End of study Project

Boost Buddy

Student's Resource Platform

Project in Business Intelligence

Student: Hadhami Abidi

Supervisor: DR Hamdi Hassen

Year : 2023-2024

TABLE OF CONTENTS

FIGURES LIST

TABLES LIST

General Introduction

Technology nowadays keeps evolving .In order to adapt to these changes humans tend to create new methods regarding these innovations .In the field of education , the use of the internet is almost a must thanks to the facilities it provides and how easy it makes the learning process .It offers a wide range of online courses and resources that play a huge role in the level of the student's comprehension .Since most of these resources are free , it reduces a lot of costs saving you time and money , not to forget , the possibility to learn at your own pace and access at any time .

It's true that a lot of resources are available ,but most of them don't match what's being taught at universities .Within this context aligns our project offering a solution that targets this specific problem that drains the student's energy every year .The idea is to create a web application named " Boost Buddy " that will hold all the resources and information necessary to boost the student's performance and help them achieve higher grades .

This web application entails different types of documents that you can access whenever you want . Due to the shortage of details and clarity about internships , i believe that adding a section to share one's experience is a crucial step to learn from others experiences .

This report will present you with a walk through my journey as i develop this web application so that you can grasp the details .

Chapitre

1 Preliminary Study

Plan

1.1 Introduction

Setting the stage for our walk through the journey , we will start by casting a gaze upon the host organization and introduce it then we're going to present the project's context .After that we will analyse the state ,explain the problematic and provide the solution for together with specifying the objectives that we desire to reach once the project comes to an end .

1.2 Host Organization Introduction

ISGS is a public university situated in Sousse and is part of Sousse University .Being a part of the educational environment ,this place was created in February 1995 . Furnished with a lot of amazing teachers with outstanding backgrounds alongside a broad selection of majors . An organization can't be formed without an objective that leads it to work harder , ISGS's objective is to provide high quality education and make their degrees recognized across the world .

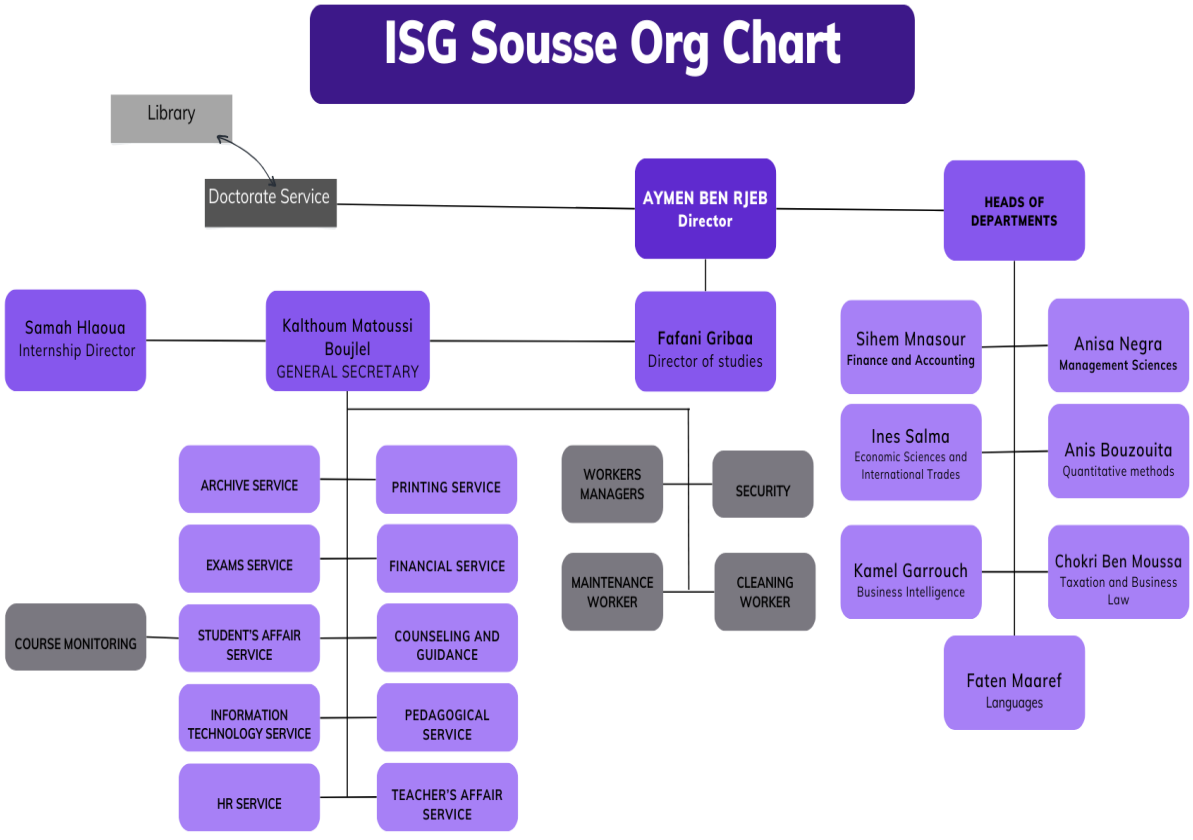


FIGURE 1.1 – ISGS Organizational Chart

1.2.1 Majors in ISGS

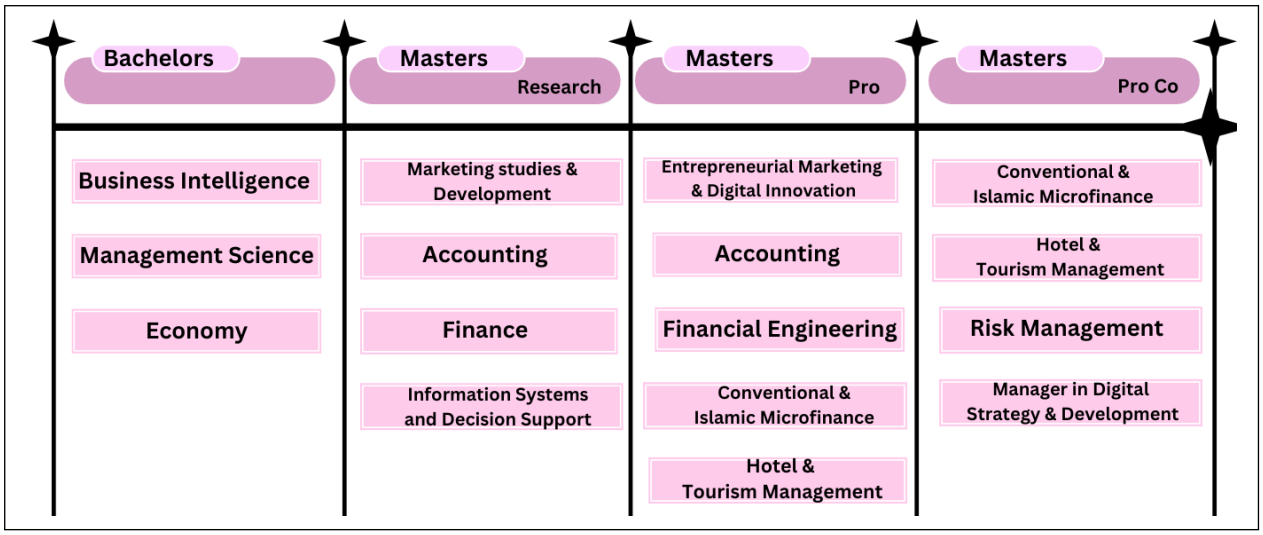


FIGURE 1.2 – Majors in ISGS

1.3 Project Context

This project was done in the context of an end of study project at the Higher Institute Of Management Of Sousse aiming to obtain a bachelor's degree in Business informatics " Business Intelligence " . This project is the result of my genuine effort i made during my internship that lasted 3 months at my university " ISGS " . We will now be discussing the problematic that led me to come up with the project's topis .

1.4 Existing State Analysis

By doing an analysis of the existing state we figure out what's exactly missing to find a solution for it .We also study previous projects that are similar to benefit from them and to avoid their mistakes .

1.4.1 Existing State Description

Innovative solutions were found in every field with the rise of internet dependency .Within the domain of education ,teaching methods have changed to what we call " Auto-Education " that signifies learning by yourself and at your own pace using resources online which i believe is very practical due to the fact that everybody's learning pace is different along with other factors like available time , geographical constraints , cost barriers ....

1.4.2 Problematic

The available resources online are not always useful since they can be very different from the courses currently instructed .Students had to look for documents from old students which is an exhausting process .Once they get the documents they find themselves in front of a pile of papers not knowing where to start so they lose motivation along the way .Most students don't seem to know a lot about the professional life and skills they need to build for their careers so they need a place that helps them acknowledge what it's like from other people's experience .

1.4.3 Solution

Students need a support in both academic and professional life , either they will get help from the platform or they will offer help for others. Our web application will be a platform that hold the necessary resources to support the students .They will be able through it to :

- Manage documents
- Manage Posts about internship experiences
- Gain study points through viewing files
- Manage their profiles
- View dashboard
- Leave feedback

1.4.4 Project objectives

The main reason to create this platform is to make sure that students no longer struggle throughout their academic journey and to help them start building their path into the professional life .They will have a simple but efficient interface that allows them to manage documents and posts . To conclude this part , our platform is going to :

- Ease the learning process of the student's , motivate them and boost their performance.
- Maximise knowledge about professional life .
- Provide meaningful insights and analysis .

1.5 Work Methodology Choice

Within the domain of web development , a work methodology is required to solve issues related to project structuring , planning and for an enhanced control . To make sure we're choosing the right methodology , we're going to compare two different ones and pick the one that aligns with our project objectives the most .

1.5.1 Work Methodology Comparison

To avoid a bad project outcome , our goal is to make a wise choice between the two main project management methodologies . We have to chose an efficient one that offers us the best approach for better project management .
In the context of decision making we begin by analysing each methodology :

- **Waterfall :** Striking with it's traditional vibe until the early 2000. It resembles the waterfall where each step needs to be done in order to pass to the next one . It's drawbacks are contact loss and difficulty to make changes in the project .As soon as the project is delivered the contact with our precious client is lost And in the case where this latter want to make a change in the project it's pretty much impossible to include it unless it

was done from the very beginning which leaves our clients unsatisfied with our work and can lead to an increase in costs followed by delivery delays .

- **Agile Approach :** Elegantly situated on the top of all approaches , agile method offers an unbeatable flexibility going by a principle that says " Clients first " . The goal here is to make sure that you have captivated the client and made them satisfied with the project output . The contrast between this approach and the waterfall approach is that in the Agile method we make sure to include the client in the development process .We divide the project into mini projects and take the client's opinion after each one is done and not at the end of the development process like the waterfall approach.

  To warp it up here's a table that summarizes the comparison between both methodologies .

Methodology	Waterfall	Scrum
Project Type	Linear	Iterative and incremental
Flexibility Level	Low	High
Client involvement	minimal until delivery	Continuous feedback
Testing	At the end of the project	Continuous testing
Choice	For simple and well defined projects	For complex and evolving projects

TABLE 1.1 – Comparison between Waterfall and Scrum

The project we aim to make is not likely to have a final version since we can keep adding more creative use cases throughout the years .After exploring the impact that the waterfall methodology can have on our projects and the limited flexibility the decision is to opt for SCRUM as our work methodology .Let's commence by understanding this method .

1.5.2 SCRUM methodology introduction

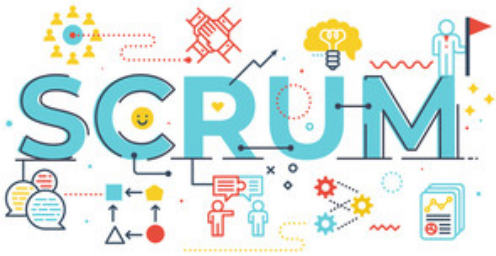


FIGURE 1.3 – SCRUM

Infused with a touch of characteristics that makes it better than the rest , " SCRUM " is one of the most popular Agile methods . It is mainly used in the filed of software development . As a proof of it's goodness we can specify some of it's advantages :

- **Valuable deliveries :** With " Clients first " in mind and " Team spirit " in heart , the team looks forward to deliver high-value products that meets it's clients needs with adaptability to market conditions .
- **Roles and Responsibilities association :** assigning roles and responsibilities offers a greater transparency and surrounds the working environment with an air of accountability.
- **Daily stand-ups :** What's a greater solution to solve problems and do a follow up on the progress that to communicate on a deeper level with the team ? These daily stand-ups are going to promote communication within the team .
- **Priority update :** Since not all of us are good at taking a clear decision from the beginning , Scrum helps us in this matter by allowing the client to define or change the priority order later in the development process .

1.5.3 Scrum Tools

Backlog	To avoid getting overwhelmed we use this tool to define in the form of a table the list of user stories the web application has to offer .There are two types a **Product Backlog** that holds all the project features and a **Sprint Backlog** that's a subset of the product backlog that holds the sprint's features we want to focus on and develop .
Scrum Board	A tool that touches the organizational spirit within the team . It is basically a visual representation divided into three parts **" To DO "** , **" Doing "** and **" Done "**. Since it's real-time there won't be any confusion or task repetition because we can visualise the whole progress in one place .
Scrum Burn Down Chart	It is a graph that contains the work done and how much it took to do it versus how much we estimated it will take . The challenge is to work hard enough until we see the graph line land to zero which means we succeeded our mission and got all the work done .

TABLE 1.2 – Scrum Tools

1.5.4 Design Language

Designing with UML offers clear visual representations of the system's structure and the different interactions within it .The design also varies from high level at the beginning of the project to detailed ones each sprint . That's why it's the perfect match for this project because it will represent with it's extended range of diagrams the static and dynamic aspect of the system in a simplified way .

1.6 Conclusion

Bringing the discussion to a close , our exploration has led us to get to know the Internship placement where within it we introduced the host organization , the problem faced and it's

solution . On top of that we specified the work methodology and design language that we'll use as our base in the next chapters . The journey continues where we'll come out with a plan for our project and we'll dive deeper into the web application's features and understand them .

Chapitre

2 Sprint 0 :Needs specification and analysis

Plan

2.1 Introduction

Starting our preparatory phase that sets the ground we will build on it our project .In this sprint we will be initiating the discussion of our web application precisely by specifying the different types of needs , designing globally our application , shedding insight on the product backlog and planning our sprints , we will also understand the physical architecture , highlight the work environment and finish with a deployment diagram to illustrate the deployment of software onto hardware .

2.2 Needs Specification

Keeping the objective of ensuring that your team understands what they're expected to do in mind .The needs specification phase is the process of identifying and documenting your clients needs , this includes the list of features and functionalities as well as the constraints .

2.2.1 Actors Identification

It is best to describe him as an external entity that interacts with our system it can be a user or another system . This entity is not in the system but their interactions have an impact on it . Now that we know what an actor is , let's move on to present our application's actors in the form of a table :

Actor	Description
User	Responsible for managing documents , Posts ,Tips and viewing his dashboard .
Admin	Responsible for managing users , documents , posts and tips .

TABLE 2.1 – Actors table

2.2.2 Functional Requirements

At the outset of each web application , we have a bunch of steps to follow , one of them being defining for each actor the features that the system should provide for them .

Feature	Description
Manage documents	Each actor will be able to add ,search , view , update and delete documents.
View and download files	the user can , as a part of the view documents feature , to open the files that the system supports and download them , they can also download unsupported file types such as word documents , power points , images etc ...
Gain study-points	the user gains a point each time they open , download or add a file as a way of motivation .
Manage Tips	another thing the actor can do is adding ,searching , viewing , updating and deleting tips they share with others .
Manage posts	each actor will be able to add ,search , view , update and delete internship experience posts .
View Dashboard	The student can view the visualisation of data on the platform.
Manage profile	In case the user needs to update a field previously filled when signing in or to change their password a manage profile page is necessary .
Leave feedback	the user's opinion is very important to keep upgrading the platform for the better , that's why there's a form that the user can fill to give their feedback .The admin will then check them and make the necessary steps to fix issues .

TABLE 2.2 – Functional requirements

2.2.3 Non Functional Requirements

Feature	Description
Security	It's really important to protect user's data to make them feel more secure and comfortable using the platform .
Regulatory Compliance	Education norms must be respected , and uploaded documents must be checked to ensure they don't provide falsified information .
User Support	Including a chat-bot can be really helpful for users to lessen the need to keep leaving the web application searching for something .
Session management	To protect users against unauthorized access by logging them out automatically when the session is over (Session Time-Out).
Usability	By providing a clear navigation bar and use friendly interfaces the web application becomes more usefull and easy to undeerstand .

TABLE 2.3 – Non functional requirements

2.2.4 Decision Requirements

Decision Requirements represents a crucial part of the Business Intelligence domain since With the help of the statistics and dashboards provided the user can take better decisions . A good way the user can know that they have studied well is by viewing how many documents they have viewed and downloaded to view later , how much time they have spent studying that document . We can tell that the project has done it's job when we see the user's opinion on the platform .

2.3 Project Management with Scrum

it's now time to implement the methodology we chose in preliminary study , we will define the scrum team that will work on this project and the product backlog .

2.3.1 Roles in SCRUM

To elicit the sense of a systematic environment and head to achieve better results we should start by identifying each role within the team . It is known that bad team composition can lead to severe outcomes some of them are :

- **Being stuck in progress :** it's happens when the team lacks the skills needed to finish the project , for this exact reason we have to be more thoughtful when picking the members of each project.
- **Bad role assignment :** it's when someone who's not skilled enough to hold the position they're at .
- **Missing deadlines :** each project has a duration .Both outcomes mentioned above can cause delivery delays therefore missing deadlines .

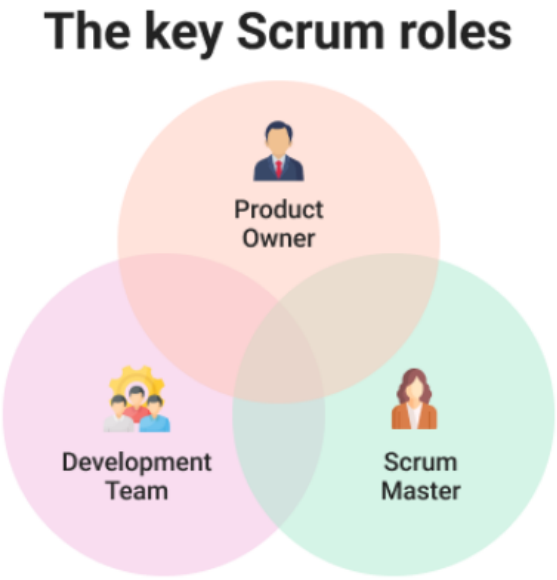


FIGURE 2.1 – Roles in SCRUM

As mentioned in the picture above we can conclude that there are three key roles in scrum and we're going to assign a role for each one of the people included in this project :

Role	Name	Description
Product Owner	Ms Kalthoum Boujlel	Responsible for the product backlog and priority order definition ,and gives their feedback after each sprint .
Scrum Master	Mr Hamdi Hassen	This latter acts as the bridge between the product owner and the development team where his role is to ensure that scrum methodology is well understood and followed .
Development team	Hadhami Abidi	Concentrating on the aspect of delivering a high quality product , the team holds the responsibility of getting the work done and delivering the product .

TABLE 2.4 – Roles in SCRUM

2.3.2 Global Use Case Diagram

Being the heart of development projects , and the map we get to follow it's roads . We are now going to focus on globally designing the use case diagram of our web application . In UML , this diagram is a visual representation that solves the issues related to understanding who interacts with our system and what are the features that they can use . It has 4 key elements : actors , Use cases , associations and system boundaries .

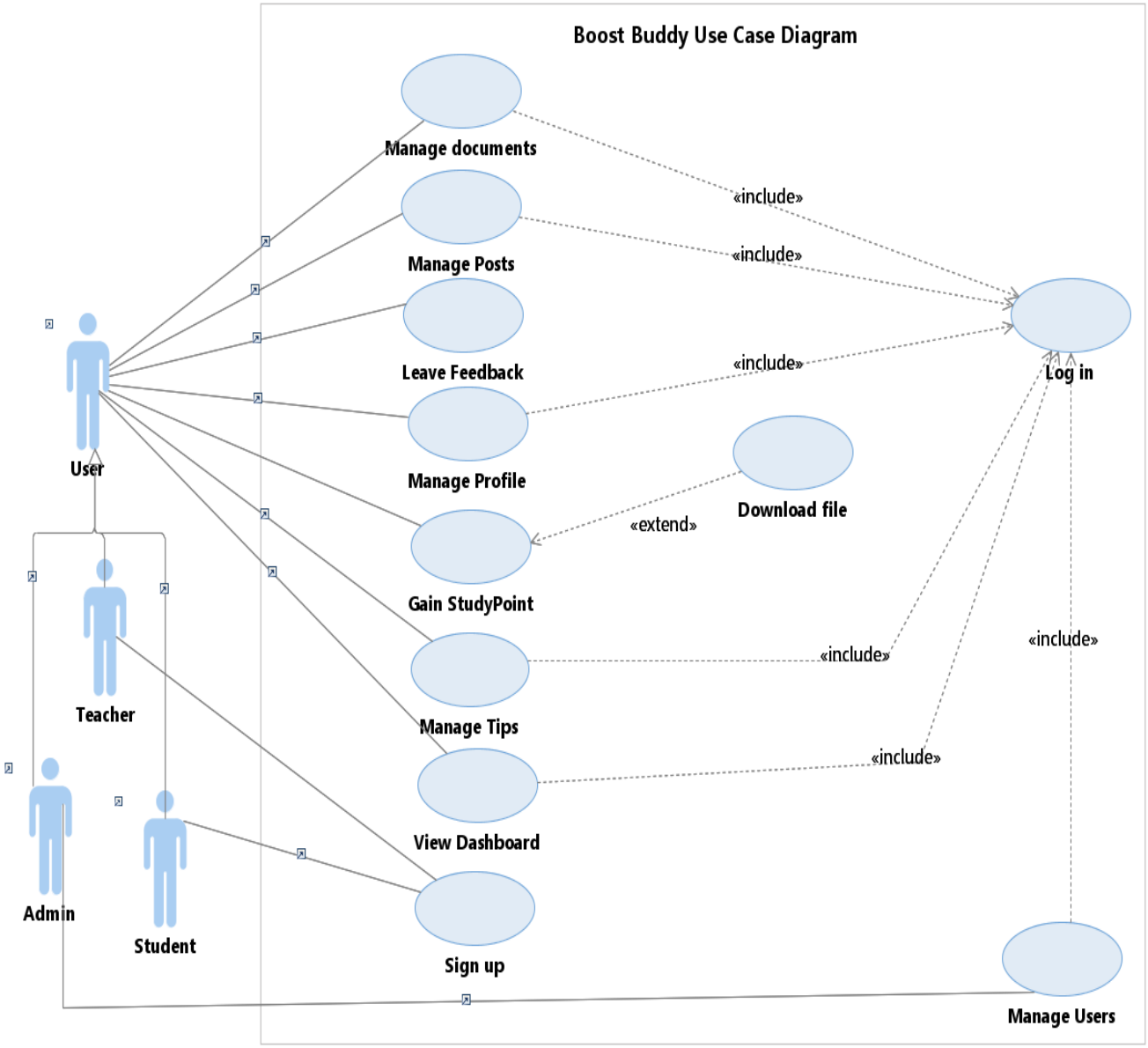


FIGURE 2.2 – Global Use Case Diagram

2.4 Use Case Processing Planning

- **Priorities :**As the quote says " No project unfolds exactly as planned " which means It's challenging to establish the final planning as we forecasted since circumstances can happen however what's important is to specify the priority order of each use case to know what to work on first .

