BSCS-S22-006

03-134182-019 Gul e Sahar

03-134182-114 Shaharyar Ahmad

FlowArt

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: Umar Hameed

Department of Computer Sciences

Bahria University, Lahore Campus

January 2023

C e r t i f i c a t e

We accept the work contained in the report titled

"**FlowArt**"

written by

Gul e Sahar

Shaharyar Ahmad

as a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: Umar Hameed

_____ (Signature)

January 9, 2023

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134182-019	Gul e Sahar	
03-134182-114	Shaharyar Ahmad	

Date : January 9, 2023

Specially dedicated to
my beloved grandfather, mother, and father
(Gul e Sahar)
my beloved grandmother, mother, and father
(Shaharyar Ahmad)

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to my research supervisor, Umar Hameed for his invaluable advice, guidance, and his enormous patience throughout the development of the research.

In addition, we would also like to express my gratitude to our loving parent and friends who had helped and given me encouragement.

Gul e Sahar
Shaharyar Ahmad

FlowArt

ABSTRACT

Every software requires a series of important steps to complete its development. Before starting to work on a certain project, the first step is to determine how the flow of work will take place. To design the flow of work and to assign responsibilities to every person involved in the production of the project, we must design flowcharts, activity diagrams or use case diagrams. Usually, project managers use different tools to make these diagrams and if a huge number of people are trying to access a same tool, there is a chance that it may get slow. Also, some of the tools available on the internet are not very economical or are not user friendly.

It is necessary to develop an application that gives its users the capability to develop all these diagrams using a single platform without having to switch between different applications. Moreover, the application should be economical so that even the start-ups or students can take advantage of it. Our goal is to develop a web application that allows users to develop their designs using an interactive environment and allows them to save and edit their designs or diagrams. The development of the application is done using MEAN stack which includes MongoDB, Express, Angular and Node JS.

TABLE OF CONTENTS

DECLARATION **ii**
ACKNOWLEDGEMENTS **iv**
ABSTRACT **v**
TABLE OF CONTENTS **vi**
LIST OF TABLES **x**
LIST OF FIGURES **xi**

CHAPTERS

1 INTRODUCTION **1**
1.1 Background 1
1.2 Problem Statements 2
1.3 Aims and Objectives 2
1.4 Scope of Project 2

2 LITERATURE REVIEW **4**
2.1 Introduction 4
2.2 Purpose 5
2.3 Functional Requirements 5
2.4 Use Case Diagrams 6
2.4.1 System Use Case 6
2.4.2 Login Use Case 7
2.4.3 Signup Use Case 7
2.4.4 Create Design Use Case 8
2.4.5 Save Design Use Case 8
2.4.6 Delete Design Use Case 9

2.4.7 Edit Design Use Case 9
2.4.8 Create User Use Case 10
2.4.9 Delete User Use Case 10
2.5 Use Case Scenarios 11
2.5.1 System 11
2.5.2 Login 12
2.5.3 Signup 13
2.5.4 Create Design 14
2.5.5 Save Design 15
2.5.6 Delete Design 16
2.5.7 Edit Design 17
2.5.8 Create User 18
2.5.9 Delete User 19
2.6 Non-Functional Requirements 19
2.6.1 Security 20
2.6.2 Performance 20
2.6.3 Compatibility 20
2.6.4 Portability 20
2.6.5 Usability 21
2.6.6 Reusability 21

3 DESIGN AND METHODOLOGY 22
3.1 Methodology 22
3.2 Sequence Diagrams 25
3.2.1 Login Sequence 25
3.2.2 Signup Sequence 26
3.2.3 Create, Save, Delete, Edit Design Sequence 27
3.2.4 Create User Sequence 28
3.2.5 Delete User Sequence 29
3.3 Class Diagram 30
3.4 Class Diagram Functions 31

4 DATA AND EXPERIMENTS 32

4.1 Technologies Used 32

4.1.1 HTML 32

4.1.2 CSS 32

4.1.3 JavaScript 32

4.1.4 TypeScript 32

4.1.5 Angular 33

4.1.6 Angular Material and Bootstrap 33

4.1.7 NodeJS and Express 33

4.1.8 MongoDB and Mongoose 33

4.1.9 BPMN IO 34

4.2 System Requirements 34

4.3 Test Cases 34

4.3.1 Login Test Case 35

4.3.2 Signup Test Case 35

4.3.3 Create Design Test Case 36

4.3.4 Save Design Test Case 36

4.3.5 Delete Design Test Case 37

4.3.6 Edit Design Test Case 37

4.3.7 Create User Test Case 38

4.3.8 Delete User Test Case 38

5 USER MANUAL 39

5.1 Home Page 39

5.2 About Page 40

5.3 Contact Page 40

5.4 Login Page 41

5.5 Signup Page 41

5.6 User Dashboard 41

5.7 Drawing Board 42

5.8 Admin Dashboard 43

6 CONCLUSION AND RECOMMENDATIONS 44

REFERENCES 45

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.1	System Use Case	11
Table 2.2	Login Use Case	12
Table 2.3	Signup Use Case	13
Table 2.4	Create Use Case	14
Table 2.5	Save Design Use Case	15
Table 2.6	Delete design Use Case	16
Table 2.7	Edit Design use Case	17
Table 2.8	Create User Use Case	18
Table 2.9	Delete User Use Case	19
Table 3.1	Class Diagram Functions	31
Table 4.1	Login Test Case	35
Table 4.2	Signup Test case	35
Table 4.3	Create design Test Case	36
Table 4.4	Save Design Test Case	36
Table 4.5	Delete Design Test Cases	37
Table 4.6	Edit Design Test Case	37
Table 4.7	Create User Test Case	38
Table 4.8	Delete User Test Case	38

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2.1	System Use Case Diagram	6
Figure 2.2	Login Use Case Diagram	7
Figure 2.3	Signup Use Case Diagram	7
Figure 2.4	Create Design Use Case Diagram	8
Figure 2.5	Save Design Use Case Diagram	8
Figure 2.6	Delete Design Use Case Diagram	9
Figure 2.7	Edit Design Use Case Diagram	9
Figure 2.8	Create User Use Case Diagram	10
Figure 2.9	Delete User Case Diagram	10
Figure 3.1	Agile Scrum Methodology	23
Figure 3.2	Scrum board	24
Figure 3.3	Login Sequence Diagram	25
Figure 3.4	Signup sequence Diagram	26
Figure 3.5	Create, Save, Delete and Edit Sequence Diagram	27
Figure 3.6	Create User Sequence Diagram	28
Figure 3.7	Delete User Sequence diagram	29
Figure 3.8	Class Diagram	30
Figure 5.1	Home Page	39
Figure 5.2	About page	40
Figure 5.3	Contact Page	40
Figure 5.4	Login Page	41

Figure 5.5 Signup Page 41

Figure 5.6 User Dashboard 42

Figure 5.7 Drawing Board 42

Figure 5.8 Admin Dashboard 43

CHAPTER 1

INTRODUCTION

1.1 Background

The proposed application will help the users to design multiple diagrams on a single page platform that requires minimum bandwidth. This single page application provides ease of work as compared to those applications on which users must switch between different pages to get their work done. Students, freelancers, software houses and other professionals will find this feature suitable. This application will prove to be time saving and efficient. Being a password protected app, its users will not only have access to multiple diagrams on single page but will also feel secure doing so.

There are many applications that make it easy for people to design their diagrams. One type of such applications deals with technical actions like creating diagrams, graphs, and flowcharts. Although such applications are source of ease, they are not without their shortcomings. An example of this type of application is Lucid [1] or Smart Draw which are two of the biggest contenders out there. But the drawback of Smart Draw is that it does not offer multiple design templates and it is not a single page application as well [2]. Every time, the user makes a certain request, The application reloads a new page. As a result, this type of application requires a higher bandwidth.

The diagrams that can be designed using FlowArt are activity, flowcharts and use case diagrams. All these diagrams require multiple actions to complete. But making these diagrams is difficult when one must switch between different software

or applications. Moreover, FlowArt also provides the advantage of portability. The user will be able to access their saved designs from any location.

1.2 Problem Statements

The process of finding an application that caters to all our needs is hectic. People face many problems such as change in pages and attention when using applications that are multiple page oriented and require more bandwidth, making it a slow process. FlowArt, as apart from these, provides a faster work completion rate as it works on a lower bandwidth and provides an interactive drag and drop feature which makes it more convenient.

1.3 Aims and Objectives

The objectives that we are aiming to accomplish are shown as following:

- To allow users to make flowcharts, use case diagrams and activity diagrams using an interactive environment with drag and drop capability.
- To allow user to create accounts protected by a password and save their designs online.
- To allow users to delete, edit or download a design.

1.4 Scope of Project

We are planning to make a single page web application which will allow its users to make flowcharts, activity diagrams and use case diagrams. Project managers are required to use different applications to design each of these diagrams. Some of these applications may not be fast enough and some of them may not be user friendly. FlowArt will provide a user interactive designing environment to its users with additional benefits and functionalities. The application will provide a drawing board. The users can simply drag a required element from the list of elements and drop it on the drawing board. This feature is provided by Visme as well [3]. However, it does not

offer multiple diagram templates. The users of FlowArt will also be able to save, export or edit their designs as well. To ensure the privacy, the users can create a password protected account and keep their previously made diagrams safe. They can access their accounts from any location to view those saved designs. They will also be able to upload a design and make changes to it. The designs can be exported in the form of an SVG or a JSON file to the desktop.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The proposed idea of the final year project is based on a single page web application which will allow its users to design their flowcharts, activity diagrams and use case diagrams using the same platform. The idea is inspired from some of the existing web applications that are available for this purpose. The reason we have decided to make this a single page application is because in case of a multiple page application, if multiple users are trying to request the same page from the server, the application may get slow because it reloads a completely new page every time they click on a certain option. A single page application eliminates this problem because it does not reload a new page every time, it simply changes a certain portion of the page. This is the reason it is faster and more reliable as compared to a multiple page application.

