

## FACULTY OF COMPUTING STAFF PUBLICATION DASHBOARD

Adib Bin Morshed

UNIVERSITI TEKNOLOGI MALAYSIA

**DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND  
COPYRIGHT**

Author's full name : Adib Bin Morshed

Date of Birth : 19/09/2001

Title : FACULTY OF COMPUTING STAFF PUBLICATION DASHBOARD

Academic Session : 2022/23

I declare that this thesis is classified as:



**CONFIDENTIAL**

(Contains confidential information under the Official Secret Act 1972)\*



**RESTRICTED**

(Contains restricted information as specified by the organization where research was done)\*



**OPEN ACCESS**

I agree that my thesis to be published as online open access (full text)

1. I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:
2. The thesis is the property of Universiti Teknologi Malaysia
3. The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of research only.
4. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

*nihal*




---

**SIGNATURE OF STUDENT**

A20EC4008

---

**MATRIX NUMBER**

---

**SIGNATURE OF SUPERVISOR**

Prof. Madya. Ts. Dr. Mohd Shahizan bin Othman

---

**NAME OF SUPERVISOR**

NOTES : If the thesis is CONFIDENTIAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction

“I hereby declare that we have read this thesis and in my opinion this thesis is sufficient in term of scope and quality for the award of the degree of Bachelor of Computer Science (Software Engineering)”

Signature : 

---

Name of Supervisor : **Prof. Madya. Ts. Dr. Mohd Shahizan bin Othman**

Date : 21 JANUARY 2025

# FACULTY OF COMPUTING STAFF PUBLICATION DASHBOARD

Adib Bin Morshed

A thesis submitted in fulfilment of the  
requirements for the award of the degree of  
Bachelor of Computer Science (Software Engineering)

Faculty of Computing

Universiti Teknologi Malaysia

June 25

## **DECLARATION**

I declare that this thesis entitled "*FACULTY OF COMPUTING STAFF PUBLICATION DASHBOARD*" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ..... *nihal*

Name : Adib Bin Morshed  
Date : 22 JANUARY 2025

## **DEDICATION**

This thesis is dedicated to my father, who taught me that hard-working is the key to success. This thesis is also dedicated to my friends and family, who has always supported me in every situation. I would like to mention my fellow teacher and supervisors for teaching and guiding me well.

## **ACKNOWLEDGEMENT**

I would like to thank my parents and my LORD for guiding me and helping me in need. I would like to thank my lecturers for teaching me so well in my previous years.

I would like to thank my supervisor for supporting me and helping me with whatever I need during this project.

Lastly, this project is far from being perfect and it has lot to achieved before it is usable. Therefore, suggestions are most welcomed for perfecting this thesis.

## **ABSTRACT**

The Faculty of Computing Staff Publication Dashboard is a comprehensive web-based platform designed to track and analyse publication metrics for the faculty members and research staff. The dashboard provides a centralized repository for storing, managing, and visualizing various publication-related data, including indexed and non-indexed publications, H-index, citation counts, grant information, income generation, commercialization activities, and networking collaborations. The project aims to streamline the process of monitoring and evaluating the faculty's publication performance, enabling better insights into research productivity and impact. By aggregating and presenting the publication metrics in the form of graphs and charts, the dashboard offers an intuitive and interactive interface for users to explore and understand the faculty's scholarly output. The system utilizes a database to store the publication data, which is scraped and processed through various data preprocessing methods. The data is then organized and displayed on the dashboard in real-time, ensuring up-to-date information for faculty members, research staff, and other stakeholders.

## **ABSTRAK**

Papan Pemuka Penerbitan Staf Fakulti Pengkomputeran ialah platform berasaskan web komprehensif yang direka untuk menjelak dan menganalisis metrik penerbitan untuk ahli fakulti dan kakitangan penyelidikan. Papan pemuka menyediakan repositori berpusat untuk menyimpan, mengurus dan menggambarkan pelbagai data berkaitan penerbitan, termasuk penerbitan berindeks dan tidak diindeks, indeks H, kiraan petikan, maklumat geran, penjanaan pendapatan, aktiviti pengkomersialan dan kerjasama rangkaian. Projek ini bertujuan untuk menyelaraskan proses pemantauan dan penilaian prestasi penerbitan fakulti, membolehkan pandangan yang lebih baik tentang produktiviti dan impak penyelidikan. Dengan mengagregat dan mempersebahkan metrik penerbitan dalam bentuk graf dan carta, papan pemuka menawarkan antara muka yang intuitif dan interaktif untuk pengguna meneroka dan memahami output ilmiah fakulti. Sistem ini menggunakan pangkalan data untuk menyimpan data penerbitan, yang dikikis dan diproses melalui pelbagai kaedah prapemprosesan data. Data tersebut kemudiannya disusun dan dipaparkan pada papan pemuka dalam masa nyata, memastikan maklumat terkini untuk ahli fakulti, kakitangan penyelidikan dan pihak berkepentingan lain.

## TABLE OF CONTENTS

	<b>TITLE</b>	<b>PAGE</b>
<b>DECLARATION</b>		ii
<b>DEDICATION</b>		iii
<b>ACKNOWLEDGEMENT</b>		iv
<b>ABSTRACT</b>		v
<b>ABSTRAK</b>		vi
<b>TABLE OF CONTENTS</b>		vii
<b>LIST OF TABLES</b>		xi
<b>LIST OF FIGURES</b>		xii
<b>LIST OF ABBREVIATIONS</b>		xiv
<b>LIST OF SYMBOLS</b>		xv
<b>LIST OF APPENDICES</b>		xvi
<b>CHAPTER 1      INTRODUCTION</b>		<b>1</b>
1.1     Introduction	1	
1.2     Problem Background	2	
1.3     Project Aim	2	
1.4     Project Objectives	3	
1.5     Project Scope	3	
1.6     Project Importance	4	
1.7     Report Organization	5	
<b>CHAPTER 2      LITERATURE REVIEW</b>		<b>7</b>
2.1     Introduction	7	
2.2     Current System Analysis	7	
2.2.1    System Pro	8	
2.2.2    Scimago Journal & Country Rank	9	
2.2.3    Google scholar	11	

2.3	Comparison between existing systems	13
2.4	Literature Review of Technology Used	14
2.4.1	IDE	14
2.4.1.1	Visual Studio Code	14
2.4.2	Coding language:	15
2.4.2.1	JavaScript	15
2.4.2.2	ReactJS	15
2.4.2.3	Nodejs	15
2.4.2.4	Puppeteer	15
2.4.2.5	MongoDB	15
2.4.3	Technology used:	16
2.4.3.1	Web Scraping	16
2.4.3.2	Web Development	16
2.5	Chapter Summary	16
<b>CHAPTER 3</b>	<b>SYSTEM DEVELOPMENT METHODOLOGY</b>	<b>17</b>
3.1	Introduction	17
3.2	Methodology Choice and Justification	17
3.3	Phases of the Chosen Methodology	18
3.4	Sprints	21
3.4.1	Sprint 1	21
3.4.2	Sprint 2	22
3.4.3	Sprint 3	22
3.5	Technology Used Description	22
3.6	System Requirement Analysis	23
3.6.1	Hardware Requirements	23
3.6.2	Software Requirements	24
3.7	Chapter Summary	24
<b>CHAPTER 4</b>	<b>REQUIREMENT ANALYSIS AND DESIGN</b>	<b>25</b>
4.1	Introduction	25
4.2	Requirement Analysis	25

4.2.1	Functional requirements	25
4.2.1.1	Use Case Diagram	25
4.2.1.2	Actor Description	26
4.2.1.3	Use Case Description	27
4.2.1.4	Sequence Diagram	30
4.2.1.5	Activity Diagrams	31
4.2.1.6	Use Case Specifications	32
4.2.2	Non-Functional Requirements:	33
4.3	Project Design	34
4.4	Database Design	36
4.5	Interface Design	38
4.6	Chapter Summary	40
<b>CHAPTER 5</b>	<b>SYSTEM DEVELOPMENT METHODOLOGY</b>	<b>44</b>
5.1	Introduction	44
5.2	Development Environment	44
5.3	Implementation	45
5.3.1	View Implementation	45
5.3.2	Coding Implementation	48
5.4	Architecture Implementation	51
5.5	System Testing	53
5.5.1	Black Box Testing	53
5.5.1.1	Test Case Design	54
5.5.2	User Acceptance Testing	58
5.5.2.1	Scraping Functionality	58
<b>CHAPTER 6</b>	<b>Conclusion</b>	<b>60</b>
6.1	Introduction	60
6.2	Achievement of Project Objectives	60
6.3	Suggestions for Future Improvement	61
<b>REFERENCES</b>		<b>62</b>



## LIST OF TABLES

<b>TABLE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
Table 2.1: Comparison between existing systems and Faculty of Computing publication Dashboard		13
Table 3.1: Technology Used		22
Table 3.2: Hardware Requirements		23
Table 4.1: User Description		26
Table 4.2: Use Case Description		27
Table 4.3: Use Case Specifications for Login		33
Table 4.4: Data Dictionary		37
Table 5. 1: Test Case 1 for login		54
Table 5. 2: Test Case 2 for login		55
Table 5. 3: Test Case 3 for login		55
Table 5. 4: Test Case 4 for login		56
Table 5. 5: Test Case 1 for H-index Data		57
Table 5. 6: User Acceptance Testing		58

## **LIST OF FIGURES**

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
Figure 2.1: Search Option		9
Figure 2.2:Details about publications		9
Figure 2.3: Homepage for scimagojr website		10
Figure 2.4: Details about the journals		11
Figure 2.5: Home page for Google Scholar		12
Figure 2.6: Display of results in google scholar		12
Figure 3.1: Workflows of Agile Methodology		19
Figure 4.1: Use Case Diagram of Dashboard		26
Figure 4.2: Sequence Diagram for Login		31
Figure 4.3: Activity Diagram for login		32
Figure 4.4: Architectural Model		36
Figure 4.5. Database Diagram		37
Figure 4.6: Dashboard Page		39
Figure 4.7:Publication Page		39
Figure 4.8: Dashboard for Networking		40
Figure 5 1:Login Page		45
Figure 5 2Grants Bar Chart		46
Figure 5 3: Grants Doughnut Chart		46
Figure 5 4: Networking data Bar Chart		47
Figure 5 5: Conferences Data Line Chart		47
Figure 5 6: MongoDB Connection setup		48
Figure 5 7: Api endpoint for Grants		49
Figure 5 8: Api endpoints for Training Data		49
Figure 5 9Frontend Component of Sidebar		50
Figure 5 10: Frontend Component for Bar Chart		50

Figure 5 11: Architecture Implementation	51
Figure 5 12: Scholars Model for database	52
Figure 5 13: Login Auth Controller	53

## **LIST OF ABBREVIATIONS**

UTM	-	Universiti Teknologi Malaysia
SRS	-	Software Requirements Specification
SDD	-	Software Development Documentation
STD	-	Software Testing Documentation

## **LIST OF SYMBOLS**

## **LIST OF APPENDICES**

<b>APPENDIX</b>	<b>TITLE</b>	<b>PAGE</b>
Appendix A System Requirements Specification		63
Appendix B System Design Documentation		137
Appendix C System Testing Documentation		185
Appendix D Gantt Chart		223

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1 Introduction**

Over the years, technology has advanced to a stage where we are able to make everything organized into a single platform with different components that have been able to solve many real-life problems. Data is one of the most important factors in the advancement of technology in the modern world. Modern technology is dependent on the data. It is without a doubt that technology has simplified our lives and has made changes in the way people used to work or think before. Without proper data collection, no system can be developed whether it is related to technology, science, or any other relevant fields. Data is extracted and refined using certain techniques and methods. Web Scraping is one such technology. Web Scraping is the process involving the extraction of data from a source using automated software and tools. It has grown more popular over the years since its utilization boosts the businesses of different organizations or individuals. In recent years, it has helped different organizations and individuals gain access to useful information that can be used for a wide range of tasks like marketing, research, analysis, etc. For businesses looking to gain a competitive edge in the industry, web scraping has become an essential tool for them.

Web scraping-related technology has advanced significantly over time. Web scraping was a labour-intensive manual procedure in the early days of the internet that entailed manually copying and pasting data from webpages into spreadsheets. The technology evolved in tandem with the growth and complexity of the internet. Various organizations and companies use this technology to analyse their data over the years and work on improving their system in the future. For universities like UTM which is a research-intensive institute, it is very important to keep track of the publications made by the staff of different faculties. One of the main faculties of UTM is the faculty of computing. Web Scraping is one of the very good approaches to keeping track of

the publications made by the faculty of computing staff. With this approach, we can analyse the research output and evaluate the current state. A dashboard created with the data collected from web scraping will be more insightful for viewing the outputs of research and publications made by the staff from the faculty of computing,

In a nutshell, a dashboard can assist faculty members with keeping track of faculty publications, identifying patterns in research output, and automating the process of updating publication data by giving them a comprehensive and up-to-date perspective of research output.

## **1.2 Problem Background**

One of the cornerstones of UTM is the Faculty of Computing. The staff of the faculty of computing not only excels in academic teaching but also in research. University Teknologi Malaysia (UTM) is a research-intensive university. The Faculty of Computing at the Universiti Teknologi Malaysia (UTM) has many faculty members who publish their research papers, articles, and other academic works. Keeping track of all the publications from each faculty member can be a challenging task, particularly when this information is dispersed over numerous websites and databases. It is important to keep track of the publications to do an analysis of the results from the staff of faculty of computing staff. Through this approach, the faculty members can keep track of their latest research updates and can analyse them accordingly to make better progress in the future. It can be challenging to obtain an accurate picture of the faculty's research output because of discrepancies in data and inaccuracies caused by manual reliance on individual faculty members. Through this, faculty members may risk missing out on funding or partnership opportunities as well as possible avenues for research advancement without a thorough and current perspective of research output.

## **1.3 Project Aim**

The purpose of this project is to extract data and develop a web-based dashboard system that will highlight publications, indexed publications, non-indexed

publications, citations and other data like trainings conducted, conferences organized, and agreements done by the faculty to improve the decision-making process and assist faculty members by keeping track of faculty publications, identifying patterns in research output, and automating the process of updating publication data by giving them a thorough and current perspective of research performance.

#### **1.4 Project Objectives**

Setting up the objectives is a key step for developing any project. The objectives of the project are given below:

- a) To extract data on the research performance of faculty members in the Faculty of Computing at UTM.
- b) To develop a Web-Based Dashboard system that will display different research performance data outputs in the form of charts that may include bar chart, pie chart etc.
- c) To validate the data and test the Dashboard system to check if it matches the user requirements gathered from the stakeholder.

#### **1.5 Project Scope**

The scope of the web scraping project for the faculty of computing staff research performance at UTM includes the collection and analysis of publication data from various sources and the creation of a user-friendly dashboard. The project will prioritize data security and privacy while also enabling the faculty to track their own publication output, identify research strengths, and inform decision-making.

- (a) Web scraping data from various sources including the UTM website, utmscholar etc.
- (b) Data Duplication will be handled by applying suitable deduplication techniques and algorithms depending on the variance of the dataset. Manual review and verification will also be done to eliminate duplicate entries of data.

- (c) Creating a dashboard that allows users to view and analyse publication data based on various criteria such as grants secured, publication year, journal, and citation count.
- (d) The project will use appropriate web scraping and data visualization tools based on the requirements and constraints of the project.
- (e) The project will be a web-based system that can be accessed through a desktop or a personal computer with Windows, MacOS and, a stable internet connection.

## **1.6 Project Importance**

The main idea of this project is to provide real-time information on faculty activities using web scraping techniques. The main aim of this project is to give an idea to improve the decision-making process by providing accurate and timely information to the administration and faculty. In today's world, many organizations use the data acquisition technique to make future decisions on the improvement of the system. The type of data that will be collected and included are grants, publications, indexed publications, non-indexed publications, and citations. This sort of data is important for analysing the effectiveness of the research published by the faculty of computing staff.

With web scraping, the analysis of the information gathered can be used to identify trends, patterns, and insights to make informed decisions. The Dashboard developed during the project can be used for a variety of purposes, including monitoring faculty performance, identifying areas for improvement, tracking research activity, and fostering collaboration among faculty. For example, a dashboard can provide information on the number of papers each faculty authored, the number of indexed and non-indexed publications, and the number of citations received. This data will help to identify highly productive faculty members and will guide the allocation of R&D resources.

Additionally, the Dashboard can be used to monitor the status of grant applications. This information can facilitate effective communication and collaboration among faculty members, leading to successful research projects. The dashboard will also provide information about other resources such as income generation, training conducted, and conferences organized. This information will help the faculty members to work on their professional development and can increase the visibility of goals set by the faculty. The administration can also allocate resources more effectively. The dashboard will highlight the networking opportunities which will help the faculty members to recognize the collaboration standards and work on them.

In summary, the project is an important attempt to boost decision-making processes, foster collaboration among faculty members, and enable effective resource management. This project uses web scraping techniques to collect real-time information from various sources that can be used to identify trends, patterns, and insights.

## **1.7 Report Organization**

Chapter 1 provides a summary of the project's background, including the introduction, problem statement, project aim, objectives, scopes, project importance, and report structure. Chapter 2 elaborates on the literature review for this project, which will be used to develop a dashboard containing research publications using web scraping. Chapter 3 describes the methodology of the system development process as well as the hardware and software required to develop the project. All the system's requirement analyses and designs are shown in Chapter 4. The system's implementation (code) and testing are discussed in Chapter 5 to ensure quality. Finally, chapter 6 is the conclusion, which includes a review of the conclusions as well as suggestions for further studies.

## **CHAPTER 2**

### **LITERATURE REVIEW**

#### **2.1 Introduction**

This chapter provides an overview of the literature on existing web-based dashboard systems that are designed to provide information about faculty publications, identify patterns in research output, and automate the process of updating publication data. This literature review will examine the importance of publications and citations in the academic world. The discussion will also cover various web-based dashboard systems that have been developed to solve problems related to the project, including their features and functionalities.

This assessment of the literature is vital to enhance decision-making and help faculty members manage their publication data more effectively by giving a comprehensive and comparative analysis of the existing web-based dashboard system. The study's findings will help in better understanding the project's area and scope as well as the methods, processes, tools, and technologies that are suitable for its design and development.

#### **2.2 Current System Analysis**

For better planning for our system, a comparison of similar systems has been done and measured in conjunction with this system development. To ensure that the new system tackles all the problems while also including all the potential good elements, the systems have been reviewed from every angle to identify their strengths and weaknesses.