- **Risks :** In order to set the stage for a successful project management , it's crucial to take some time and identify the potential risks that could lead to project failure . In our project , it's the project's complexity compared to the available time .

2.5 Product Backlog

Before we jump into the process of planning, as the methodology suggests , we will commence by gathering the outcome of our meetings with the client and address the features they asked for , then we compact all these features in one place called " The Product Backlog " . This latter has for each feature it's priority to help plan the work and the estimated complexity level .

ID	Feature	Priority	Complexity Level
1	Sign Up	1	Moderate
2	Log In	1	Moderate
3	Manage Users	1	Moderate
4	Manage Document	2	Complex
5	Gain study points	2	Easy
6	Manage Tips	2	Complex
7	Manage Posts	3	Complex
8	Manage Profile	3	Moderate
9	Leave feedback	3	Moderate
10	View Dashboard	4	Hard

TABLE 2.5 – Product Backlog

Now that we're done creating the product backlog we can initiate the discussion of a road map to success by carefully planning what use cases we'll be dealing with in each sprint .

2.6 Sprints Planning

FIGURE 2.3 – Sprints Planning

2.7 Work Environment

The notable achievements we will acquire once we're done with this part is a finer idea on both parts used in the development process , the hardware and the software .

2.7.1 Hardware

The table below covers the description for the specs of the machine used for the development during the project .

Owner	Hadhami Abidi
PC Brand	Asus
Processor	Intel Core i7-8750H
RAM	16 Go
Hard Disk	255 GB SSD
Operating System	Windows 11

TABLE 2.6 – Development machine specs

2.7.2 Software

Since the project requires a development process we need a software tools combination to get us through until we achieve our goal . Let's initiate the exploration of these softwares.

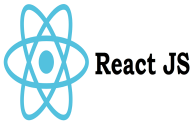
	For the backend of my application I've decided to opt for one of the most famous and powerful Python frameworks " Django " .Not only it is open source but it also follows the "Batteries Included " philosophy which implies that within it exists a lot of built-in features like an admin interface , authentication , security algorithms ... With the help of these features the development time is less and we're able to build a robust web application .
	One of the tools that when combined with django backend is React JS as they offer "Best of both worlds" .React is developed by facebook and uses a virtual DOM (Document Object Model) that helps it update faster making the application more efficient .React is component based ,that means we can reuse the code in different part of out application
	PostgreSQL is an open source and free relational database system .It offers advanced features like full-text search and complex queries .PostgreSQL is also known for it's scalability making it suitable for large projects .It integrates very well with Django framework making it a popular choice due to it's compatibility.
	It has a french name that means quick , it is a server to ease the development process and make it faster of front-end applications built with Java-Script frameworks or libraries such as React , Vue.js , Angular .....
	Visual Studio Code is an open source code editor developed by Microsoft .It provides a vast and powerful environment for coding with a support for all programming languages and a rich ecosystem of extensions that makes the working process a lot better .
	Trello is an Agile project management tool , the reason behind choosing to work with it Comme outilde gestiondeprojet agile, nous avonsutiliséTrelloquiestuneapplicationde gestion de projetgratuite permettant d'organiser des projets sous forme de tableaux composés delistes en colonnes, qui répertorient des tâches sous forme de cartes.
	It's an Integrated Development Environment (IDE) for both developers and architects based on UML diagrams to facilitate modeling complex systems .RSA also integrates easily with a vast range of tools , supports code generation to granting the ability to create end-to-end softwares .

TABLE 2.7 – Development environment

2.8 Physical Architecture

The part where we explore and investigate an application's components is called " Physical Architecture " . In the heart of our application is situated a user , a server and a database . We should be able to understand better how it looks like by observing the figure below .

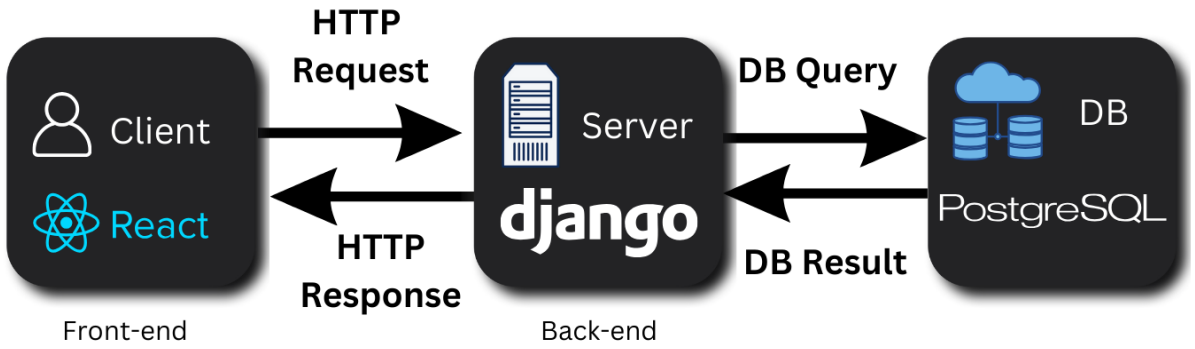


FIGURE 2.4 – Physical Architecture

2.8.1 Architectural Pattern

As we move forward , our focus now shifts to the architectural pattern which entails a solution for a reccurent architectural problem we face either in the front-end or back-end .

2.8.2 MVC architecture

As the seasons change and the time passes by , MVC architecture (Model-View-Controller) remains the top most used architecture among all for creating web applications . The reason behind that is the 3 parts it's composed of that serves as a guide for a better project structure . Moving on to addressing what's each part's responsibilities :

- **Model :** this component is privileged in contrast to the user with the ability to interact with the data base and to manipulate data . On top of that it can respond to instructions given from the entity responsible of extracting or updating data , the model then follows these instructions precisely .

- **View :** the view entails the communication with the controller in order to receive the retrieved data and to display it to the user in an adequate format , which indicates that this part is what is viewed by the user / what the user can see .
- **Controller :** this component takes control over the management of user interactions , that signifies receiving and handling requests sent by the user such as GET , POST etc ... It manages the communication between the view and the model by supplying the model with instructions when an update or data retrieval is needed and commands the view to perform an update .

2.8.3 Superiority of an MVC architecture[?] :

- **Modularity :**Since each component of the 3 handles a specific application aspect we get a modular ,easily understandable and maintained code .
- **Better collaboration :** The components are distinct thus different teams can work on them at the same time leading to a sped up development process
- **Adaptability to change :** we're able thanks to this architecture to make a change in one of the components without affecting the others

To strengthen our understanding on how these components in the MVC architecture communicate between eachother and with the user we should take a glance at the figure below .

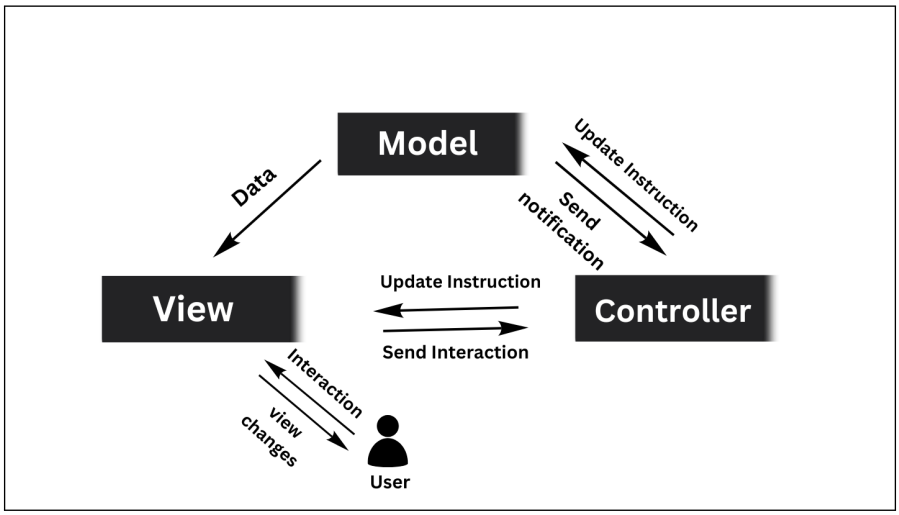


FIGURE 2.5 – MVC Architecture

2.8.4 BI section Architecture

2.8.5 BI tools

Our primary goal is to make our web application's visualisation dynamic and not static , for this particular reason i believe that **Recharts** , a react library , is a good option to choose considering that it's made specifically for React and how well it works with the tools used to develop our web application .

2.9 Conculsion

Ascertaining the outcome of this chapter , we identified the actors ,the functional and non functional requirements along with the decision requirements . After that we drew the lines on our project's global use case diagram and class diagram .Then we shed the light on the product backlog followed by planning our sprints and presented our Data Base schema .Successively we came to an end with precising and explaining the physical architecture and the development environment that we're going to use .We can now proudly announce that we took the first steps and it's now time to dive deeper into the project and unveil in each chapter a feature .

Chapitre

3 Sprint 1 :Log-In , Sign-Up , Manage Users

Plan

3.1 Introduction

As we commence the first sprint , we should get to know better what's a sprint . Having a name inspired from the world of sports since in both fields , a sprint is an iteration that has a duration and an objective . In scrum we choose features from the " Product Backlog " to develop in each sprint and deliver a significant part of the project at the end of it and most notably we can't move to the next sprint unless we're done with the current one as that's against the " iterative and incremental " concept of scrum . I've decided to start with a gentle breeze while i get used to the tools i will be using in the project .In this sprint I'll be accomplishing two basic yet very important features Sign up and Log In along with Users management .

3.2 Sprint Backlog

In this matter , we present in a table the user and his user story along with the priority order. A user story consists of describing a feature from the user's point of view , it has three key elements : a role , an action and a benefit . Here's an outline of the table :

Feature	**User Story**	**Priority**	**Estimated Duration**
Sign Up	As a user I want to be able to sign up so that I can access the platform	1	4
Log In	As a user I want to be able to log in so that I can access the platform	1	2
ADD Users	As an Admin I want to be able to add users	2	2
View Users	As an Admin I want to be able to view users who registered on the platform	2	1
Delete Users	As an Admin I want to be able to delete users from the platform	2	2

Search users	As an admin I want to be able to search for users	3	1
Update user information	As an admin I want to be able to update users information	3	2

TABLE 3.1 – Sprint 1 Backlog

3.3 Functional specification

3.3.1 Sprint 1 Use Case Diagram

In the figure below we demonstrate the detailed use case diagram of this sprint that we'll later move on to writing a specification for each of the use cases shown in the diagram .

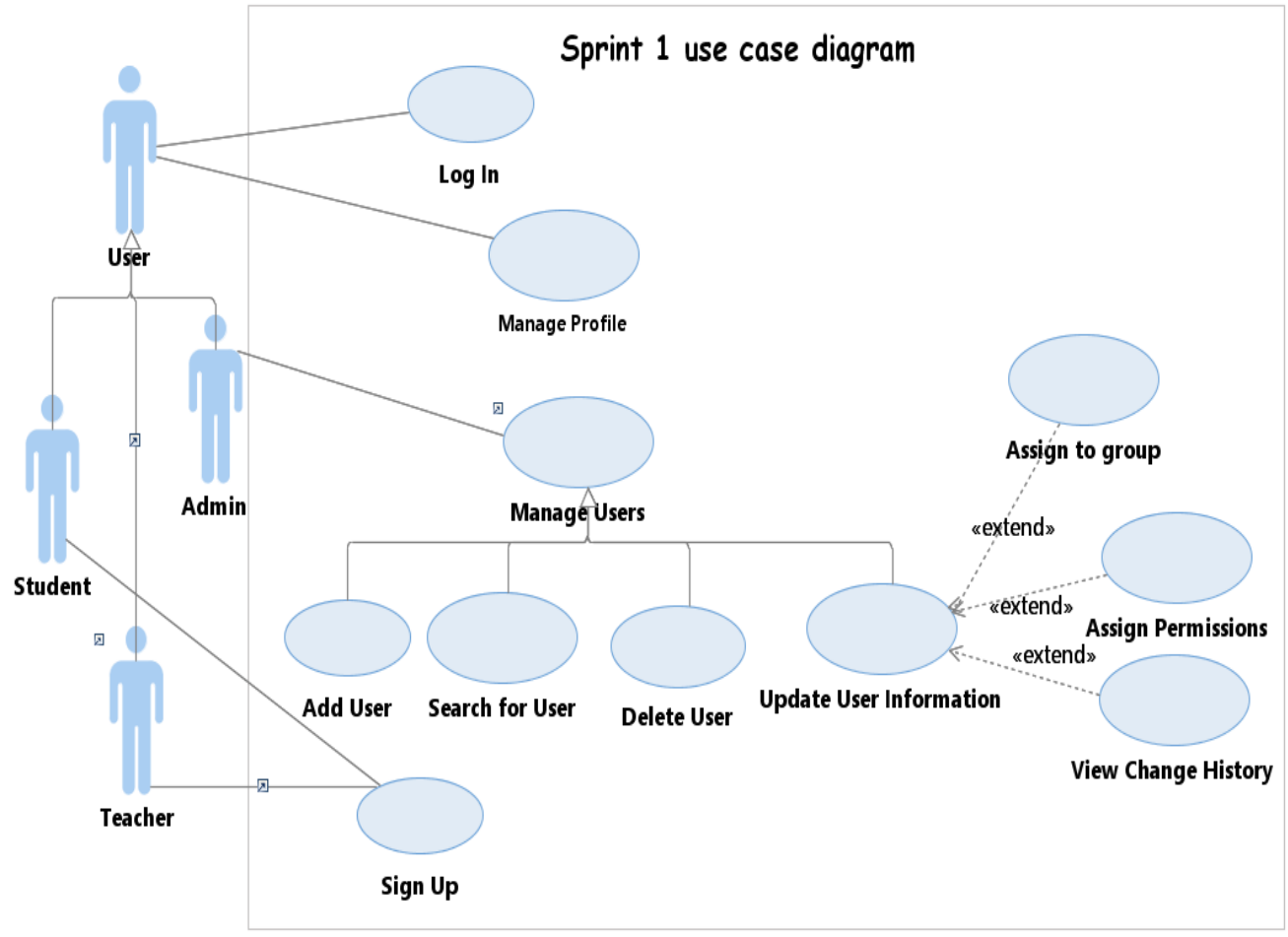


FIGURE 3.1 – Sprint 1 Use Case Diagram

Now that we've represented our use cases for this sprint in the diagram it's time to detail each one of them for an enhanced comprehension .

3.3.2 Use Case « Sign Up » Textual Description

Use Case	Sign Up
Actor	User
Pre-condition	The user is on the sign up page
Post-condition	New user added to the system
Main Scenario	1.The user selects the sign up option . 2. The system displays the sign up form . 3.The user fills the form . 4. The system verifies the user's entries. 5. The system saves the data .
Alternative Scenario	4.a. User data already exists 1- The system displays the message " user with this data already exists ". 2- The system goes back to step 3 .

TABLE 3.2 – Use Case « Sign Up » Textual Description

3.3.3 Use Case « Log In » Textual Description

Use Case	Log In
Actor	User (Student ,teacher , Admin)
Pre-condition	- The user is on the log in page . - The user has an account
Post-condition	User is authenticated
Main Scenario	1. The user selects the log in page. 2. The system displays the log in form. 3. The user fills the form with his credentials. 4. The user clicks on the log in button. 5. The system verifies the user's credentials . 6. The system redirects the user to the home page.
Alternative Scenario	5.a. Wrong credentials entered : 1- The system shows an error message 2-The system goes back to step 2

TABLE 3.3 – Use Case « Log In » Textual Description

3.3.4 Use Case « Manage Users » Textual Description

3.3.4.1 Use Case « ADD User » Textual Description

Use Case	ADD User
Actor	Admin
Pre-condition	The admin is authenticated
Post-condition	New User added
Main Scenario	1. The admin selects the user's list. 2. The system displays the list of users . 3. The admin selects the add User button . 4. The system displays the add user form. 5. The admin fills the form . 6. The system verifies the data . 7. The system saves the data .
Alternative Scenario	7.a. user data already exists : 1- The system shows an error message 7.b. The data format is not valid : 1- The system displays an error message . 2- The system goes back to step 5.

TABLE 3.4 – Use Case « ADD user » Textual Description

3.3.4.2 Use Case « Delete User » Textual Description

Use Case	Delete User
Actor	Admin
Pre-condition	- The admin is authenticated . - Number of users >=1
Post-condition	User deleted
Main Scenario	1. The admin selects users list. 2. The system displays the user's list . 3. The admin selects the User to delete . 4. The admin clicks on delete . 5. The system displays a confirmation message . 6. The admin confirms the deletion . 7. The system deletes the user . 8. The system updates the user's list .
Alternative Scenario	6.a. The admin cancels the deletion : 1- The system goes back to step 2

TABLE 3.5 – Use Case « Delete user » Textual Description

3.3.4.3 Use Case « Search for User » Textual Description

Use Case	Search for Users
Actor	Admin
Pre-condition	The admin is authenticated
Post-condition	Search results displayed
Main Scenario	1. The admin selects users list. 2. The system displays the user's list . 3. The admin types the desired name in the search bar. 4. The system searches for the user. 5. The system displays the search results .
Alternative Scenario	5.a. User doesn't exists : 1- The system displays a message "0 users" .

TABLE 3.6 – Use Case « Search for user » Textual Description

3.3.4.4 Use Case « Update User » Textual Description

Use Case	Update User
Actor	Admin
Pre-condition	- The admin is authenticated . -User exists
Post-condition	User information updated
Main Scenario	1. The admin selects users list. 2. The system displays the user's list . 3. The admin selects the desired user form the list . 4. The system displays the user's information . 5. The admin updates the desired information . 6. The admin clicks on save . 7. The system saves the changes.
Alternative Scenario	3.a. No accounts to update : 1- The system shows an empty list 7.a. The admin doesn't save the changes : 1- The system doesn't save the changes .

TABLE 3.7 – Use Case « Update user » Textual Description

3.3.4.5 Use Case « View User » Textual Description

Use Case	View User
Actor	Admin
Pre-condition	- The admin is authenticated . -User exists
Post-condition	User information displayed
Main Scenario	1. The admin selects users list. 2. The system displays the user's list . 3. The admin selects the desired User . 4. The system displays the user's information .
Alternative Scenario	6.a. No users in the system : 1- The system displays an empty list .

TABLE 3.8 – Use Case « View user » Textual Description

3.3.5 Use Case « Manage Profile » Textual Description

Use Case	Manage Profile
Actor	User
Pre-condition	- The user is authenticate.
Post-condition	User information updated
Main Scenario	1. The user selects the manage profile page. 2. The system displays the manage profile page. 3. The user updates the desired information. 4. The user clicks on update profile . 5. The system updates the user's information . 6. The system refreshes the page.
Alternative Scenario	2.a. User not logged in : 1- The system displays an empty form

TABLE 3.9 – Use Case « Log In » Textual Description

3.4 Prototypes

Before we delve into developing our web application , we give the client a glimpse on how our application would look like and do a quick check on whether it matches the requirements discussed before , if there's something missing or if they want to add or remove something . Not only does it help the client but also the developers since it gives them a visualization of the application and sets the ground they'll be working on .

FIGURE 3.2 – Sign Up Prototype

FIGURE 3.3 – Log In Prototype

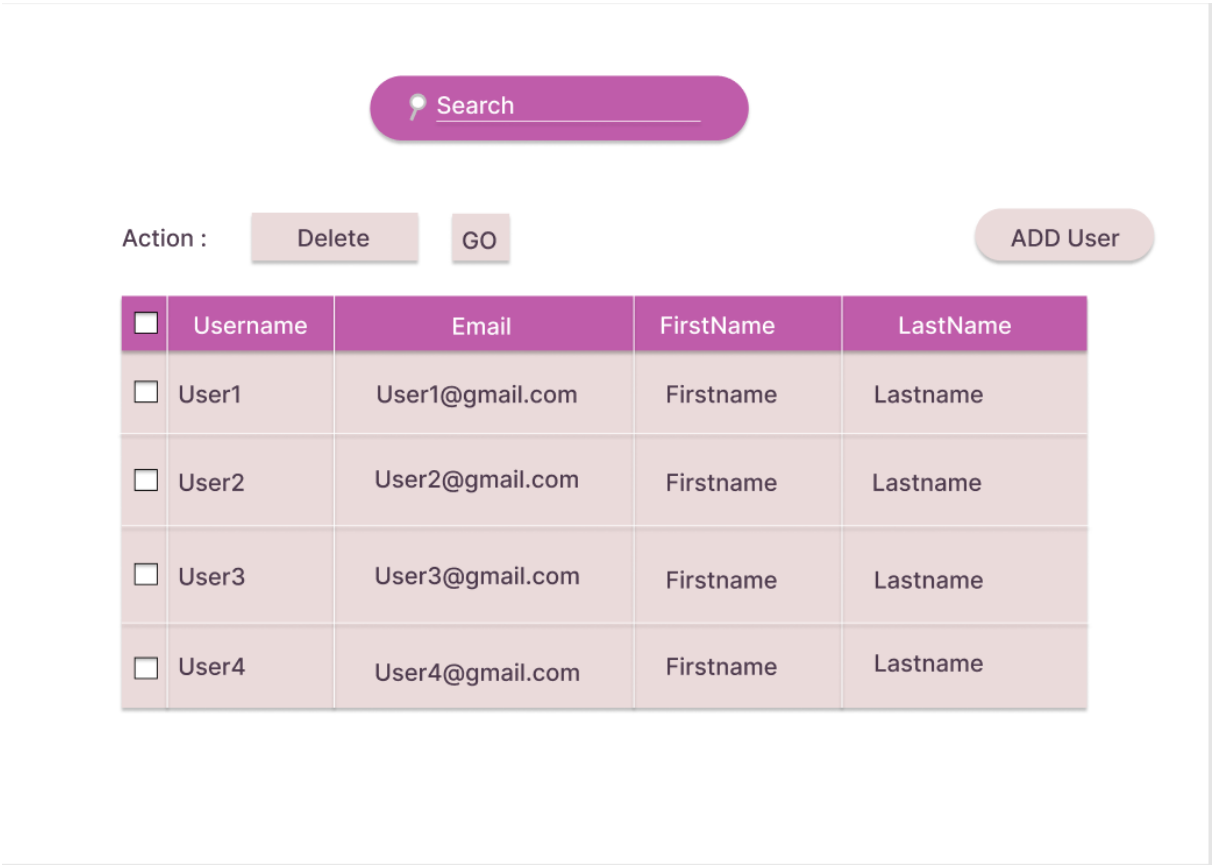


FIGURE 3.4 – Manage Users Prototype

3.5 Design

With the help of UML design we can understand better the system features that we envisioned .Each diagram plays an important role , the class diagram captures the static aspect of the system while the sequence diagram captures the dynamic aspect of it .