The application also provides the option of exporting the designs. These designs can be sent to other people and can also be edited. This application will hopefully prove to be a better option rather than using different applications to design different diagrams.

2.2 Purpose

The purpose of this web application is to allow its users to design activity diagrams, use case diagrams and flowcharts with ease. People with various backgrounds, sometimes, find the need to use different applications to design their content but these may either be slow or not easy to use. Different from these, FlowArt will provide an immersive experience that will deal with the previously mentioned issues and will also provide further benefits. With a simple drag and drop feature, users can connect various elements on a single page. It will ensure password protection and users can thus, save their data and will also be able to edit and export it in JSON file or SVG.

2.3 Functional Requirements

The functional requirements of this project include inducing a user-friendly single page interface with an interactive designing environment equipped with a drag and drop feature. The beneficiaries will be:

- Software houses
- Students
- Freelancers

It will allow users to make flowcharts, use case diagrams and activity diagrams with ease. Users will be provided with the options to make free accounts, protected with passwords, that will allow them to save, delete, edit, or download their designs. The performance will be faster as compared to other applications due to the sole reason that it consumes less bandwidth. As the application allows users to make accounts to save their content, they can access it from anywhere at any time hence, securing its portability. It is compatible with every operating system with minimum system requirements at any available internet speed. It is reliable and secure due to its password protection and users can also share their designs with others as they desire.

2.4 Use Case Diagrams

Use case diagrams show how the different actors in the system interact with the functionalities. There are two actors in our system. These actors are user and admins.

2.4.1 System Use Case

The following is the use case diagram for the whole system. As mentioned above, there are two types of actors interacting with the system. The admin has the authorization of creating and deleting users. Users can sign up and login to perform their actions.

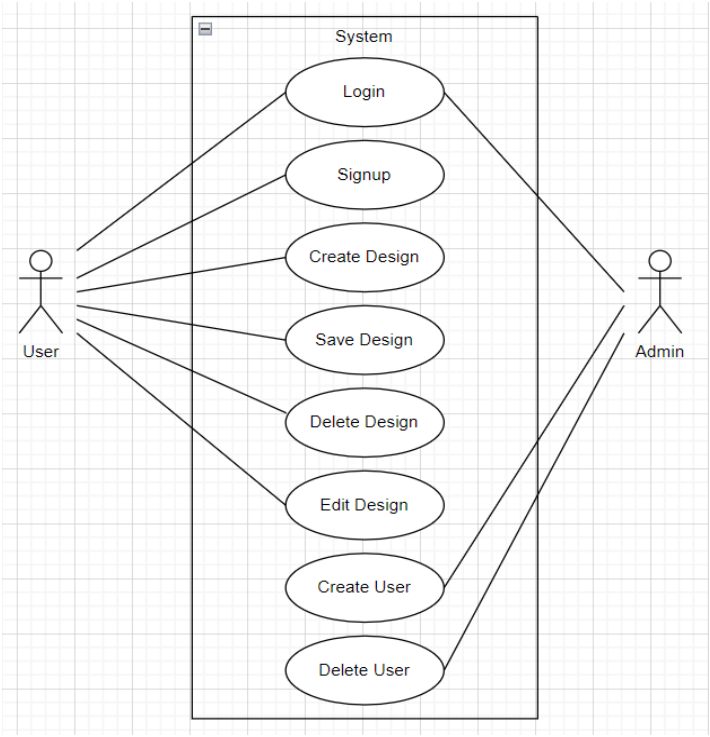

Figure 2.1 System Use Case Diagram

2.4.2 Login Use Case

To login, both users and the admin must enter their email address and password into the system. Once the user is logged in, a message is shown on the screen.

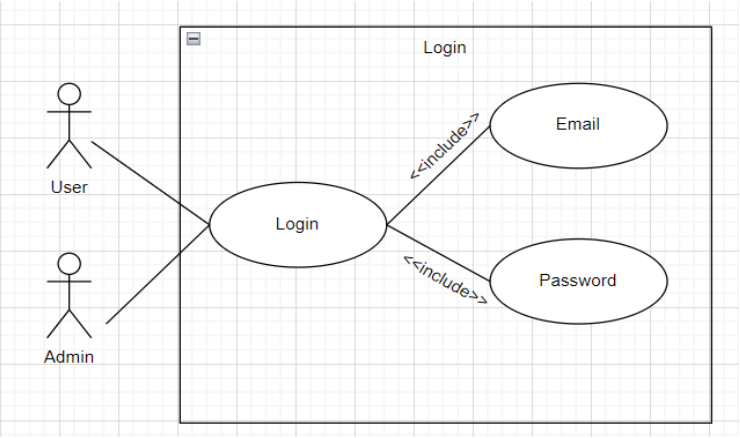

Figure 2.2 Login Use Case Diagram

2.4.3 Signup Use Case

To create an account or to signup, the user must enter his email address, username and create a secure password. Once the account is created, a message is shown on the screen.

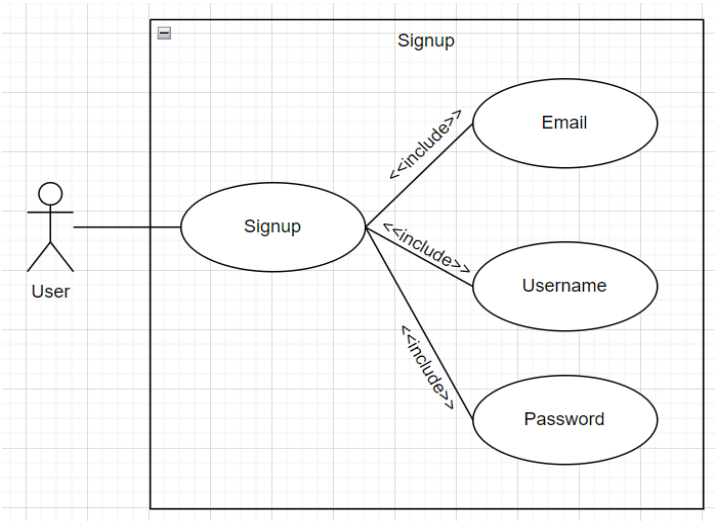

Figure 2.3 Signup Use Case Diagram

2.4.4 Create Design Use Case

Once the user logs in into his account, he can create a new design.

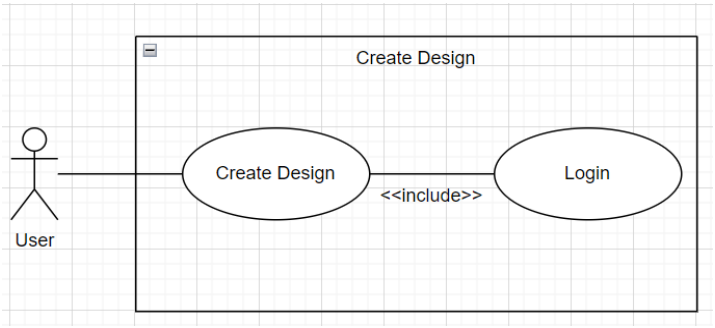

Figure 2.4 Create Design Use Case Diagram

2.4.5 Save Design Use Case

Once the user is done making a new design, he can save it. The design can be saved online, or the user can download it on his system in the form of JSON file or an SVG.

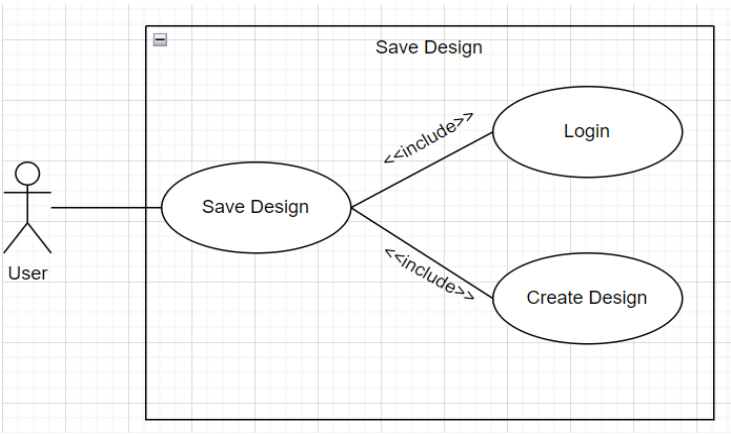

Figure 2.5 Save Design Use Case Diagram

2.4.6 Delete Design Use Case

When the user logs in into his account, he can see all his designed that were saved previously online. The user can delete any of these designs.

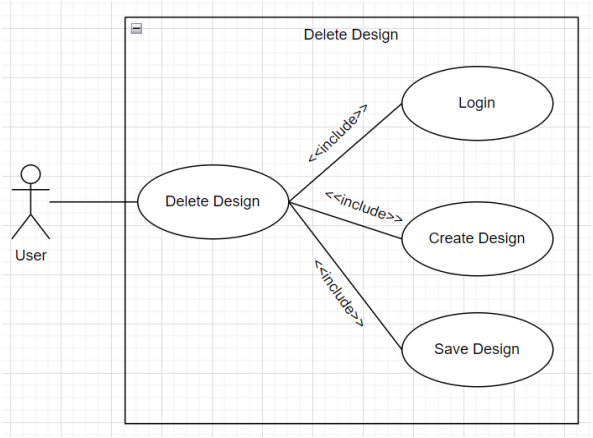

Figure 2.6 Delete Design Use Case Diagram

2.4.7 Edit Design Use Case

If the user wants to edit a design, he must login first. Once logged in he can edit any of the design saved online. The user can also upload design from his computer that is in the form of a JSON file.