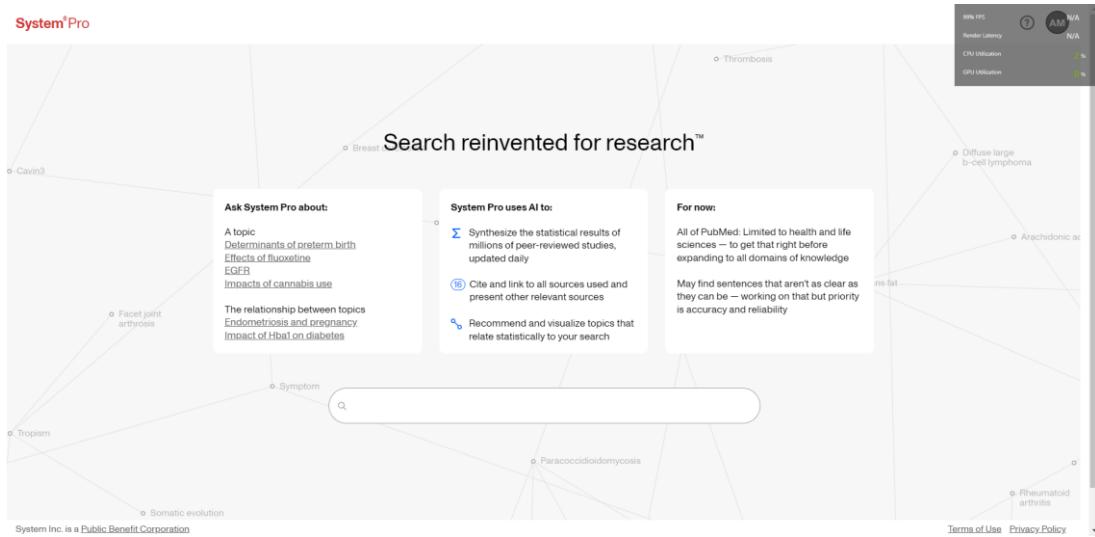
### **2.2.1 System Pro**

System Pro is a web-based tool that enables users to examine and display publishing success indicators for institutions and academics. Users may view the number of publications and partnerships, as well as other performance indicators like citation counts and journal impact factors. Data is gathered from a variety of sources, including academic databases, social networking sites, and institutional archives, by System Pro using web scraping algorithms. For users to analyse and explain their data, the platform provides a variety of visualization tools like network graphs, and scatter plots. Users may focus their study on certain areas of interest by filtering and searching their data using a range of parameters, including author, publication year, and research field. System Pro users may easily share and interact with peers by creating custom reports and exporting data in several formats. The platform provides a selection of subscription packages, including choices for independent researchers, organizations, and funders, as well as specialized business solutions.

In Figure 2.1, we can see that we can search for research articles and publications. It opens to another page which we can see in Figure 2.2, from where we can get an idea about the publications based on our search. From there we can get more detailed information on the publications.

#### **Shortcomings:**

There are no major flaws on the website. But the user interface is not up to the mark. It can be made better with proper analysis of the design. However, it is a commercial platform for which users must pay a fee to access the features and services which makes it less accessible to researchers who have limited funding or resources.



**Figure 2.1: Search Option**

STUDY TITLE	JOURNAL	AUTHORS	RELEVANT FINDINGS	POPULATION	STUDY TYPE	CITED BY	DOI
Student Satisfaction and Self-Efficacy in a Novel Online Clinical Clerkship Curriculum Delivered During the COVID-19 Pandemic	Advances in medical education and practice	Lashley, P., et al	Online teaching [vs face-to-face instruction] is associated with a 40.0 unit decrease in the mean of Effectiveness of online teaching for developing social competencies +1 more	Fourth- and fifth-year medical students who completed an online survey in January 2021.	N/A	3	2022
Academic self-concept mediates the effect of online learning engagement on deep learning in online courses for Chinese nursing students: A cross-sectional study	Nurse Education Today	Zhang, S., et al	Deep learning in online courses is associated with no change in online learning engagement	617 nursing students in 14 schools in eastern, central, and western China from September 2021 to October 2021.	N/A	5	2022
(3) Self-efficacy, good perceived health status is associated with a 4.5 unit decrease to a 0.25 unit increase in preference for e-learning.						2 studies	▼
(4) Use of technology is associated with a 3.01 to 3.08 unit decrease in preference for e-learning.						1 study	▼
(5) Acceptance is associated with a 5.96 unit decrease in preference for e-learning.						1 study	▼
(6) Convenience of e-learning is associated with both decreases and no change in adoption of e-learning.						2 studies	▼
(7) Confidence in using IT devices is associated with a 84.0% decrease in the odds of adoption of e-learning.						1 study	▼
(8) Internet access is associated with a 87.0% decrease in the odds of adoption of e-learning.						1 study	▼
(9) Combine after-hours house-call (ahhc) duties with regular-hours general practice is associated with no change in the odds of online-only participants.						1 study	▼
(10) Satisfaction with online class is associated with a 0.65 unit increase in Cyber-class flow.						1 study	▼
(11) Social networking is associated with a 12.0 to 68.0% increase in the odds of online gaming.						1 study	▼

**Figure 2.2:Details about publications**

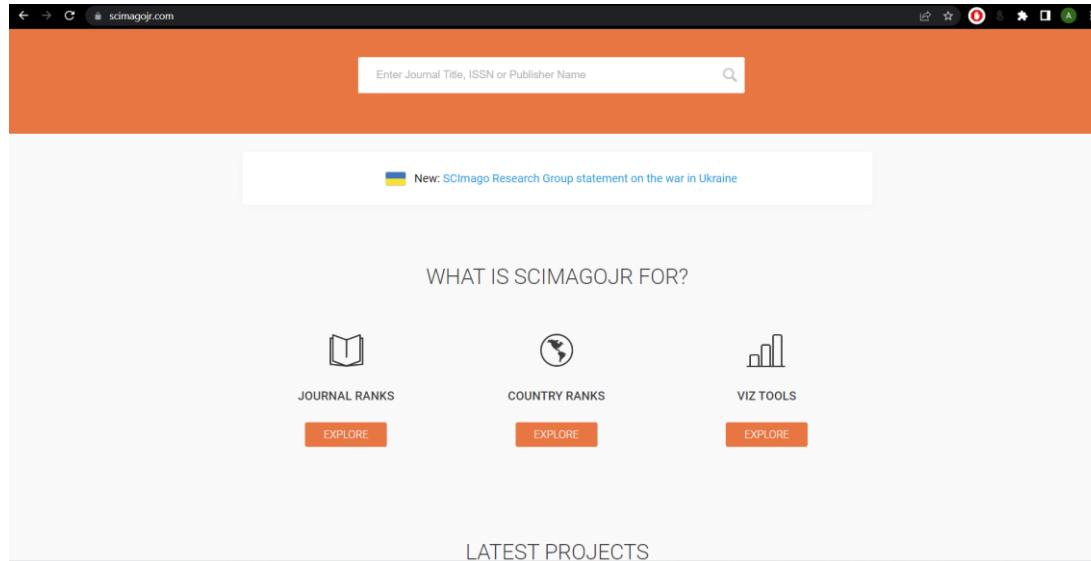
## 2.2.2 Scimago Journal & Country Rank

The website Scimago Journal & Country Rank have several tools that are available to aid researchers in assessing the scientific output of nations, journals, and individuals. The website Scimago Journal & Country Rank offers a platform for comparing the scientific output of nations, journals, and researchers. Using online scraping techniques, it collects data from several sources, such as Scopus, PubMed, and online Science. Scientific journals are ranked according to their impact, influence, and quality

by Scimago Journal & Country Rank. Based on information gathered from the Scopus database, which includes publications from many publishers, the rankings are created. Scimago Journal & Country Rank gives ranking to nations according to their contributions to science, citations, and partnerships. The rating is based on information gathered from different sources.

### **Shortcomings:**

The interface of the website is user-friendly, but it can be more organized to help the users navigate more easily. The primary focus of Scimago Journal & Country Rank is on the production of scientific research in the engineering and natural sciences. The website may not be as helpful for researchers in other domains when assessing the effectiveness of their work. Overall, the website can be helpful for anyone to go through but there are sides open for development including the setup of a dashboard showing the statistics.



**Figure 2.3: Homepage for scimagojr website**

SJR		Scimago Journal & Country Rank					Enter Journal Title, ISSN or Publisher Name				
		Home	Journal Rankings	Country Rankings	Viz Tools	Help	About Us				
<b>Online Learning Journal</b>											
A journal with a high impact factor, indicating its influence and quality in the field of education.											
<b>COUNTRY</b>	<b>SUBJECT AREA AND CATEGORY</b>			<b>PUBLISHER</b>	<b>H-INDEX</b>						
United States	Computer Science — Computer Networks and Communications Social Sciences — Education			The Online Learning Consortium	<b>60</b>						
Universities and research institutions in United States											
Media Ranking in United States											
<b>PUBLICATION TYPE</b>	<b>ISSN</b>	<b>COVERAGE</b>		<b>INFORMATION</b>							
Journals	24725730	2015-2022		<a href="#">Homepage</a> <small>Check if available in this journal</small>							

**Figure 2.4: Details about the journals**

### 2.2.3 Google scholar

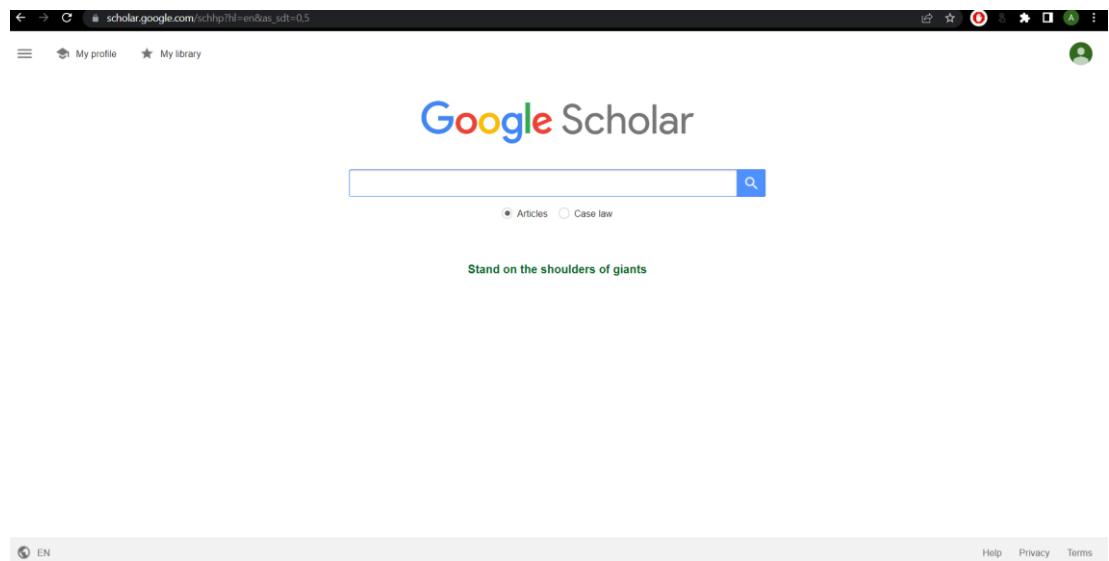
Google Scholar is an extremely popular web-based search engine that searches scholarly materials like academic publications and indexes them. The scholarly literature, including articles, theses, conference papers, and other publications, is covered by Google Scholar. It has an advanced search option through which users may search for articles that reference authors or papers as well as filter their search results by author, publisher, date, and keyword. The best part of using google scholar is that anyone with an internet connection can use Google Scholar without paying a fee. The search results are displayed in an accessible and organized way, and the interface is intuitive and simple to use. Its integration with a variety of university library management systems has enabled its users to quickly access full-text articles and other resources that may be offered by their institution.

#### Shortcomings:

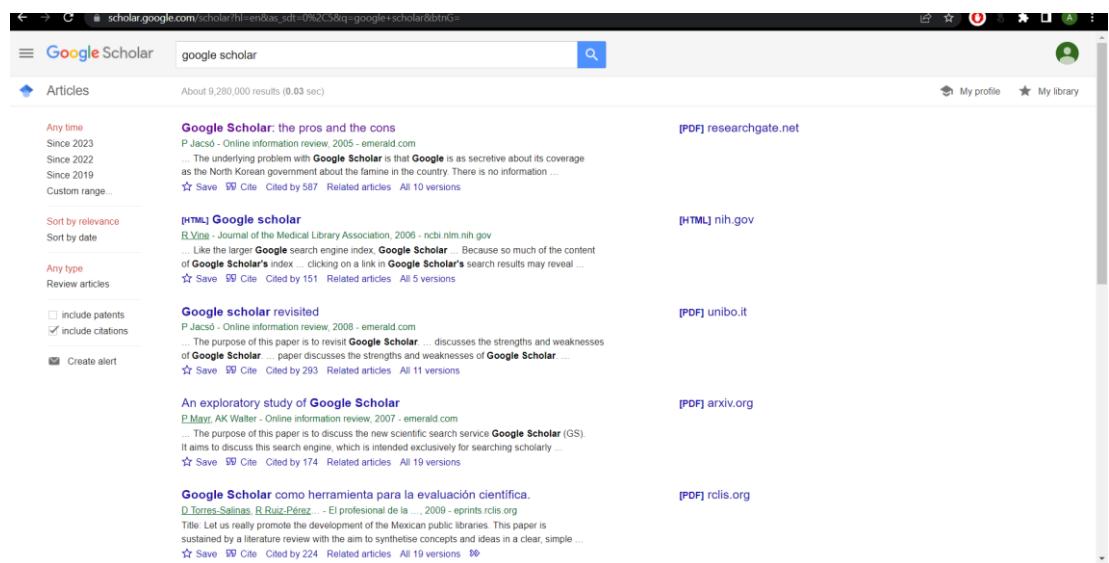
Although Google Scholar provides a very intuitive interface and has numerous good features, it doesn't do data analysis based on articles or publications. The coverage provided by Google Scholar is extensive but not exhaustive. The standards by which Google Scholar determines whether a result is "scholarly" are not stated. The researcher must decide whether outcomes are appropriate for their objectives because results are frequently of varying quality. Users of Google Scholar are unable to filter

results by discipline, full-text, or peer-reviewed sources. Overall, it doesn't help any organization/individual by providing them a thorough analysis of their research and doesn't provide any data analysis dashboard based on them.

The dashboard developed for the faculty of computing staff will do data analysis based on the publications made which makes it unique and different.



**Figure 2.5: Home page for Google Scholar**



**Figure 2.6: Display of results in google scholar**

### 2.3 Comparison between existing systems

Each of System Pro, Scimago Journal & Country Rank, and Google Scholar have benefits and drawbacks of their own. But what sets each of them apart from the others is their area of expertise. Here are some comparisons of the characteristics and functions of the three systems, therefore.

**Table 2.1:Comparison between existing systems and Faculty of Computing publication Dashboard**

Properties	System Pro	Scimagojr	Google Scholar	FC Dashboard
<b>System Type</b>	Web-based	Web-based	Web-based	Web-based
<b>Interface</b>	Data visualization is a limited	User-friendly Interface but lacks organization	Very user-friendly interface	User-friendly interface with beautiful themes and organized features
<b>Language</b>	English	English	English	English
<b>Data Analysis</b>	Yes	Yes	No	Yes
<b>Reliability</b>	Fair	Average	Average	Good
<b>Uniqueness</b>	Examines and displays publishing success for institutions and academics	Assess scientific research of individuals and organizations	Display scholarly materials	Dashboard that'll provide insight on publication performance among Faculty of Computing researchers
<b>Analytical Dashboard</b>	No	Yes	No	Yes

Based on the comparison Table 2.1, it can be concluded that the Dashboard developed for the faculty of computing staff to check on their publication performance is an excellent platform when compared to the alternatives. The faculty of computing staff will be benefitted to do a study of their publication works and thus think of further development. This project aims to help the faculty of computing staff to get an insight into their work and assess the current condition of their publication methods. In comparison to similar systems, it can be said that our system will be useful for a certain organization to follow up with their work in their own field.

## **2.4 Literature Review of Technology Used**

Fast advancements in technology are occurring. With the aid of technology, the world is now limitless, and everything is possible. Software development has grown to the point that it is the most sophisticated and in demand in the ever-evolving world of technology. To stay up with the development of time and technology, a technical study was done to determine the best technologies accessible for this project. In this project, the following technologies will be utilized:

### **2.4.1 IDE**

#### **2.4.1.1 Visual Studio Code**

Microsoft's Visual Studio Code (VS Code) is a free and open-source code editor, and it is a popular choice among developers because of its versatile nature, making it one of the favourite choices for web development, app development, and machine learning projects.

## **2.4.2 Coding language:**

### **2.4.2.1 JavaScript**

We will be using JavaScript for our project to carry out web scraping from different sources like utm scholar, utm research website etc. JavaScript is a popular programming language used by developers for web scraping and development. There are many libraries and tools available for scraping data from websites in JavaScript. Some of the notable libraries are puppeteer, cheerio, etc.

### **2.4.2.2 ReactJS**

We will be using ReactJS for the frontend of the system. ReactJS is a popular JavaScript library. The user interfaces of the system will be built using ReactJS.

### **2.4.2.3 Nodejs**

Nodejs is a popular JavaScript library. It is open source so the usage of it is very wide and popular. It is very fast, secure and highly scalable.

### **2.4.2.4 Puppeteer**

Puppeteer is a popular web scraping tool that can be used to scrap data from any website. It is popular JavaScript library that is used for web scraping.

### **2.4.2.5 MongoDB**

MongoDB is a popular database management system. It is a NoSQL database management system that is designed for scalability, flexibility and high performance.

### **2.4.3 Technology used:**

#### **2.4.3.1 Web Scraping**

Web Scraping is the process involving the extraction of data from a source using automated software and tools. This technology helps in doing data analysis since it collects data from different sources and puts them together in one place. The main purpose of this technique is to have a thorough understanding and analysis of the data collected from different sources.

#### **2.4.3.2 Web Development**

For developing the web-based system, we'll be using technology like Nodejs and ReactJS library. ReactJS is used for developing user interfaces. Meanwhile, Nodejs can be used to work on the backend of the system. MongoDB will be used for the database management of the system.

### **2.5 Chapter Summary**

In conclusion, this chapter described similar systems related to my project that have been studied and researched for strengths and shortcomings that need to be acknowledged. This research will aid in the development of the dashboard that'll come in handy for the faculty of computing staff and will provide insight into their publications.

## **CHAPTER 3**

### **SYSTEM DEVELOPMENT METHODOLOGY**

#### **3.1 Introduction**

Software development methodology implies to the structured processes involved while working on a project. In simple words, the major objective of the methodologies is to provide a well-structured approach to software development. Planning is very important when it comes to executing a project. It can save time and money at the same time. Software development is a complex process that involves proper planning, sufficient resources, a budget, and skilled developers to complete the project. It can even lead to software failure if proper planning for software development is chosen. As a result, it is important to have excellent software planning for the project.

To achieve and ensure the development process of the software and its related products, a software development process is described as a collection of processes, actions, activities, and controls Young (2013). There are several software development approaches. Some of them have shown effective results and are used extensively in developing software by developers. Some of the notable approaches are Waterfall method, Incremental method, Prototyping, Agile, and Rational Unified Process (RUP). These methods consist of seven different stages which is known as the Software Development Life Cycle (SDLC). In this chapter, we'll be discussing the approach that'll be taken for implementing the project.

#### **3.2 Methodology Choice and Justification**

The system development methodology is an orderly process used in executing the basic steps necessary for analyzing, designing, implementation, and maintaining the system efficiently.

Quite a few numbers of approaches can be utilized for effective project planning as well as execution. Considering the underlying system requirements, the Agile methodology would be the most feasible approach toward software development for this project.