3.5.1 Sprint 1 Sequence Diagrams

3.5.1.1 Use Case « Sign Up » Sequence Diagram

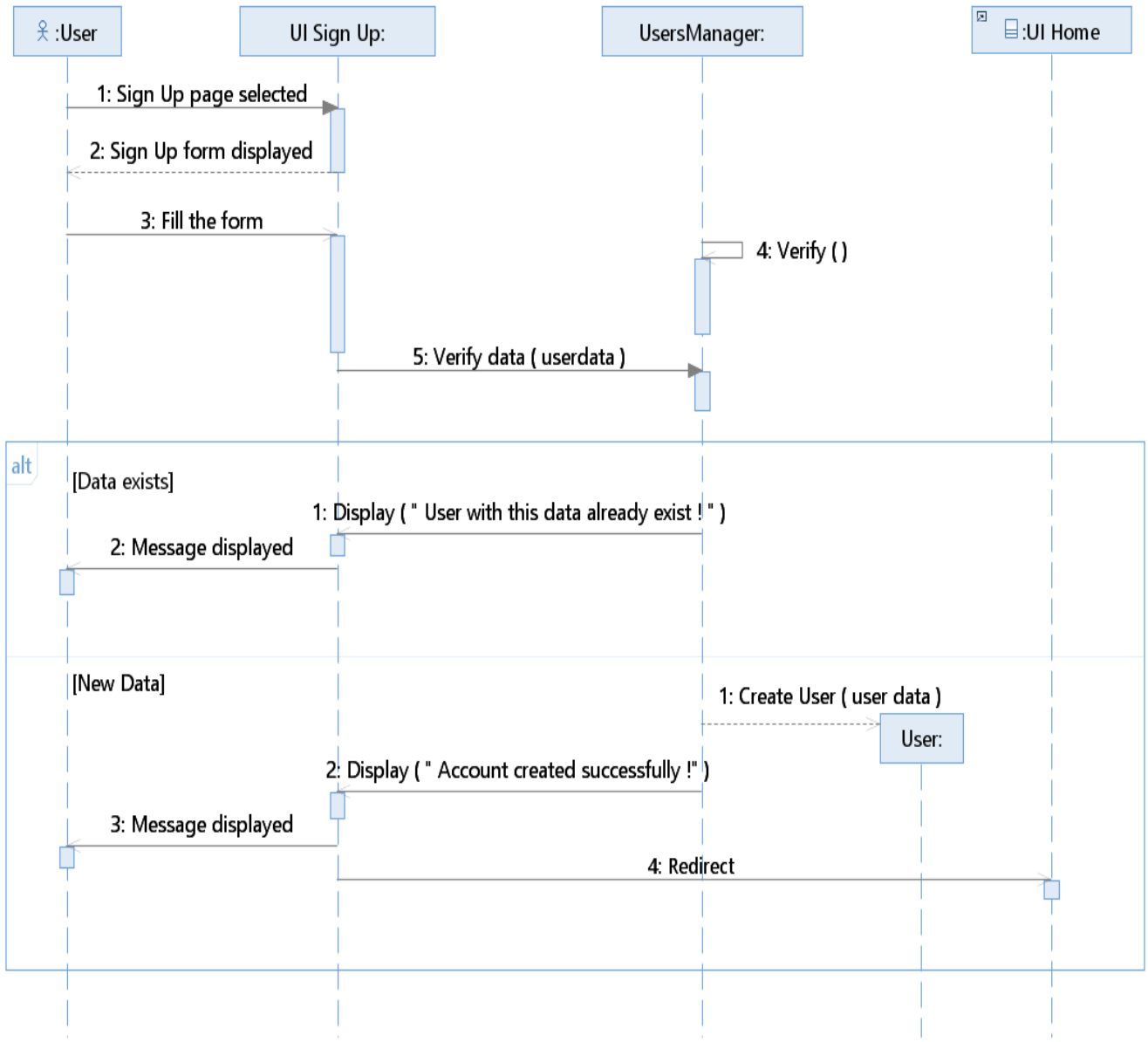


FIGURE 3.5 – Use Case « Sign Up » Sequence Diagram

3.5.1.2 Use Case « Log In » Sequence Diagram

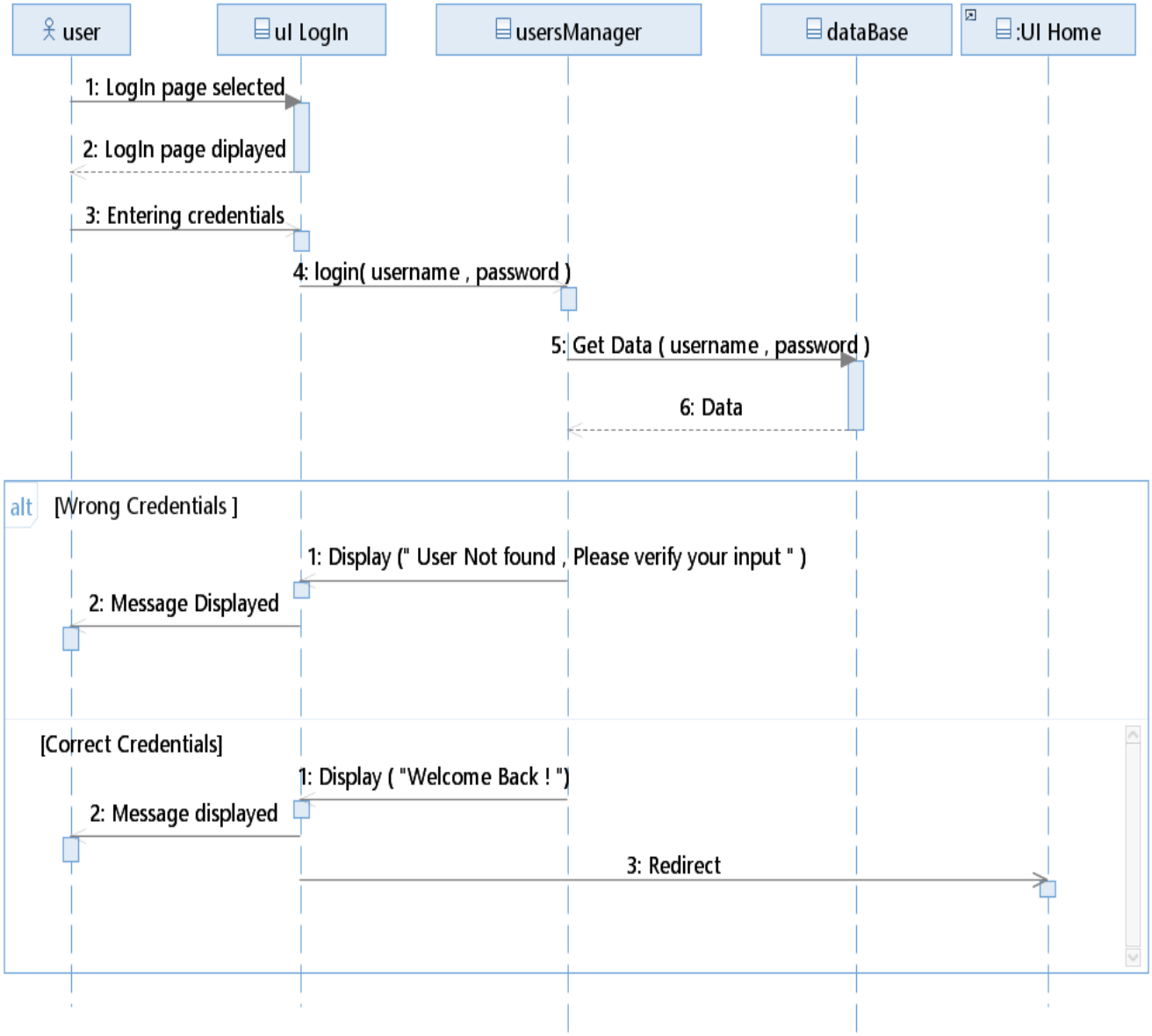


FIGURE 3.6 – Use Case « Log In » Sequence Diagram

3.5.1.3 Use Case « ADD User » Sequence Diagram

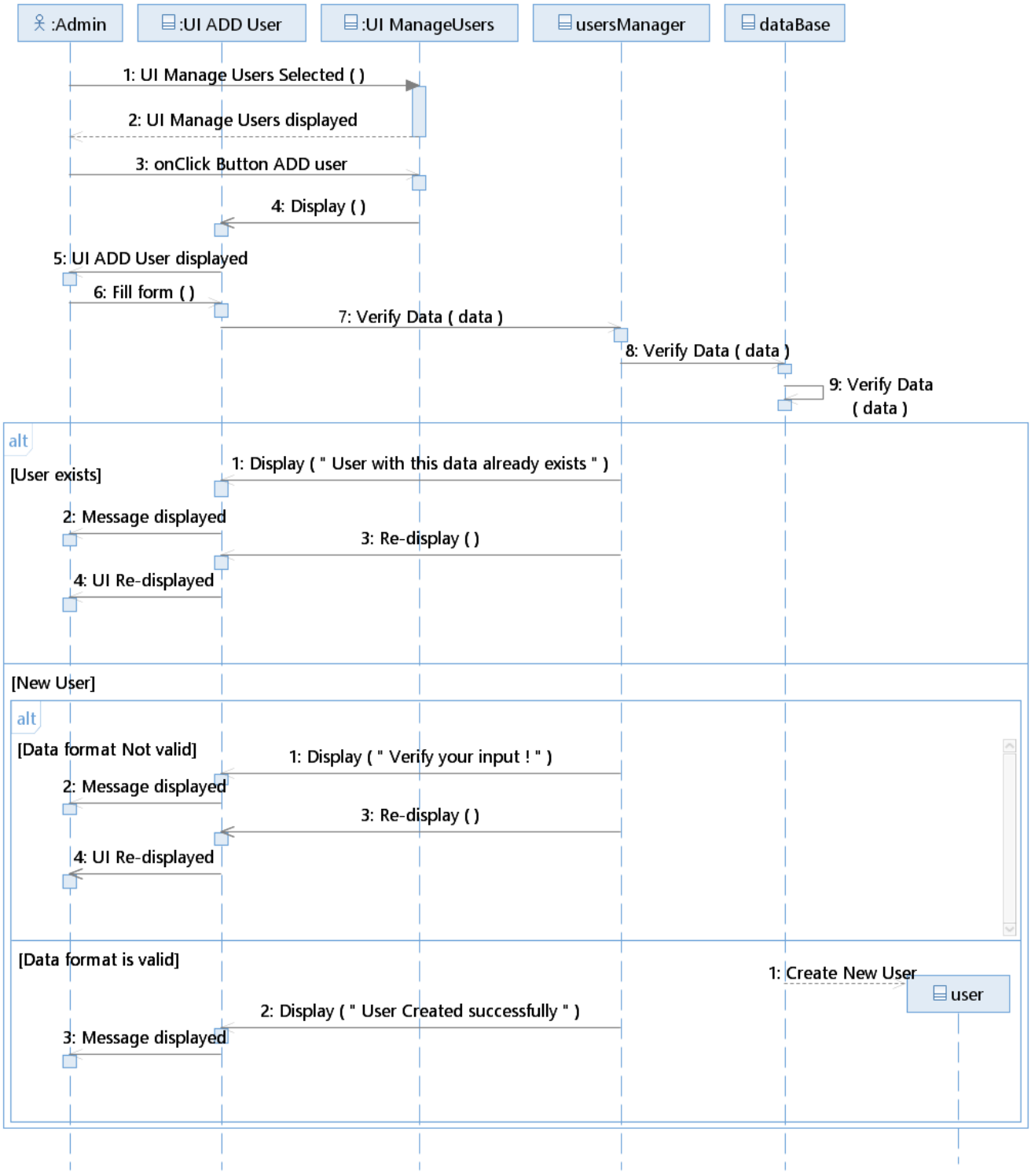


FIGURE 3.7 – Use Case « ADD User » Sequence Diagram

3.5.1.4 Use Case « Delete Users » Sequence Diagram

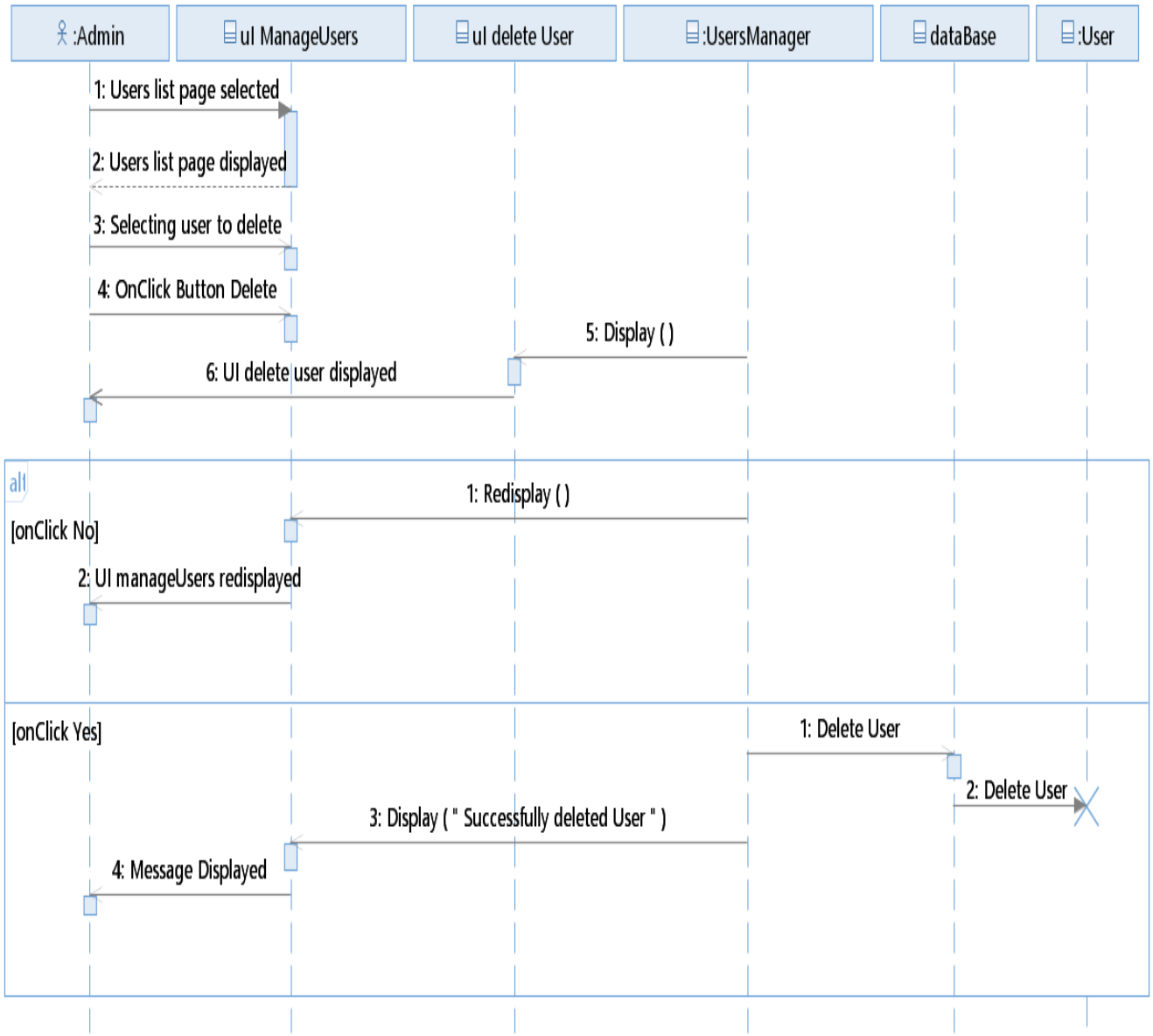


FIGURE 3.8 – Use Case « Delete Users » Sequence Diagram

3.5.1.5 Use Case « Manage Profile » Sequence Diagram

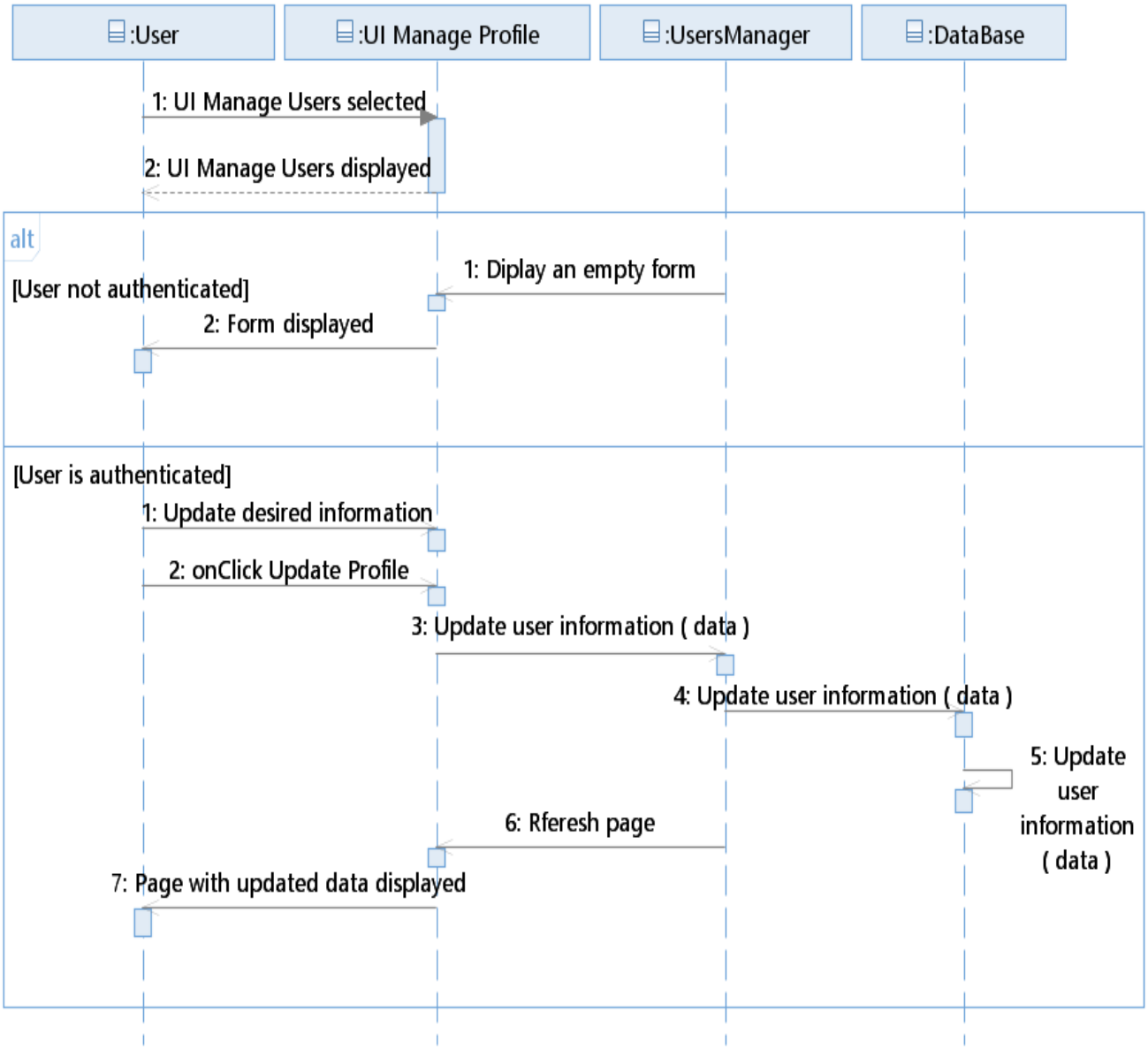


FIGURE 3.9 – Use Case « Manage Profile » Sequence Diagram

3.5.2 Sprint 1 Class Diagram

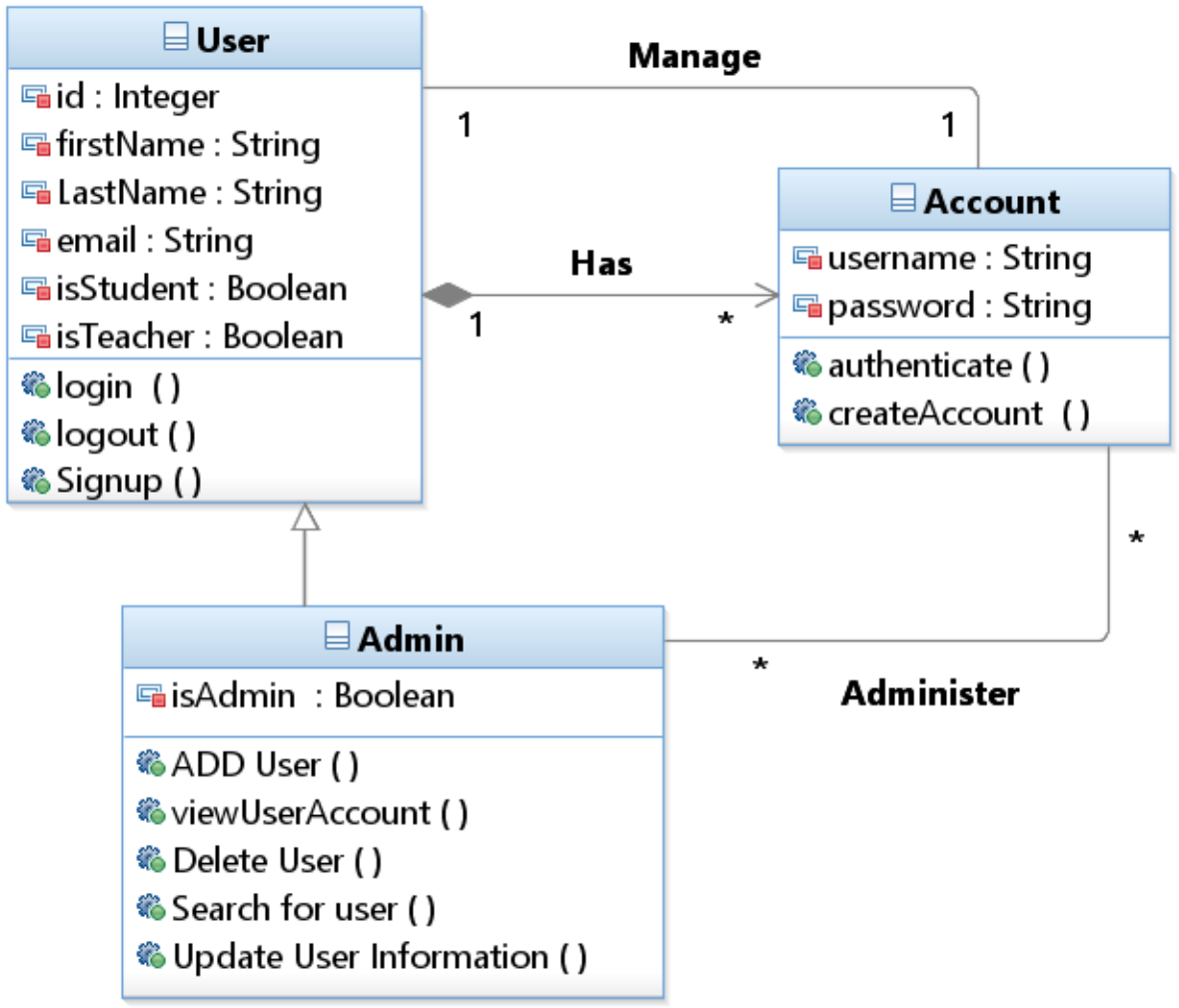


FIGURE 3.10 – Sprint 1 Class Diagram

3.5.3 Sprint 1 Traceability

3.5.3.1 « Log In » Use Case Traceability

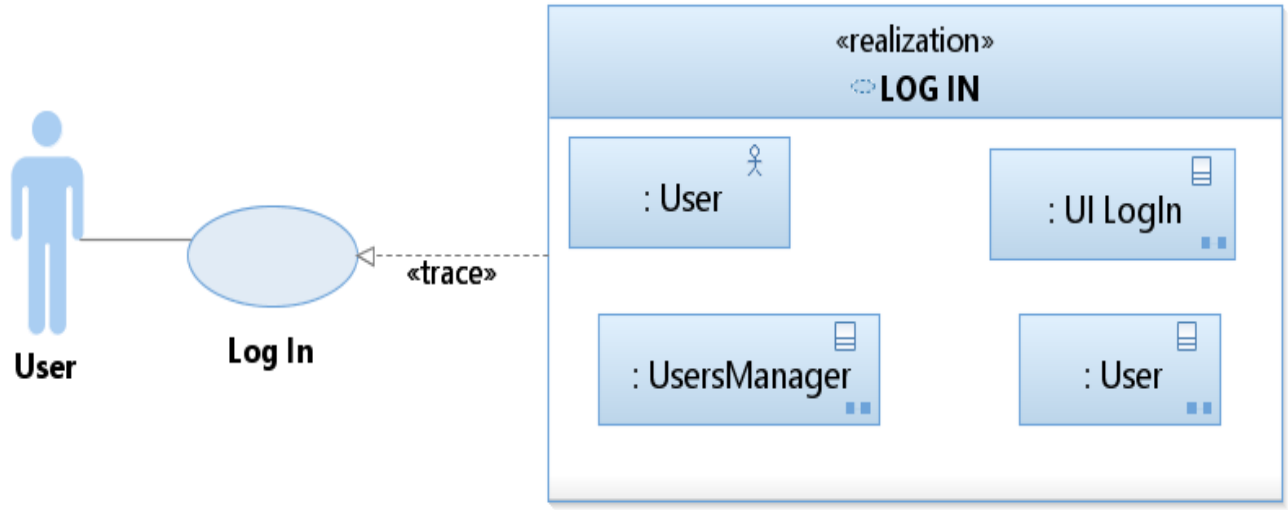


FIGURE 3.11 – « Log In » Use Case Traceability

3.5.3.2 « Sign Up » Use Case Traceability

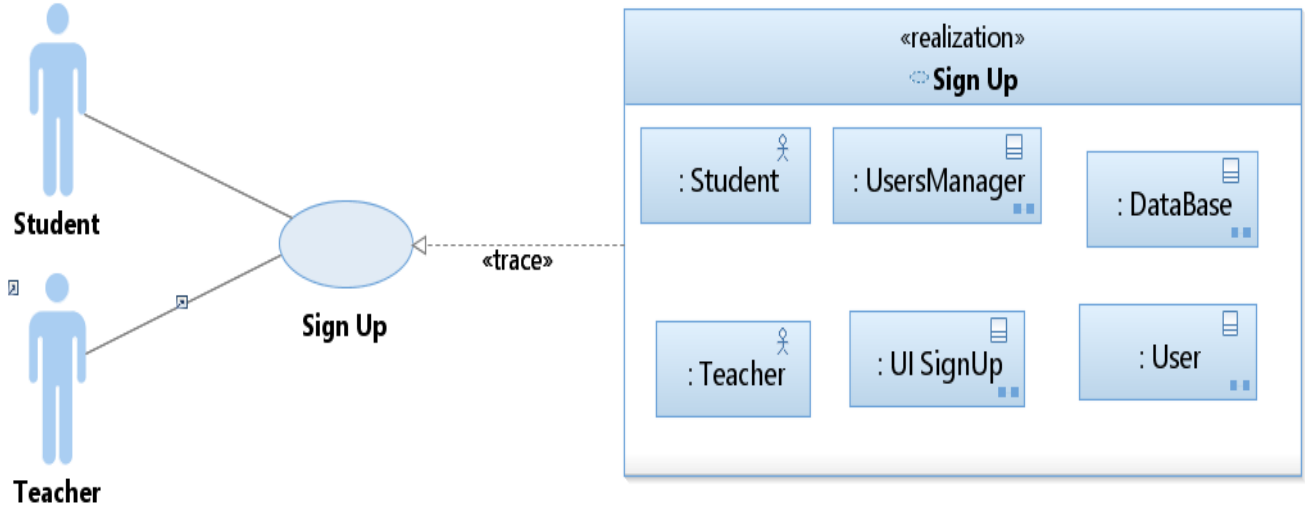


FIGURE 3.12 – « Sign Up » Use Case Traceability

3.5.3.3 « Manage Users » Use Case Traceability

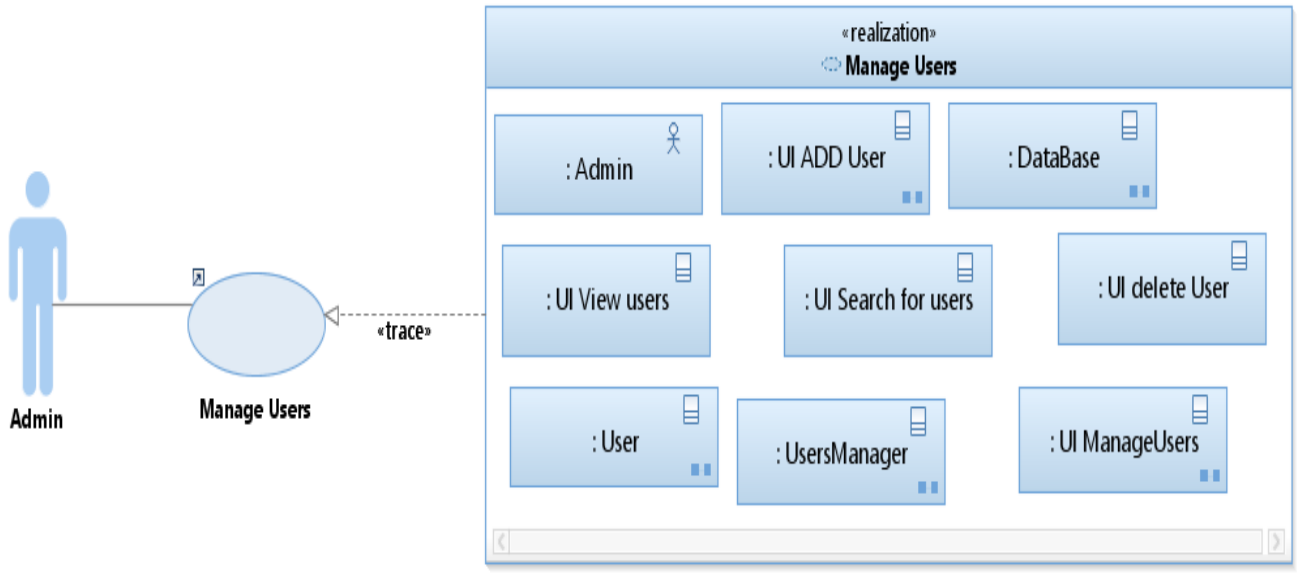


FIGURE 3.13 – « Manage Users » Use Case Traceability

3.6 Implementation and Tests

3.6.1 Sign Up

3.6.2 Log In

3.6.3 Manage Users

3.7 Scrum Tools implementation

3.7.1 Scrum Board

In the course of two weeks and after getting the designs necessary for this sprint done this is what the scrum board , where we specified the list of tasks we have to get done , looked like .

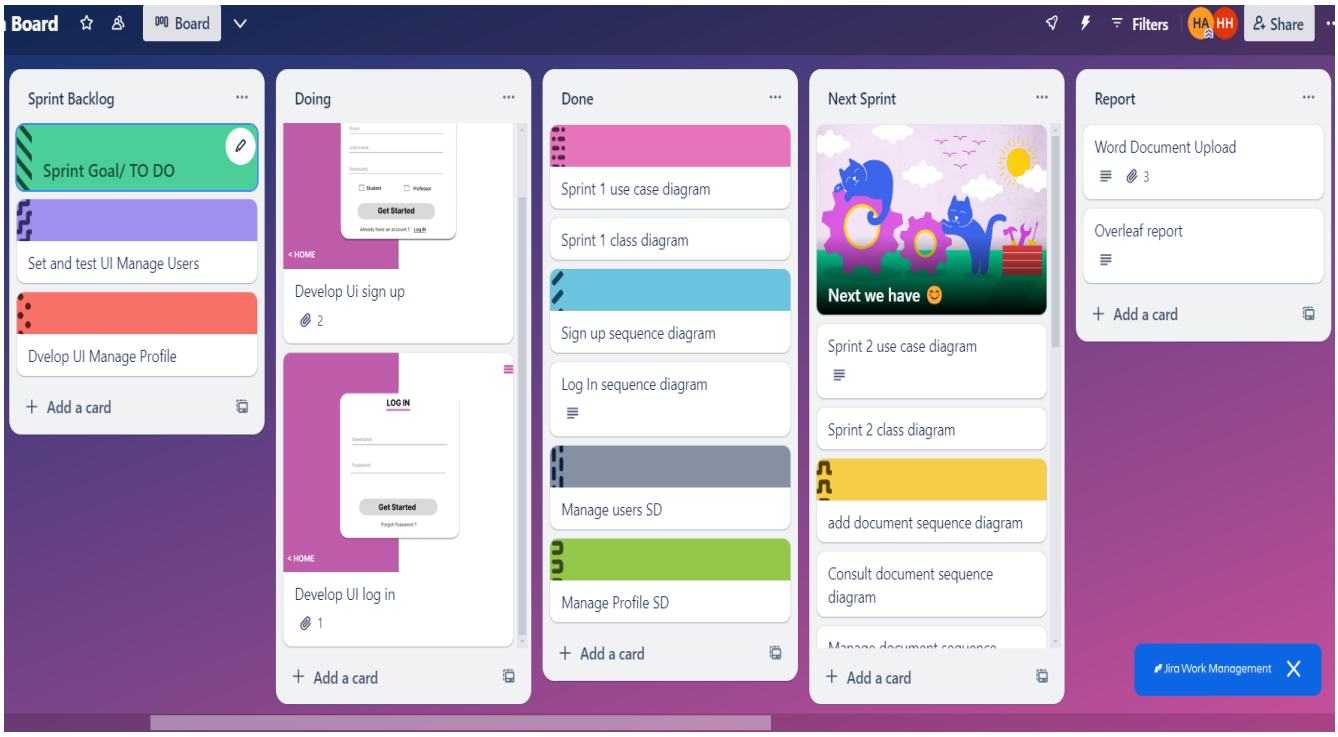


FIGURE 3.14 – Sprint 1 Scrum Board

And this is how it looked like at the end of the first sprint of this project :

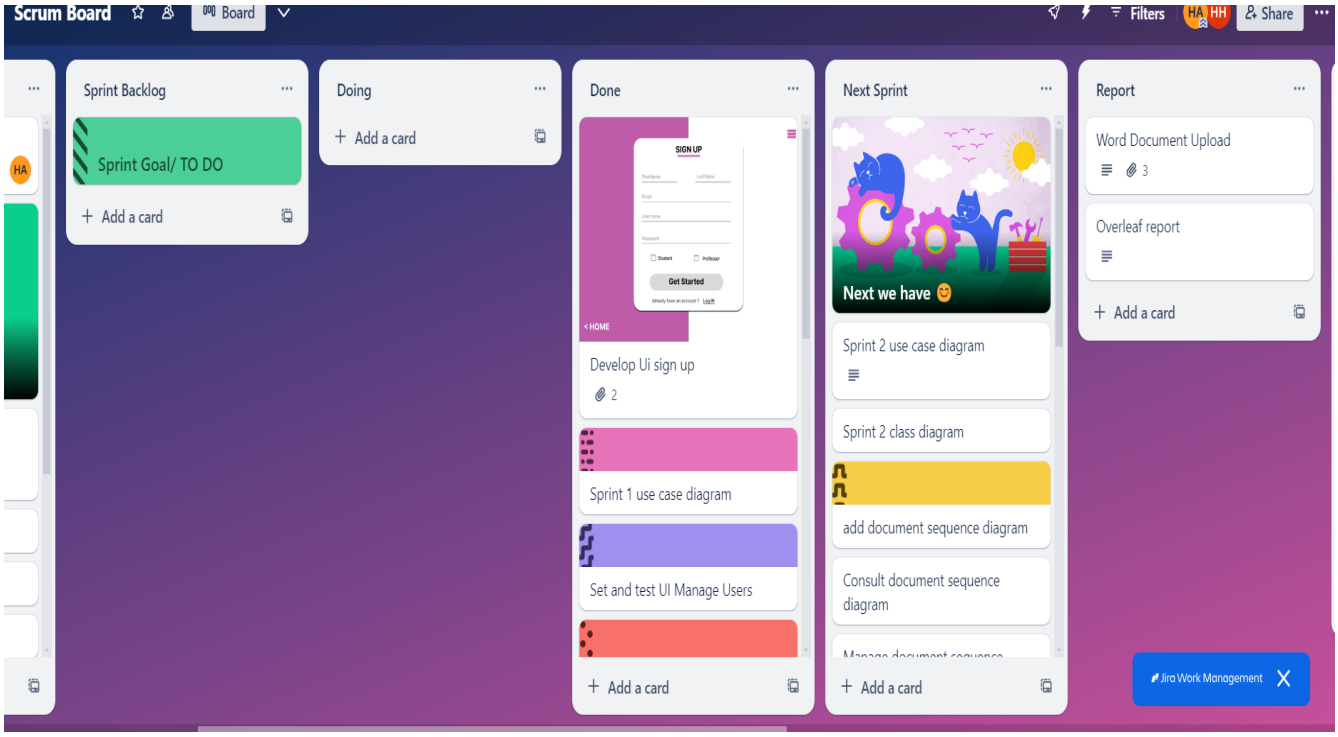


FIGURE 3.15 – Sprint 1 Scrum Board

3.7.2 Scrum Burn-Down Chart

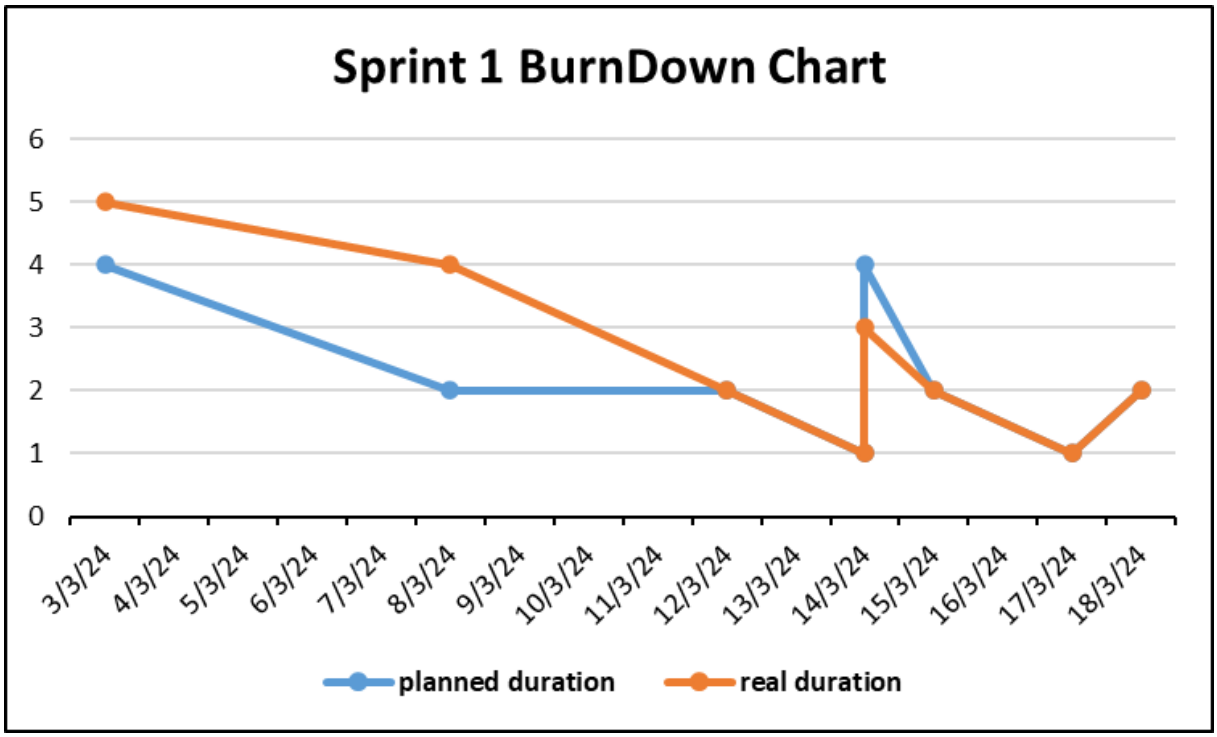


FIGURE 3.16 – Sprint 1 Burn Down Chart

3.8 Sprint Review

3.8.1 Sprint Delivery

Since my plan was to start slow on the development process , this sprint has as an output a login , sign up and manage profile interfaces besides a user management interface for the admin . During this first iteration we're able to create an account , authenticate , and perform different options on users by the admin like add , delete , update ....

3.8.2 Difficulties faced

In the course of working on this sprint , i have faced a couple of difficulties some of them are :

— Technical difficulties seeing that it's my first time using this combination of tools for development it took me some time to learn how to link the front end to the back end and send requests correctly .

