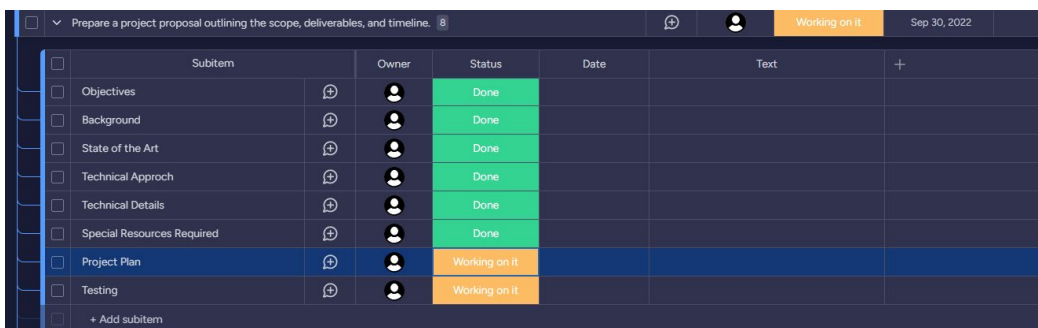
Each month, I have planned to put in a great amount of work and effort towards the overall project. With the structure of Monday.com, it allows me to put in a Title for the work I want to do. This work is going to include smaller sub items which I will have to tick off along the way, a visualized status bar will be provided towards the overall progress of each month, and a date on each titled task will provide a deadline for which I would have to strive for.

1.SEP – 2.OCT: Project Initiation and Research
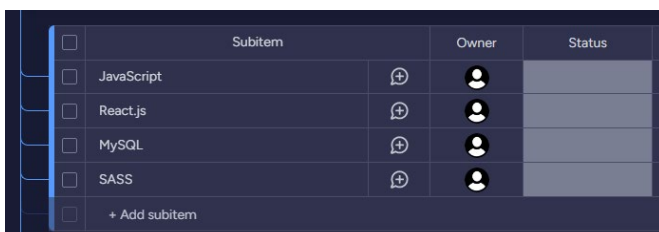
During this month, I have set out for myself to gain a clearer understanding of the project, requirements, and objectives. As it can be seen in the image above, the status of this goal is complete. I had to provide an initial project proposal outlining the scope, deliverables, and timeline. The below image will show the items that I am working on, it can be seen as "working on it" status.

For the remainder of this month, I plan to create a video presentation which will provide a more general understanding of my project and summarizing the project proposal. For the entirety of the next month, I will be conducting intensive research on the technologies I will want to be using, such as Java Script, React.js, MySQL and SASS.

Month 3.NOV - 5.JAN: Midpoint Presentation and Prototype Development

Taking into the account what I have thought myself regarding the technology I will be using, I will plan to create and design the prototype and its core features. I have split the development into subitems consisting of the home page, login page, registration page, post page and the event page.

For the midpoint presentation, a video presentation is going to be made to showcase the prototype and its functionalities.

For the start of December, I have given myself time to write up the midpoint report in order to showcase the progress and the direction of the project.

Taking into consideration that these months consist of many holydays, I have planned my project in a way that it would not collide with any of the holiday events and give myself enough time to work on the deliverables of this project.

Month 6.FEB: Development

During the sixth month of this project, I have given myself more than enough time to work on the project, keeping in mind that starting is the most difficult part of the development. I have written out the names of the pages I want to add and what features I want to incorporate. At the end of each month, from now on, I

want to test any features that I have incorporated and document them on my testing report.