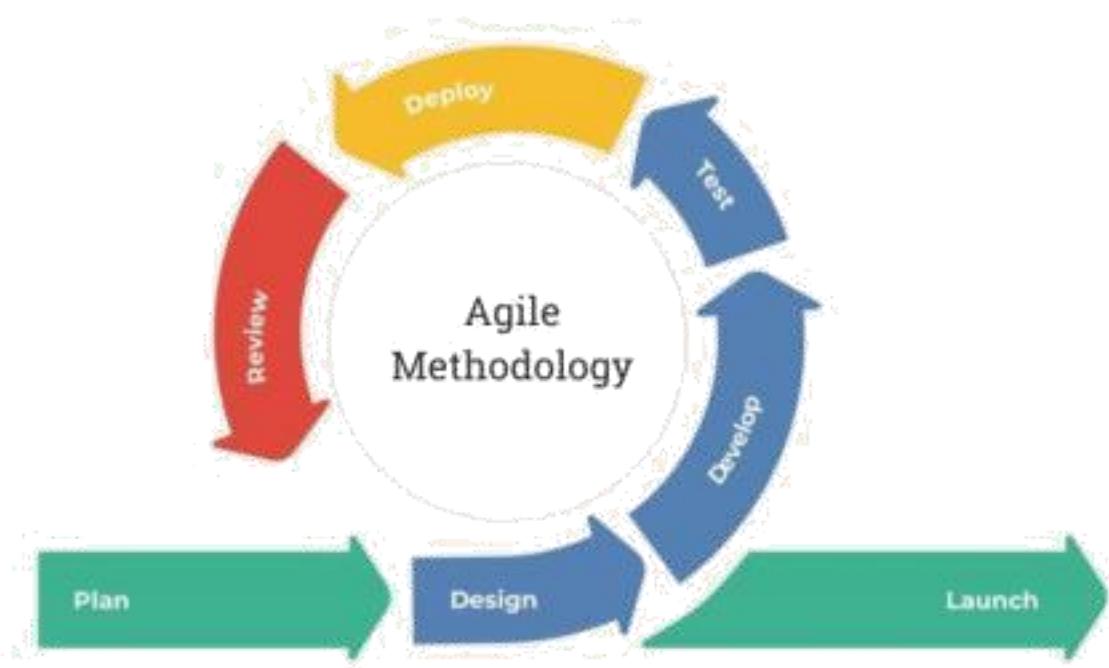
In Agile methodology, the project is broken down into small pieces of user functionalities. These functionalities can be organized by prioritizing their importance and setting a delivering time limit to complete them. They can be divided into weekly cycles which known as iterations or sprints. The goal of the sprint needs to be specified before every cycle. By making an analysis of the requirements, the functionalities will be divided and developed sequentially by giving highest priority to the most important ones. In this way, the development process will be easier and smooth for the developer to develop.

Agile methodology is best suited for our project because of its phases. It is easier to implement the agile methodology as well. The use of sprints to complete tasks in Agile methodology makes it more suitable for the implementation of the project since it will help in saving time, money and dividing the tasks according to their priority. It has been seen from previous projects that by embracing the Agile Methodology, organizations can experience numerous benefits. They include more adaptability, a shorter time to market, higher customer satisfaction, better software quality, and lower project risks. Due to its endless benefits, it can be concluded that Agile methodology is the best suited for my project.

### **3.3 Phases of the Chosen Methodology**

Agile methodology follows a cyclical and iterative approach to implement the development of the software. It typically consists of six phases. The phases include the planning, Designing, development, testing, deployment and reviewing. The flow can be viewed from the figure below.

# AGILE METHODOLOGY



**Figure 3.1: Workflows of Agile Methodology (source: interqualitybg.com)**

### 3.3.1. Planning

Planning is the stage of the system where requirements set by the stakeholder are reviewed by the developers. The developer team reaches a conclusion with the client on how the software/system should look like. After that, the developer divides the various tasks that are needed to develop the system. In this stage, the priority and division of the tasks is done. A detailed planning for the iterations or sprints is made during the planning stage. The items for the development can be updated and viewed in the backlog. The backlog is updated during this phase.

as well. For each sprint and iteration, a goal is set to be completed by the end of that sprint. This includes the timeline for completing the tasks, the percentage of tasks that need to be completed and any other constraints or dependencies. During this phase, communication with the stakeholder is done profusely to meet the goals set for each sprint.

### **3.3.2. Design**

Design is the stage where the system blueprint is designed based on the analysis of the requirements and results of the analysis. The prototype of the system is developed in the design stage. A detailed description of different features and operations of the system is carried out in this phase as well. Developing the prototype is a key task that is to be completed in the design phase. It is also important to develop the prototype based on the requirements set by the stakeholders but also based on the flexibility of the developer. So, developing the prototype, validating the results of the analysis pinned during the planning stage are to be done during this phase.

### **3.3.3. Development**

This is the part where the development of software is carried out according to the plan by the developer. Developers will start developing the system using the programming language that has been chosen for developing the system. During this stage, it involves the development of different modules for building the system. For example, it involves connecting the database to the backend and designing the frontend interface with which the services or logic will be communicated at the backend. Development of software is done accordingly as mentioned in requirements and guidelines devised in the previous steps of the project.

### **3.3.4. Testing**

In this stage, the testing of the software is done. The testing of the software is very important to avoid software failure. It has been seen in the case of many software companies that due to the lack of proper testing; it has failed during the launch of the software. That is why this phase plays an essential role to avoid software failure and to ensure software success. Software quality is assured and maintained in this phase.

### **3.3.5. Deployment**

Deployment is the stage where the software is ready to be placed in the market after getting clearance from the software quality control team. It comes after the testing phase since the quality and percentage of failure/success are determined in the testing phase. The software will be ready for deployment once the requirements are met.

### **3.3.6. Review**

The review stage involves user feedback, either on bugs that have not been detected during testing or the addition of more features. This can be considered a paramount stage for the success of launching software, since the basis for developing any kind of software is getting user input. The system is for the user, and any assurance of the user experience will determine if positive reviews come forth. This marks an entry for the commencement of another software development cycle, taking into consideration bug fixes, plans for iterative development, or feature updates for subsequent releases.

## **3.4 Sprints**

### **3.4.1 Sprint 1**

The implementation of the design of the system was done in first sprint. As this project involves web scraping, the inspection of the website for data extraction was done in the first sprint. Besides, building up the codebase including the setting up different libraries and dependencies for the

framework was done in this sprint as well. Some of the front-end components were organized in this sprint as well.

### **3.4.2 Sprint 2**

The second sprint involves the creation of the project's backend structure, creating a dashboard and a few other modules for displaying the data that was scraped. The Mongoose Models were created for database integration with MongoDB that helped in loading the data from the database and display in the form of graphs. The visualization of the data was mostly done in this sprint.

### **3.4.3 Sprint 3**

The correct flow of the system was utilized during this sprint. Scraping of the website with zero error and with proper data storage into the database was achieved within this period. The visualization of data in the dashboard was achieved as well during this sprint. Finally, testing with stakeholders' feedback was carried out by using the system in count toward further system improvements.

## **3.5 Technology Used Description**

This section will briefly explain the required tools and technology used for the development of the proposed system which is the faculty of computing staff publication dashboard.

**Table 3.1: Technology Used**

<b>Technology</b>	<b>Purpose</b>	<b>Type</b>
Windows	Operating System	OS
Google Drive	Documentation	Cloud Storage

Draw.io	Design	UML Diagram Editor
Figma		UI and UX Design
Visual Studio Code	Development	IDE
ReactJS	Stack	Frontend, JavaScript Library
Material UI		UI component Library
Tailwind CSS		CSS framework
NodeJS		Backend, JavaScript Library
MongoDB		DBMS
Puppeteer	Scraping Tool	JavaScript Scraping Library

### 3.6 System Requirement Analysis

The system involves the use of several hardware and software peripherals for the design, development, and evaluation of the system. Correct usage of hardware and software tools assures the correctness in the working of the system and compatibility.

#### 3.6.1 Hardware Requirements

A computer or laptop is needed for documentation and, development.

**Table 3.2: Hardware Requirements**

Component	Requirements
Processor	Ryzen 7
RAM	8GB
Operating System	Windows 7 / 8 / 10 / 11 (32-bit or 64-bits)
HDD	20GB
Available Disk Space	100GB

Internet connection	Wifi
---------------------	------

### **3.6.2 Software Requirements**

Software Requirements must be met for developing the system.

#### **3.5.2.1. Vs Code**

VS Code, or Visual Studio Code, is a lightweight code editor by Microsoft that is open source. It supports various programming languages with complete syntax highlighting, debugging, and integration with Git. With huge libraries for a variety of extensions, VS Code, therefore, offers a customized smooth way to build and manage applications for developers.

#### **3.5.2.2. Microsoft Word**

Microsoft Word is the best and commonly used software for documentation. It has very powerful features which give the user a way of writing more effectively and efficiently. The documentation of the project will be done using Microsoft Word software.

### **3.7 Chapter Summary**

The agile method which is our system methodology that was adopted for the development of our system has been described in this chapter. The phases that come with the method have also been described. Moreover, the utilization of the technologies that are required for developing this project along with the hardware requirements have been described in this part of the report as well.

## **CHAPTER 4**

### **REQUIREMENT ANALYSIS AND DESIGN**

#### **4.1 Introduction**

This chapter describes the requirement analysis and the design of the system. Analysis of essential functional and non-functional requirements gathered during the design process is also discussed in this chapter. Essential Design Elements are discussed in this chapter. For example, a use case diagram is used to understand the requirements of the whole system. Other diagrams like sequence and activity diagrams are used to represent the use case in further detail. The chosen architecture style, database design and the interface designs will also be discussed in this document.

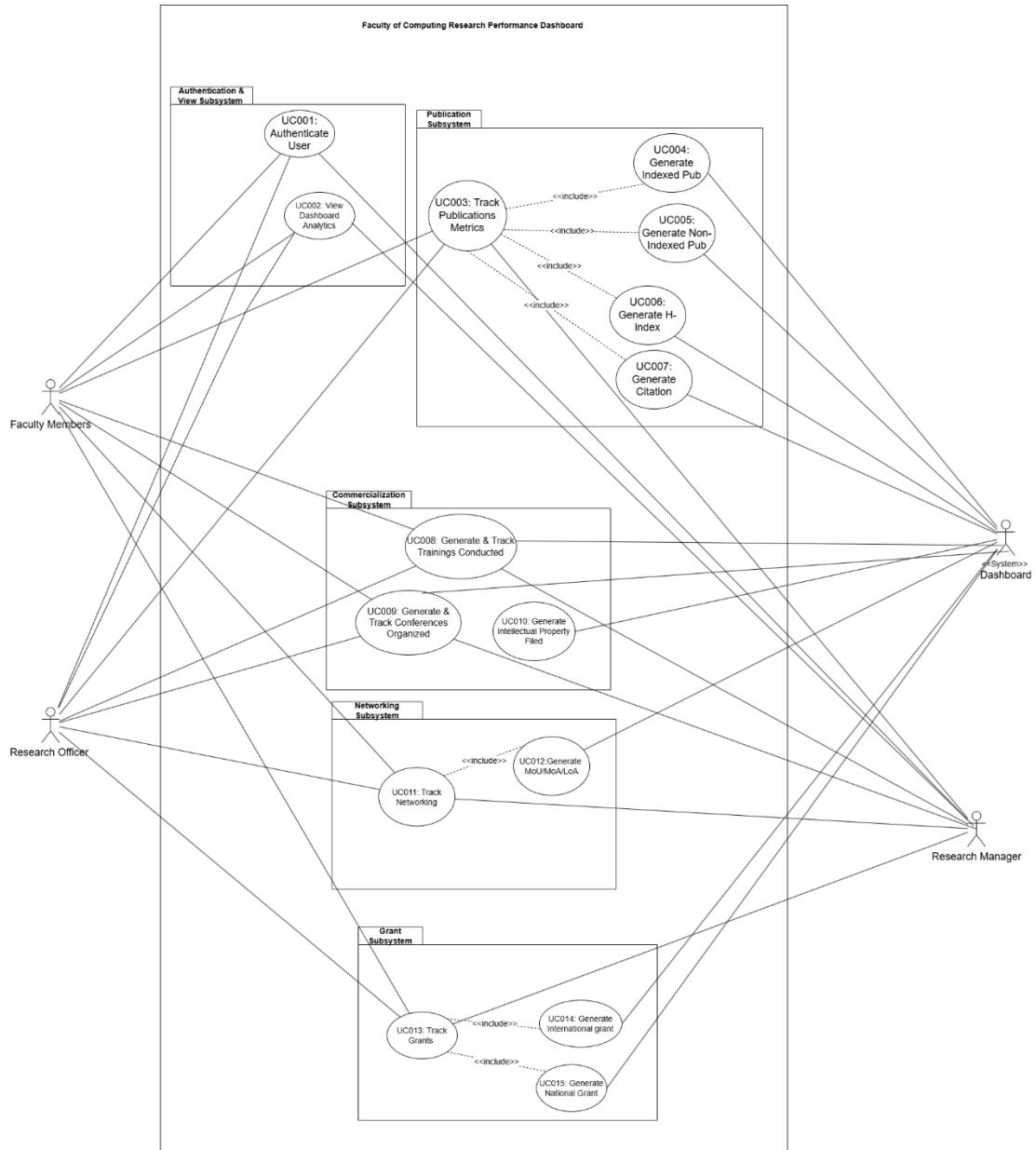
#### **4.2 Requirement Analysis**

The functional and non-functional requirements of the system are discussed in this section. These requirements were gathered from the stakeholders of the system.

##### **4.2.1 Functional requirements**

###### **4.2.1.1 Use Case Diagram**

The use case diagram represents all the requirements of the system. The system features and functions are shown in the use case diagram as well. The requirements are set after finalizing with the stakeholder. The use case diagram is the primary diagram for depicting all the necessary features and functions of our system.



**Figure 4.1:** Use Case Diagram of Dashboard

#### 4.2.1.2 Actor Description

The description of the actors of the use case given above is given below:

**Table 4.1:** User Description

No.	User	Characteristics
-----	------	-----------------

1.	Faculty Members	This user is the main part of the system. The system is basically built for the usage of the faculty members. The faculty members will be able to get an insight of their research work and what measures can be taken to do improvement on research by the faculty of computing staff.
2.	Research Officer	Research officer plays a crucial role in our system design. All the data that will be collected will have to go through the research officer for filtering it properly before displaying in the system. This role is one of the major parts of the project. Research officer will help in data filtering before displaying in the form of a dashboard.
3.	Research Manager	Research manager will help in collecting the data that will be displayed in the dashboard. Research manager is the one who manages the data related to research. So, the role of the research manager is very important for carrying out the development of the system.

#### 4.2.1.3 Use Case Description

The elements of the use case are described in the table below:

**Table 4.2: Use Case Description**

Module	Product Function	Description
Authentication and View	Authenticate User	It will allow the user to login to the system.
	View Dashboard Analytics	It will allow the user to view the dashboard

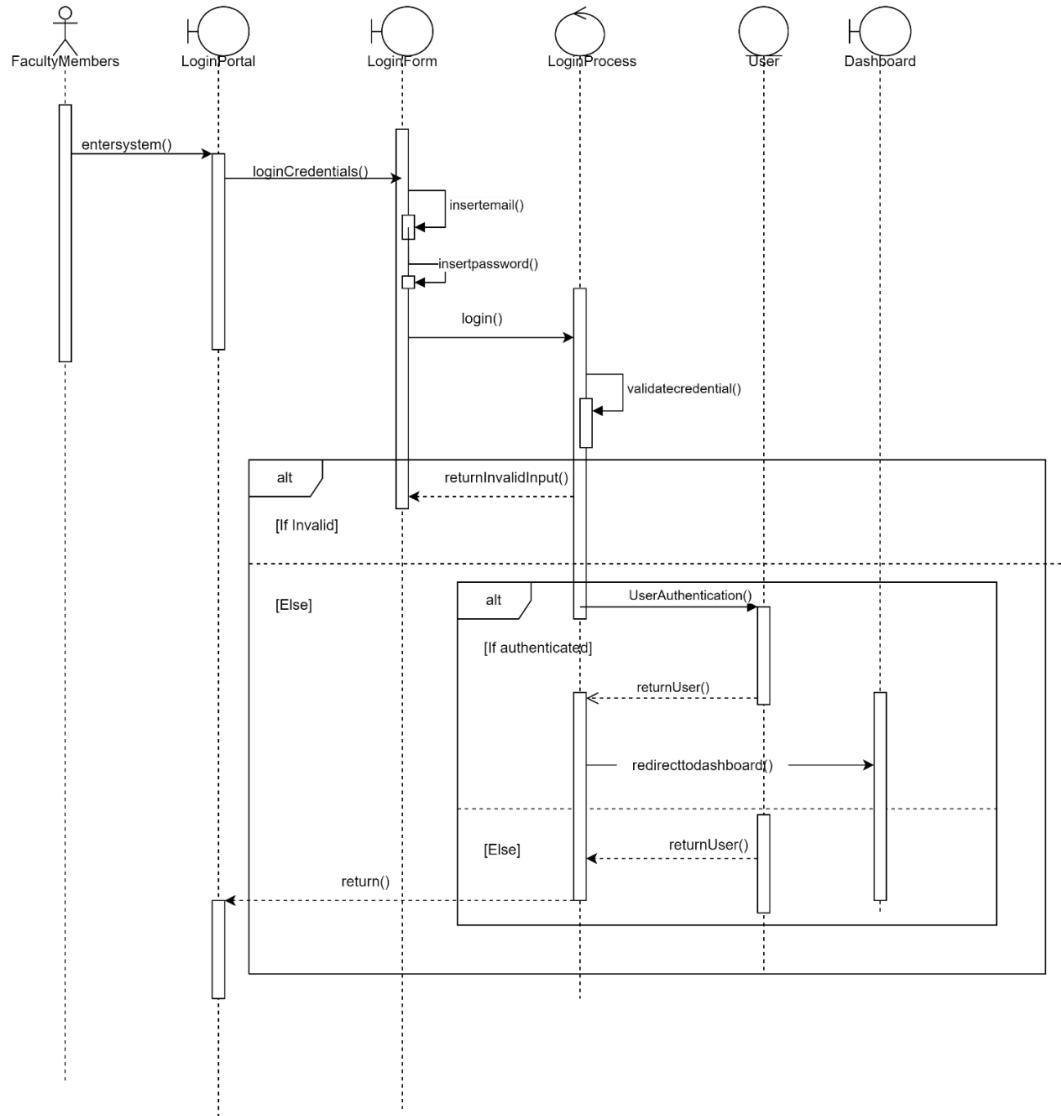
		analytics of different research related data.
Publication	Track Publication Metrics	This will allow the research staff to track the publication metrics by reviewing the data related to publication.
	Generate Indexed Publication	This will be generated by the system dashboard by using data science techniques for the users to view their update.
	Generate Non-Indexed Publication	This will be generated by the system dashboard by using data science techniques for the users to view non-Indexed publications.
	Generate H-Index	This will be generated by the system dashboard by using data science techniques for the users to view H-Index data generated.
	Generate Citation	This will be generated by the system dashboard by using data science techniques for the users to view Citations made data.
Commercialization	Generate & Track Trainings Conducted	This will allow the research staff to track the Income generation by

		reviewing the data related to trainings organized.
	Generate & Track Conferences Organized	This will allow the research staff to track and edit the Conferences data by reviewing the data related to Conferences.
	Generate Intellectual Property Filed	This will be generated by the system dashboard by using data science techniques for the users to view the number of intellectual property filed for the researchers.
Networking	Track Networking	This will allow the research staff to track and edit the Networking data by reviewing the data related to networking.
	Generate MoU/MoA/LoU data	The system dashboard will generate data related to the agreement/letter of undertaking by using data science techniques for the users to view and evaluate on the current condition.
Grant	Track Grant	This will allow the research staff to track and edit the Grant data by reviewing the data related to Grant management.
	Generate International grant Data	The system dashboard will generate data related to the

		international grant that has been received by using data science techniques for the research staff to view and evaluate on the current condition.
	Generate National grant Data	The system dashboard will generate data related to the national grant that has been received by using data science techniques for the research staff to view and evaluate on the current condition.