— After developing the login page i realized that it only worked in that page but struggled to keep the user logged in through all the application to grant them access to other options .

— It was a little bit challenging to make the manage profile feature because the back end uses a password hashing algorithm .

3.9 Conclusion

I'm pleased to report now that during this sprint we successfully designed and implemented the user stories selected from the product backlog . Next on the plan is the second sprint , more effort is needed as the complexity is higher .

Chapitre

4 Sprint 2 : Manage Documents , Gain Study-Point , Manage Tips

Plan

4.1 Introduction

As we progress , attention will be given to the manage documents feature .Looking forward to achieving tangible progress , this sprint is a key step in the success of our project so let's navigate through it together and embrace it's challenges and opportunities .

4.2 Sprint Backlog

Feature	**User Story**	**Priority**	**Estimated Duration**
Add document	As a user I want to be able to add documents to the platform	1	4
View document	As a user I want to be able to view documents on the platform	1	7
Update document	As a user I want to be able to update documents on the platform	2	4
Delete document	As a user I want to be able to delete documents in the platform	2	3
Search for document	As a user I want to be able to search for documents in the platform	3	2
Gain study point	As a user I want to be able to gain points when i contribute on the platform	3	2
Add tip	As a user I want to be able to tips to the platform	3	2
Delete Tip	As a user I want to be able to delete tips i added on the platform	3	1
Update Tip	As a user I want to be able to update tips i added on the platform	3	2

Search for tip	As a user I want to be able to search for tips about a specific topic	3	1
View tips	As a user I want to be able to view tips added on the platform	3	3

TABLE 4.1 – Sprint 2 Backlog

4.3 Functional Specification

4.3.1 Sprint 2 Detailed Use Case Diagram

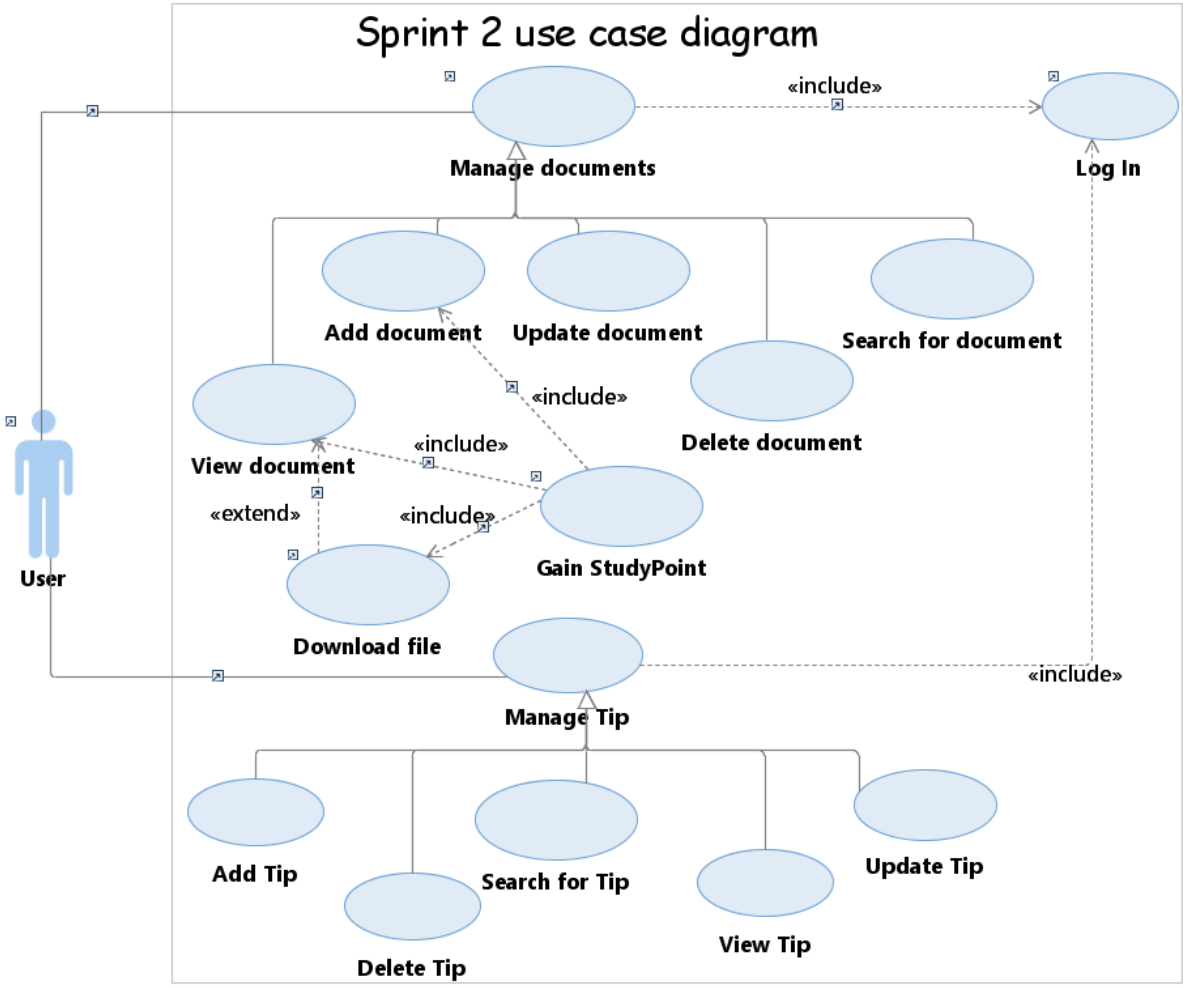


FIGURE 4.1 – Use Case « Manage documents » Use Case Diagram

4.3.1.1 Use Case « ADD document » Textual Description

Use Case	ADD document
Actor	User
Pre-condition	The user is logged in
Post-condition	New document added
Main Scenario	1. The user selects manage documents. 2. The system displays the manage documents UI . 3. The user selects the add button . 4. The system displays the add document form . 5. The user fills the form . 6. The system verifies the data . « Include » Gain Study Point . 7. The system saves the data .
Alternative Scenario	2.a. user not logged in : 1- The system redirects the user to the login page . 6.a. No document was added : 1- The system displays an error message . 2- The system goes back to step 4.