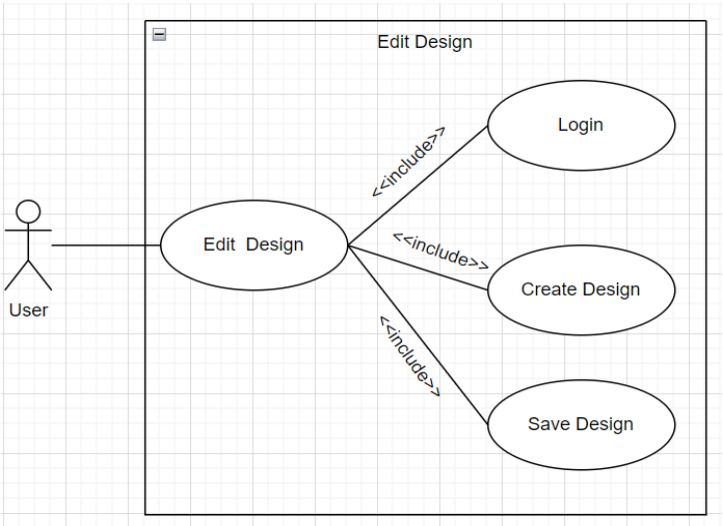

Figure 2.7 Edit Design Use Case Diagram

2.4.8 Create User Use Case

To create a new user, the admin must enter an email, username and create a secure password for the user.

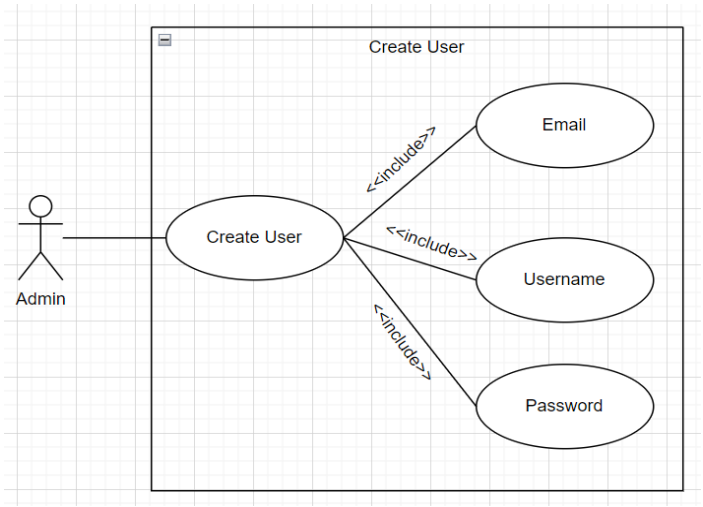

Figure 2.8 Create User Use Case Diagram

2.4.9 Delete User Use Case

Once the admin logs in, he can see the list of all the users. He can delete any user he wants from his account. Once a user has been deleted by the admin, that user would not be able to access his account anymore.

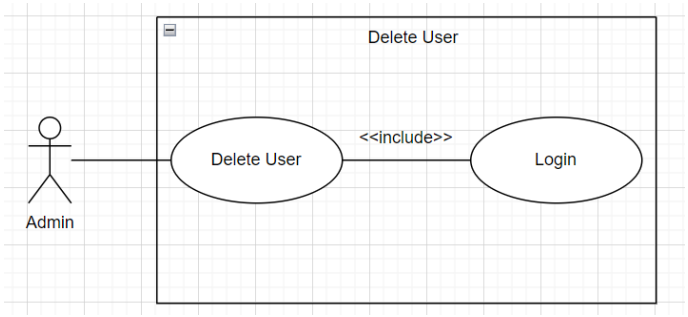

Figure 2.9 Delete User Case Diagram

2.5 Use Case Scenarios

2.5.1 System

Use Case ID:	1
Name	System use case
Actors	User, admin
Description	Users can sign up, login, create, save, delete, and edit designs. The admin can add or delete a user.
Pre-Conditions	Users and admin have to login into their account to perform required actions
Post-Conditions	Once designs are created by users they will be stored in the database or the computer of the user. Once the admin creates an account for the user, that user can login into that account. If the admin deletes an account, that user would not be able to access his account.

Normal Course	**Information For Steps**
User logging into his account	To login, the user must enter his email and password
User signing up	To signup, the user must enter his email, username and create a password
Creating, saving, deleting, and editing designs	These actions are only possible once the user accesses his account
Creating and deleting users	Only the admin has the authorization to create and delete a user. He can perform these actions once he logs into his account
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.1 System Use Case

2.5.2 Login

Use Case ID	2
Name	Login use case
Actors	User, admin
Description	Users and admin can log into their accounts
Pre-Conditions	Users and admin must enter their email and password to login
Post-Conditions	Once logged in, users and admin can perform the required actions

Normal Course	**Information For Steps**
User or admin goes to the login page	Once clicked on the login option, the login form will be shown on the screen
Entering email address and password	The system verifies that the credentials are correct and logs in the user
Alternative Course	
The user can also go the login page by clicking on the “login” option on the signup page	
Exception	
Wrong Credentials were entered	
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.2 Login Use Case

2.5.3 Signup

Use Case ID	3
Name	Signup use case
Actors	User
Description	Users can create a new account
Pre-Conditions	Users must enter email address, name, and password to create an account
Post-Conditions	Once the account has been created, the user can login

Normal Course	**Information For Steps**
User goes to signup page	Once clicked on the signup option, the signup form will be shown on the screen
Entering email address, name, and password	The account is created
Alternative Course	
The user can also go the signup page by clicking on the “signup” option on the login page	
Exception	
Account already exists	
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.3 Signup Use Case

2.5.4 Create Design

Use Case ID	4
Name	Create design use case
Actors	User
Description	Users can create a new design
Pre-Conditions	Users must login to create a new design
PostConditions	Once the design is completed, the user can save it

Normal Course	**Information For Steps**
User logs into his account	Once logged in, the user can see the option of creating a new design
User creates a new design	The drawing board provides the functionality of creating a design
Alternative Course	
User can make multiple designs	
Exception	
User should be logged in	
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.4 Create Use Case

2.5.5 Save Design

Use Case ID	5
Name	Save design use case
Actors	User
Description	Users can save the design
Pre-Conditions	Once the user creates a design, he can save it
Post-Conditions	User can save online, or on his computer in JSON or SVG

Normal Course	**Information For Steps**
User creates a new design	User goes to the drawing board by clicking on the new design button
User saves the design	User clicks the save button
Alternative Course	
User can export the design to his computer	
Exception	
User should create the design first	
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.5 Save Design Use Case

2.5.6 Delete Design

Use Case ID	6
Name	Delete design use case
Actors	User
Description	Users can delete the design
Pre-Conditions	Once the user creates a design, he can delete it
Post-Conditions	Once the design is deleted, it cannot be brought back

Normal Course	**Information For Steps**
User creates a new design	User goes to the drawing board by clicking on the new design button
User saves the design	User clicks the save button
User deletes the design	The design is deleted from the database
Alternative Course	
User can also delete the design from his computer	
Exception	
User should create the design first	
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.6 Delete design Use Case

2.5.7 Edit Design

Use Case ID	7
Name	Edit design use case
Actors	User
Description	Users can edit the design
Pre-Conditions	Once the user creates a design, he can edit it later
Post-Conditions	After editing a design, the user can save it

Normal Course	**Information For Steps**
User creates a new design	User goes to the drawing board by clicking on the new design button
User saves the design	User clicks the save button
User edits the design	The user clicks on the open design option
Alternative Course	
User can also upload a JSON file from his computer and edit it	
Exception	
User should create the design first	
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.7 Edit Design use Case

2.5.8 Create User

Use Case ID	8	
Name	Create user use case	
Actors	Admin	
Description	Admin can create a new user	
Pre-Conditions	The admin must enter an email, name, and password to create a new user	
Post-Conditions	Once the user is created, that user can access his account	
Normal Course		**Information For Steps**
Admin creates a new user		Admin clicks on the new user option in his account and enters email address, name and password
Alternative Course		
Admin can create as many users as required		
Exception		
User already exists		
Created By		Shaharyar Ahmad, Gul e Sahar

Table 2.8 Create User Use Case

2.5.9 Delete User

Use Case ID	9
Name	Delete user use case
Actors	Admin
Description	Admin can delete the user
Pre-Conditions	Admin can delete any user of his choice
Post-Conditions	Once the user is deleted, it cannot be brought back

Normal Course	**Information For Steps**
Admin deletes the design	The user account is deleted
Alternative Course	
Admin can also delete the user from the database	
Exception	
The user should exist	
Created By	Shaharyar Ahmad, Gul e Sahar

Table 2.9 Delete User Use Case

2.6 Non-Functional Requirements

Non-functional requirements are as important for a system as functional requirements. These are the quality attributes that make a system reliable. If a software completes all the functional requirements but fails to fulfil non-functional requirements, there is a strong chance that the system may fail to please the users. Some non-functional requirements that we kept in mind during development are following:

2.6.1 Security

Security is the most common requirement nowadays. To make the designs of the users safe, we have given the option of creating a password protected account. No one else will be able to see the user's designs. We also include password encryption. Even if an unauthorized user somehow manages to get access of the database, he won't be able to see the user passwords.

2.6.2 Performance

We also made sure that the application works on as low requirements as possible. It is a single page application which does not reload at all. Therefore, even if a user has a slow internet connection, the application never fails to fulfil its users' requirements.

2.6.3 Compatibility

We made the system highly compatible. It means that it can be used in different operating systems and on different web browsers. The application does not have very high computer requirements either.

2.6.4 Portability

The user can log in into his account no matter where he is located. The designs that are made by the users are portable as well. Once the user downloads a design from his account into his computer, he can give that design to any other user. That user can reupload that design in his account and can view or make changes to it.

2.6.5 Usability

The interface of the application is simple and user friendly. It is easy to learn how to use it and even a non-technical user can use the application without any problem.