#### **4.2.1.4 Sequence Diagram**

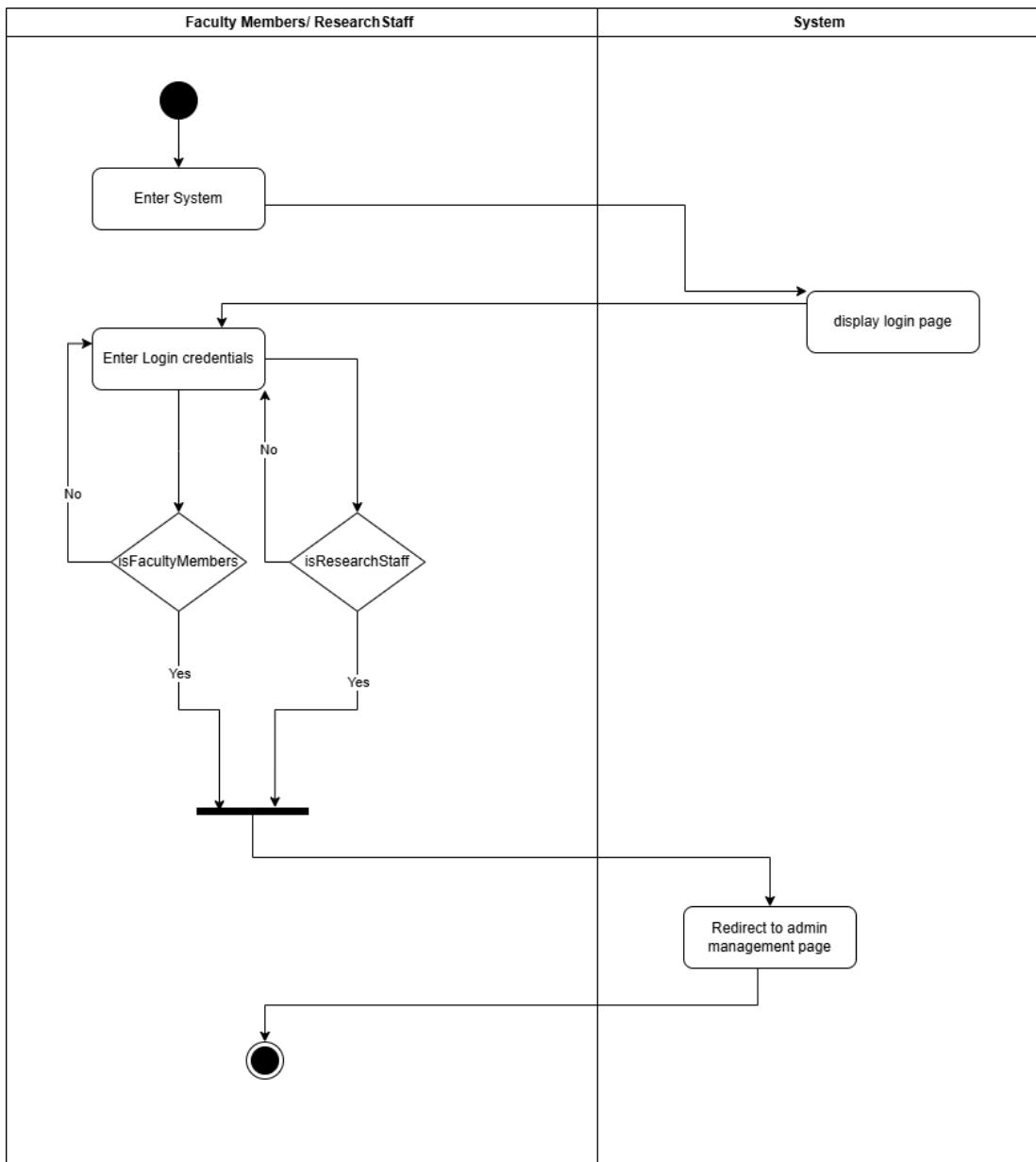
Sequence diagrams are detailed design of a use case should work. The sequence of events that'll be happening while using the system are depicted in sequence diagrams. For each use case, a sequence of events is maintained. The sequence of events for each use case are depicted in sequence diagrams. Detailed sequence diagrams are given in the SRS.



**Figure 4.2: Sequence Diagram for Login**

#### 4.2.1.5 Activity Diagrams

The activity diagrams are illustrations of the workflow of a use case diagram. The activity diagram depicts the flow of the system while a user is making the system. Below is just one example of the activity diagram for this system.



**Figure 4.3: Activity Diagram for login**

#### 4.2.1.6 Use Case Specifications

Use case specifications are tabulated form of in-depth analysis of each of the use cases. The Use Case Specifications consist of the normal, alternative and exception flow through which flow of the use case are organized. It also consists of all the details related to the use case. The actors related to the use case are also shown in the Use Case Specifications. Below is an example of the Use Case Specifications for our system:

**Table 4.3: Use Case Specifications for Login**

<b>Use case ID</b>	UC001
<b>Use case name</b>	Authenticate User
<b>Description</b>	This use case describes flow for user log in into the system.
<b>Actor</b>	Faculty Members, Research officer and Research Manager
<b>Pre-condition</b>	1. There is an active network connection to the platform
<b>Normal Flow(s)</b>	<ol style="list-style-type: none"> <li>1. User enters the system.</li> <li>2. Login page is displayed.</li> <li>3. User enters the email and password.</li> <li>4. User clicks on the ‘Login’ Button.</li> <li>5. System validates the user.</li> <li>6. If user enters wrong or invalid email or password, then exception flow 1 and exception flow 2 are performed. Otherwise proceed to normal flow 7</li> <li>7. User is logged in.</li> <li>8. Use case ends</li> </ol>
<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<ol style="list-style-type: none"> <li>1. <b>Invalid email</b> <ol style="list-style-type: none"> <li>1.1 The system displays invalid email message.</li> <li>1.2 Normal Flow 3 is executed again.</li> </ol> </li> <li>2. <b>Invalid Password</b> <ol style="list-style-type: none"> <li>2.1 The system displays invalid password message.</li> <li>2.2 Normal Flow 3 is executed again.</li> </ol> </li> </ol>
<b>Post-Conditions</b>	1. User successfully login to the system.
<b>Related Requirement</b>	1. View Dashboard Analytics

#### **4.2.2 Non-Functional Requirements:**

The non-functional requirements are those requirements that help a system to work better. Some of the non-functional requirements required for our system are given below:

- 1) Reliability: Reliability: The system should be dependable, with consistent and accurate operation. It should elegantly manage mistakes and exceptions, recover from failures, and minimise data loss or corruption. The system should also have backup and recovery techniques to prevent data loss and assure system availability.
- 2) Security: The system must prioritise data and user information security. It should put in place proper authentication and authorisation methods to ensure that only authorised people may access the data. In addition, the system should use secure data transmission methods to protect data during transit and comply with applicable data protection and privacy standards.
- 3) Usability: The system should be simple to use and intuitive, allowing faculty and research staff to explore and engage with the dashboard and its features. The user interface should be developed in way that will ensure users can simply obtain needed information, perform actions, and comprehend visualisations without confusion or excessive complexity.
- 4) Portability: The system should be built to be portable, allowing it to work on multiple operating systems or environments. It should make use of technologies and frameworks that are cross-platform compatible, ensuring flexibility in deployment options.

### **4.3 Project Design**

The chosen architectural style for this system is the MVC architecture. The MVC architecture refers to the Model-View-Controller architecture. The MVC architecture brings numerous benefits to the system. The system component is divided into three major parts. The three parts are model, view and controller. This architecture pattern makes it easier to manage code because of its structure.

The description of its three different parts is given below:

#### **Model:**

The business logic of the application is represented by the model layer. It focuses on the data structures and data operations which is one of the crucial factors for this system. Its sustainability is one of the big factors in choosing this architectural style.

- Represents the business logic of the application.
- Data storage and retrieval is handled in this layer.
- If there is any change in data, this layer notifies the observer.

#### **View:**

The presentation layer is represented by the view layer. It is responsible for displaying the interface and data to the user. It receives data from the model and works as a presentation layer of the collected data in the form of tables, charts and so on.

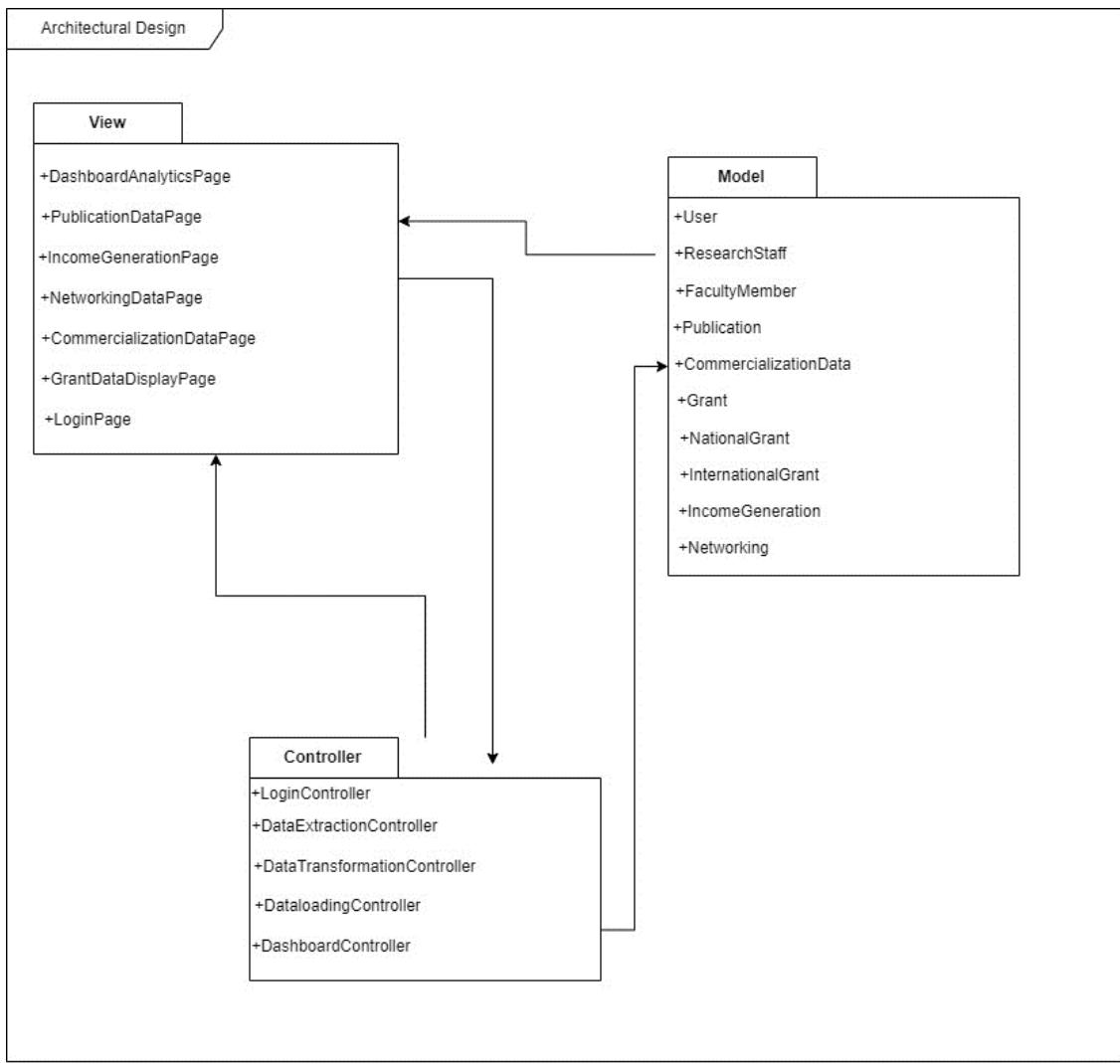
- The data is displayed by this layer.
- Works as a presentation layer.
- User interface elements are rendered in this layer.
- It passes user input to the controller.

#### **Controller:**

The controller acts as a bridge between the model and view layer. It interacts with the model layer to update or retrieve data and interacts with the view layer to reflect the changes made.

- It receives user input, and it performs actions based on the user input.
- Interacts with the model to collect data.
- Interacts with the view layer to reflect changes made in the model layer.
- It controls the data flow between the model and the view layer.

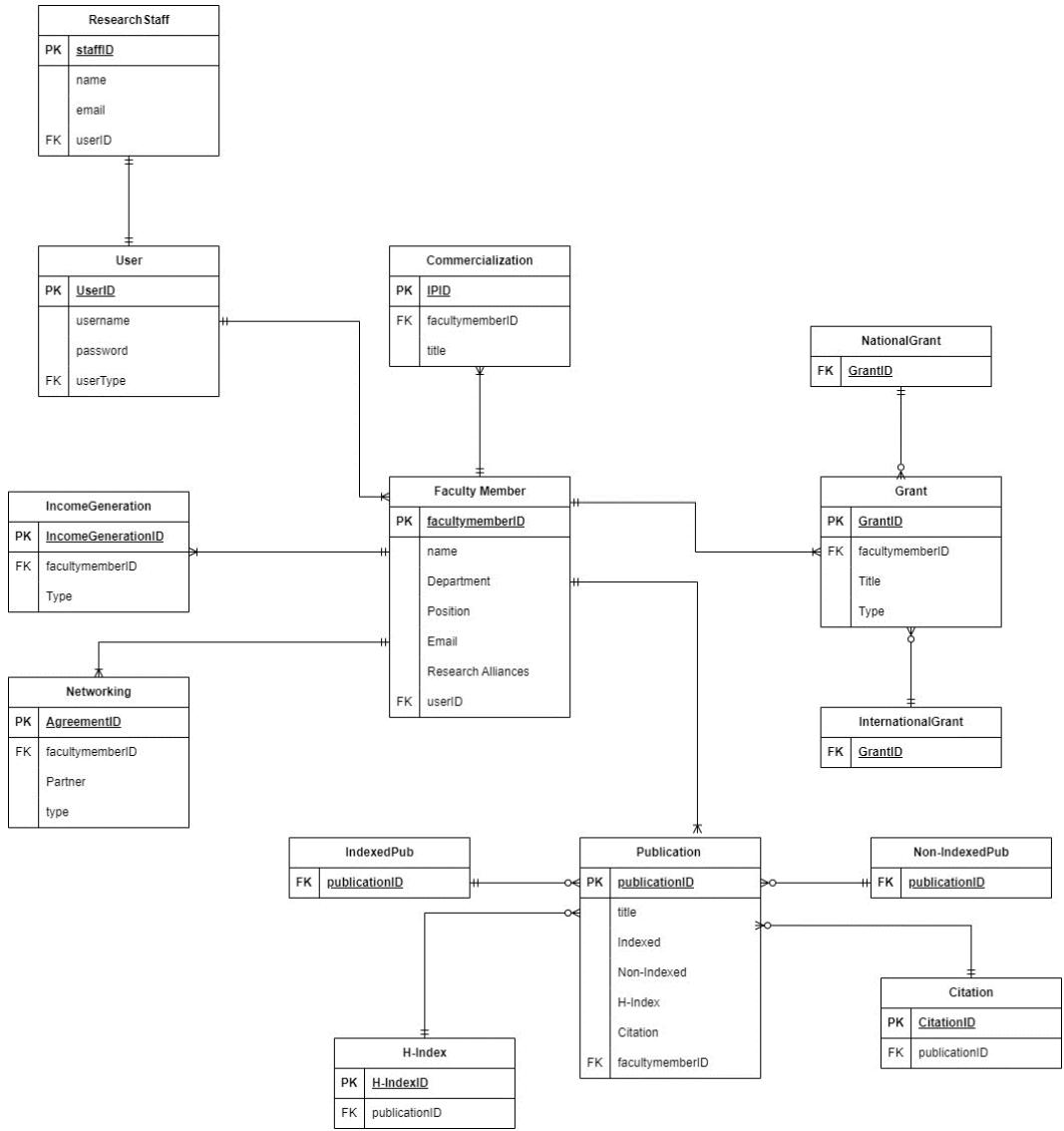
The reason for choosing this architectural model is due to its durability, testability, reusability, and flexibility.