TABLE 4.2 – Use Case « ADD document » Textual Description

4.3.1.2 Use Case « Delete document » Textual Description

Use Case	Delete document
Actor	User
Pre-condition	- The user is authenticated . - Document exists
Post-condition	Document deleted
Main Scenario	1. The user selects manage document. 2. The system displays the manage document UI . 3. The system displays the list of documents added by the user. 4. The user selects the document to delete . 5. The user clicks on delete . 6. The system displays a confirmation message . 7. The user confirms the deletion . 8. The system updates the documents list .
Alternative Scenario	3.a. No documents were added by the user : 1- The system informs the user that no documents were found . 2-The system re-displays the manage documents UI . 7.a. The user cancels deletion : 1- The system informs the user that no changes were made. 2-The system re-displays the manage documents UI .

TABLE 4.3 – Use Case « Delete document » Textual Description

4.3.1.3 Use Case « Search for document » Textual Description

Use Case	Search for document
Actor	User
Pre-condition	The user is authenticated
Post-condition	Search results displayed
Main Scenario	1. The user selects manage documents. 2. The system displays the manage documents UI . 3. The user types the desired document name in the search bar. 4. The user clicks on search . 5. The system searches for the document . 6. The system displays the search results .
Alternative Scenario	2.a.User not authenticated : 1- The system redirects the user to the login page . 6.a. Document not found : 1- The system displays an error message .

TABLE 4.4 – Use Case « Search for document » Textual Description

4.3.1.4 Use Case « Update document » Textual Description

Use Case	Update document
Actor	User
Pre-condition	- The user is authenticated . -Document exists
Post-condition	Document information updated
Main Scenario	1. The user selects manage documents. 2. The system displays the manage documents UI . 3. The system displays the list of documents added by the user . 4. The user selects the desired document form the list . 5. The user clicks on update . 6. The system displays the form . 7. The user makes changes . 8. The user clicks on save. 9. The system saves the changes.
Alternative Scenario	7.a. User forgot to save : 1- The system informs the user that they forgot to save the changes .

TABLE 4.5 – Use Case « Update document » Textual Description

4.3.1.5 Use Case « View document » Textual Description

Use Case	View document
Actor	User
Pre-condition	- The user is logged in. -Document exists
Post-condition	Document displayed
Main Scenario	1. The user selects View document. 2. The system displays the list of documents . 3. The user selects the document they want to view . « Include » Gain Study Point 4. The system displays the document . « Extend » Download file .
Alternative Scenario	1.a. User not authenticated : 1- The system redirects the user to the login page .

TABLE 4.6 – Use Case « View document » Textual Description

4.3.1.6 Use Case « ADD Tip » Textual Description

Use Case	ADD Tip
Actor	User
Pre-condition	The user is authenticated
Post-condition	New Tip added
Main Scenario	1. The user selects Share Tip. 2. The system displays the Share Tip UI . 3. The user selects the add Button . 4. The system displays the add Tip form . 5. The user fills the form . 6. The system verifies the data . 7. The system saves the data .
Alternative Scenario	2.a. user not logged in : 1- The system redirects the user to the login page .

TABLE 4.7 – Use Case « ADD Tip » Textual Description

4.3.1.7 Use Case « Delete Tip » Textual Description

Use Case	Delete Tip
Actor	User
Pre-condition	- The user is authenticated . - Tip added >= 1
Post-condition	Tip deleted
Main Scenario	1. The user selects Share Tip. 2. The system displays the Share Tip UI . 3. The system displays the list of tips added by the user. 4. The user selects delete from the tip menu . 5. The system displays a confirmation message . 6. The user confirms the deletion . 7. The system updates the documents list .
Alternative Scenario	3.a. No tips were added by the user : 1- The system displays an empty UI . 7.a. The user cancels deletion : 1- The system goes back to step 2.

TABLE 4.8 – Use Case « Delete Tip » Textual Description

4.3.1.8 Use Case « Search for Tip » Textual Description

Use Case	Search for Tip
Actor	User
Pre-condition	The user is logged in
Post-condition	Search results displayed
Main Scenario	1. The user selects share Tip. 2. The system displays the share Tip UI . 3. The user types the desired Tip topic name in the search bar. 4. The system searches for the tip . 5. The system displays the search results .
Alternative Scenario	2.a.User not authenticated : 1- The system redirects the user to the login page . 6.a. Tip not found : 1- The system displays an empty UI .

TABLE 4.9 – Use Case « Search for Tip » Textual Description

4.3.1.9 Use Case « Update Tip » Textual Description

Use Case	Update Tip
Actor	User
Pre-condition	- The user is authenticated. -Tip exists
Post-condition	Tip information updated
Main Scenario	1. The user selects share Tip. 2. The system displays the share Tip UI . 3. The system displays the list of tips added by the user . 4. The user selects the desired tip to update form the list . 5. The user selects update from the menu . 6. The system displays the form . 7. The user makes changes . 8. The user clicks on update. 9. The system saves the changes.
Alternative Scenario	7.a. User forgot to save : 1- The system goes back to step 2 .

TABLE 4.10 – Use Case « Update Tip » Textual Description

4.3.1.10 Use Case « View Tips » Textual Description

Use Case	View Tips
Actor	User
Pre-condition	- The user is logged in . - Tips >= 1
Post-condition	Tips displayed
Main Scenario	1. The user selects View Tips UI . 2. The system displays the list of Tips . 3. The user selects the Tip they want to view . 4. The system displays the Tip's full content .
Alternative Scenario	1.a. User not authenticated : 1- The system redirects the user to the login page .