2.6.6 Reusability

The components that are used in the application during its development are reusable. It means that if we want to implement the same kind of system in the future, we can easily use the same components.

CHAPTER 3

DESIGN AND METHODOLOGY

3.1 Methodology

We have decided to follow agile scrum methodology. Scrum was first introduced in 1997 and has since become the most widely applied agile software development framework [4]. It involves an iterative approach and incremental delivery. It means that it will allow us to divide our work into smaller parts or modules. We can complete each module one by one. After completing a particular module, we will be able to review and check if the quality is up to the mark before proceeding on to the next one. With the help of agile scrum methodology, we will be able to achieve high accuracy and high-quality product.

- **Requirement gathering and analysis**

  In this phase, we will identify the functional and non-functional requirements of our project. Functional requirements include the functionalities and the tasks that the application is expected to perform. Non-functional requirements include quality constraints which make the application reliable and efficient. We will also make sure that we have the required hardware resources like computers or laptops, internet availability and working environment.

- **Design**

  In this phase, we will develop architecture of our project and also design the flow of work. The whole project will be divided into smaller sub tasks or modules. We will assign a priority to each of these modules.

- **Development**

  The coding will start from this phase. We will start the development phase with the front end of the application. All the modules involved in the frontend will be developed based on their priority. After completing a certain module, we will check it and make sure that it is working fine before moving on to the next one. Once all the modules of frontend are completed, we will integrate them and move towards the development of backend. The backend development will take place exactly like frontend development. After completing the development of backend, we will integrate it with the frontend. The last phase of development would be to integrate the database.

- **Testing**

  This phase involves a series of steps which are completed to make sure that the application is doing what it is supposed to do. We will make sure that the outcomes align with the objectives of our project.

- **Deployment**

  After the testing is done, we will run the project in local host. We are also planning to deploy the application in the future.

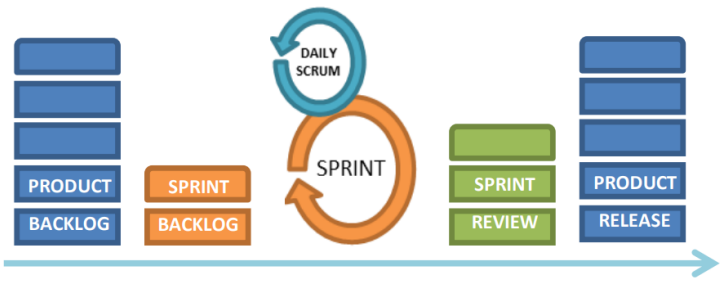

Figure 3.1 Agile Scrum Methodology

- **Product Backlog:**

  We will make a dynamic list of all the features that we have to achieve to complete the development of the application.

- **Sprint Planning:**

  We will select a priority for each feature and divide it into sub tasks.

- **Spring Backlog:**

  The priority of each subtask will be decided in this phase.

- **Scrum Team:**

  The team members will start the development of the project.

- **Increment:**

  When the task with the highest priotiy is completed, the first sprint will be completed and we will go ahead towards the next one.

- **Sprint Review:**

  After comepleting a certain sprint, we will review and make sure that it is working fine.

- **Sprint Retrospective:**

  The team will discuss about those things that were learned during the development.

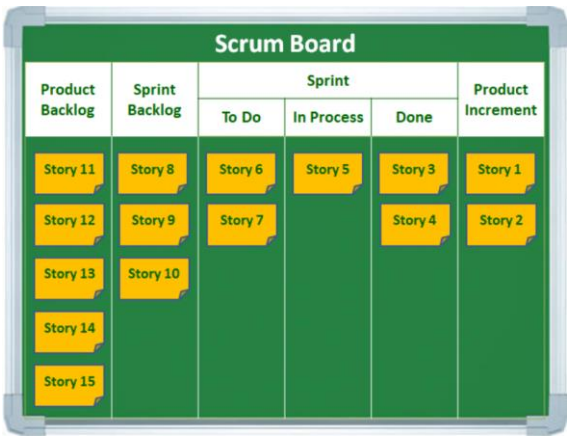

Figure 3.2 Scrum board

3.2 Sequence Diagrams

The series of actions that take in place in a system are shown in a sequence diagram. This diagram also shows how each method is invoked. The sequential diagram can be used to understand the behaviour of the system.

3.2.1 Login Sequence

This diagram shows the working of login operation. When the user clicks on the login option, he must enter the details to login. Once the system validates that the data entered by the user is correct, the user logs in.

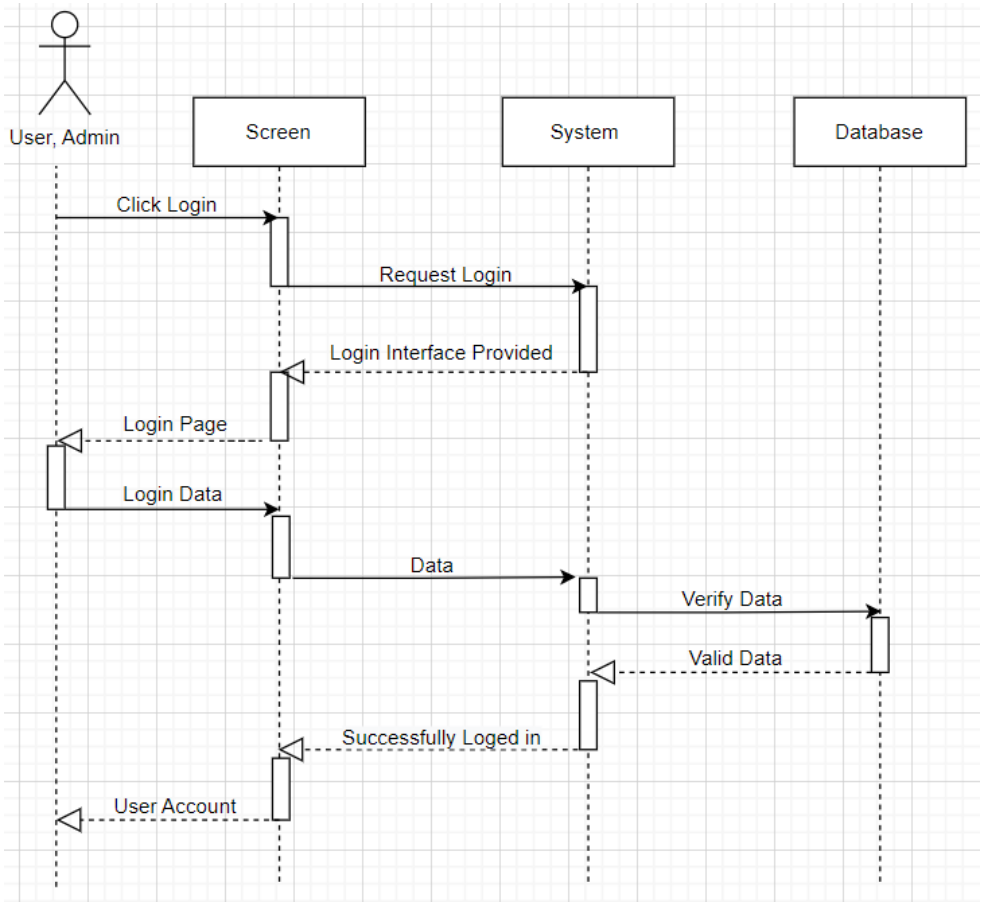

Figure 3.3 Login Sequence Diagram

3.2.2 Signup Sequence

This diagram shows the working of signup operation. When the user clicks on the signup option, he must enter the details to create a new account. Once the system validates that the data entered by the user is correct, the account is created.

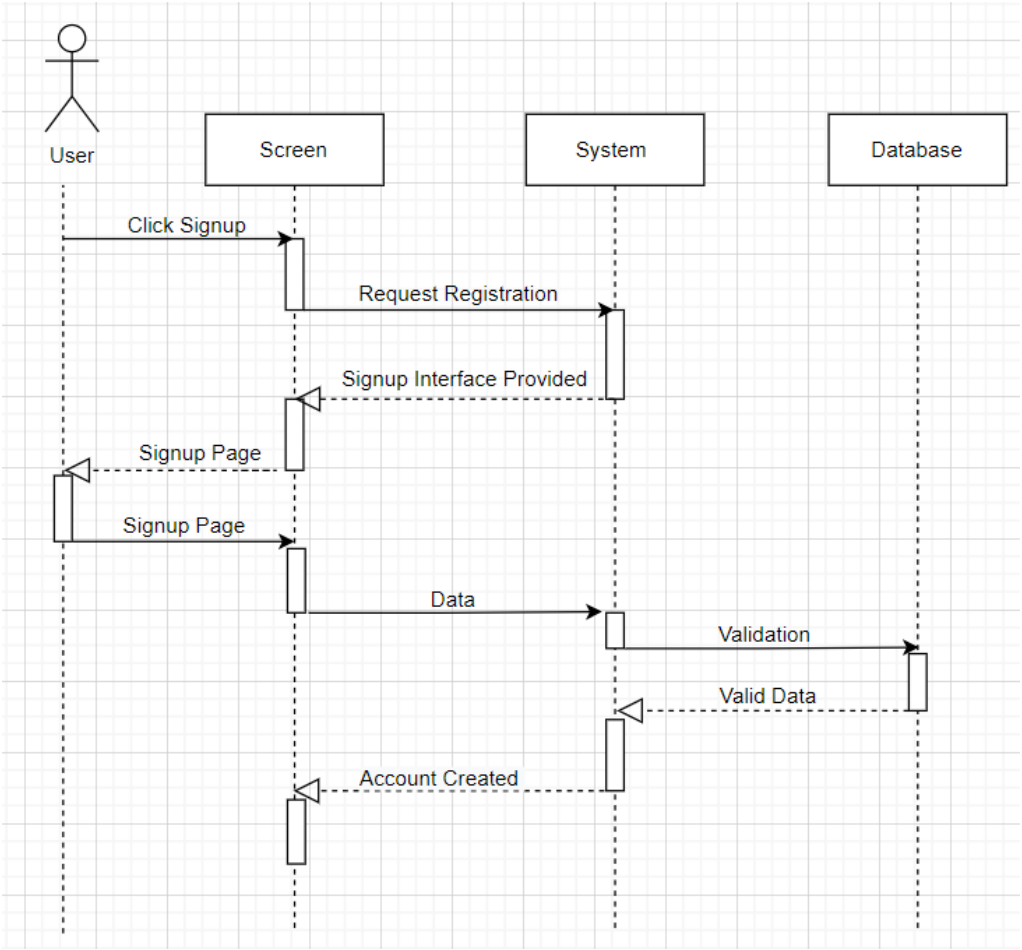

Figure 3.4 Signup sequence Diagram

3.2.3 Create, Save, Delete, Edit Design Sequence

Once the user accesses his account, he has the option of creating, deleting, saving, and editing designs. When he selects a required option in his account, the request is sent to the system.