**Figure 4.4: Architectural Model**

#### 4.4 Database Design

The database design is one of the main parts of our system. Database requires the data to be saved in a proper format. The format is depicted in the database design. For developing our system, database design is a major part. The database design of our system is given below:



**Figure 4.5. Database Diagram**

Data Dictionary:

Below is the data dictionary designed for three fields:

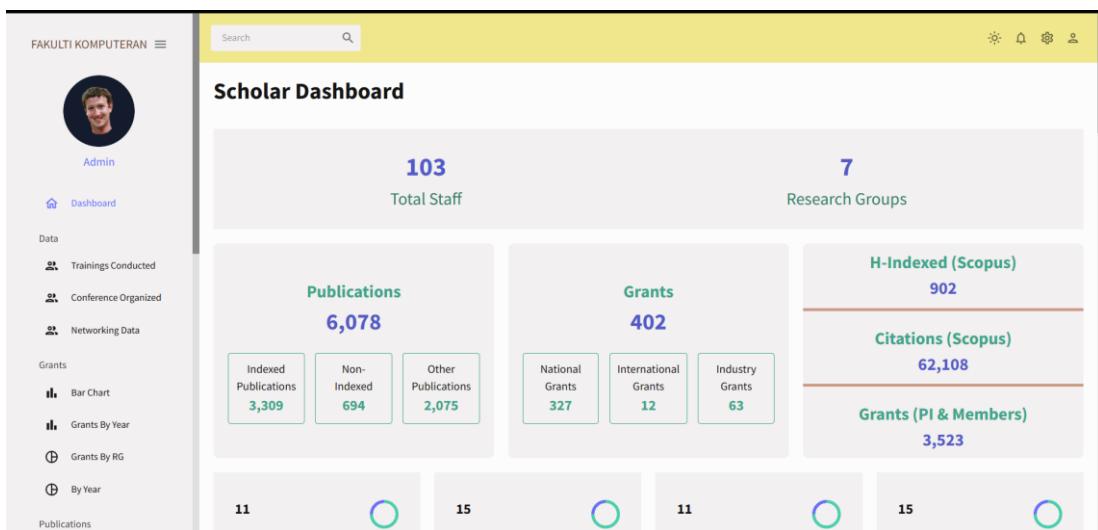
**Table 4.4: Data Dictionary**

Field Name	Datatype	Constraint	Description
<b>User</b>			
UserId	INT	Primary Key	Unique ID for user
username	VARCHAR	Not NULL	Username of the user

password	VARCHAR	Not NULL	Password of the user
userType	VARCHAR	Foreign Key	Unique type of user
<b>Research Staff</b>			
staffID	INT	Primary Key	Unique ID for user
name	VARCHAR	Not NULL	Name of the user
email	VARCHAR	Not NULL	Email of the user
userID	INT	Foreign Key	Unique ID for user
<b>Income Generation</b>			
IncomeGenerationID	INT	Primary Key	Unique ID for Income Generation field
facultymemberID	INT	Foreign Key	Unique ID of the faculty member
Type	VARCHAR	Not NULL	Type of Income Generation

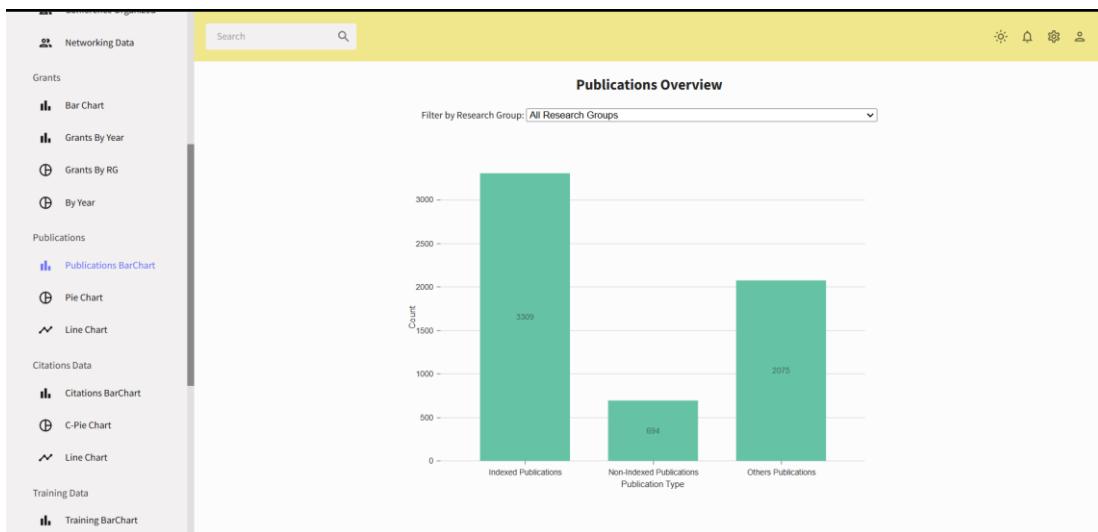
#### 4.5 Interface Design

Users will be taken to a login page to enter the system. From the login page, the user will be taken to a dashboard home page.



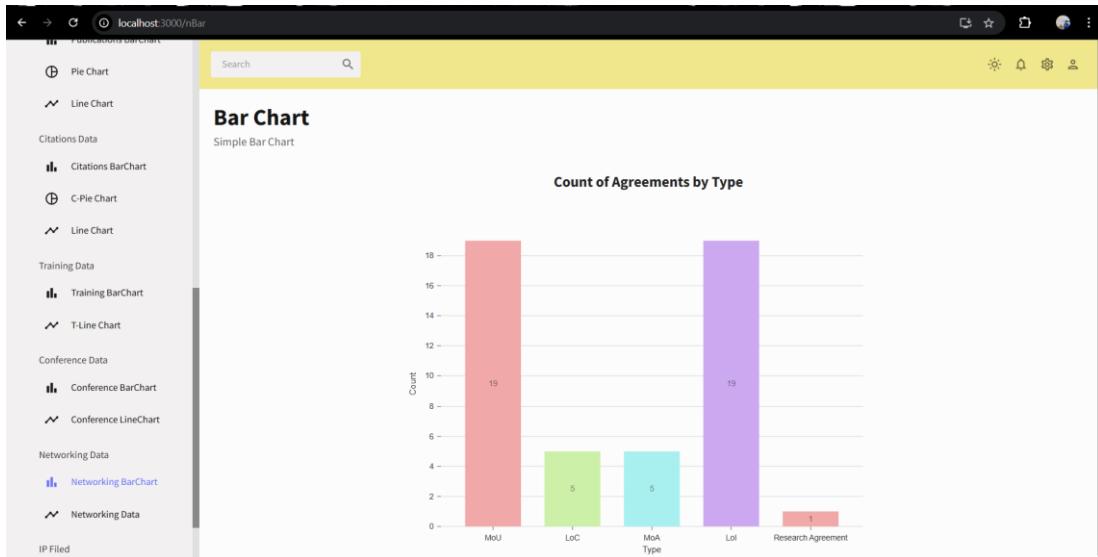
**Figure 4.6: Dashboard Page**

From the dashboard, the user can choose what data they want to view. If they want to view the publication data, they'll be able to view it.



**Figure 4.7: Publication Page**

The user can view other data like networking data as well.



**Figure 4.8: Dashboard for Networking**

## 4.6 Chapter Summary

This chapter discusses all the important diagrams and methods that are required to develop the system. The requirements are the initial phase of the system. But it is just a part of the process. For developing the system, the development process is broken down into different phases. The diagrams depict a detailed description of the system features and functions. To summarize, this chapter includes all the important steps that are required to develop the system.

# **CHAPTER 5**

## **SYSTEM DEVELOPMENT METHODOLOGY**

### **5.1 Introduction**

This chapter focuses on the description of the development process, methodologies adopted, and design issues related to the Research Data Dashboard for the Faculty of Computing. The implementation involves designing the front-end interface, setting up the backend, managing the database, and coding the functionalities to meet the system requirements. Key components of the implementation are presented, including code snippets of important functions and explanation of system architecture for the overall view of development.

The testing procedures are presented in this chapter as well that are used in making sure that the functionalities work correctly, and the errors are well validated.

### **5.2 Development Environment**

The Faculty of Computing's Research Data Dashboard is a data-dependent system reliant on data extraction from target websites, followed by data analysis. The MERN Stack employed in this project involves MongoDB, Express.js, React, and Node.js. Among the libraries of Node.js, Puppeteer is preferred as the most suitable tool because of its usefulness in the extraction of data. Extracted data are stored in MongoDB in JSON format, which is further accessed by the backend and served to the frontend.

The design of the backend part will be based on Node.js using the Express framework. It will be implemented on the server side, providing necessary APIs. The React-based frontend will fetch values from it and plot, using the library React Chart.js to create interactive charts - bar charts, pie charts, doughnut charts, and line charts, depending on the requirements.

The frontend and backend are on two different ports; still, they use libraries like Axios on them to handle API requests and middleware like CORS to make the communication smooth between them.

### 5.3 Implementation

The implementation stage means putting into code the system design and architecture mapped out through the requirement analysis such that the system can work and perform the intended goals set out for it. After setting up the backend and frontend, the development of the system commences. The front-end contains numerous React components that are spread across different pages to show different types of information. Many of these components have JavaScript logic that processes the data. The back end is basically the core of the Puppeteer scripts which are the living vital source of data for the whole system. These scripts form the basis of the Research Data Dashboard for the Faculty of Computing.

#### 5.3.1 View Implementation

The front-end interfaces are displayed in this section that includes the statistical diagrams like pie chart, bar chart generated and the login page.

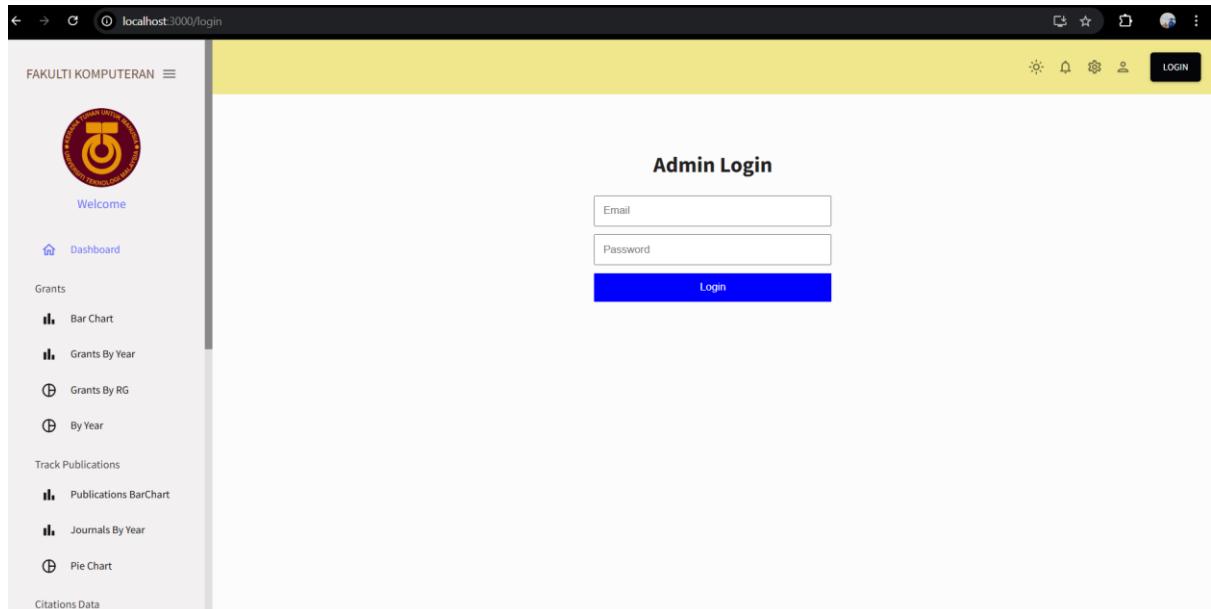
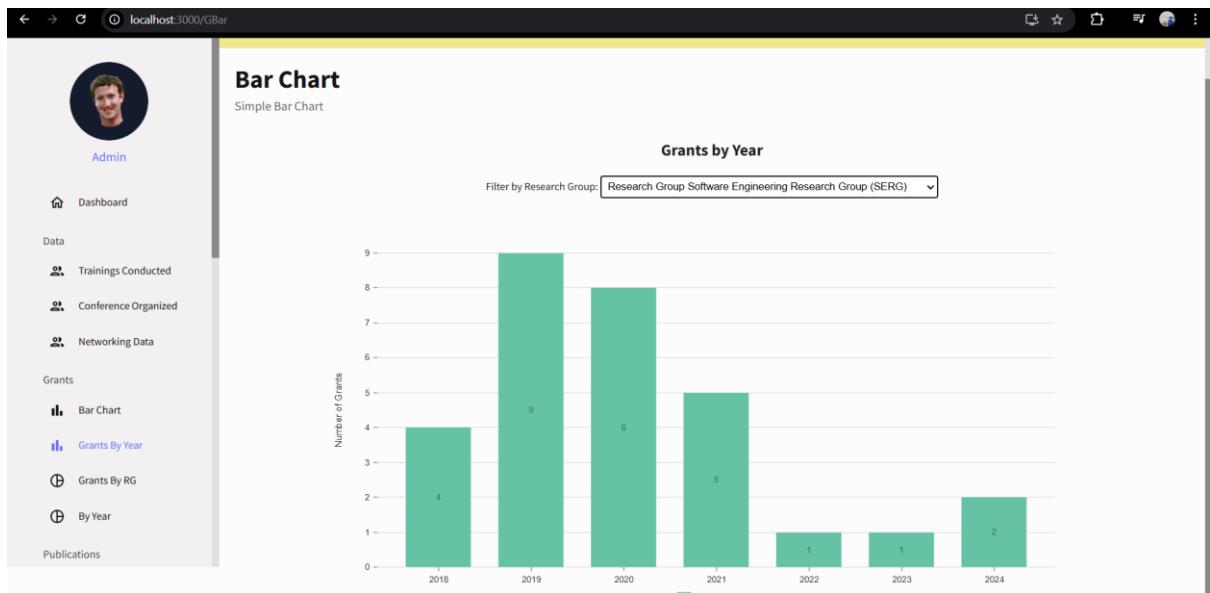
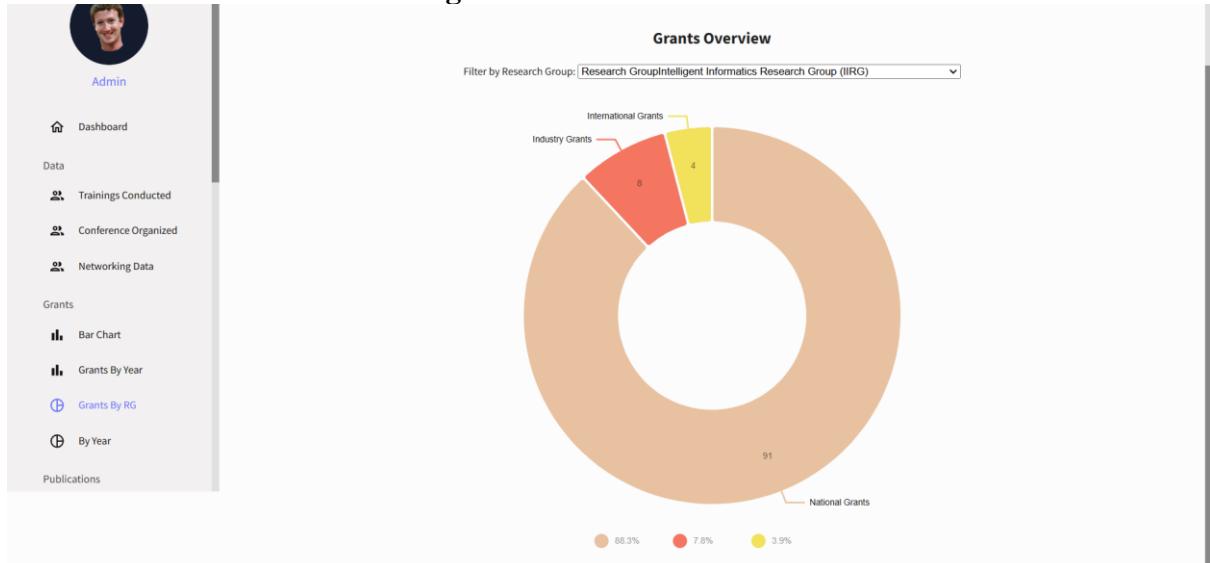


Figure 5 1:Login Page

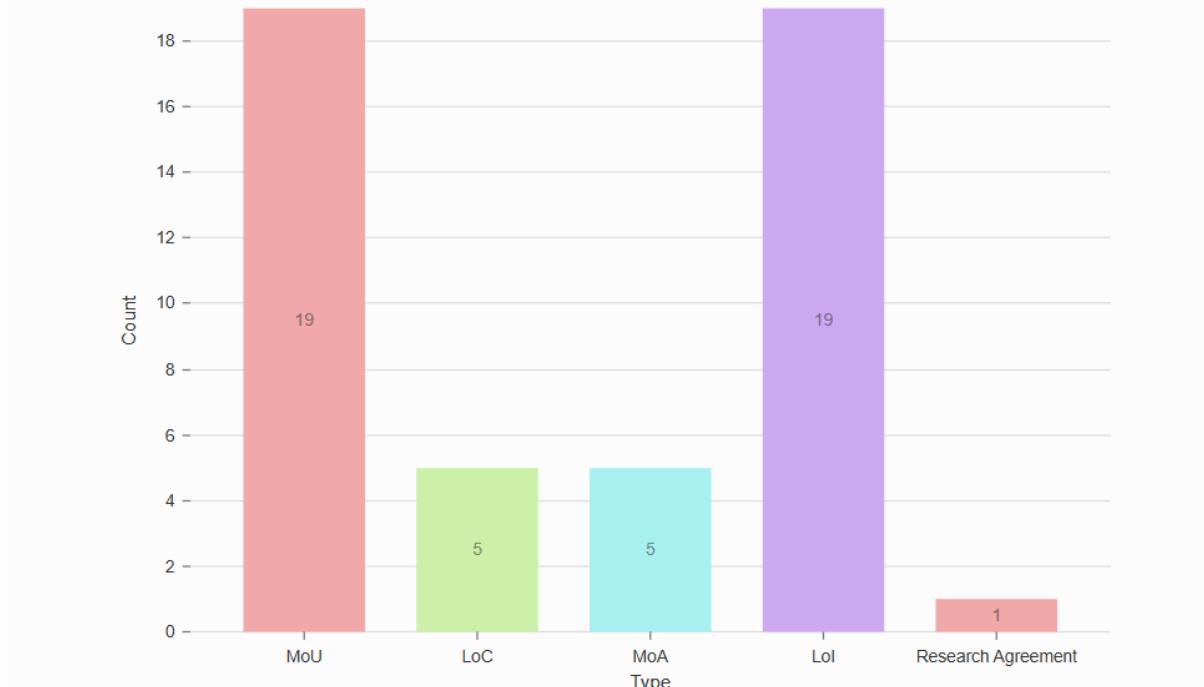


**Figure 5 2Grants Bar Chart**

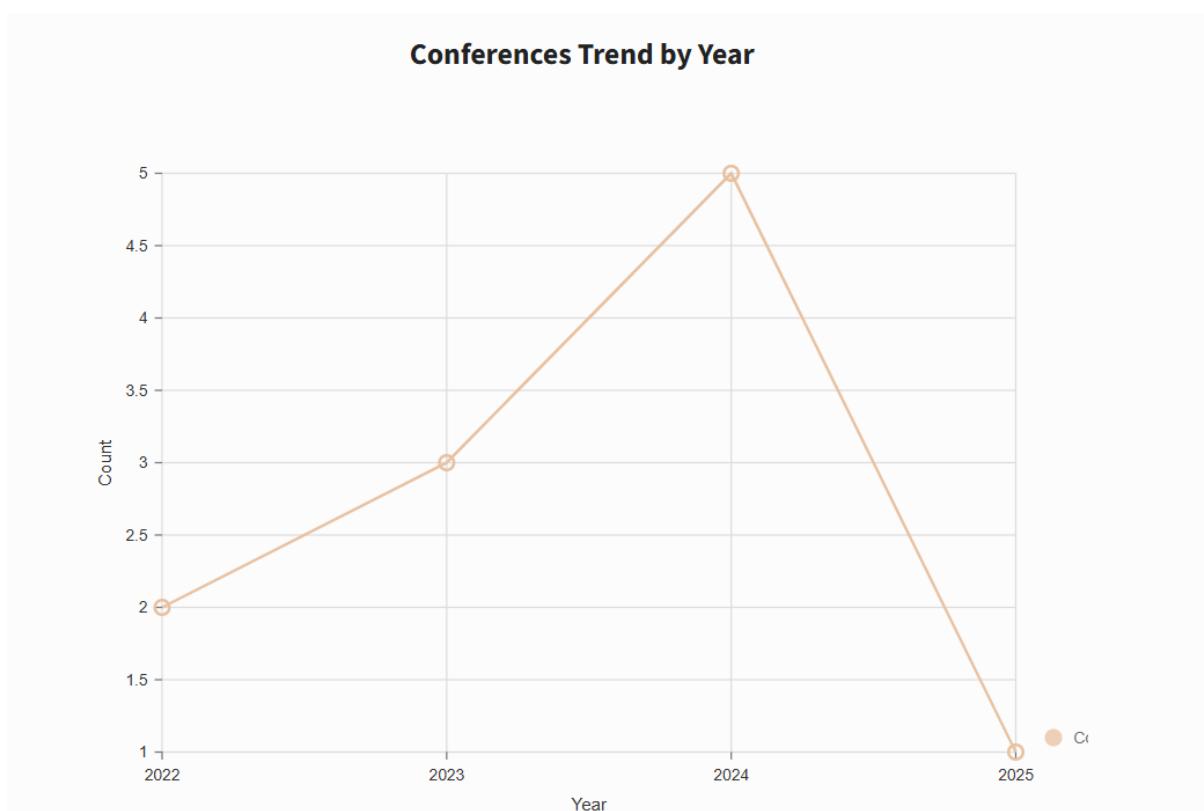


**Figure 5 3: Grants Doughnut Chart**

### Count of Agreements by Type



### Figure 5 4: Networking data Bar Chart



### Figure 5 5: Conferences Data Line Chart

### 5.3.2 Coding Implementation

This part includes the structure of the code in the system. The database integration, backend setup also including the components used in the frontend will be discussed in this section.

```
// Middleware
app.use(cors());
app.use(express.json());

// MongoDB Connection
mongoose
  .connect("mongodb://127.0.0.1:27017/utm_scholars", {
    useNewUrlParser: true,
    useUnifiedTopology: true,
  })
  .then(() => console.log("Connected to MongoDB: utm_scholars"))
  .catch((err) => console.error("MongoDB connection error:", err));
```