TABLE 4.11 – Use Case « View Tips » Textual Description

4.4 Design

4.4.1 Use Case « Manage documents » Sequence Diagram

4.4.1.1 Use Case « ADD document » Sequence Diagram

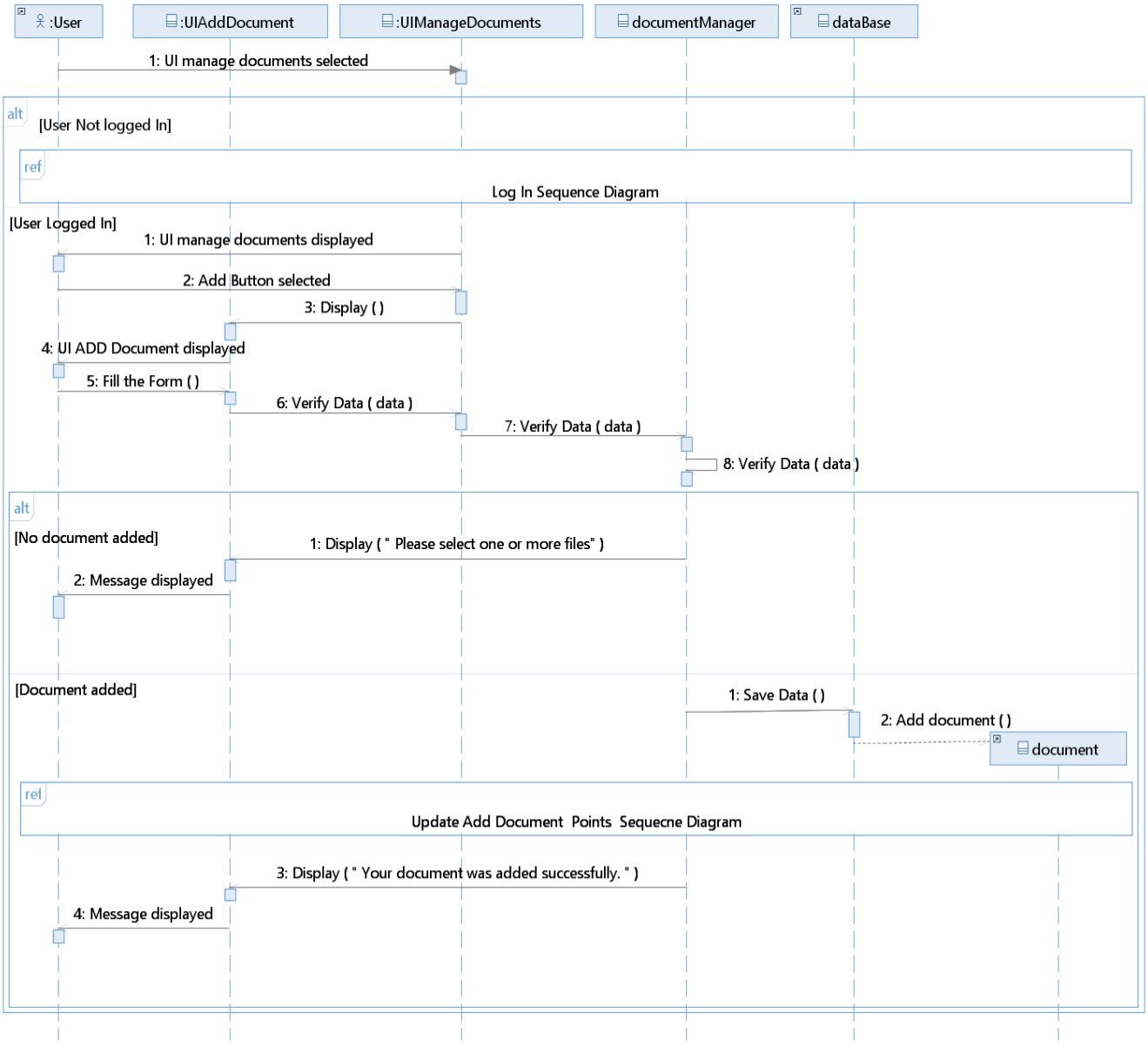


FIGURE 4.2 – Use Case « ADD document » Sequence Diagram

4.4.1.2 Use Case « Delete document » Sequence Diagram

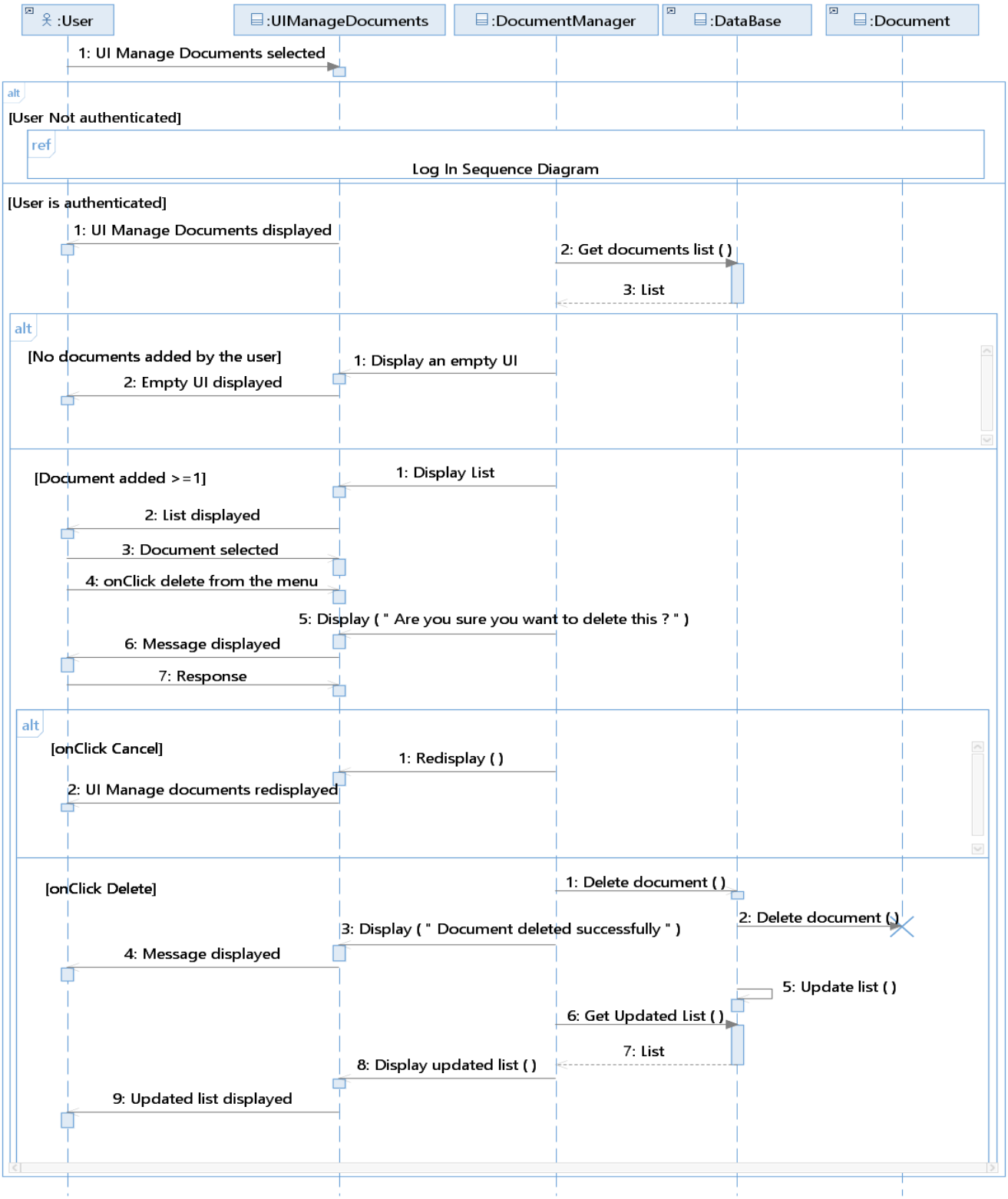


FIGURE 4.3 – Use Case « Delete document » Sequence Diagram

4.4.1.3 Use Case « Search for document » Sequence Diagram

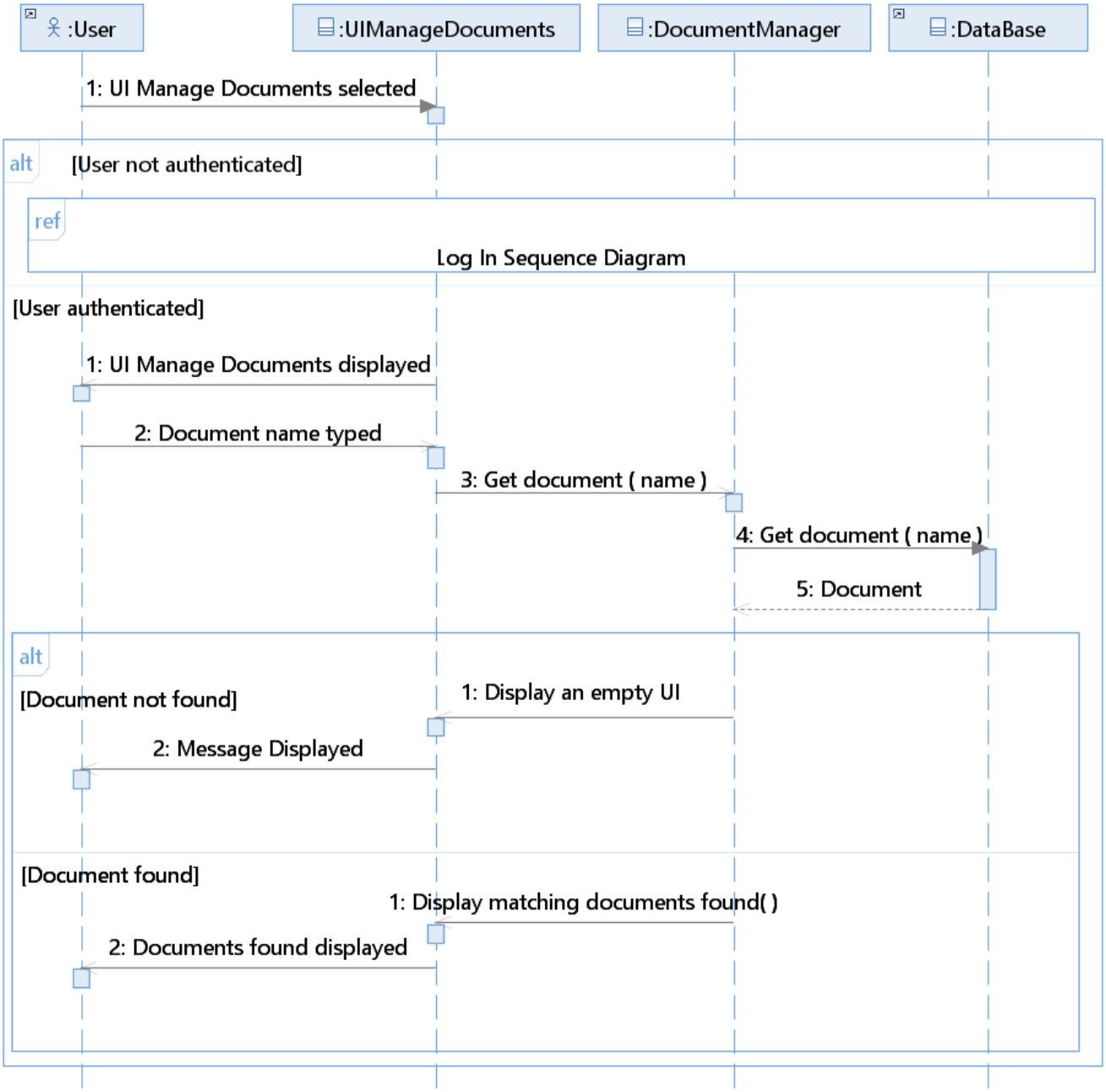


FIGURE 4.4 – Use Case « Search for document » Sequence Diagram

4.4.2 Use Case « ADD Tip » Sequence Diagram

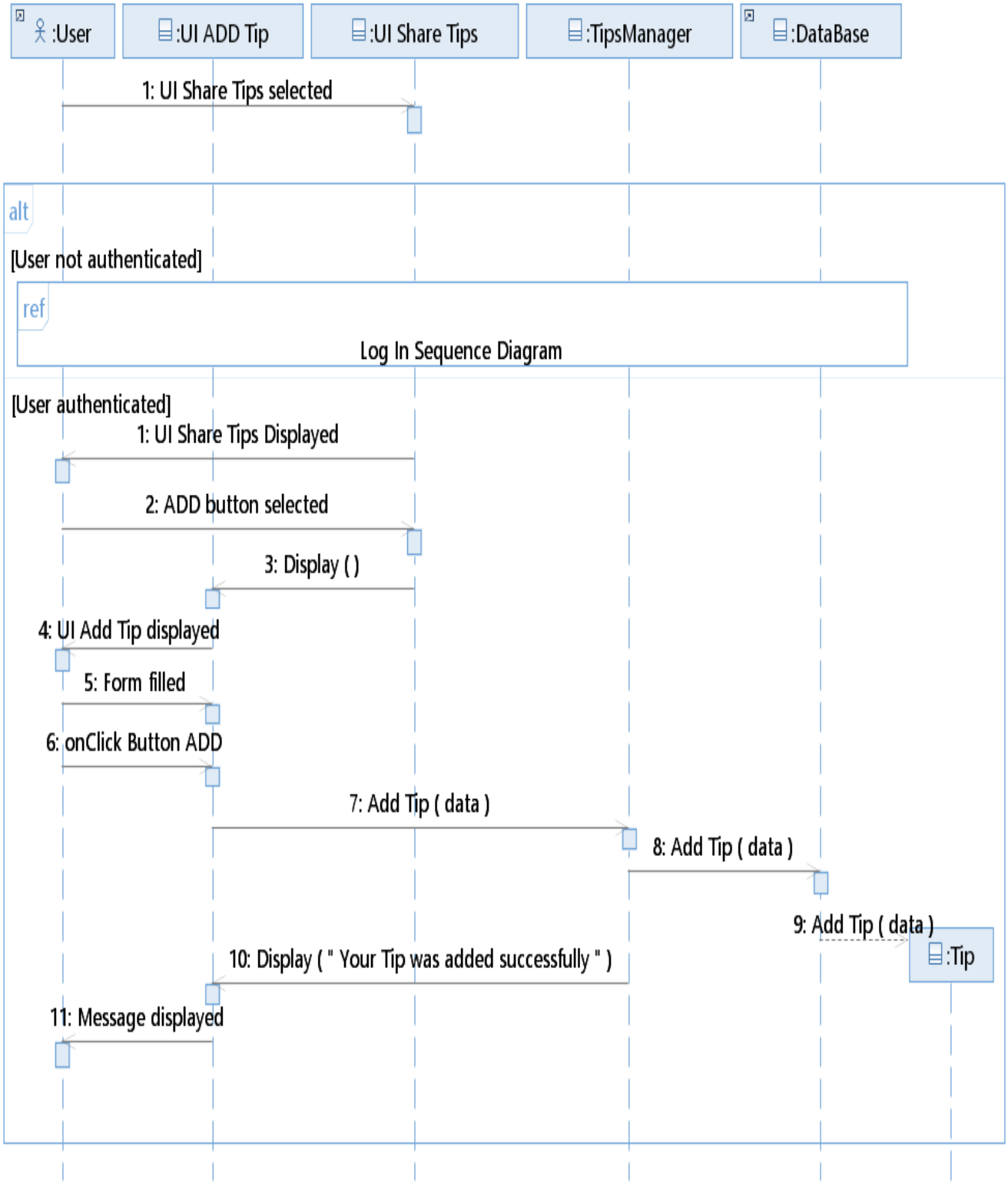


FIGURE 4.5 – Use Case « ADD Tip » Sequence Diagram

4.4.3 Use Case « Delete Tip » Sequence Diagram

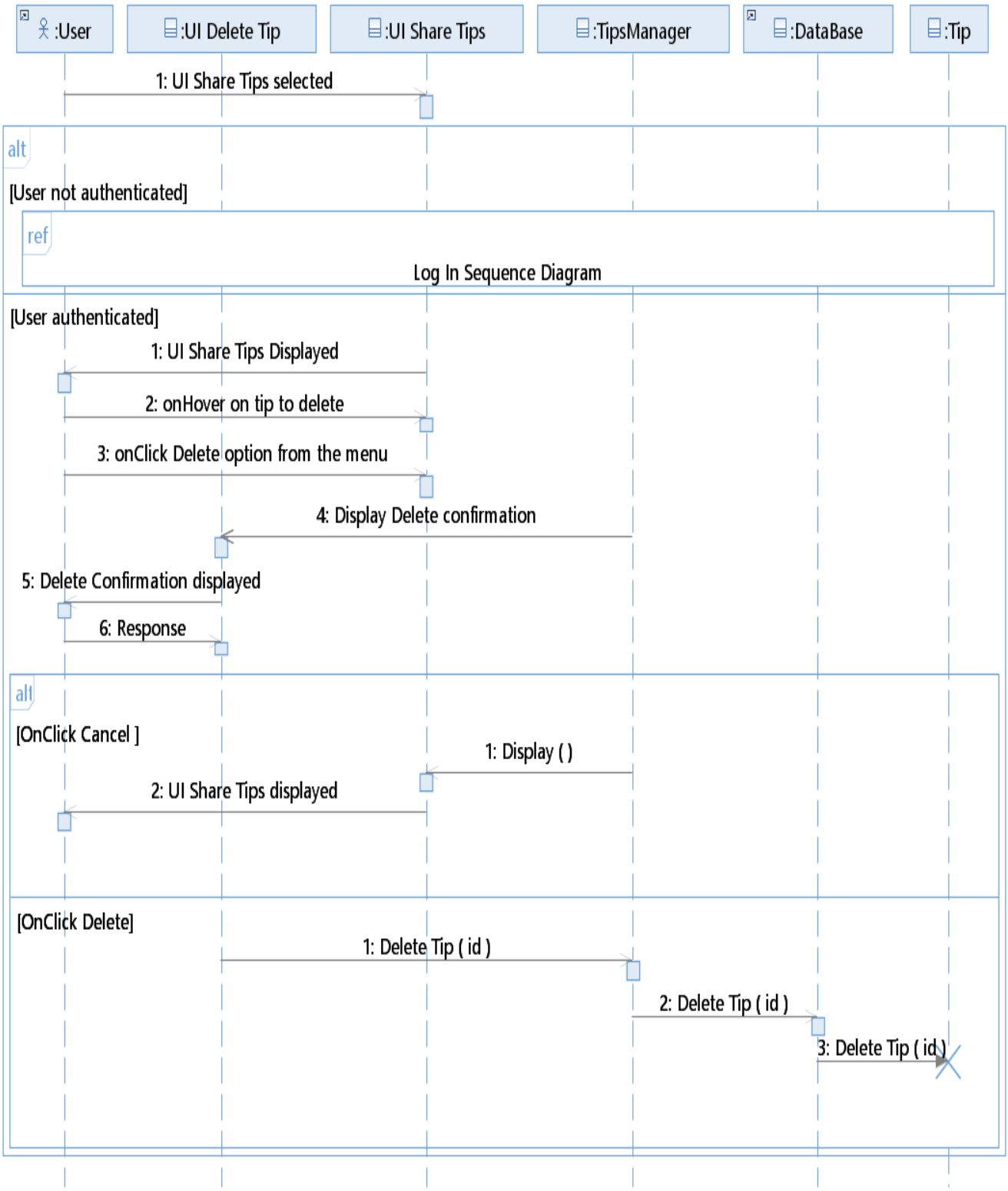


FIGURE 4.6 – Use Case « Delete Tip » Sequence Diagram

4.4.4 Sprint 2 Class Diagram

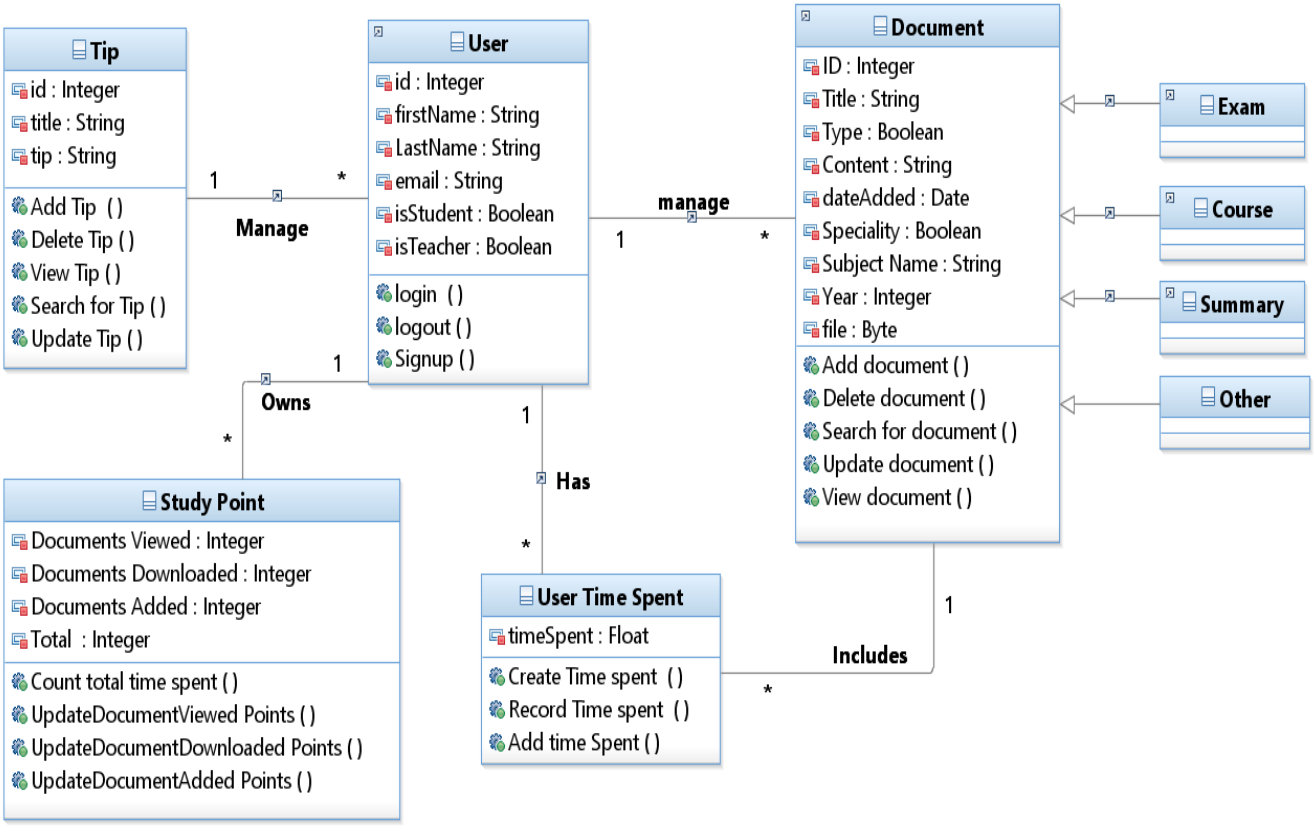


FIGURE 4.7 – Sprint 2 Class Diagram

4.4.5 Use Case « Manage documents » Traceability

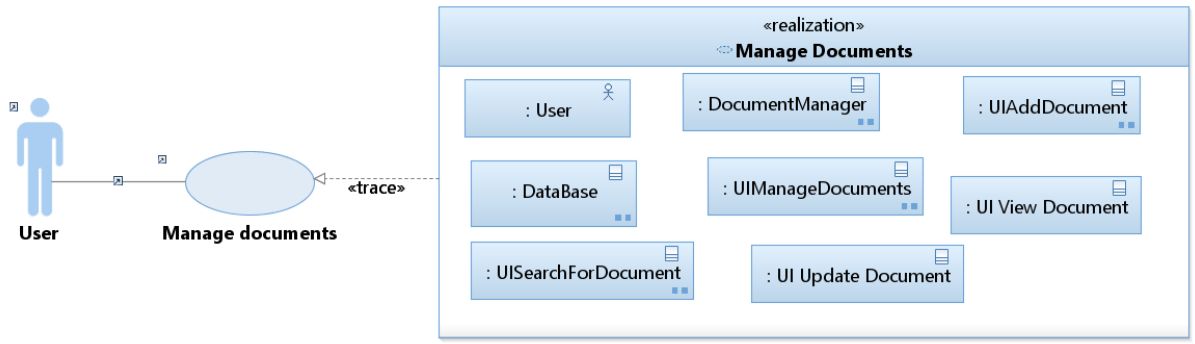


FIGURE 4.8 – Use Case « Manage documents » Traceability

4.5 Implementation and Tests

4.5.1 Add document

4.5.2 View document

4.5.3 Update document

4.5.4 Search for document

4.5.5 Delete document

4.6 Scrum Tools implementation

4.6.1 Scrum Board

We began the sprint 2 by setting every task we need to get done counting tests to do as well . Here's the visual representation of it at the beginning and another one at the end of the sprint :

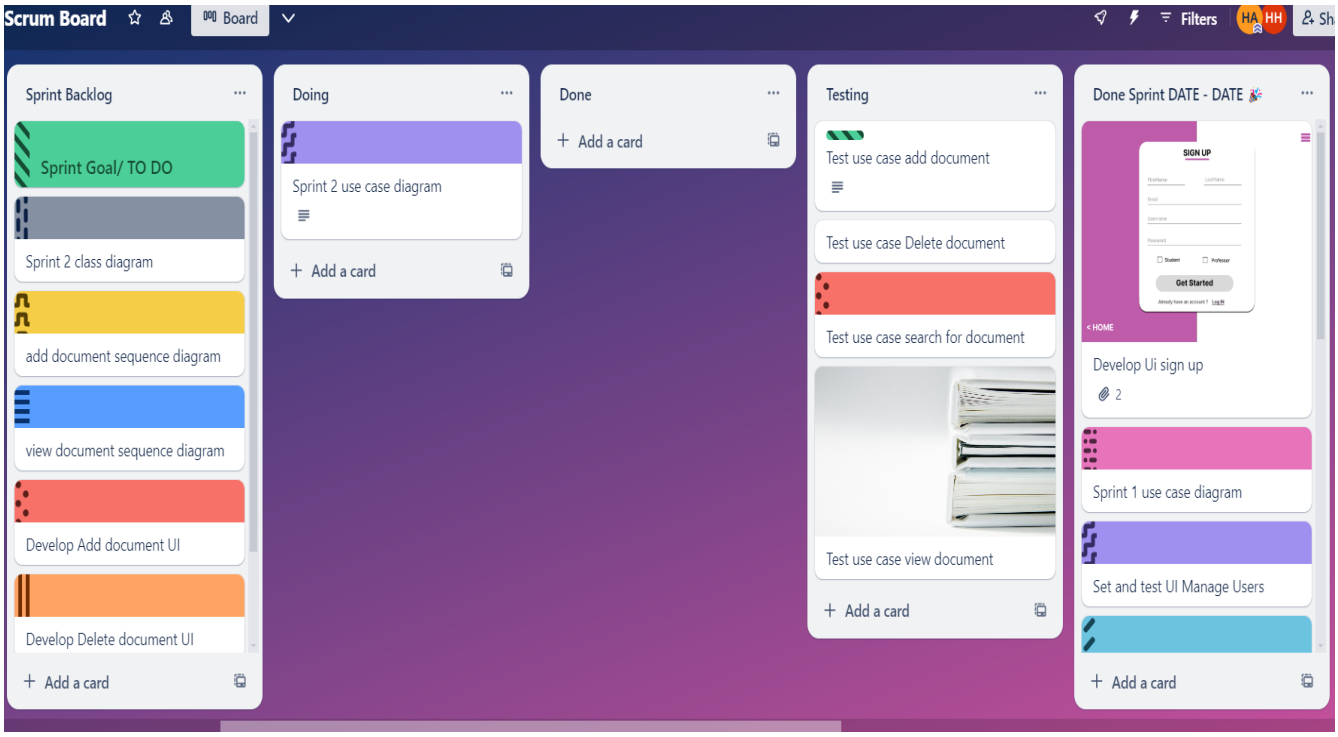


FIGURE 4.9 – Sprint 2 Scrum Board

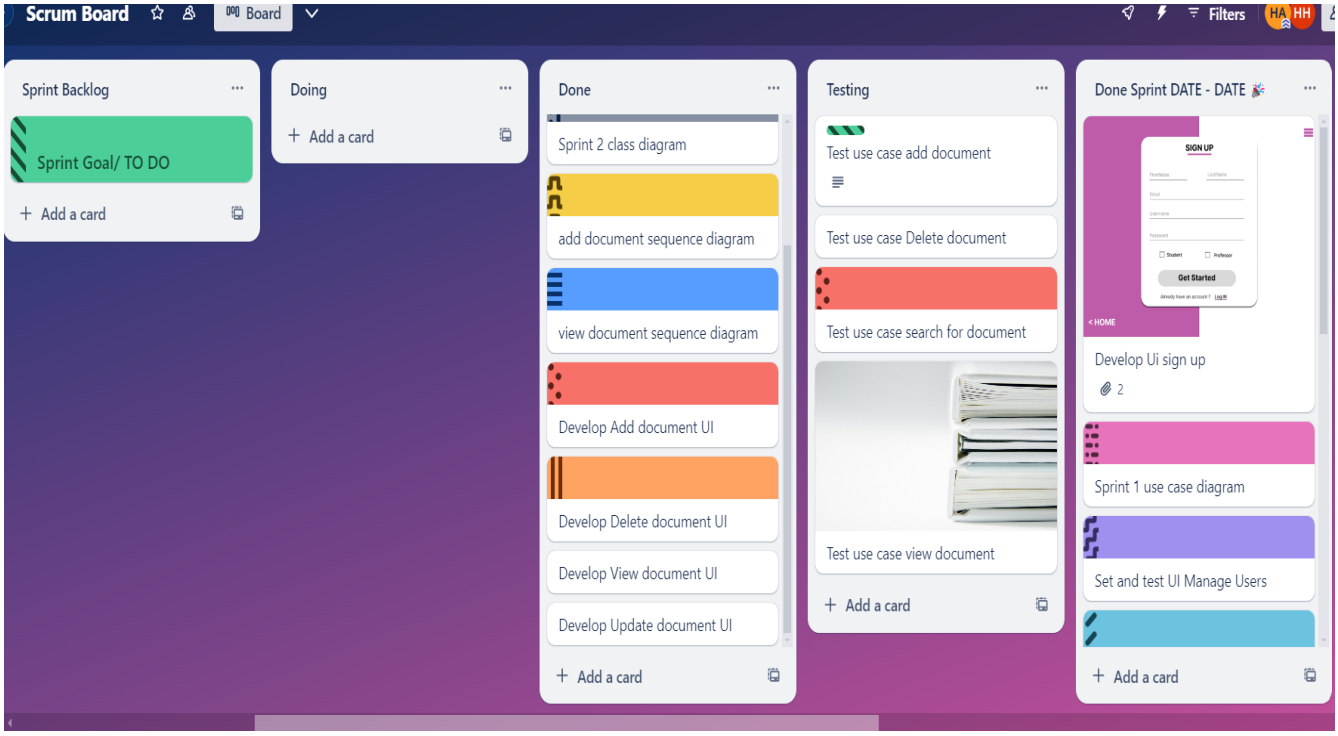


FIGURE 4.10 – Sprint 2 Scrum Board

4.6.2 Scrum Burn-Down Chart

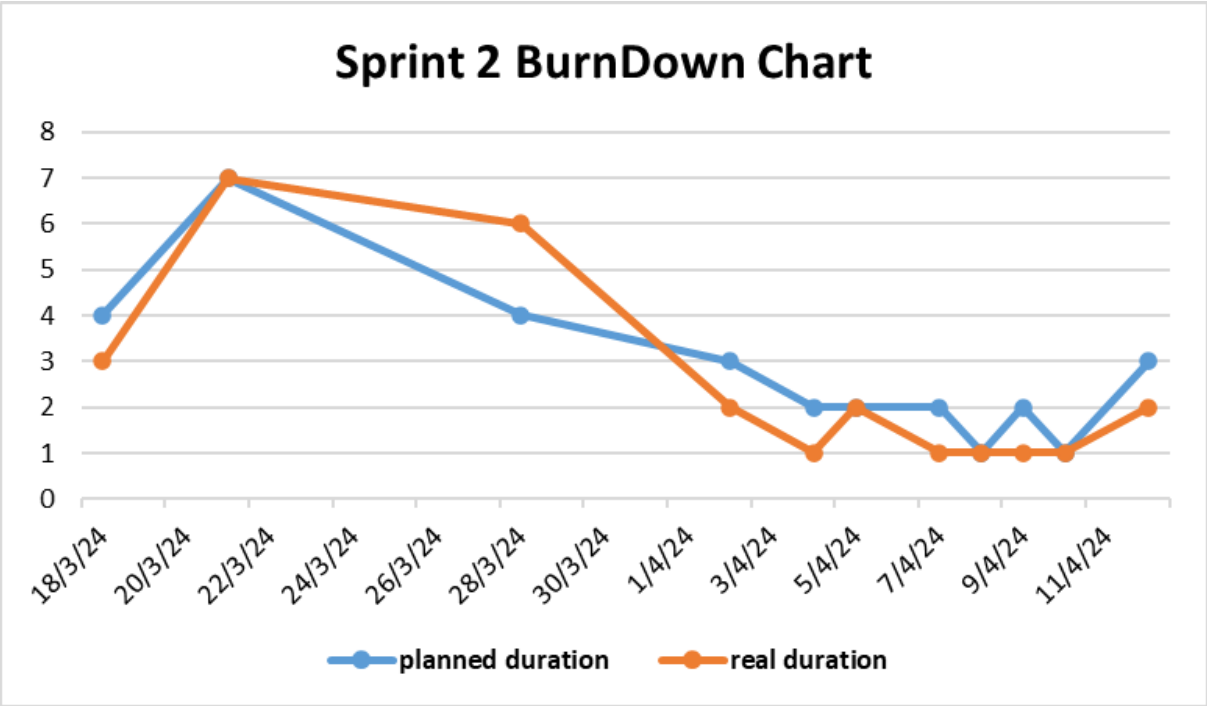


FIGURE 4.11 – Sprint 2 Burn Down Chart

4.7 Sprint Review

4.7.1 Sprint Delivery

The development process got so much easier compared to the first sprint where i was getting used to the tools .The outcome of the second sprint is more that what we got at the previous one .We granted the user with the ability to add , update , search and delete documents they added and view documents they added along with documents added by other and gain study points from them

4.7.2 Difficulties faced

Mid working on this sprint i faced a bunch of challenges like :

— Learning to fetch from the back end specific information i need and not display everything like when i worked on the manage profile use case .

— Working on the view document was very challenging but i learned through it how to display a PDF file in my component .

4.8 Conclusion

In closing , this sprint we worked on manage documents and it's key point gain study point along with manage tips . We can now tell that a big progress was made upon finishing this sprint and it's time to move onto the third one where we deal with managing posts and leaving feedback use cases .

Chapitre

5 Sprint 3 : Manage Posts , Leave feedback

Plan

5.1 Introduction

5.2 Sprint Backlog

Feature	**User Story**	**Priority**	**Estimated Duration**
Add Post	As a user I want to be able to add a Post to the platform	1	4
View Post	As a user I want to be able to view Posts on the platform	1	5
Update Post	As a user I want to be able to update a Post i added on the platform	2	2
Delete Post	As a user I want to be able to delete a Post i added on the platform	2	1
Search for a Post	As a user I want to be able to search for a Post on the platform	3	1
Leave feedback	As a user I want to be able to leave a feedback about the platform and express my opinion	3	2

TABLE 5.1 – Sprint 3 Backlog

5.3 Use Cases Specification

5.3.1 Sprint 3 Detailed Use Case Diagram

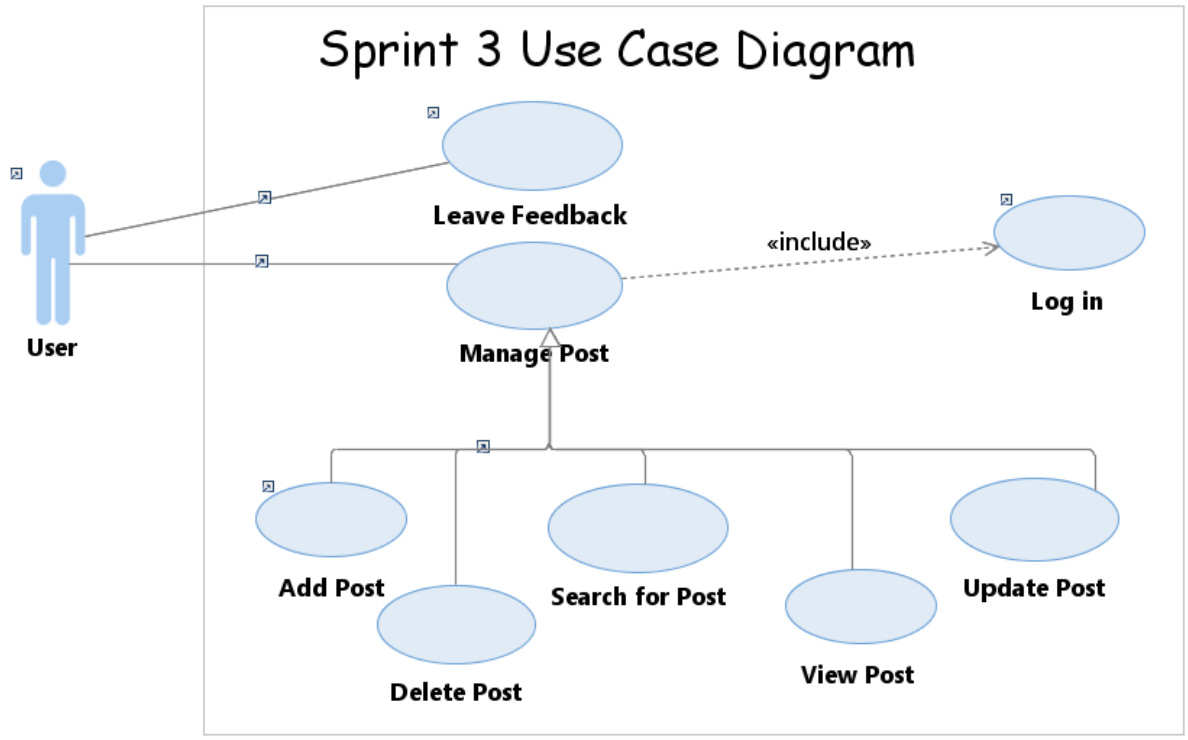


FIGURE 5.1 – Sprint 3 detailed Use Case Diagram

5.3.1.1 Use Case « ADD Post » Textual Description

Use Case	ADD Post
Actor	User
Pre-condition	The user is logged in
Post-condition	New Post added
Main Scenario	1. The user selects manage Post. 2. The system displays the manage Post UI . 3. The user selects the add Post option. 4. The system displays the add Post UI . 5. The user fills the form . 6. The system verifies the data . 7. The system saves the data .
Alternative Scenario	2.a. user not logged in : 1- The system redirects the user to the login page . 6.a. Form not filled : 1- The system informs the user that the form is not filled.