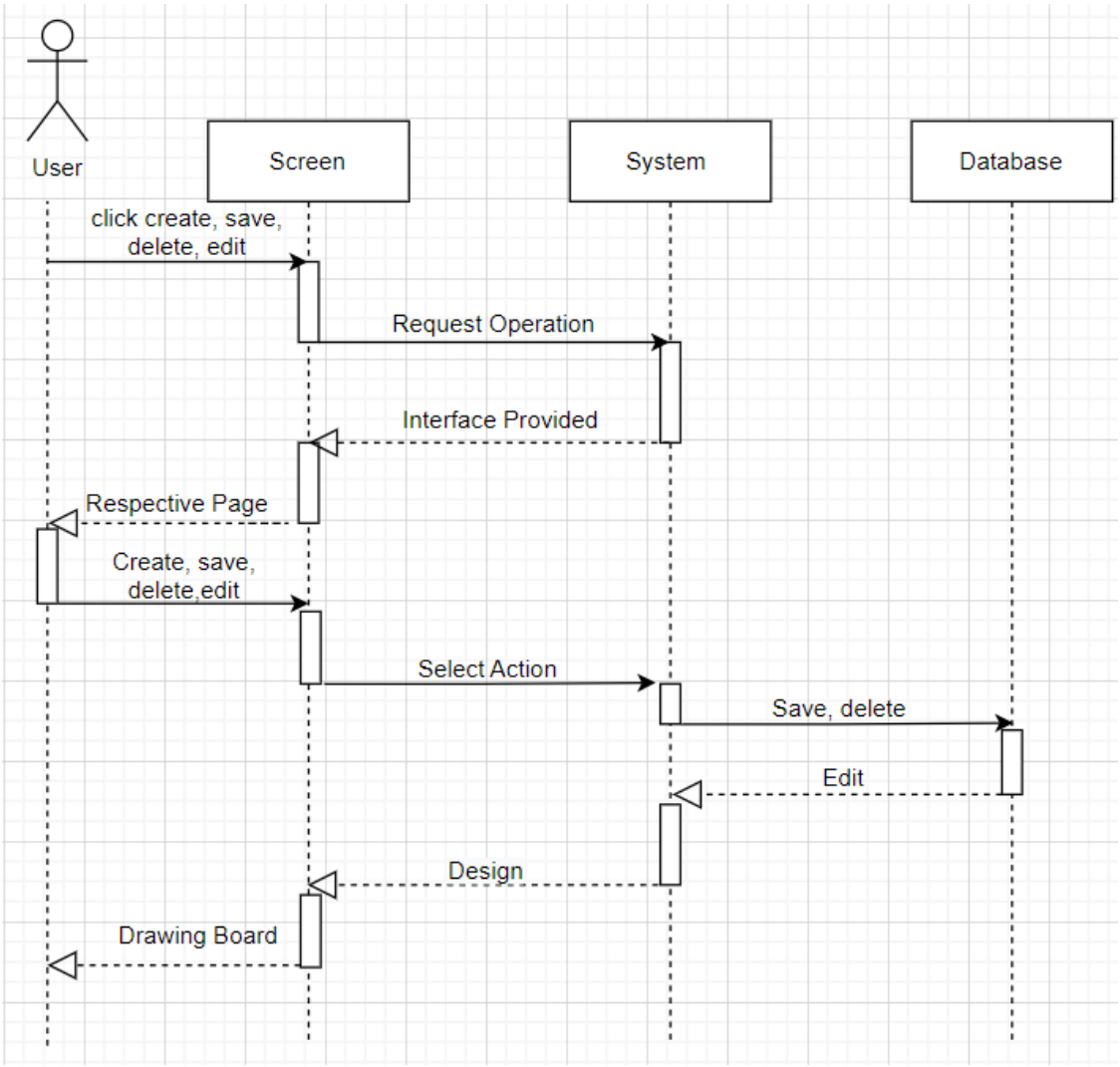

Figure 3.5 Create, Save, Delete and Edit Sequence Diagram

3.2.4 Create User Sequence

The admin can create a new user once he is logged in. He must enter the name, email address and password for the new user

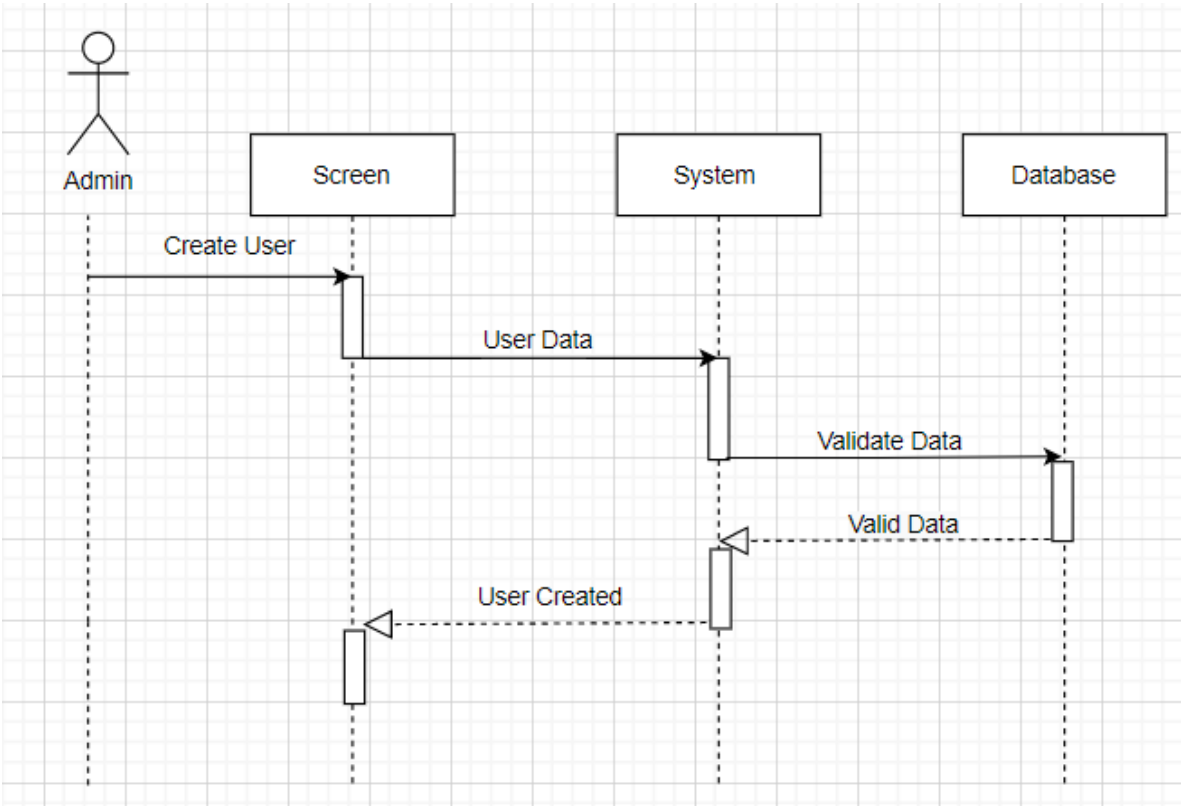

Figure 3.6 Create User Sequence Diagram

3.2.5 Delete User Sequence

The admin can delete a selected user at any time. He must log in first to delete a user. Once the delete operation is perform by the admin, it cannot be undone.