Figure 5 6: MongoDB Connection setup

The api endpoints were called to display the data from the database.

```
app.get("/api/grants", async (req, res) => {
  try {
    const grantsData = await Scholar.aggregate([
      {
        $addFields: {
          NATIONAL_GRANTS_INT: { $toInt: "$NATIONAL_GRANTS" },
          INTERNATIONAL_GRANTS_INT: { $toInt: "$INTERNATIONAL_GRANTS" },
          INDUSTRY_GRANTS_INT: { $toInt: "$INDUSTRY_GRANTS" },
        },
      },
      {
        $group: {
          _id: null,
          totalNationalGrants: { $sum: "$NATIONAL_GRANTS_INT" },
          totalInternationalGrants: { $sum: "$INTERNATIONAL_GRANTS_INT" },
          totalIndustryGrants: { $sum: "$INDUSTRY_GRANTS_INT" },
        },
      },
    ]);
    const {
      totalNationalGrants = 0,
      totalInternationalGrants = 0,
      totalIndustryGrants = 0,
    } = grantsData[0] || {};
    res.json({
      nationalGrants: totalNationalGrants,
      internationalGrants: totalInternationalGrants,
      industryGrants: totalIndustryGrants,
    });
  } catch (err) {
    console.error("Error fetching grants data:", err);
    res.status(500).send("Error fetching grants data");
  }
}
```

**Figure 5 7: Api endpoint for Grants**

```
// 4. Training Projects Endpoint (training_2024 data)
app.get("/api/trainingProjects", async (req, res) => {
  try {
    const projects = await TrainingProject.find({});
    res.json(projects);
  } catch (err) {
    console.error("Error fetching training projects:", err);
    res.status(500).send("Error fetching training projects");
  }
});
app.post("/api/trainingProjects", async (req, res) => {
  try {
    console.log("Request body:", req.body); // Debug incoming data
    const newProject = new TrainingProject({
      Bil: Number(req.body.Bil),
      KetuaProjek: String(req.body.KetuaProjek),
      Vot: Number(req.body.Vot),
      TajukProjek: String(req.body.TajukProjek),
      Klien: String(req.body.Klien),
      KosProjek: Number(req.body.KosProjek),
      Year: Number(req.body.Year), // Ensure Year is a number
    });

    const savedProject = await newProject.save();
    console.log("Saved project:", savedProject); // Debug saved data
    res.status(201).json(savedProject);
  } catch (err) {
    console.error("Error saving training project:", err);
    res.status(500).send("Error saving training project");
  }
});
```

**Figure 5 8: Api endpoints for Training Data**

Here's an example of the frontend react component of the sidebar.

```

const Sidebar = () => {
  const theme = useTheme();
  const colors = tokens(theme.palette.mode);
  const [isCollapsed, setIsCollapsed] = useState(false);
  const [selected, setSelected] = useState("Dashboard");

  return (
    <Box
      sx={{
        "& .pro-sidebar-inner": {
          background: `${colors.primary[400]} !important`,
        },
        "& .pro-icon-wrapper": {
          backgroundColor: "transparent !important",
        },
        "& .pro-inner-item": {
          padding: "5px 35px 5px 20px !important",
        },
        "& .pro-inner-item:hover": {
          color: "#868dfb !important",
        },
        "& .pro-menu-item.active": {
          color: "#6870fa !important",
        },
      }}
    >
      <ProSidebar
        collapsed={isCollapsed}
        style={{}}
      </ProSidebar>
    
```

**Figure 5 9Frontend Component of Sidebar**

```

import React, { useEffect, useState } from "react";
import { ResponsiveBar } from "@nivo/bar";

const BarChart = () => {
  const [data, setData] = useState([]);
  const [loading, setLoading] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() => {
    const fetchGrantsData = async () => {
      try {
        const response = await fetch("http://localhost:5000/api/grants");
        if (!response.ok) throw new Error(`Error: ${response.statusText}`);
        const jsonData = await response.json();
        // Map API response to chart data format
        const chartData = [
          { type: "National Grants", count: jsonData.nationalGrants },
          { type: "International Grants", count: jsonData.internationalGrants },
          { type: "Industry Grants", count: jsonData.industryGrants },
        ];
        setData(chartData);
      } catch (err) {
        console.error("Error fetching grants data:", err);
        setError("Failed to load chart data.");
      } finally {
        setLoading(false);
      }
    };
    fetchGrantsData();
  }, []);

  if (loading) return <p>Loading chart...</p>;
  if (error) return <p>{error}</p>;
}

return (

```

**Figure 5 10: Frontend Component for Bar Chart**

## 5.4 Architecture Implementation

The architecture followed by the Dashboard is the Model-View Controller architecture. A model holds the database models of the system. The controller holds the logic and enables the communication between the front-end and back-end while the views are the React components.

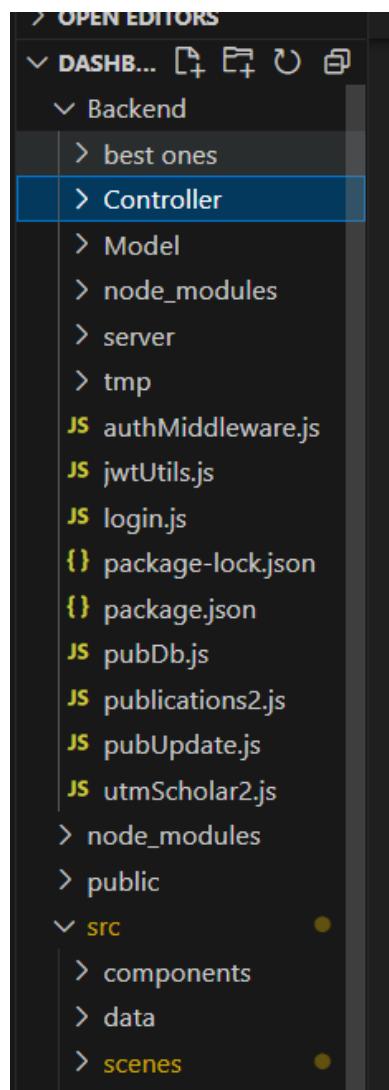


Figure 5 11: Architecture Implementation

```
Backend > Model > JS scholarsModel.js > [🔗] scholarSchema
1  const mongoose = require("mongoose");
2
3  const scholarSchema = new mongoose.Schema({
4      link: { type: String, required: true },
5      GRANT_PI_MEMBERS: String,
6      PUBLICATIONS: Number,
7      INDEXED_PUBLICATION: String,
8      TOTAL_STUDENTS: String,
9      H_INDEXED_SCOPUS: String,
10     CITATIONS_SCOPUS: String,
11     INDUSTRY_GRANTS: String,
12     INTERNATIONAL_GRANTS: String,
13     NATIONAL_GRANTS: String,
14     UNIVERSITY_FUND: String,
15     TOTAL: Number,
16     NON_INDEXED_PUBLICATION: String,
17     OTHERS_PUBLICATION: String,
18     MASTER: Number,
19     PHD: Number,
20 });
21
22 module.exports = mongoose.model("Scholar", scholarSchema);
23
```

Figure 5 12: Scholars Model for database

```

const authService = require("../login");

async function login(req, res) {
  try {
    const { email, password } = req.body;

    // Verify the admin login
    const adminUser = await authService.verifyAdmin(email, password);

    // Return a success response with the admin's info
    res.json({
      message: "Login successful",
      admin: {
        id: adminUser._id,
        email: adminUser.email,
        name: adminUser.name,
      },
    });
  } catch (error) {
    res.status(401).json({ message: error.message }); // Return error messages directly
  }
}

module.exports = { login };

```

**Figure 5 13: Login Auth Controller**

## 5.5 System Testing

The immediate vital activity after the implementation of the Research Data Dashboard for the Faculty of Computing would be the testing of the system for whether the system works as it was required and designed. Tests can be performed in several ways: black box testing, white box testing, and user acceptance testing. Each of them has different advantages and helps in making the system robust. Research Data Dashboard for the Faculty of Computing shall use Black Box Testing and User Acceptance Testing during the testing process.

### 5.5.1 Black Box Testing

Black-box testing is normally a software testing approach that focuses on assessment functionality based on application behavior. It doesn't require much use of the system code. It always involves investigations of input and output values of the software system regarding every test case. The objective of this testing is to ascertain whether the system acts as documented.

### 5.5.1.1 Test Case Design

#### 1.1 Test TC001 for Module Authentication & View: Login(UC001)

<b>Test Case ID</b>	TC001_1		
<b>Test Case Description</b>	Able to login Successfully	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	There is an active internet connection		

Test execution steps:

**Table 5. 1: Test Case 1 for login**

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass
2	User clicks on the email field and enters his registered email	Email: example@gmail.com	Email field works correctly entered by the user	Email Field is Working	Pass
3.	User clicks on the password field and enters the password	Password: 1234admin	Password field is taking inputs	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Successful	Pass
5.	User is taken to their respective dashboard by the system	-	System verifies the user successfully by checking with the database	Page loaded successfully	Pass

<b>Test Case ID</b>	TC001_2		
<b>Test Case Description</b>	Display error message for Unsuccessful Login	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	There is an active internet connection		

Test execution steps:

**Table 5. 2: Test Case 2 for login**

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass
2	User clicks on the email field and enters his registered email	Email: example@gmail.com	Email field works correctly and will display the username entered by the user	Email Field is Working	Pass
3.	User clicks on the password field and enters the password	Password: 1234	Password field is taking inputs and user can view the entered password	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Unsuccessful	Pass
5.	User is displayed an error message for unsuccessful login.	-	System displays error message for unsuccessful login.	Invalid email or Password is displayed	Pass

<b>Test Case ID</b>	TC001_3			
<b>Test Case Description</b>	Display error message for invalid input	<b>Test Priority</b>	High	
<b>Pre-Requisite</b>	There is an active internet connection			

Test execution steps:

**Table 5. 3: Test Case 3 for login**

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass
2	User clicks on the email field and enters his	Email: abc	Email field works correctly and	Email Field is Working	Pass

	registered email		will display to write @ in email		
3.	User clicks on the password field and enters the password	Password: 1234	Password field is taking inputs and user can view the entered password	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Unsuccessful	Pass
5.	User is displayed an error message for unsuccessful login.	-	System displays error message for Invalid input.	Message Displayed	Pass

<b>Test Case ID</b>	TC001_4			
<b>Test Case Description</b>	Display error message for Empty Input	<b>Test Priority</b>		High
<b>Pre-Requisite</b>	There is an active internet connection			

Test execution steps:

**Table 5. 4: Test Case 4 for login**

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass
2	User clicks on the email field and leaves it blank	Email:	Email field works correctly	Email Field is Working	Pass
3.	User clicks on the password field and enters the password	Password: 1234	Password field is taking inputs and user can view the entered password	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Unsuccessful	Pass
5.	User is displayed an error message for unsuccessful login.	-	System displays error message for empty input.	Message Displayed	Pass

#### Test TC006 for Module Publication: Generate H-Indexed Publication(UC006)

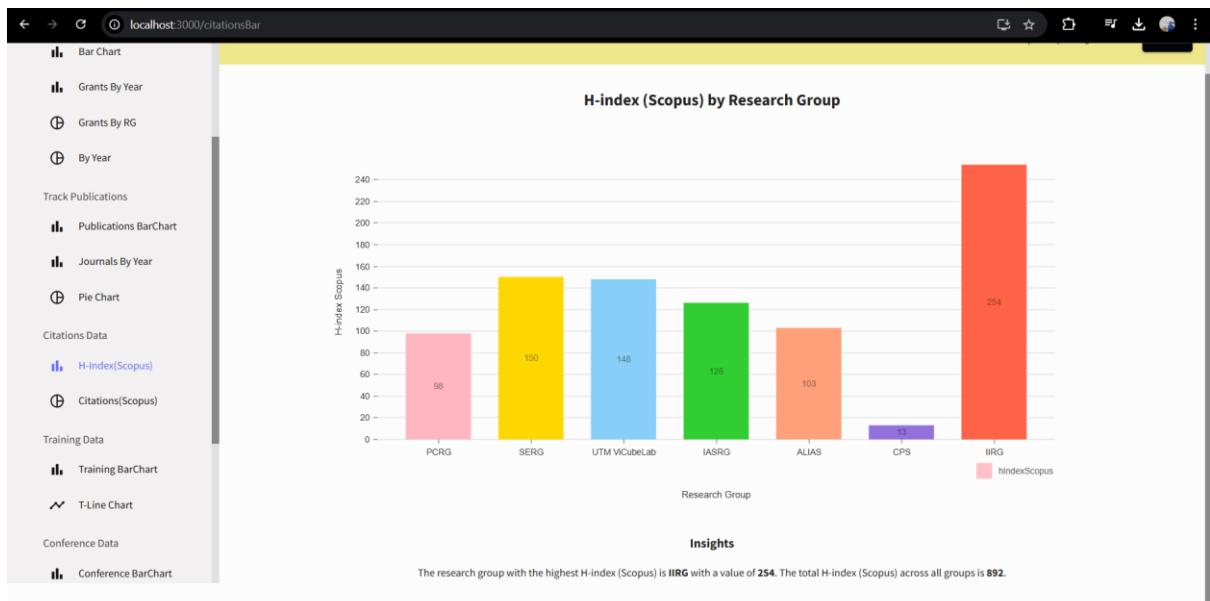
<b>Test Case ID</b>	TC006_1
---------------------	---------

<b>Test Description</b>	Case Verify that system is successfully retrieving data from database	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is secured		

Test execution steps:

**Table 5. 5: Test Case 1 for H-index Data**

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Citations section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the H-Index (Scopus) Data button.	-	Button works properly and user is taken to H-Index metrics page.	Button and page are working	Pass
3.	User can view the generated data from the database	Non-Indexed Publication Data	Data has been successfully retrieved from database.	Data displayed in the form of a bargraph	Pass



**Figure 5 14: H-Index Bar Chart**

## **5.5.2 User Acceptance Testing**

User Acceptance Testing is one form of testing wherein the stakeholder goes on to review himself in the system functionalities to make sure they are all up to specifications. In this GRF project, the stakeholder needed to do UAT for the functionality of scraping.

### **5.5.2.1 Scraping Functionality**

No.	Function	Description	Performed By
1	Scraping the data from the website	This function ensures the scraping of the desired data	Admin

**Table 5. 6: User Acceptance Testing**

No.	Actions	Expected Results	Pass/Fail	Comment
1.	Admin logs in	Login is successful	Pass	
2.	Admin clicks on the start scraping button	Button works fine	Pass	
3.	Admin clicks on the start scraping button	Scraping is starting and ends with a successfully completed message	Pass	

## **CHAPTER 6**

### **Conclusion**

#### **6.1 Introduction**

This chapter is the conclusion of the development of our system which is dashboard for the faculty of computing researchers. The dashboard will display various data related to the project and will help in evaluating for the future improvements. This document has discussed all the necessary details that are needed for the project.

#### **6.2 Achievement of Project Objectives**

The need for the system was discussed with the stakeholders in PSM1. A literature review of the existing system like the proposed system is conducted and comparisons between the proposed system and the existing system are outlined in Chapter 2. The procedure that is utilized in the creation of the system is shortly described and discussed, provided in Chapter 3. The system requirement is deeply discussed and thoroughly specified within the SRS. SDD contains information about the general design of the system. STD also has a listing of the system's test cases.

Documentation of PSM1 was a key source in the successful development of the system for PSM2. Following the design methodology and mentioning requirements from the documentation of PSM1, the dashboard has been developed, and all key functionalities have been implemented.

The system was tested as a part of the Agile development stage and the comments of the stakeholder were considered during the development of the system. The outcome was the successful implementation of the documentation from psm1 to build a complete web-based dashboard that displays different research data for the faculty of computing and allows users to get an insight on the research output by the scholars.

### **6.3 Suggestions for Future Improvement**

The current industry is rapidly changing, and the introduction of new technologies requires every application to be updated over time. There will always be room for improvements and suggestions for any application. The improvements that can be made are:

- Including more websites for scraping research related data
- Integrating Ai tools for future predictions based on the available data
- Including for visualization tools that are suitable for displaying research data
- Making this system usable for a large number of users

## REFERENCES

- Pauwels, K., Ambler, T., Clark, B. H., LaPointe, P., Reibstein, D. J., Skiera, B., Wierenga, B., & Wiesel, T. (2009). Dashboards as a Service. *Journal of Service Research*, 12(2), 175–189. <https://doi.org/10.1177/1094670509344213>
- Susnjak, T., Ramaswami, G. S., & Mathrani, A. (2022). Learning analytics dashboard: a tool for providing actionable insights to learners. *International Journal of Educational Technology in Higher Education*, 19(1). <https://doi.org/10.1186/s41239-021-00313-7>
- Elsevier. (n.d.). *My Research Dashboard*. <https://www.elsevier.com/connect/editors-update/my-research-dashboard>
- Matheus, R., Janssen, M., & Maheshwari, D. (2020). Data science empowering the public: Data-driven dashboards for transparent and accountable decision-making in smart cities. *Government Information Quarterly*, 37(3), 101284. <https://doi.org/10.1016/j.giq.2018.01.006>
- Kadijević, Đ. (2020). *Modern statistical literacy, data science, dashboards, and automated analytics and its applications*. <https://ipir.ipisr.org.rs/handle/123456789/322>
- Power, D. J. (2016). Data science: supporting decision-making. *Journal of Decision Systems*, 25(4), 345–356. <https://doi.org/10.1080/12460125.2016.1171610>
- Bhardwaj, A. (2014, September 2). *DataHub: Collaborative Data Science & Dataset Version Management at Scale*. arXiv.org. <https://arxiv.org/abs/1409.0798>

## **Appendix A System Requirements Specification**



---

# **Software Requirements Specification**

Faculty of Computing Staff Publication Dashboard

Version 1.0

25/06/2023

Faculty of Computing, Software Engineering

Prepared by: Adib Bin Morshed

# **Revision Page**

## **a. Overview**

This document is the first draft of the Software Requirements Specification (SRS) for the Faculty of Computing Staff Publication Dashboard.