Month 7.MAR-8.APR: Card Functionality and Event Page Development

Month 7.MAR–8.APR: Card Functionality and Event Page Devel…	5 Projects / 35 Subitems			
Project		**Person**	**Status**	**Date**
∨ Design and implement the functionality for users to post cards with options to edit, delete, and add. 8	⊕	👤		Mar 10

Subitem		Owner	Status	Date	Text	+
Write Page	⊕	👤				
Single Page	⊕	👤				
Menu Component	⊕	👤				
Update Home Page	⊕	👤				
Add post database	⊕	👤				
Update Backend API	⊕	👤				
Set up upload image backend	⊕	👤				
Test new pages and components	⊕	👤				
+ Add subitem						

Project	Person	Status	Date
＞ Develop a profile page, and update page to allow the user to update user profile and delete user profile 4	⊕ 👤		Mar 24
＞ Develop the event page, allowing users to add and delete events. 9	⊕ 👤		Mar 31
＞ Incorporate features like likes and comments for posts. 5	⊕ 👤		Apr 15
＞ Testing and fix any issues that arise during development. 9	⊕ 👤		Apr 29
+ Add project			

During month 7 and 8 I want to do the majority of the development during this time. This will include all the core features I would like to add, I have given myself more the enough time to develop all the core functionality. Each page and component are already named in this project plan.

Month 9.MAY: Testing, Finalization, and Reporting

∨ Month 9.MAY: Testing, Finalization, and Reporting				
Project		**Person**	**Status**	**Date**
∨ Conduct thorough testing throughout the development process, identifying and fixing any bugs or issues. 11	⊕	👤		May 12

Subitem		Owner	Status	Date	Text	+
Home Page Testing	⊕	👤				
Navbar testing	⊕	👤				
Footer testing	⊕	👤				
Single Page Testing	⊕	👤				
Write Page Testing	⊕	👤				
Profile page testing	⊕	👤				
Event page testing	⊕	👤				
Events page testing	⊕	👤				
Login page testing	⊕	👤				
Register page testing	⊕	👤				
Backend testing	⊕	👤				
+ Add subitem						

Project	Person	Status	Date
Design a poster for the website.	⊕ 👤		May 14
＞ Write the final project report. 12	⊕ 👤		May 14
＞ Create final video presentation 2	⊕ 👤		May 14
+ Add project			

During the final month, I gave myself time from Apr 29th till May 14th to conduct final testing and completion of the write up for all of the deliverables for the project.

Testing

For future testing, a well-defined strategy will be implemented to ensure the application's stability and functionality. Integration tests will be conducted every time there is a major feature release. The focus will remain on testing the features developed at the end of each month to ensure their correct implementation.

The project plan will serve as a reference to ensure that all the planned features are developed and integrated properly. When conducting the tests, a detailed report will be created, consisting of functional testing conducted internally. The report will include essential information such as a test ID, test priority, test steps, and a user story. Finally, it will also document the results of each test.

In the case of a test failure, this will be promptly reported, and a defect will be raised with a unique ID. Any issue or problem will be thoroughly analysed, and a solution will be efficiently developed. Once the defect has been addressed, the fixed solution will be uploaded to GitHub, with the defect ID mentioned in the commit message. The test will be re-executed to ensure that the defect has been successfully resolved.

Throughout each month, the testing process will heavily rely on the reports and the written-out test cases. This approach will ensure that the development deliverables for each month do not introduce any new issues or bread that have previously developed code. By retesting the existing functionality, any regressions or conflicts arising from new features will be detected and resolved promptly.

Supervision & Reflection Template	

Student Name	Maciej Sobiecki
Student Number	19457323
Course	BSH Computing (Software Development)
Supervisor	William Clifford

Month: October

What?

This month I started working on my project for the first time. In the first week I have put together a basic project pitch video which explains the basic fundamentals of the project. The project got accepted with amendments. Once the results were given and I was allocated a supervisor, I initiated to contact him about my result from the project pitch video. We have scheduled together a meeting on Friday 28th October. During the meeting I further explained the idea of my project to provide a better understanding. Together with William we have agreed on which project management website/tools to use. We also have agreed on what to do before our next meeting. I have been given a set of tasks to do over the next two weeks of which will be included in the next month's journal. We have agreed to adapt the sprint agile methodology to the project, and we are going to host monthly scrums. Today on the 29th of October I have finished writing the first stage of my project proposal and submitted my ethical form.

So What?

During this month with the tasks that I have done, I got a better understanding of what I want to do with the project and how I am going to do it. I am glad me and William communicate so early, I feel like we are going to make a great team. Finishing the project pitch and project proposal gave me more insight on what I am going to do with the project. Following this now I need to understand more about the technical details of the project including the languages and tools I am going to use. I also need to take a better look at all the competitors that are currently on the market and see how I can stand out from them.