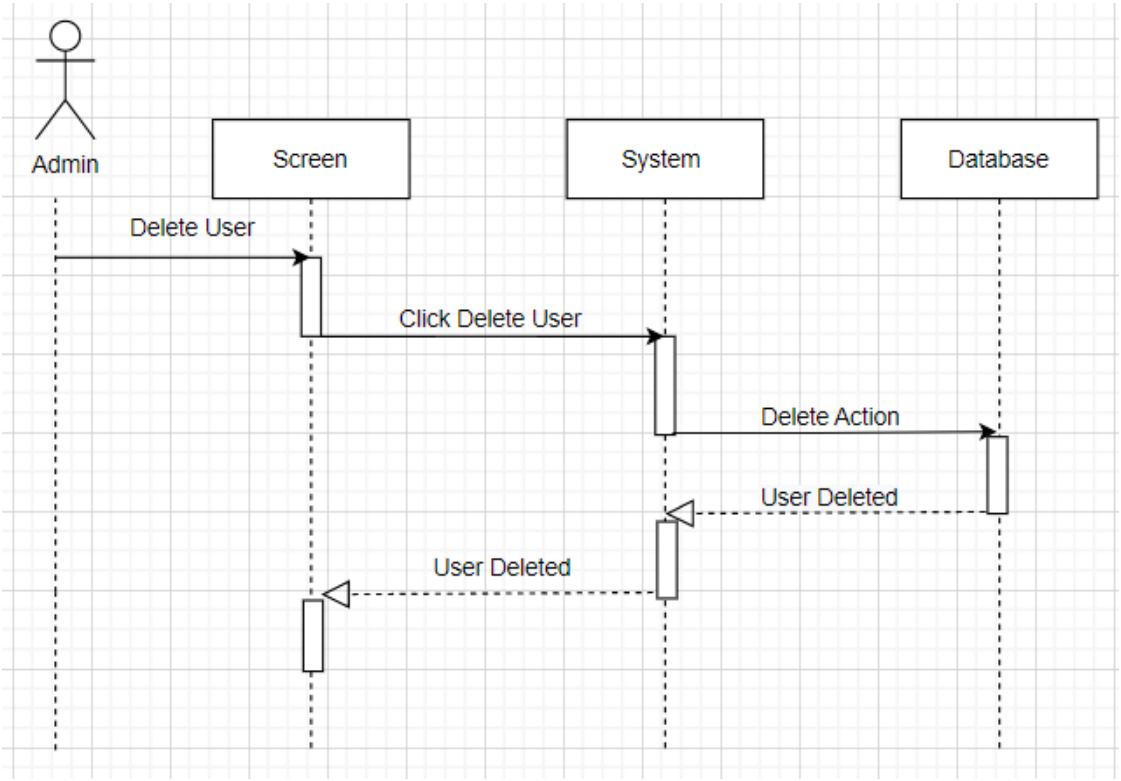

Figure 3.7 Delete User Sequence diagram

3.3 Class Diagram

Class diagram is used to explain the structure of an application. All the classes and their functions are mentioned in the class diagram.

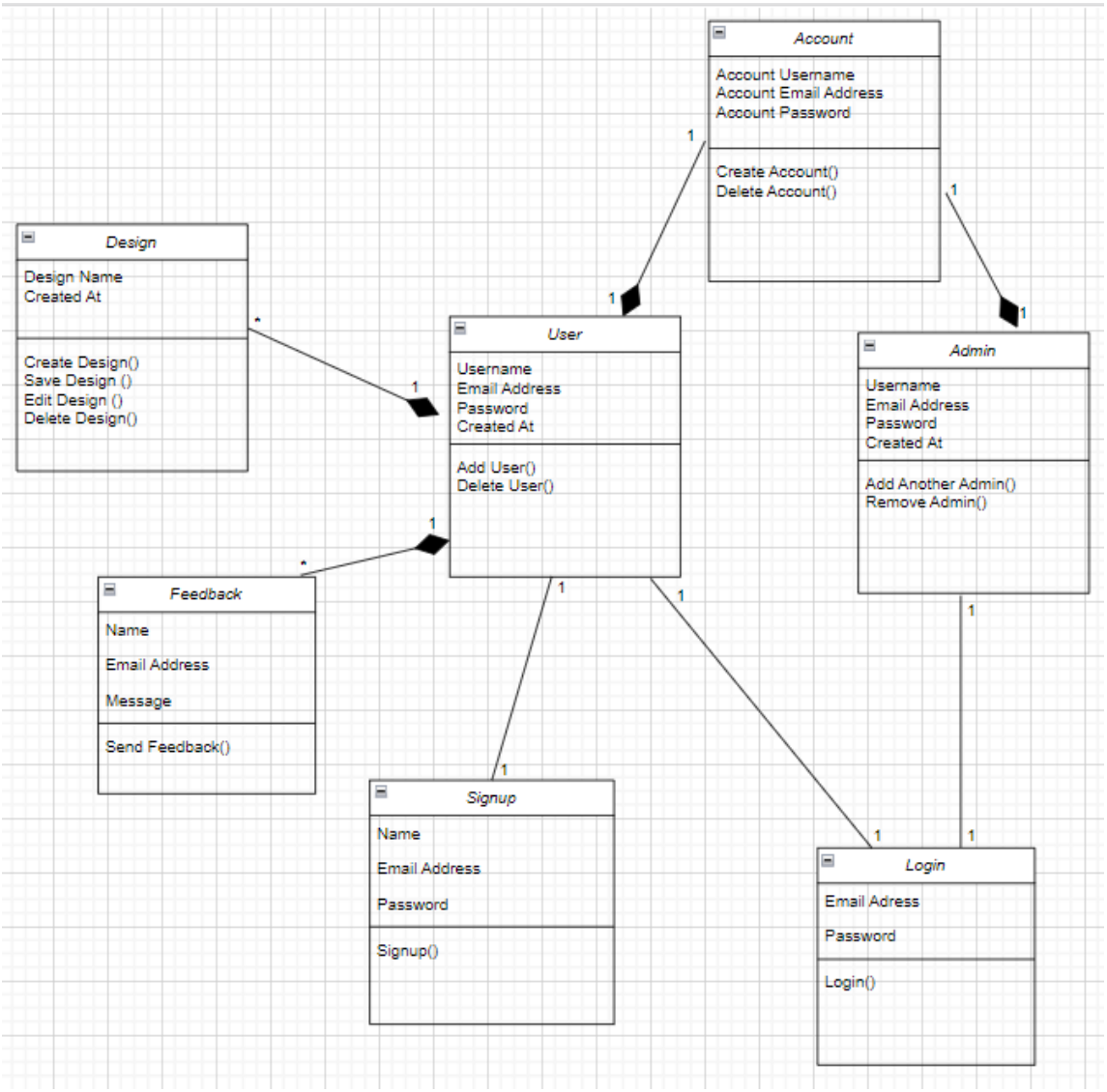

Figure 3.8 Class Diagram

3.4 Class Diagram Functions

The following table provides the description about every function in the class diagram.

Class	Function	Overview
Admin	Add Another Admin () Remove Admin ()	There can be more than one admins of the application. Each admin can create another user admin or remove an admin
User	Add User () Delete User ()	The admin can create a new user or delete any user
Login	Login ()	Once the correct credentials are entered, the user or admin can login.
Signup	Signup ()	To use the application, the users have to signup first. To sign up, they must enter their name, email address and password
Design	Create Design () Save Design () Edit Design () Delete Design ()	The user can create new designs, save them once created, delete them, and edit them later. The user must login first to perform any of these tasks
Feedback	Send Feedback ()	Even if a user does not have an account, he can still send messages and queries
Account	Create Account () Delete Account ()	The user can signup to create a new account

Table 3.1 Class Diagram Functions

CHAPTER 4

DATA AND EXPERIMENTS

4.1 Technologies Used

We have used the following technologies or languages to develop the project:

4.1.1 HTML

HTML is Hypertext Markup Language. It is used mainly used to provide structure to web page. We have used HTML to provide a structure to our components.

4.1.2 CSS

Cascading style sheet is being used to provide designs to out components. It is being used to make out project presentable**.**

4.1.3 JavaScript

It is an object-oriented programming language which is used in the project develop a logic and interaction between components.

4.1.4 TypeScript

It is a superset of JavaScript. Angular mainly uses TypeScript. It is used to improve the developing experience. The benefit of Typescript is that it detects errors before compilation. Each Angular component has a Typescript file in which we can write the working of the components, variables, objects, and functions.

4.1.5 Angular

Angular is a framework which is used to develop scalable web applications. It includes a wide variety of libraries and features. Angular uses components which work together as building blocks for an application. These components are reusable. Each Component has its own HTML, CSS, and Typescript file.

4.1.6 Angular Material and Bootstrap

Angular material provides readymade components which can be used anywhere within the application. An example of Angular Material component used that is used in our application is Snack bar.

Bootstrap is a CSS framework. We have used this to make our user interface attractive and engaging. Some of the Bootstrap components used in our application are forms and buttons etc.

4.1.7 NodeJS and Express

NodeJS allows us to write JavaScript on server side. It is used to allow the user to interact with the backend. Angular does not communicate with the database directly. NodeJS is a platform that connects the frontend with the database.

Express is a NodeJS framework which we have used to write REST APIs to make our code efficient and minimal.

4.1.8 MongoDB and Mongoose

Mongo DB is a NoSQL database. It is more flexible and convenient as compared to SQL. The benefit of Mongo DB is that it is schema less. However, some of the

operations might be difficult to perform with Mongo DB but Mongoose makes it easier for us.

4.1.9 BPMN IO

BPMN IO is a JavaScript library which provides the rendering functionality. It is an open-source library embeddable with Angular.[5] It can used or embedded into any type of web application. With the help of this library, we were able to develop a drag and drop interface and the drawing board.

4.2 System Requirements

The following requirements are required to be fulfilled to run the application:

- Computer/Laptop
- Web browser
- Stable Internet connection

4.3 Test Cases

Test cases are designed to make sure that each functionality of the application is working fine. It includes a set of actions or instructions that are used to perform these cases. The system determines whether the test case fails or passes.