## **b. Target Audience**

- **Prof Madya Dr. Siti Zaiton Bt. Mohd Hashim, Faculty of Computing, Universiti Teknologi Malaysia (UTM).**
- **Faculty of computing Staff, UTM.**

## **c. Project Team Members**

- Adib Bin Morshed

## **d. Version Control History**

<b>Version</b>	<b>Primary Author(s)</b>	<b>Description of Version</b>	<b>Date Completed</b>
Version 1.0	Adib Bin Morshed	SRS of Faculty of Computing Staff publication Dashboard	25 June,2023

### **Note:**

This Software Requirements Specification (SRS) template is based on IEEE Std 830-1998, organized by modules according to system features (Appendix A.5 of the IEEE Std, 830-1998, Section 5) and customized to meet the need of SCSJ2203 course at Faculty of Computing, UTM. Compiled and checked by Shahida Sulaiman, PhD on 20 March 2016. Examples of models are from Satzinger (2011).

# **Table of Contents**

## **1 Introduction**

- 1.1 Purpose
- 1.2 Scope
- 1.3 Definitions, Acronyms and Abbreviations
- 1.4 References
- 1.5 Overview

## **2 Overall Description**

- 2.1 Product Perspective
  - 2.1.1 System Interfaces
  - 2.1.2 User Interfaces
  - 2.1.3 Hardware Interfaces
  - 2.1.4 Software Interfaces
  - 2.1.5 Communication Interfaces
  - 2.1.6 Memory
  - 2.1.7 Operations
  - 2.1.8 Site Adaptations Requirements
- 2.2 Product Functions
- 2.3 User Characteristic
- 2.4 Constraints
- 2.5 Assumption and Dependencies
- 2.6 Apportioning of Requirements

## **3 Specific Requirements**

- 3.1 External Interface Requirements
  - 3.1.1 User Interfaces
  - 3.1.2 Hardware Interfaces
  - 3.1.3 Software Interfaces
  - 3.1.4 Communication Interfaces
- 3.2 System Features
- 3.3 Performance Requirements
- 3.4 Design Constraints

3.5 Software System Attributes

3.6 Other Requirements

# **1. INTRODUCTION**

---

## **1.1 Purpose**

The main purpose of this document is to describe the necessary requirements and specifications of the system. The description of the requirements is based on functional, non-functional features and functionalities. The requirements were gathered from our stakeholders. This SRS is important for the development of the system since it contains all the requirements that have been discussed with the stakeholders. This document is developed for the ease of understanding of the system by the developer and stakeholders.

## **1.2 Scope**

The proposed system is named the research metrics dashboard for faculty of computing staff. This system is specifically designed for the stakeholders who are researchers and research staff of the faculty of computing at Universiti Teknologi Malaysia (UTM). This system will help to recognize patterns and performance measures to evaluate the current situation of the research department. Although it'll have all the necessary data required to make the evaluation, there will be a few limitations as well. This system will not include some data related to researchers which is a limitation of the system. Nevertheless, it'll be used for the performance measure of the publications by the researchers and research staff. This document is needed for the better understanding of the system by the stakeholders and developer.

## **1.3 Definitions, Acronyms and Abbreviation**

Acronyms	Definition
SRS	Software Requirements Specification
UCS	Use Case Specification
UTM	Universiti Teknologi Malaysia

## **1.4 References**

1. Department of English, L. (2020). Software Requirements Specification (SRS). *Slcc.pressbooks.pub*.  
<https://slcc.pressbooks.pub/technicalwritingatslcc/chapter/software-requirements-specification-srs>

## 1.5 Overview

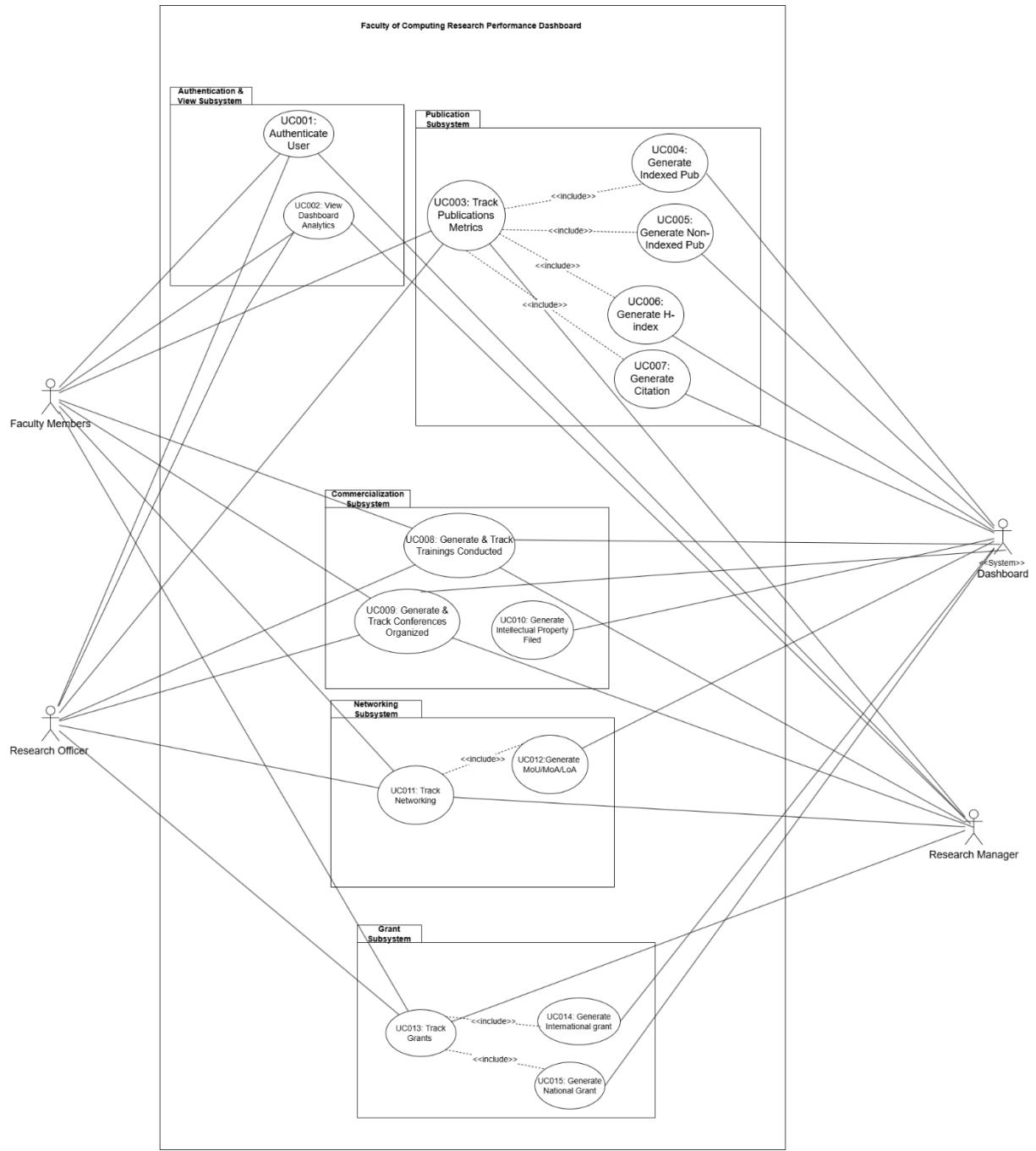
The requirements for the project have been divided into three main sections. The purpose of this document has been described in the first section. An overview of the system's features, interactions, and functionality has been described in Section 2. Additionally, assumptions and dependencies, system constraints are also discussed in section 2. The system's specific requirements including the use case specifications, descriptions, interfaces, sequence, and activity diagrams have been depicted in section 3.

## **2. OVERALL DESCRIPTION**

---

An overview of the system will be described in this section. The use case will be shared and detailed description of the use cases will be documented in this section as well. For better understanding of the system, flow of the use case will be provided in this section as well.

The overall use case contains several subsystems that are considered as accepted after discussing with the stakeholder. The use case is divided into 5 different subsystems consisting of authentication and view subsystem, publication subsystem, commercialization subsystem, networking subsystem and grant subsystem. The four actors for this use case are the faculty members, research officer, research manager and dashboard. Figure 2.1 presents the use case diagram for the whole system.



**Figure 2.1: Use Case Diagram of Faculty of Computing Staff Publication Dashboard**

## 2.1 Product Perspective

This SRS describes a dashboard system for the faculty of computing that will help to recognize patterns of the publications made by the faculty members. The dashboard system will display data from different sources that'll be used to evaluate the performance of the faculty members

in the research field. This product will help to display index, non-index publication, H-index and citations generated by taking data from other sources such as the Scopus, google scholar, UTM scholar and so on. My system will also display data related to income generation and commercialization. Commercialization data will include intellectual property filed. The system will generate networking data including the agreement data as well. Research officer and manager will be able to track the data for making reports and recommendations for improvement and evaluate the current situation. Faculty members will be able to see the condition of the research data as well. The purpose of this system is to improve performance measures of the faculty of computing staff.

### **2.1.1 System Interfaces**

The publication dashboard for the faculty of computing staff will be developed with a web-based approach. The web-based system will be developed by implementing the ReactJS for frontend and Django for backend. We will be using MongoDB for storing scraped data after filtering and cleaning it from an excel file.

### **2.1.2 User Interfaces**

The system will have user interfaces that will be easy for the users to navigate and understand while using the system. The user interfaces will be made attractive for the users to have a good impression while using the system. Below are descriptions of some of the interfaces that will be developed for our system.

#### **2.1.2.1 Login Interface**

The system will have a login interface for the users that will allow the users to access the dashboard page after giving their valid email address and password.

#### **2.1.2.2 Dashboard Interfaces**

The dashboard interfaces are the main interfaces of the system that will display data taken from different sources. The user will be taken to the respective dashboard after logging into the system. Inside the dashboard there will be different pages showing different data that

can be viewed by the faculty members and staff. The pages include publication data page, networking data page, commercialization data page, and grant data page.

### **2.1.3 Hardware Interfaces**

This system can be accessed using desktop computers as well as laptops that have web browsers installed on it. A stable internet connection is required to access the system. The development is done for the mentioned devices only.

### **2.1.4 Software Interfaces**

The interfaces that are related to the system are web-based. Internet connection is required to access the system. After ensuring that, the system can be accessed using google chrome, Mozilla Firefox, Microsoft Edge, and Apple Safari. The above-mentioned browsers are required to use the system. It can also be accessed by both the Windows and the Mac operating system.

### **2.1.5 Communication Interfaces**

TCP/IP protocols like HTTP, HTTPS, and FTP are used to carry out communication over the internet. The usage of such protocols makes it easier to ensure the highest level of compatibility and dependability possible.

### **2.1.6 Memory**

The utilization of memory is subject to the usage of the hardware platform required for our system. The accuracy of the usage of memory will depend on the of the capabilities of the hardware platform that is designed to operate this system.

### **2.1.7 Operations**

End-users will be navigated to the system login page at first. From there on, they must log in using their valid email address and password. They will be given initial information

about logging into the system. They will be taken to a user dashboard where they'll be able to visualize different data analytics related to the publication. For using this system, only registered users by the developer will be able to use the system at first. The operation of the dashboard analytics will be done on the database end. The data will be collected from different sources and will be displayed in the dashboard after cleaning, refining, and transforming the data. Users will be able to view the data in the that'll be displayed in the dashboard. Their navigation operation will be conducted after logging into the system.

### **2.1.8 Site Adaptation Requirements**

The system is compatible with different platforms and multiple operating systems. It can be accessed through different web browsers that are compatible with the platform's operating systems. It is adaptable to different platforms and systems. It is quite adaptable for the users to navigate through the system. A website link will be created for the users to access the website. Overall, the system is adaptable for users from different backgrounds.

## **2.2 Product Functions**

Table 2.2 provides the explanation of the product functions.

Table 2.2: Product Functions

<b>Module</b>	<b>Product Function</b>	<b>Description</b>
Authentication and View	Login	It will allow the user to login to the system.
	View Dashboard Analytics	It will allow the user to view the dashboard analytics of different research related data.
Publication	Track Publication Metrics	This will allow the research staff to track the publication metrics by reviewing the data related to publication.
	Generate Indexed Publication	This will be generated by the system dashboard by using

		data science techniques for the users to view their update.
	Generate Non-Indexed Publication	This will be generated by the system dashboard by using data science techniques for the users to view non-Indexed publications.
	Generate H-Index	This will be generated by the system dashboard by using data science techniques for the users to view H-Index data generated.
	Generate Citation	This will be generated by the system dashboard by using data science techniques for the users to view Citations made data.
Commercialization	Generate & Track Trainings Conducted	This will allow the research staff to track the Income generation by reviewing the data related to trainings organized.
	Generate & Track Conferences Organized	This will allow the research staff to track and edit the Conferences data by reviewing the data related to Conferences.
	Generate Intellectual Property Filed	This will be generated by the system dashboard by using data science techniques for the users to view the number

		of intellectual property filed for the researchers.
Networking	Track Networking	This will allow the research staff to track and edit the Networking data by reviewing the data related to networking.
	Generate MoU/MoA/LoU data	The system dashboard will generate data related to the agreement/letter of undertaking by using data science techniques for the users to view and evaluate on the current condition.
Grant	Track Grant	This will allow the research staff to track and edit the Grant data by reviewing the data related to Grant management.
	Generate International grant Data	The system dashboard will generate data related to the international grant that has been received by using data science techniques for the research staff to view and evaluate on the current condition.
	Generate National grant Data	The system dashboard will generate data related to the national grant that has been received by using data science techniques for the research

		staff to view and evaluate on the current condition.
--	--	------------------------------------------------------

## 2.3 User Characteristics

The user characteristics are described in the table below:

Table 2.3: User Characteristics

No.	User	Characteristics
1.	Faculty Members	This user is the main part of the system. The system is basically built for the usage of the faculty members. The faculty members will be able to get an insight of their research work and what measures can be taken to do improvement on research by the faculty of computing staff.
2.	Research Officer	Research officer plays a crucial role in our system design. All the data that will be collected will have to go through the research officer for filtering it properly before displaying in the system. This role is one of the major parts of the project. Research officer will help in data filtering before displaying in the form of a dashboard.
3.	Research Manager	Research manager will help in collecting the data that will be displayed in the dashboard. Research manager is the one who manages the data related to research. So, the role of the research manager is very important for carrying out the development of the system.

## 2.4 Constraints

The constraints of this system are given below:

- The system must be compatible with multiple web browsers and operating systems.

- The system will be easily accessible by the users with an active internet connection.
- The user interface will be designed in a way that'll be easy and intuitive for the users to use.
- The dashboard will be visualized with correct data collected from different data sources and the visualization will be done in an attractive manner.

## **2.5 Assumption and Dependencies**

It is important to have an active internet connection while using this system. This system must be used online. A big part of this system is to have an up-to-date database system. The database will contain all the necessary data required for displaying in the dashboard. Events like an outage must be considered in case the database fails to load. The system will not function properly without having proper internet connection, web browser access and database access.

Depending on the stakeholders, new functionalities and features maybe added to the application during development, which may alter the interdependencies of the requirements. The system may be modified depending on these changes. The changes are also dependant on the availability of the resources. Enough data must be collected for displaying in the dashboard. The collection of data depends on the availability and usability of the data. It is important to evaluate the validity of the data as well. As we can see, all the functions are inter-dependent on each other so any changes made will have an effect on the system as well.

## **2.6 Apportioning of Requirements**

Several requirements may be delayed until future versions of the system are released. The requirements will be identified in later phase of the document. The next part of the

document is SDD which will identify the basic requirements needed for the next version of the application. However, depending on the ongoing process and time the requirements might be delayed like the dashboard features specification and so on.

### 3. SPECIFIC REQUIREMENTS

---

This section will contain both the functional and non-functional requirements. This section includes the use case specifications with the sequence and the activity diagrams as well.

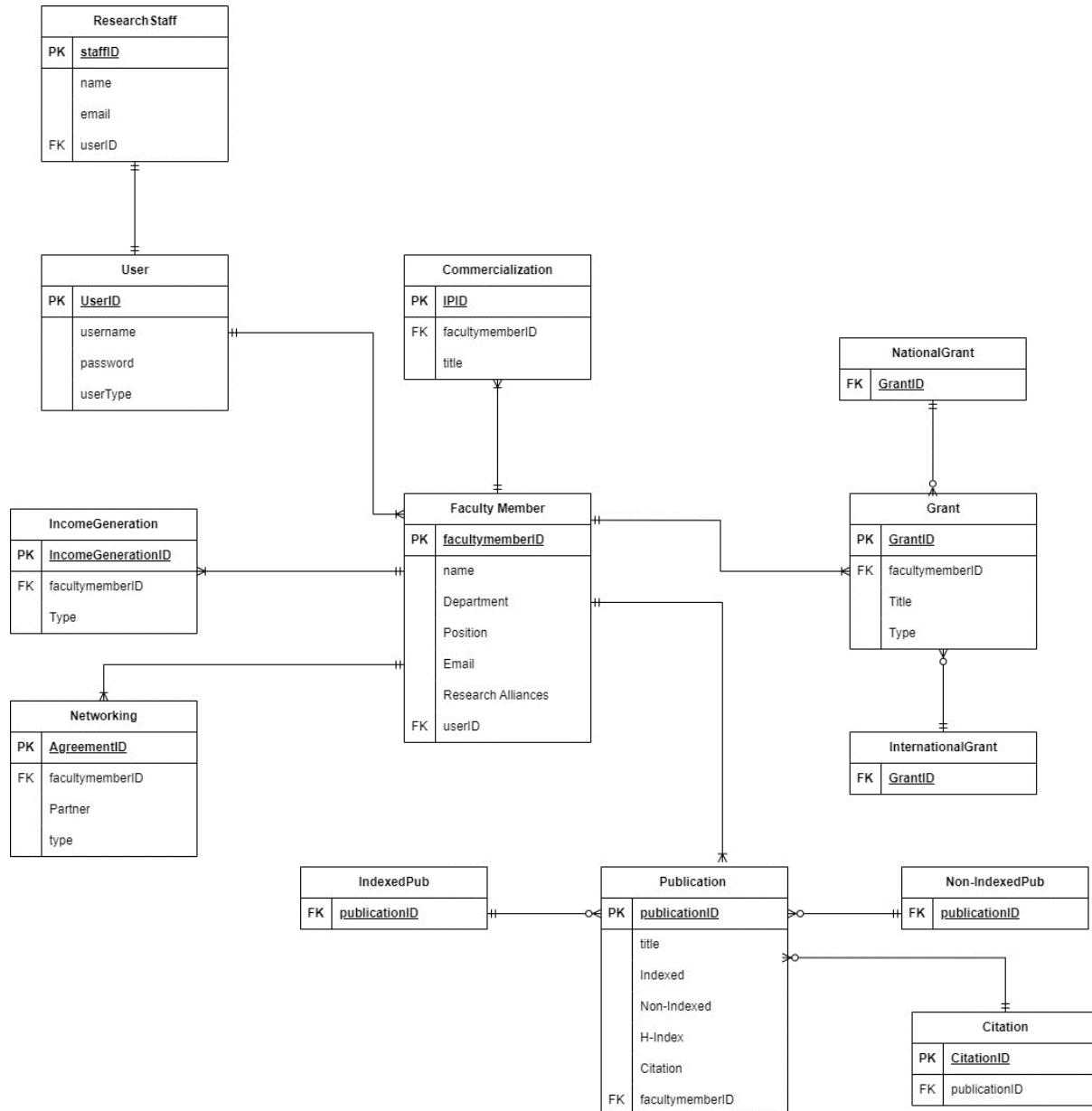
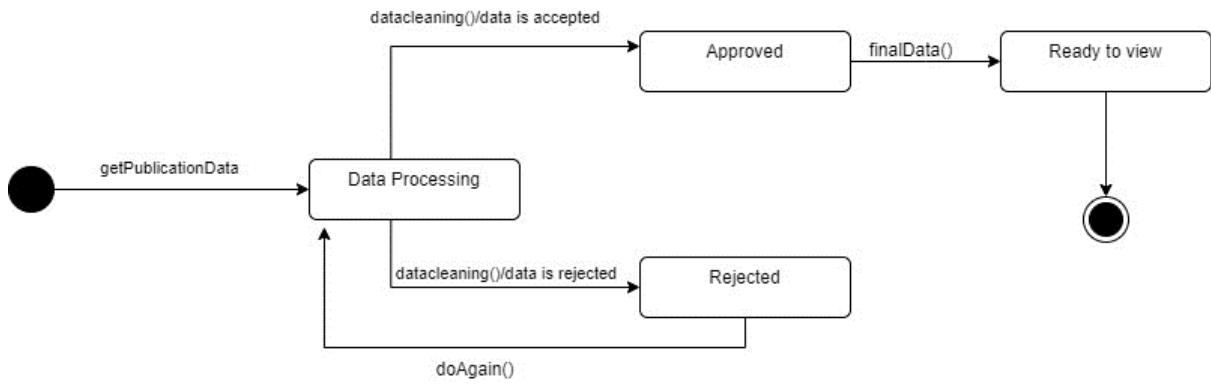


Figure 3.1: Domain Model of Faculty of Computing Staff Publication Dashboard



**Figure 3.2: State Machine Diagram of Publication**

### 3.1 External Interface Requirements

#### 3.1.1 User Interfaces

The user interfaces are described based on different roles of the interfaces:

##### 3.1.1.1 Login Interface

This interface will be designed to make sure the user enters the system with valid email address and password. The login interface is crucial part of the interface design. Users will be taken to the login portal once they enter the system. From there on, they'll have to key in their login credentials to enter the system. They'll be taken to their respective dashboard for visualization of the dashboard.