TABLE 5.2 – Use Case « ADD Post » Textual Description

5.3.1.2 Use Case « Delete Post » Textual Description

Use Case	Delete Post
Actor	User
Pre-condition	- The user has an account . - Post exists
Post-condition	Post deleted
Main Scenario	1. The user selects manage Post. 2. The system displays the manage Post UI . 3. The system displays the list of Post added by the user. 4. The user selects the Post to delete . 5. The user clicks on delete . 6. The system displays a confirmation message . 7. The user confirms the deletion . 8. The system updates the Post list .
Alternative Scenario	3.a. No Post were added by the user : 1- The system informs the user that no Internship Experiences were found . 7.a. The user cancels deletion : 1- The system informs the user that no changes were made . 2-The system re-displays the manage documents UI .

TABLE 5.3 – Use Case « Delete Post » Textual Description

5.3.1.3 Use Case « Search for Post » Textual Description

Use Case	Search for Post
Actor	User
Pre-condition	The user is authenticated
Post-condition	Search results displayed
Main Scenario	1. The user selects manage Post. 2. The system displays the manage Post UI . 3. The user types the desired Post name in the search bar. 4. The user clicks on search . 5. The system searches for the Post . 6. The system displays the search results .
Alternative Scenario	2.a.User not authenticated : 1- The system redirects the user to the login page . 6.a. Post not found : 1- The system displays an error message .

TABLE 5.4 – Use Case « Search for Post » Textual Description

5.3.1.4 Use Case « Update Post » Textual Description

Use Case	Update Post
Actor	User
Pre-condition	- The user is logged in. -Post exists
Post-condition	User information updated
Main Scenario	1. The user selects manage Post. 2. The system displays the manage Post UI . 3. The system displays the list of Post added by the user. 4. The user selects the desired Post form the list . 5. The user clicks on update . 6. The system displays the form . 7. The user makes changes . 8. The user clicks on save. 9. The system saves the changes.
Alternative Scenario	8.a. User forgot to save : 1- The system cancels the changes . 2-The system goes back to step 6 .

TABLE 5.5 – Use Case « Update Post » Textual Description

5.3.1.5 Use Case « View Post » Textual Description

Use Case	View Post
Actor	User
Pre-condition	- The user is authenticated . -Posts >= 1
Post-condition	Post displayed
Main Scenario	1. The user selects View Posts. 2. The system displays the view Posts UI.
Alternative Scenario	1.a. User not authenticated : 1- The system redirects the user to the login page .

TABLE 5.6 – Use Case « View Post » Textual Description

5.3.1.6 Use Case « Leave Feedback » Textual Description

Use Case	Leave Feedback
Actor	User
Pre-condition	The user accessed the leave feedback page
Post-condition	New Feedback added
Main Scenario	1. The user fills the form . 2. The user click on send feedback button . 3. The system verifies the data . 4. The system saves the data .
Alternative Scenario	3.a. form data missing : 1- The system displays an error page .

TABLE 5.7 – Use Case « Leave Feedback » Textual Description

5.4 Design

5.4.1 Use Case « ADD Post » Sequence Diagram

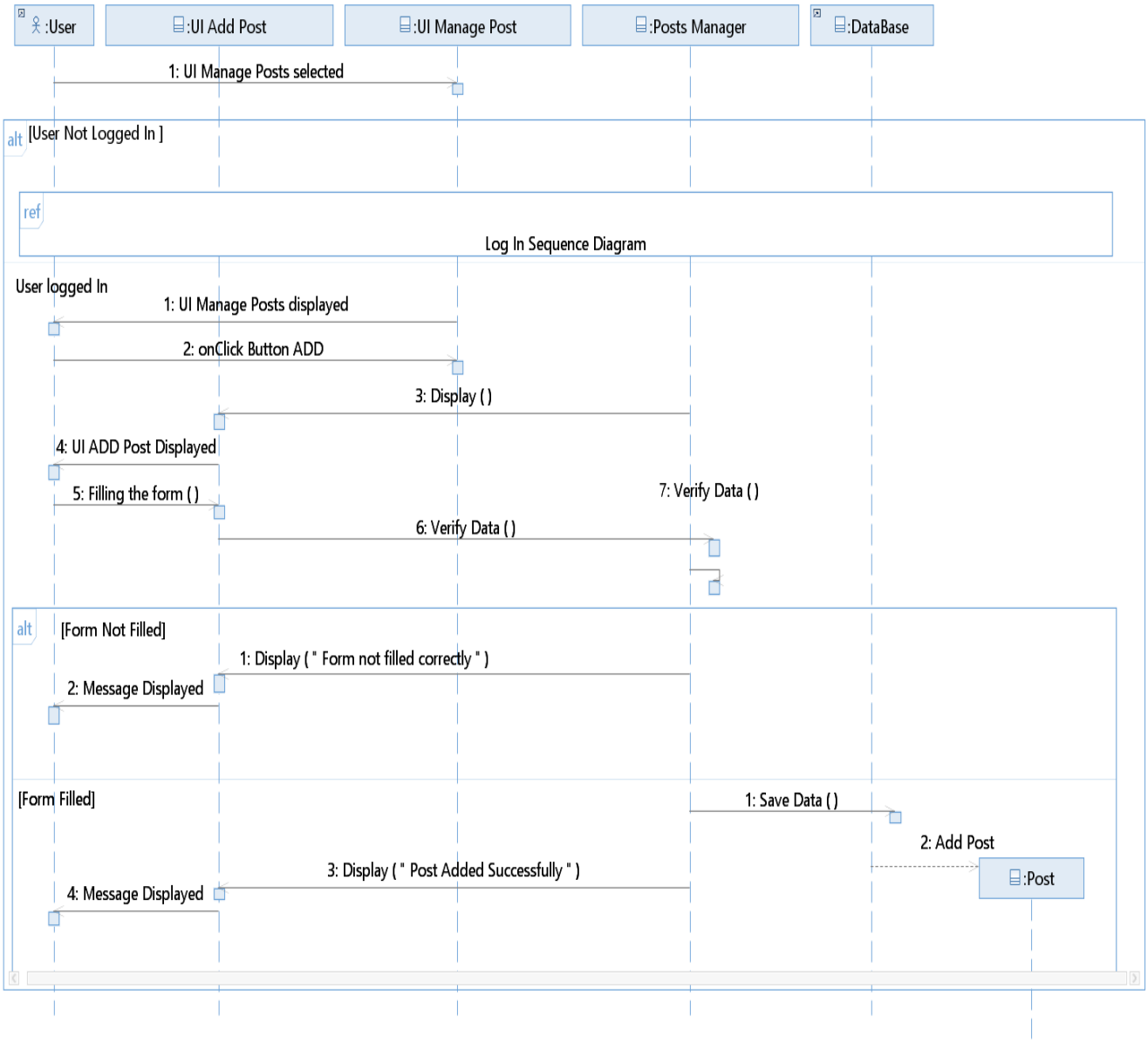


FIGURE 5.2 – Use Case « ADD Post » Sequence Diagram

5.4.2 Use Case « Update Post » Sequence Diagram

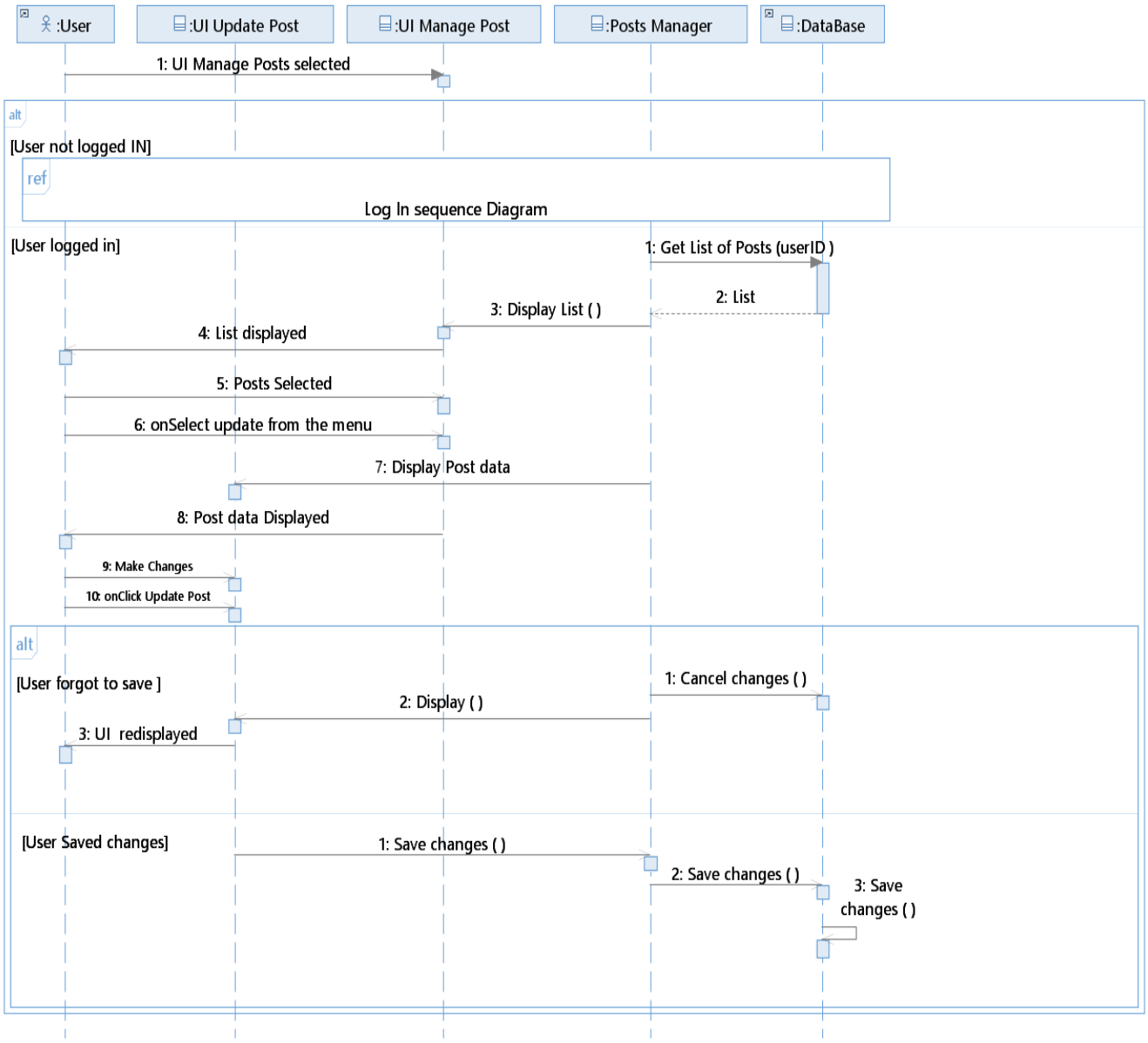


FIGURE 5.3 – Use Case « Update Post » Sequence Diagram

5.4.3 Use Case « View Post » Sequence Diagram

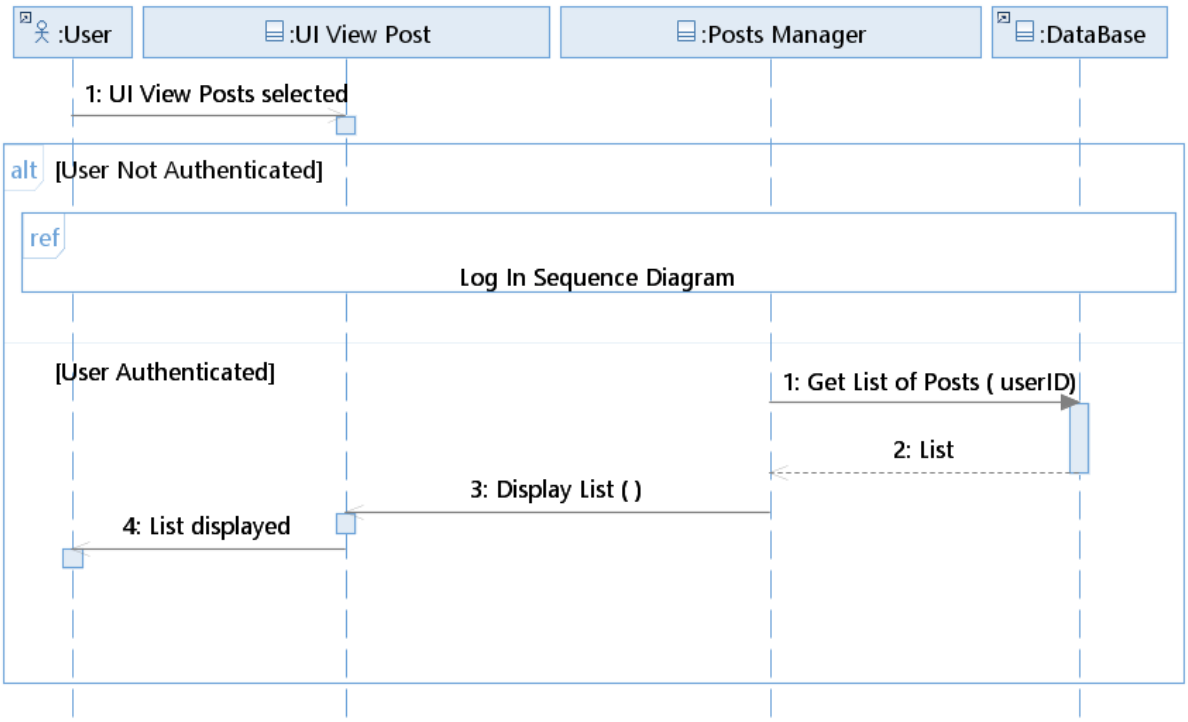


FIGURE 5.4 – Use Case « View Post » Sequence Diagram

5.4.4 Use Case « Leave Feedback » Sequence Diagram

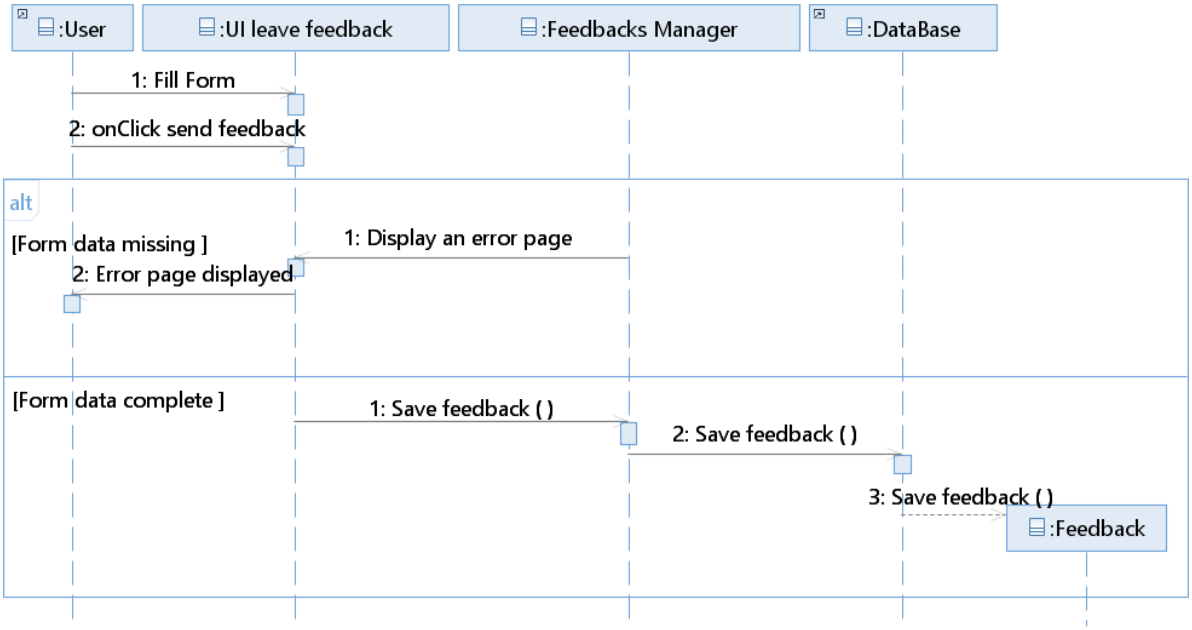


FIGURE 5.5 – Use Case « Leave Feedback » Sequence Diagram

5.4.5 Sprint 3 Class Diagram

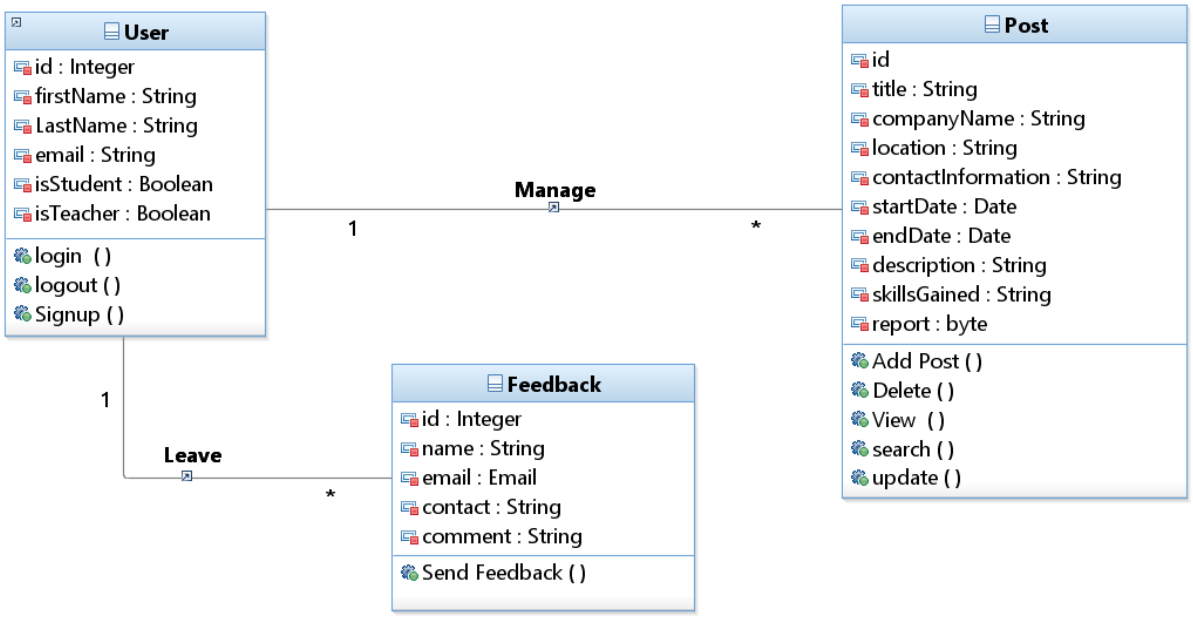


FIGURE 5.6 – Sprint 3 Class Diagram

5.4.6 Global Class Diagram

The reason we create a class diagram is to encapsulate and illustrate our system's architecture .It contains the classes and their relationships within our system .Each of these classes has attributes , methods and association . Below we can find the class diagram of our project .

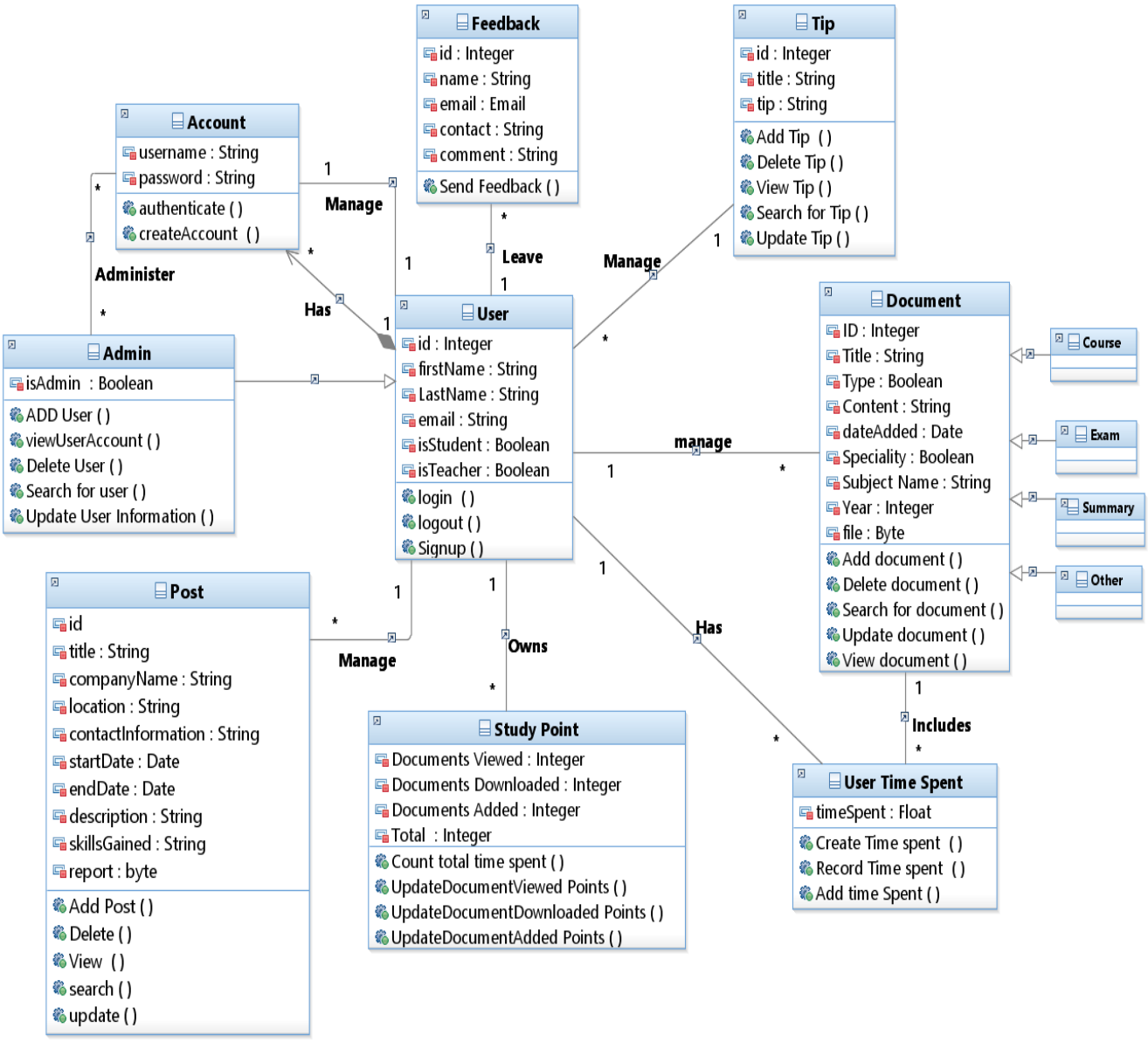


FIGURE 5.7 – Global Class Diagram

5.4.7 Use Case « Manage Post » Traceability

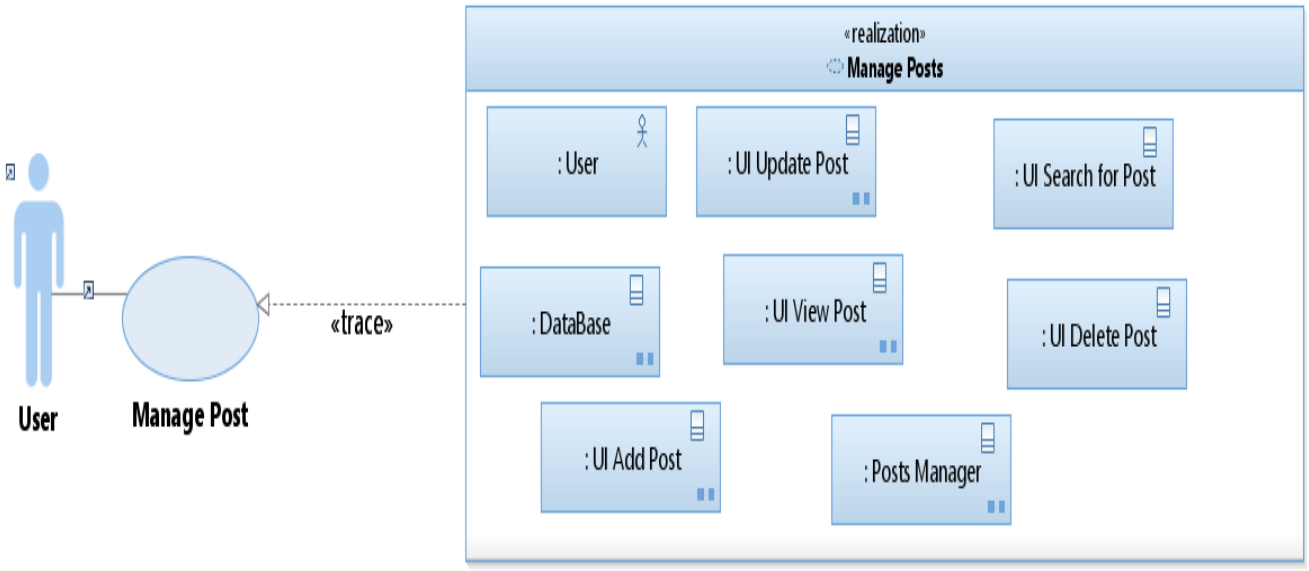


FIGURE 5.8 – Use Case « Manage Post » Traceability

5.5 Implementation and Tests

5.5.1 Add Post

5.5.2 View Post

5.5.3 Update Post

5.5.4 Search for Post

5.5.5 Delete Post

5.6 Scrum Tools implementation

5.6.1 Scrum Board

A significant progress was made up until now and we're very close to finishing the project .This is how our scrum board pictured the tasks to get done after two weeks of work along with at the end of this sprint :