4.3.1 Login Test Case

Test Case ID	1
Test Name	Login test case
Actors	User, admin
Pre-Conditions	User or admin wants to login into their account
Test Steps	**Information For Steps**
1- Enter Email Address	The system verifies that the email Address is correct
2- Enter Password	The system verifies that the password is correct
3- Click login	The login request is sent to the backend
Expected Result	Successful login
Post Condition	The user or admin can access the account
Status	Pass

Table 4.1 Login Test Case

4.3.2 Signup Test Case

Test Case ID	2
Test Name	Signup test case
Actors	User
Pre-Conditions	User wants to create a new account
Test Steps	**Information For Steps**
1- Enter Email Address	The system validates the email address
2- Enter Username	The system validates the username
3- Create Password	The system validates the password
4- Click Signup	The signup request is sent to the backend
Expected Result	Successful login
Post Condition	The user or admin can access the account
Status	Pass

Table 4.2 Signup Test case

4.3.3 Create Design Test Case

Test Case ID	3
Test Name	Create design test case
Actors	User
Pre-Conditions	User wants to create a new design
Test Steps	**Information For Steps**
1- User logs into his account	The dashboard is shown to the user
2- User clicks on the new design option	The drawing board is shown
Expected Result	The design is created
Post Condition	The user or admin can access the account
Status	Pass

Table 4.3 Create design Test Case

4.3.4 Save Design Test Case

Test Case ID	4
Test Name	Save design test case
Actors	User
Pre-Conditions	User wants to save a newly created design
Test Steps	**Information For Steps**
1- User creates a new design	The design is ready to be saved
2- User clicks on the save design option	The design is saved in the database
Expected Result	The design is saved
Post Condition	The user can now access the design in the future
Status	Pass

Table 4.4 Save Design Test Case

4.3.5 Delete Design Test Case

Test Case ID	5
Test Name	Delete design test case
Actors	User
Pre-Conditions	User wants to delete a design
Test Steps	**Information For Steps**
1- User clicks on the delete design option	The design is deleted
Expected Result	The design is deleted from the database
Post Condition	The user won't be able to see the design anymore
Status	Pass

Table 4.5 Delete Design Test Cases

4.3.6 Edit Design Test Case

Test Case ID	6
Test Name	Edit design test case
Actors	User
Pre-Conditions	User wants to edit a design
Test Steps	**Information For Steps**
1- User clicks on open design option	The design is loaded on the drawing board
Expected Result	The user can now edit the design
Post Condition	The user can save edited design in place of the old design
Status	Pass

Table 4.6 Edit Design Test Case

4.3.7 Create User Test Case

Test Case ID	7
Test Name	Create user test case
Actors	Admin
Pre-Conditions	Admin wants to create a new user
Test Steps	**Information For Steps**
1- User logs into his account	The dashboard is shown to the admin
2- User clicks on the new user option	User can now create a new user
3- Add username	The system validates the username
4- Add email address	The system validates the email address
5- Create password	The system validates the password
Expected Result	The user is created
Post Condition	That user can now access his account
Status	Pass

Table 4.7 Create User Test Case

4.3.8 Delete User Test Case

Test Case ID	8
Test Name	Delete user test case
Actors	Admin
Pre-Conditions	Admin wants to delete a user
Test Steps	**Information For Steps**
1- User logs into his account	The dashboard is shown to the admin
2- User clicks on the delete user option	The user is deleted
Expected Result	The user is deleted from the database
Post Condition	That user cannot access his account anymore
Status	Pass

Table 4.8 Delete User Test Case

CHAPTER 5

USER MANUAL

5.1 Home Page

The home screen is the first page that the user sees when he runs the application. The user can see a navigation bar which gives him the option of navigating to the different pages of the application. On home screen, the user can also see a demonstration video which gives him an idea about how to the application works. The user can watch this video before signing up.

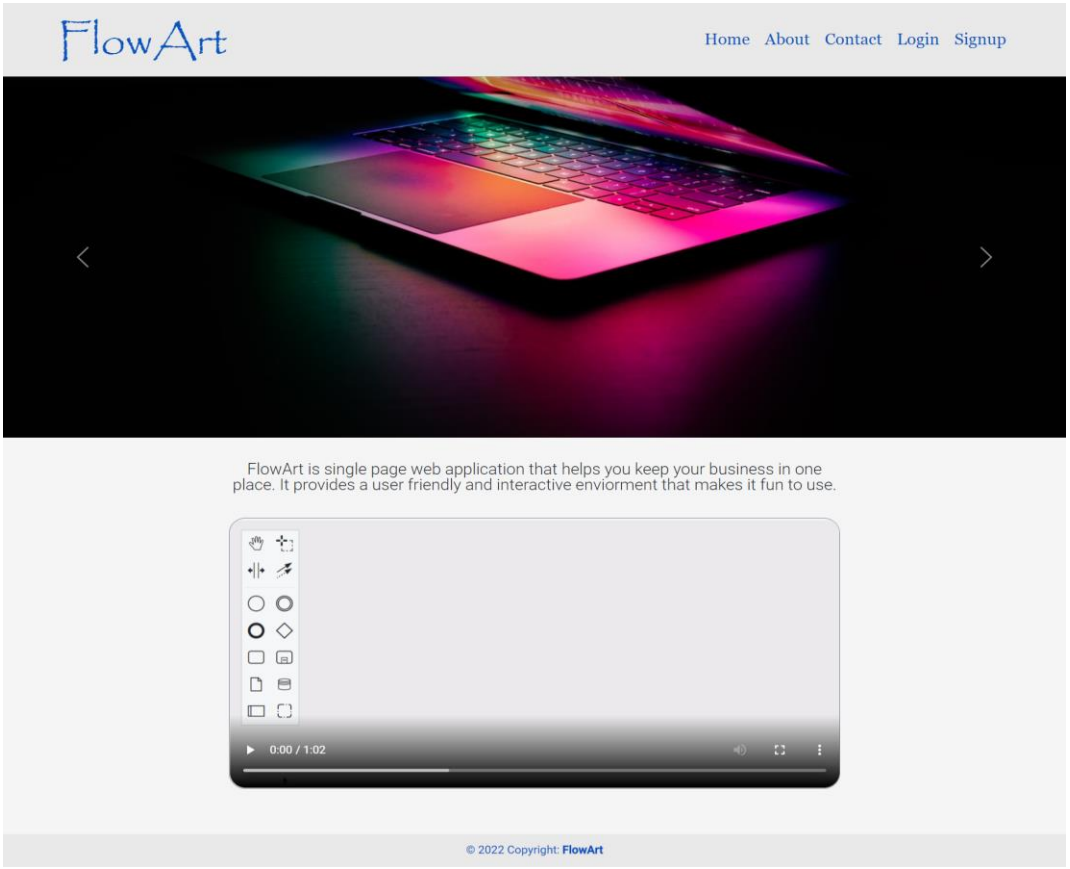

Figure 5.1 Home Page

5.2 About Page

The about page shows some of the important features that we have included in the application along with their description.

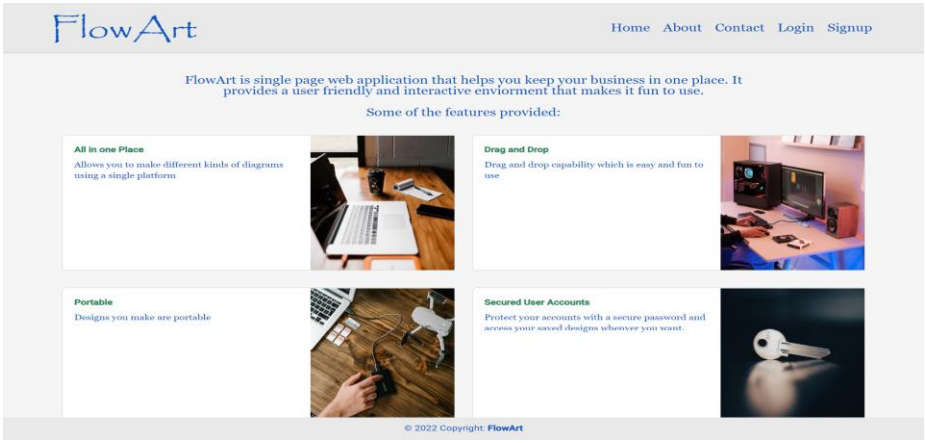

Figure 5.2 About page

5.3 Contact Page

Users can send messages or queries through the contact page. These messages can be seen by the admin only. The address, contact number and the email address can be seen on the contact page as well.

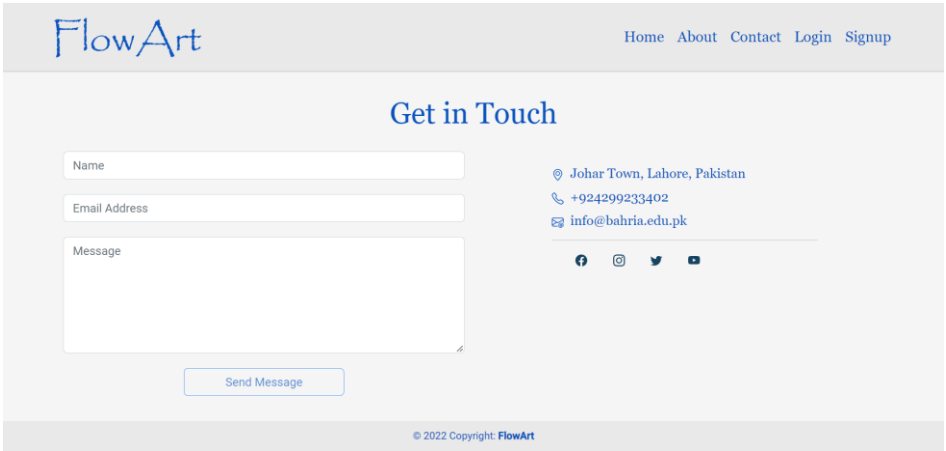

Figure 5.3 Contact Page

5.4 Login Page

To login into the account, the user must navigate to the Login page. Once the user enters his email address and password and clicks on the login button, he is navigated into his account. The user can also click on “Signup” if he already has an account.