Now What?

Over the next few weeks, I am going to conduct thorough research on a few languages and tools I am less experienced with that I believe I might use. This includes JavaScript, React.js, MySQL and SASS. I also need to create a competitor analysis to understand what features competitors are offering and what I believe they are missing.

Student Signature	*Maciej Sobiecki*

Supervision & Reflection Template

Student Name	Maciej Sobiecki
Student Number	19457323
Course	BSH Computing (Software Development)
Supervisor	William Clifford

Month: November

What?

During this month I have set out to use what I have learned in the previous month and start creating my prototype for the midpoint presentation. I feel like over the course of last month I have gained a substantial amount of research on the technologies I am going to use. I have spent a large amount of my time practicing using the technologies I learnt about so I can get more familiar with them. I have created a mock web application that allows me to drag and drop an image and upload it to my server. I was also able to create a practice web app that consisted of basic CRUD functionalities for a user registration system and login. Last month I have had a few delays and was quite behind on my project plan. So far most of my work done in this month has been on schedule according to the Project Plan.

So What?

Using what I have learn I was able to create a basic prototype that included the home page, login page, registration page. Unfortunately, I have been struggling trying to develop the core functionalities of the website being Events page and Post page where a user can upload an image of a celebration card and details to it. Overall, I was quite happy with the work I did on the login and registration page to get that to work. I only have been using react for a month, so I thought that was a good start.

Now What?

I will continue trying to work on the prototype, however if I end up with not having enough time, I will have to consider focusing on the midpoint presentation to achieve more marks. Since the release date for the midpoint presentation is coming up so soon, I will have to start working on the and start creating my video for the showcase of my prototype.

Student Signature	*Maciej Sobiecki*

Supervision & Reflection Template

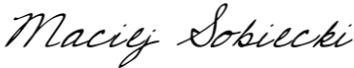| Student Name	Maciej Sobiecki
Student Number	19457323
Course	BSH Computing (Software Development)
Supervisor	William Clifford

Month: December

What?

This month I have finished up the prototype to the best of my abilities so far. I am quite happy with it but unfortunately, I was not able to add in the core functionalities I wanted. I started on the midpoint presentation early this month and it was not my best piece of work; I was not able to properly identify my functional requirements well enough to be able to write much about. I also misunderstood the whole purpose of the midpoint presentation; I thought that I can only write information based on the prototype I have submitted and that meant I was missing a lot of the functional and non-functional requirements I would have wanted to put in. I had a conversation with the supervisor in which we both agreed it was not the best work.

So What?

Based on what I did with the prototype, I feel like I am quite behind on schedule meaning I will have to put a lot more effort into conducting more research over the course of the semester break and figure out where it went wrong. The challenge from now on is going to be getting back at it with motivation after the poor performance during my mid-point presentation.

Now What?

I will have to conduct more research about the technologies I am using in order to complete and start working on my web application. I should strategies more and identify the functional requirements and non-functional requirements.

Student Signature	*Maciej Sobiecki*

January

Supervision & Reflection Template

Student Name	Maciej Sobiecki
Student Number	19457323
Course	BSH Computing (Software Development)
Supervisor	William Clifford

Month: January

What?
During this month I took my time and developed a lot of practice web application to expand my understanding of the programming languages and frameworks I am using. I have created 3 web application and a online portfolio so I could get better understanding of the react framework. Knowing how to properly code using react helped me understand what and how the functionalities should be implemented.

So What?
The work that I have put into research will definitely help me speed up my development time and will allow me to get back on schedule according to my project plan on Monday.com. I found great success making tutorial web sites getting a much better understanding of the code I was using. There are still going to be challenges as this made me be quite behind on schedule, but I believe that over the course of next month I will be able to get back on track.

Now What?
I will start developing the code intended towards my final project submission. I will start by creating the home page with the navbar and footer. I will start developing an improved Login and Registration Page. I will create a MySQL database and make a backend API so the frontend of the app can communicate with the backend. I will conclude the next month with testing.