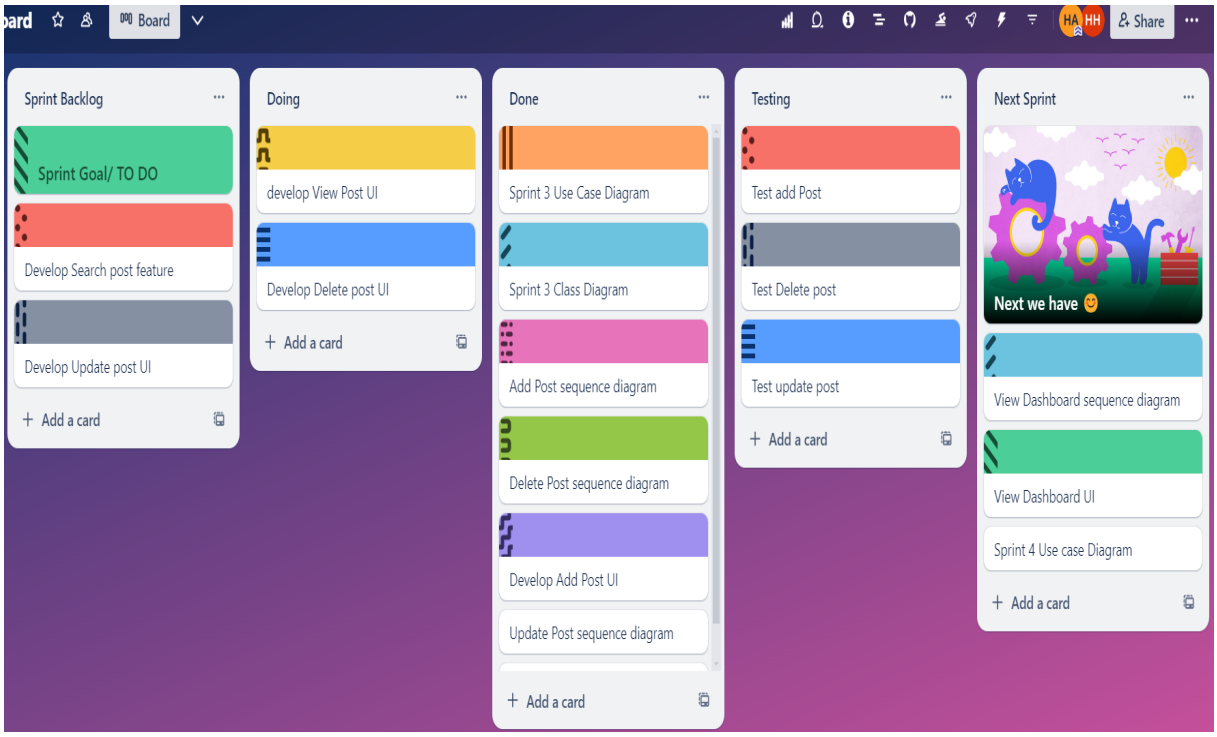


FIGURE 5.9 – Sprint 3 Scrum Board

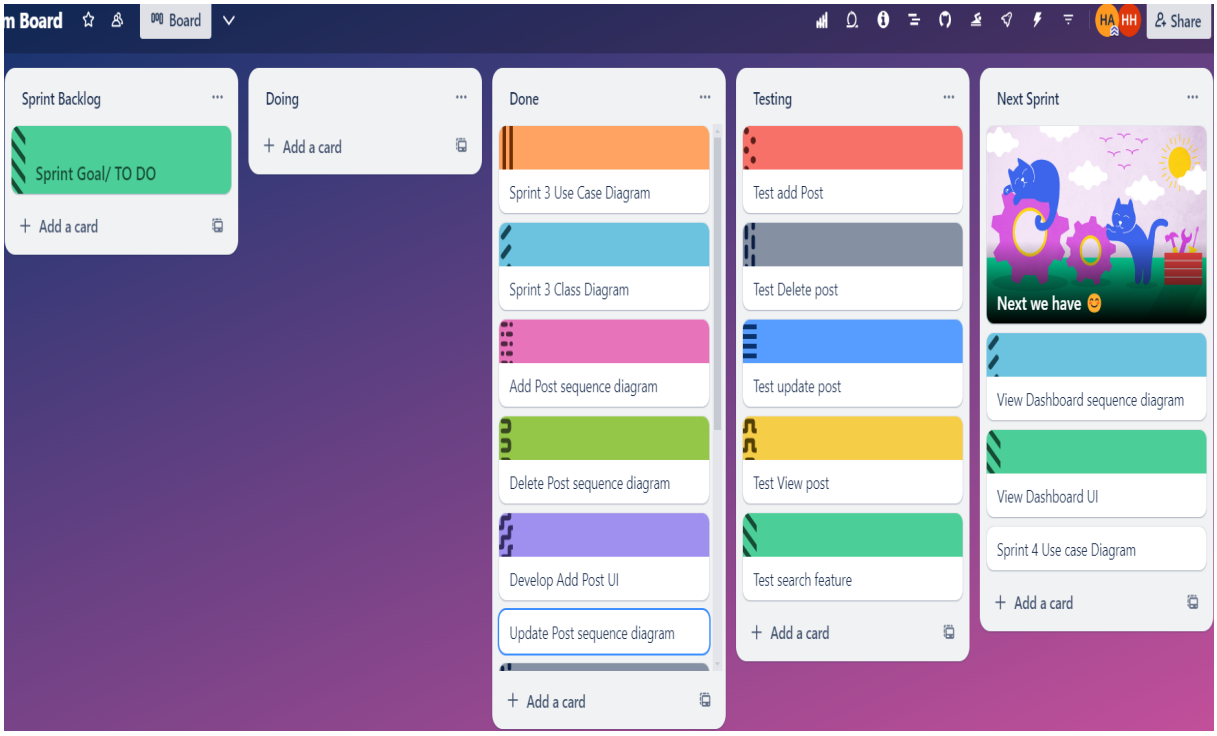


FIGURE 5.10 – Sprint 3 Scrum Board

5.6.2 Scrum Burn-Down Chart

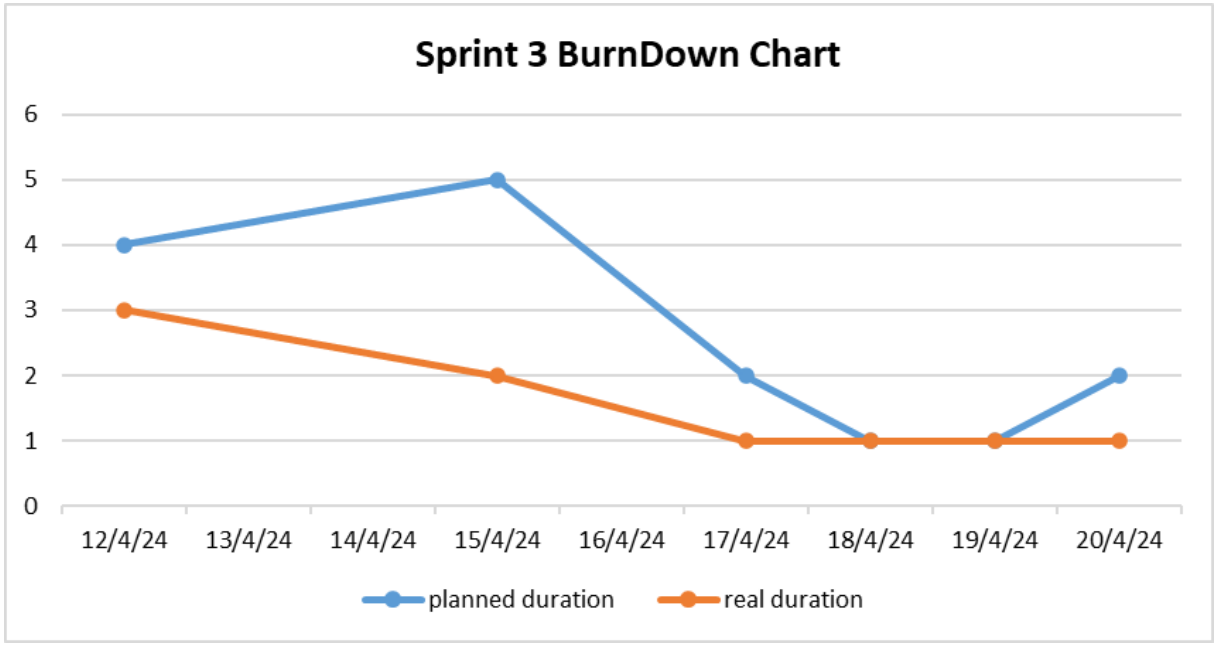


FIGURE 5.11 – Sprint 3 Burn Down Chart

5.7 Sprint Review

5.7.1 Sprint Delivery

The primary goal of this third sprint was to address the professional life of students with a section for posts about internships experiences and what they learned from it .They are now able to add , delete, update , search by company name or skills learned and view these posts .We also granted them a place to leave their feedback which eventually helps improving the platform for the better .

5.7.2 Difficulties faced

As i celebrate the success of completing the sprint like usual i want to highlight the struggle i faced :

— Since in the rest of the forms i made the fields were compulsory to fill i didn't have any problems however while making the leave feedback feature i got the chance to work on sending a form with optional fields .

5.8 Conclusion

We are pretty much done with this project .I believe we should center our focus on the last and most important part about view dashboard use case .The objective for the next sprint is to create a dashboard for users with different kind of visualisations .

Chapitre

6 Sprint 4 : View Dashboard

Plan

6.1 Introduction

6.2 Sprint Backlog

Feature	User Story	Priority	Estimated Duration
View dashboard	As a user I want to view statistics concerning my activities on the platform	1	7

TABLE 6.1 – Sprint 4 Backlog

6.3 Use Cases Specification

6.3.1 Sprint 4 Detailed Use Case Diagram

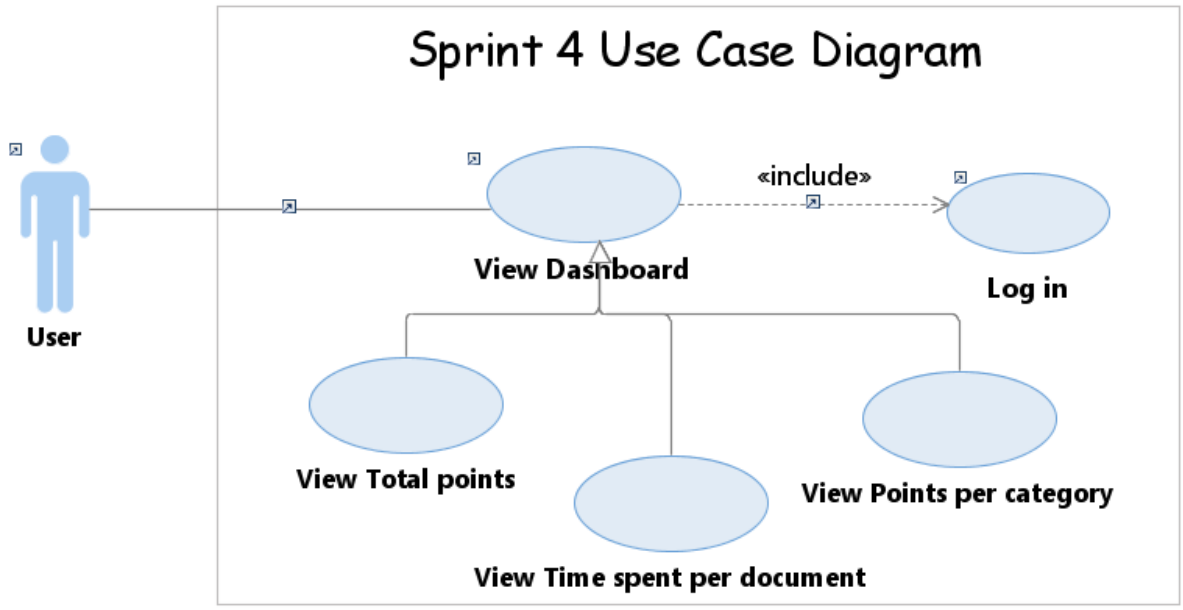


FIGURE 6.1 – Sprint 4 Detailed Use Case Diagram

6.3.2 Use Case « View Dahsboard » Textual Description

Use Case	View Statistics
Actor	User
Pre-condition	The user has an account
Post-condition	Dashboard viewed
Main Scenario	1. The user selects View Dashboard from the menu. 2. The system displays the User's dashboard .
Alternative Scenario	1.a. user not logged in : 1- The system redirects the user to the login page .

TABLE 6.2 – Use Case « View Dashboard » Textual Description

6.4 Design

6.4.1 Use Case « View Dashboard » Sequence Diagram

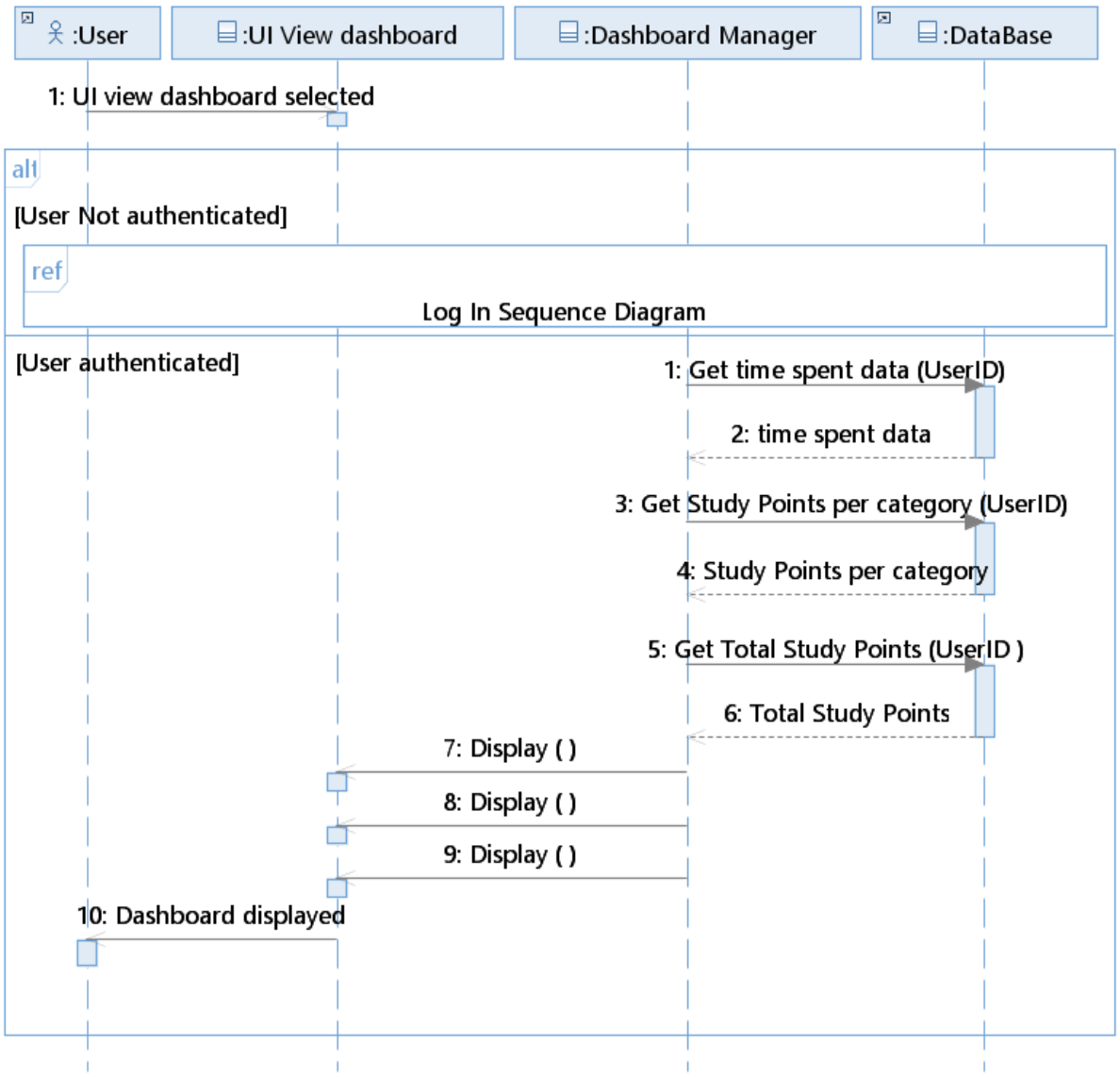


FIGURE 6.2 – Use Case « View Dashboard » Sequence Diagram

6.5 Implementation and Tests

6.5.1 View Dashboard

6.6 Conclusion

6.6.1 Data Base Schema

User (**userID**, firstName , lastName , email , type)

Account(**#userID**, username , password)

Subject(**subjectID** , name , #professorID)

Teacher(**teacherID**, #userID , department)

Student(**studentID**, #userID , major)

Document(**docID** , dateAdded , #subjectID)

Course(**#docID** , year , #teacherID , #subjectID)

Summary (**#docID** , #userID , #subjectID)

Exam(**#docID** , examYear , #teacherID , #subjectID)

Post(**#experienceID** , title , companyName , location , startDate , endDate , description , skillsGained , contactInfo , report)

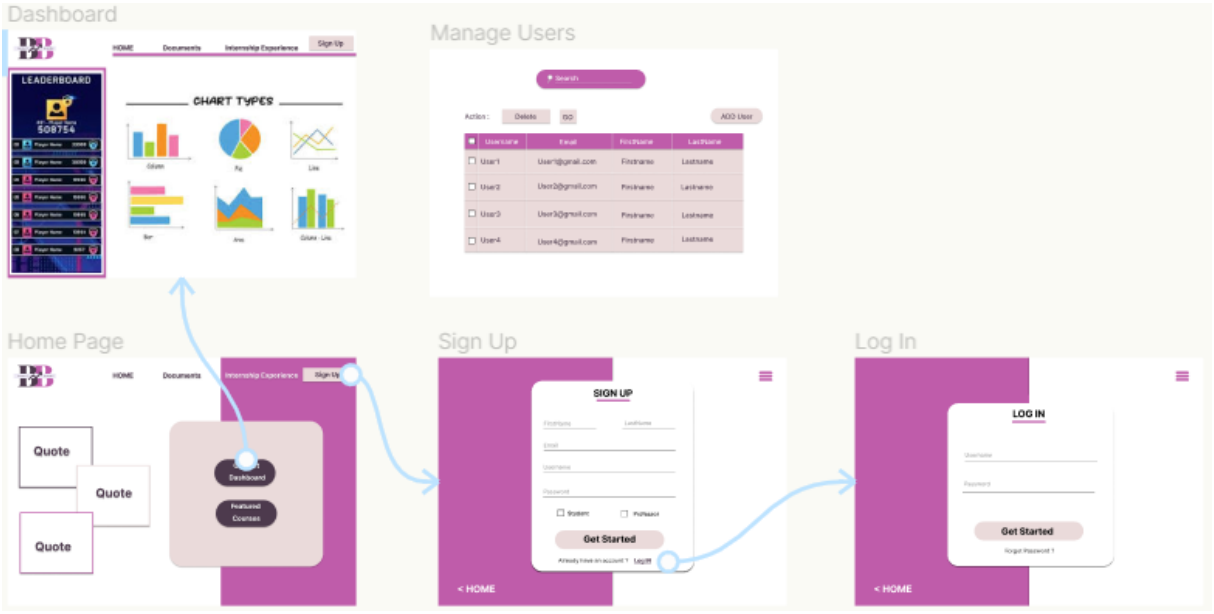


FIGURE 6.3 – Prototype

A use case specification is a textual description of the interactions between the user and the system . We document the goal behind the use case in addition to the how the system should response to the different interactions .