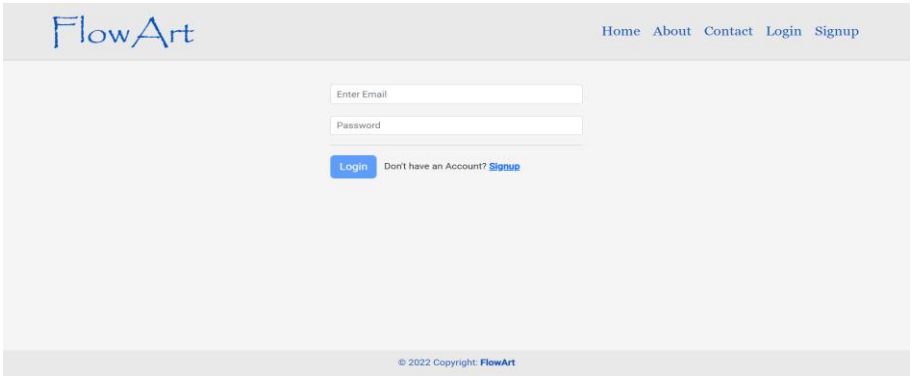

Figure 5.4 Login Page

5.5 Signup Page

If the user does not have an account and wishes to sign up, he can go to the signup page and create a new account after entering his email address, username, and creating a secure password. Once the user’s account has been created, he is automatically navigated to the login page

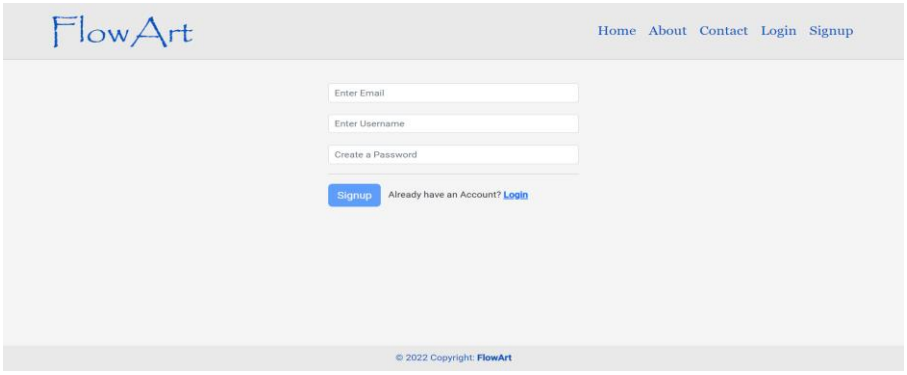

Figure 5.5 Signup Page

5.6 User Dashboard

Once the user logs into his account, he can see the list of all the previously saved designs. The username is written on the navigation bar. If the user wants to create a new design, he can click on the “New Design” button. He can also open or delete a

saved design by clicking on the respective buttons. If the user wants to logout, he can click on the logout button located at the top right corner.

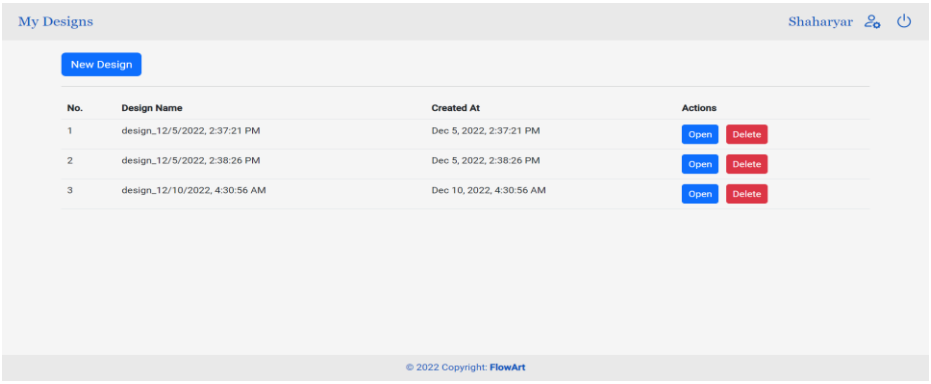

Figure 5.6 User Dashboard

5.7 Drawing Board

When the user clicks on the "New Design" button, he is navigated to the drawing board page. A template is given on the drawing board. The user can modify it to create designs or diagrams of his choice. After completing the design, the user can either save it into the database online or download it on his computer in the form of JSON or SVG. This page also gives the option of uploading a JSON file from the computer so the user can edit it.

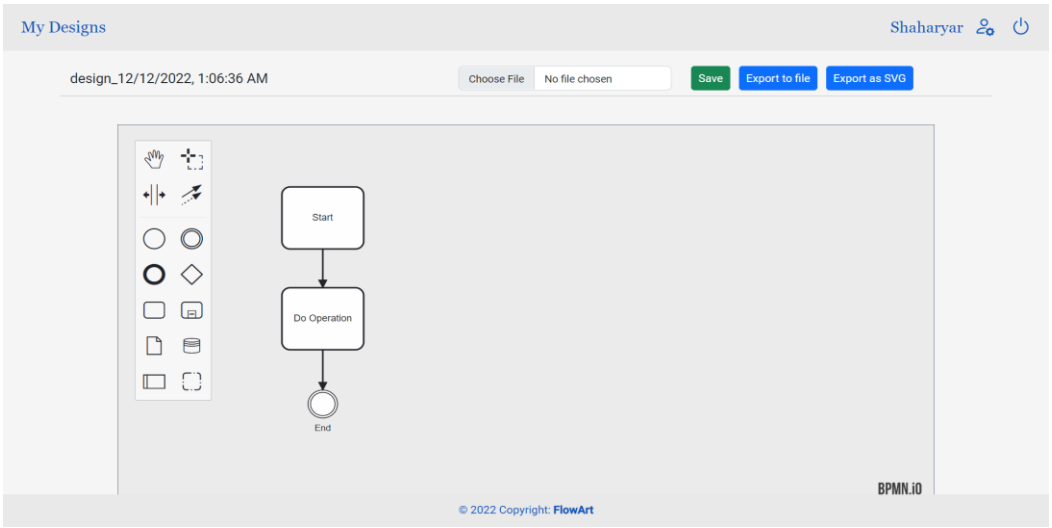

Figure 5.7 Drawing Board

5.8 Admin Dashboard

Once the admin logs into his account, he can see the list of all the users. He has the option of creating a new user and deleting the user. There can be more than one admins of the application. Every admin can see his name on the navigation bar. If the admin wants to logout, he can click on the logout button located at the top right corner.

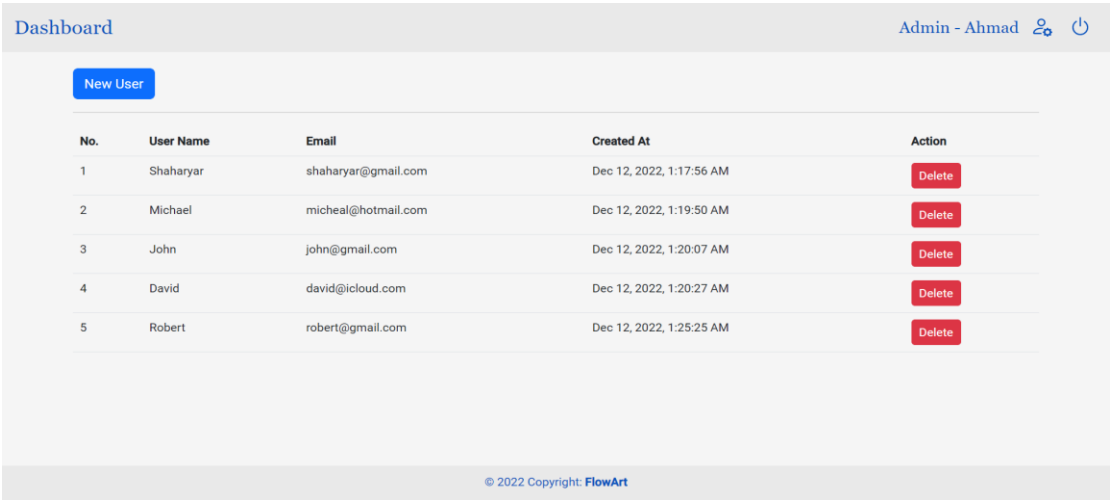

Figure 5.8 Admin Dashboard

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

We were able to complete the objectives of the application. Before starting the development of the project, we were required to learn technologies such as Angular, Node JS and Express. We also learned the procedure of integrating MongoDB with the project. After learning these technologies, we also learned frontend frameworks such as Angular Material and Bootstrap to develop an attractive user interface. Along with the objectives of the application it also has some of the major quality attributes. These attributes make the application reliable and efficient. During the development of the application, we tried to adopt efficient coding practices. The code written for the application is separated by folders and is highly manageable. We were also able to integrate the required library to complete our objectives. A long period of time was consumed while learning the required technologies and frameworks. Since we are now familiar with these technologies, we are planning to increase the functionality of the application in the future.

REFERENCES

[1] "Online Diagram Software & Visual Solution | Lucidchart." https://www.lucidchart.com/pages/landing?utm_source=google&utm_medium=cpc&utm_campaign=_chart_en_tier3_mixed_search_brand_exact_&km_CPC_CampaignId=1484560207&km_CPC_AdGroupID=60168114191&km_CPC_Keyword=lucidcharts&km_CPC_MatchType=e&km_CPC_ExtensionID=&km_CPC_Network=g&km_CPC_AdPosition=&km_CPC_Creative=442433234360&km_CPC_TargetID=aud-381457345638:kwd-84176206937&km_CPC_Country=1011082&km_CPC_Device=c&km_CPC_placement=&km_CPC_target=&gclid=CjwKCAjw9e6SBhB2EiwA5myr9gBHqH2dyY2j5q8ZUIy3x4wDHILWiH7a-7MfzEjCz6prLozeiUF0oxoCCiMQAvD_BwE (accessed Dec. 12, 2022).

[2] "Free UML Diagram Tool - Free Templates Make UML Design Easy." https://www.smartdraw.com/uml-diagram/uml-diagram-tool.htm (accessed Dec. 12, 2022

[3] "Free Online Flowchart Maker - Create Flowcharts Online | Visme." https://www.visme.co/flowchart-maker/ (accessed Dec. 12, 2022)

[4] M. Hron and N. Obwegeser, *Scrum in practice: an overview of Scrum adaptations*. [Online]. Available: http://hdl.handle.net/10125/50568

[5] "Web-based tooling for BPMN, DMN, CMMN, and Forms | bpmn.io." https://bpmn.io/ (accessed Dec. 12, 2022).