Student Signature	
	Maciej Sobiecki

February

Supervision & Reflection Template

Student Name	Maciej Sobiecki
Student Number	19457323
Course	BSH Computing (Software Development)
Supervisor	William Clifford

Month: February

What?
During this month I was able to design my own logo. I was able to develop the basic frontend pages being the Home Page which includes the Navbar and the Footer Components, the login and register page. Logout function has also been included. I created a basic MySQL database that consists of the user's table. I was able to create an API that communicates the front-end application to the database. I have included security measures from the beginning being JWT (Jason web token) and the use of cookies. When a user logs in to the website his details are safely stored into the local storage of the browser and deleted when signed out. To the registration system I have add an encryption to the password so that when the user registers, his password is stored hashed with no one having access to it. Once I was happy with the functionality of all these pages, I have started to style the page using SASS. Once my overall development work was done during this month, I started creating functional testing test cases. So, I can test the code. Which was a great success.

So What?
The progress I have made this month was very good, I have done what I intended to do according to the project plan. I am a quarter way done with the development of my website and I felt like I was understanding the code very well. I found great success with designing the backend API of my web application, it was very motivational when the entries from the frontend where successful being stored in the database and sent back to the frontend when required. The testing of the development was also a great success. There are still challenges ahead being the write post page and single post page. Its going to be a lot more in-depth development work with the backend and frontend for these two pages.

Now What?
Over the next month I intend to develop the core functionality of my page where the user is able to create a post being the post page. This page is going to have to include CRUD features while also making sure its secure with authentication. I will also be intending to develop a profile page where the user is able to see his profile details. In this page the user will also be able to edit his username and email. The user will also be able to delete his profile. I will also be intending to develop the events page where the user can store his planned events with also the basic crude functionality.

Student Signature	*Maciej Sobiecki*

March

Supervision & Reflection Template

Student Name	Maciej Sobiecki
Student Number	19457323
Course	BSH Computing (Software Development)
Supervisor	William Clifford

Month: March

What?

I have developed a write page and single page; in the write page the user can upload an image which is stored in the local code. The user they can add information such as the title and description. For the description I have implemented a rich text editor which allows the user to customise the description text with different heading and text decoration such as Bold or Italic. The user can also choose which category the post is under, so it helps with other users sorting which content they want to see. The single page is opened when the user decides to click on the read more button to read about the post. If the user is the owner of the post two buttons are available to him the edit and delete. When the users press on the edit button the user is brough back to the write page with it being prepopulated with the posts contents that the user is allowed to edit. The delete button is also available for the owner for the owner user to delete his post. I have also developed a profile page where the user can edit his details and upload his profile picture.

So What?

I think I made great progress on the project by adding the first core functionality to my web app. Even though I didn't get done what I set out in the project plan I still think it was more then enough for the month. I have made great success with developing the backend for these features making sure it is very secure. The challenge that still remains is to properly test the written code for this month to make sure everything is working as expected.

Now What?

I will continue to work on the event pages and incorporate the like & comment feature for the posts. Following that I will have to style the pages and test everything I have done throughout the development stage being closer to finishing this project

Student Signature	
	Maciej Sobiecki

April

Supervision & Reflection Template

Student Name	Maciej Sobiecki
Student Number	19457323
Course	BSH Computing (Software Development)
Supervisor	William Clifford

Month: April

What?

This month I continued working on the event page, I have made so that each user has his own events page. I added an option for the user to add and delete the events that he has made which are secured by authentication. I haven added additional features such as like/dislike under the posts on the single post page. Additionally, I have added on the ability for the user to post a comment under the post. I have tested all features that have been incorporated and fixed any defects that have risen. However, I have just realised that running MySQL database on my local machine will not work for other people grading the assignment. So, I will have to quickly think of a fix for this.

So What?

Good progress has been made regarding the overall project. Great success has been found in learning as I develop the web application. Using MySQL for the moment is okay but when uploading the source code to GitHub for the grading committee to use, they won't be able to run the application successful as there won't be a connection to the database established.

Now What?

I am going to have to think of a new way to create a database so who ever uses the code can use the full functionality of the website.

Student Signature	
	Maciej Sobiecki