##### 3.1.1.2 Main interface for the researchers and research staff

Once the user enters the system, they'll be taken to the dashboard analytics from where they can view different data analytics related to the publication. The dashboard analytics will contain data related to the publication, commercialization, networking, income generation and grants. Meanwhile the publication interface will contain data which are part of the publication. The data for publication interface will include the indexed pub, non-indexed pub, H-index, and the citation. The data for commercialization interface will include the number of intellectual properties filed. There'll be a networking interface as well which will contain data related to

the agreements made. The grant interface will contain data which will include both the international and national grants received by the faculty of computing staff.

### **3.1.2 Hardware Interfaces**

For using this system, web browser and proper running desktop computer or laptop of any specifications can be used. A reliable internet connection is required for accessing the system.

### **3.1.3 Software Interfaces**

The interfaces that are related to the system are mainly web-based. Internet connection is required to access the system. After ensuring that, the system can be accessed using google chrome, Mozilla Firefox, Microsoft Edge, and Apple Safari. The above-mentioned browsers are required to use the system. It can also be accessed by both the Windows and the Mac operating system.

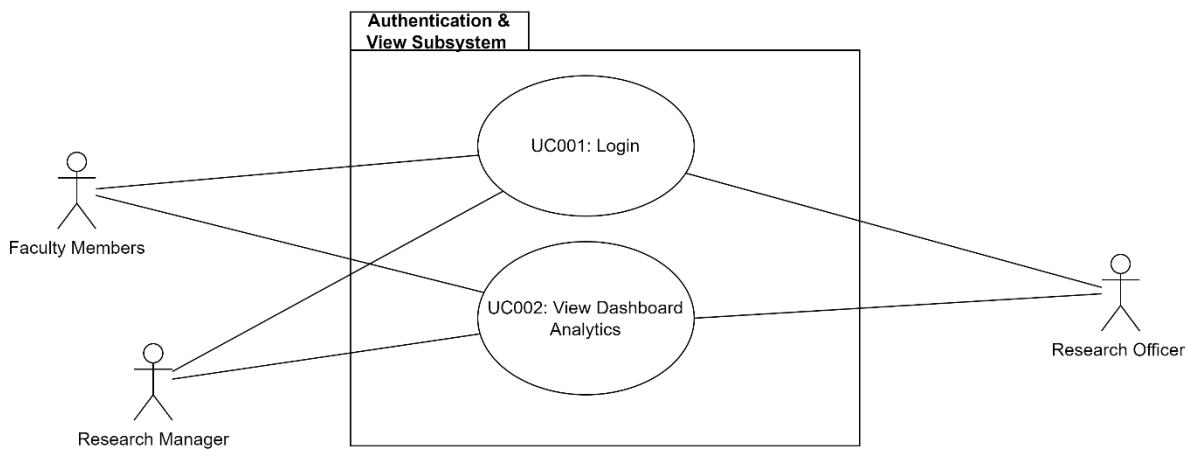
### **3.1.4 Communication Interfaces**

The system will work in a connection made with the LAN (Local Area Network). TCP/IP protocols like HTTP, HTTPS and FTP are utilized for internet-based communication for its durability and compatibility.

## **3.2 System Features**

Detailed description of the use case specifications with the relevant sequence and activity diagrams are outlined in this section of the document. Functional requirements of each module are also given in this section as well.

### **3.2.1 Module Authentication and View Subsystem**



**Figure 3.3: Module Login and View Subsystem**

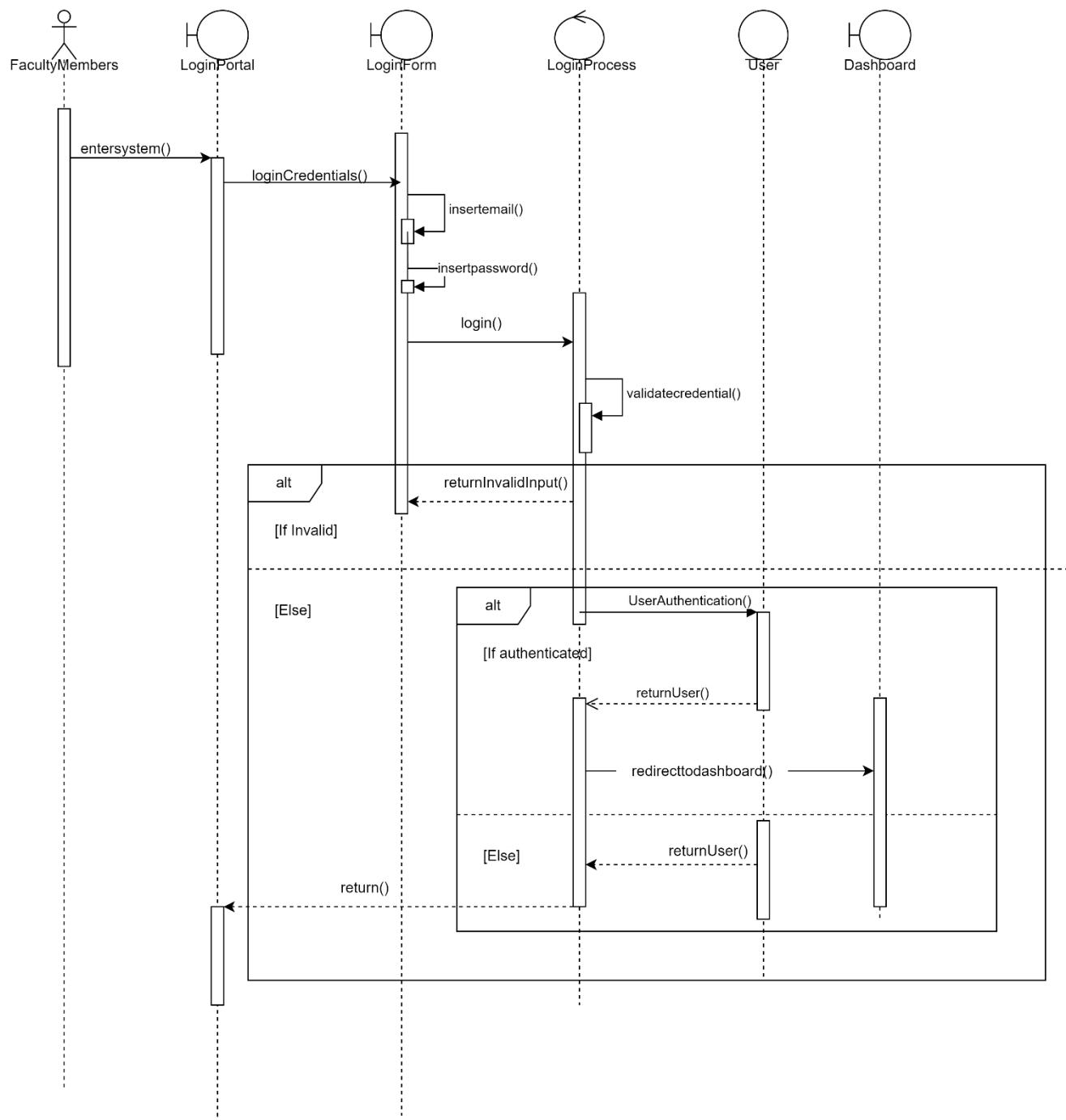
- I. FR001: Login- The system shall allow the user to login successfully to the system.
- II. FR002- View Dashboard Analytics- The system shall allow the user to view dashboard analytics in the form of charts/ graphs by providing an insight into the development made in research by the faculty.

### 3.2.1.1      UC001: Use Case Login

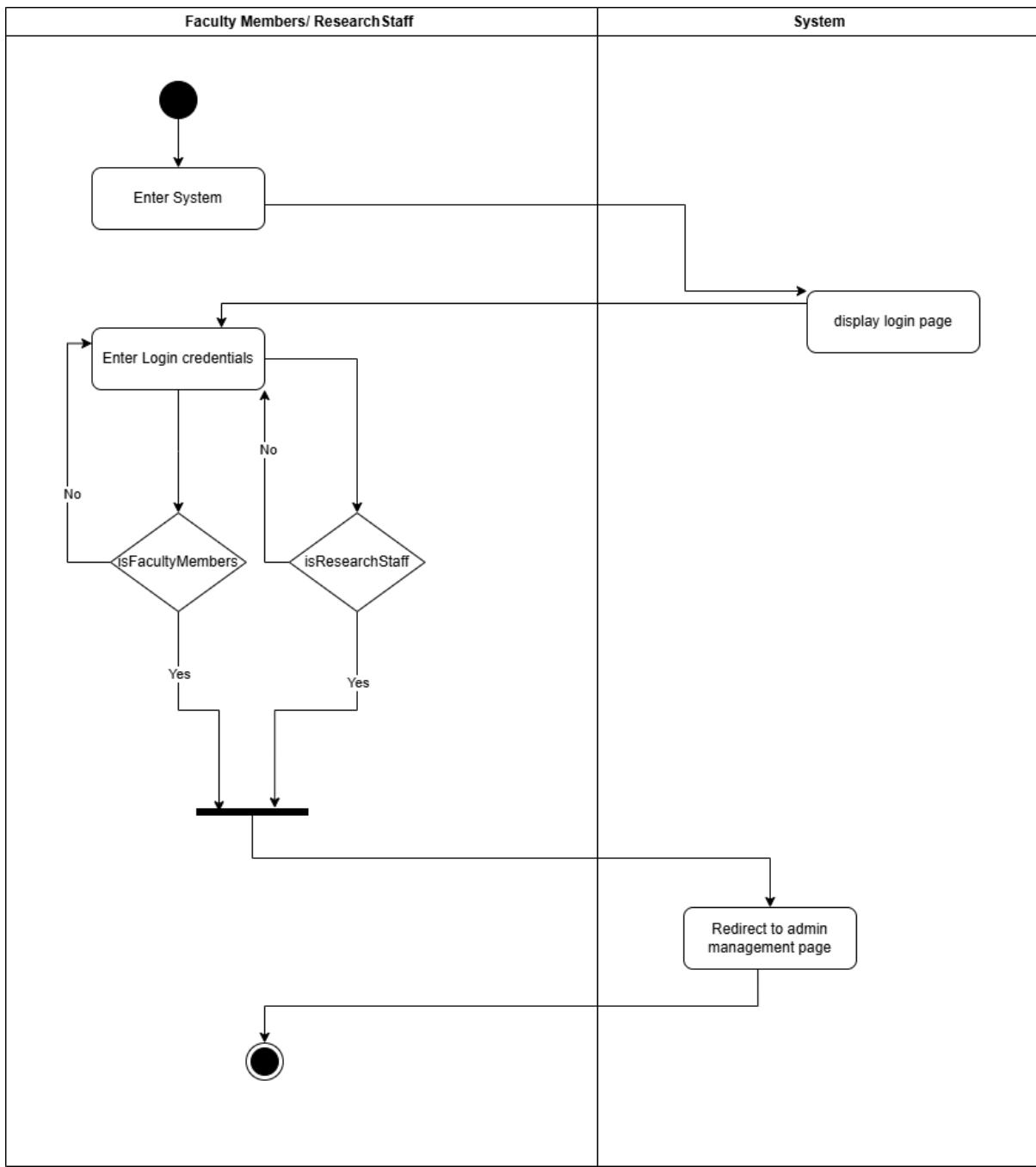
**Table 3.1: Use Case Description for Login**

<b>Use case ID</b>	UC001
<b>Use case name</b>	Login
<b>Description</b>	This use case describes flow for user log in into the system.
<b>Actor</b>	Research officer and Research Manager
<b>Pre-condition</b>	1. There is an active network connection to the platform
<b>Normal Flow(s)</b>	<ol style="list-style-type: none"> <li>1. User enters the system.</li> <li>2. Login page is displayed.</li> <li>3. User enters the email and password.</li> <li>4. User clicks on the 'Login' Button.</li> <li>5. System validates the user.</li> <li>6. If user enters wrong or invalid email or password, then exception flow 1 and exception flow 2 are performed. Otherwise proceed to normal flow 7</li> <li>7. User is logged in.</li> <li>8. Use case ends</li> </ol>

<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p>1. <b>Invalid email</b></p> <p>    1.3 The system displays invalid email message.</p> <p>    1.4 Normal Flow 3 is executed again.</p> <p>2. <b>Invalid Password</b></p> <p>    2.3 The system displays invalid password message.</p> <p>    2.4 Normal Flow 3 is executed again.</p>
<b>Post-Conditions</b>	User successfully login to the system.
<b>Related Requirement</b>	View Dashboard Analytics



**Figure 3.4: System Sequence Diagram of Login**



**Figure 3.5: Activity Diagram of Authenticate User**

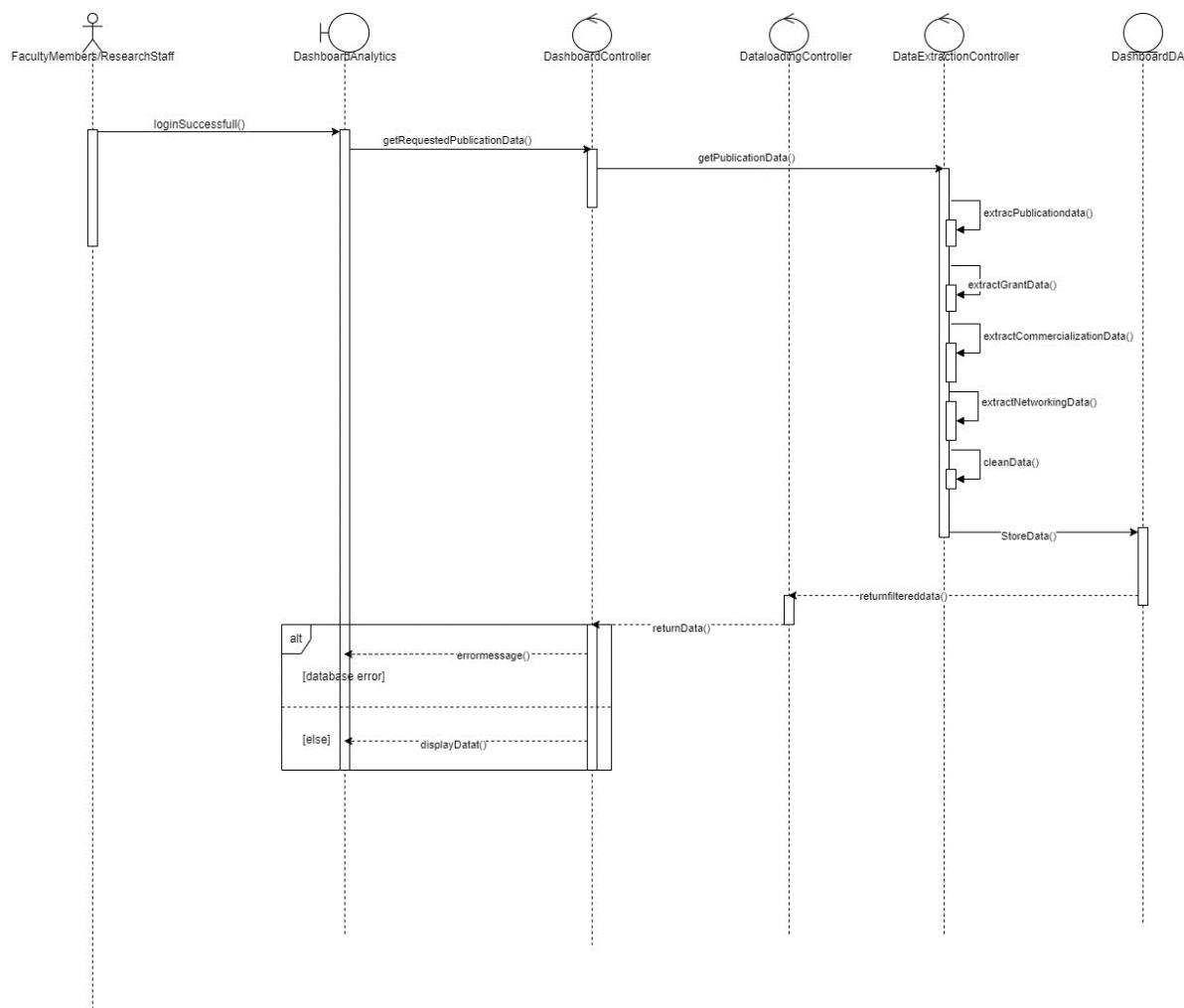
### 3.2.1.2 UC002: Use Case View Dashboard Analytics

**Table 3.2: View Dashboard Analytics**

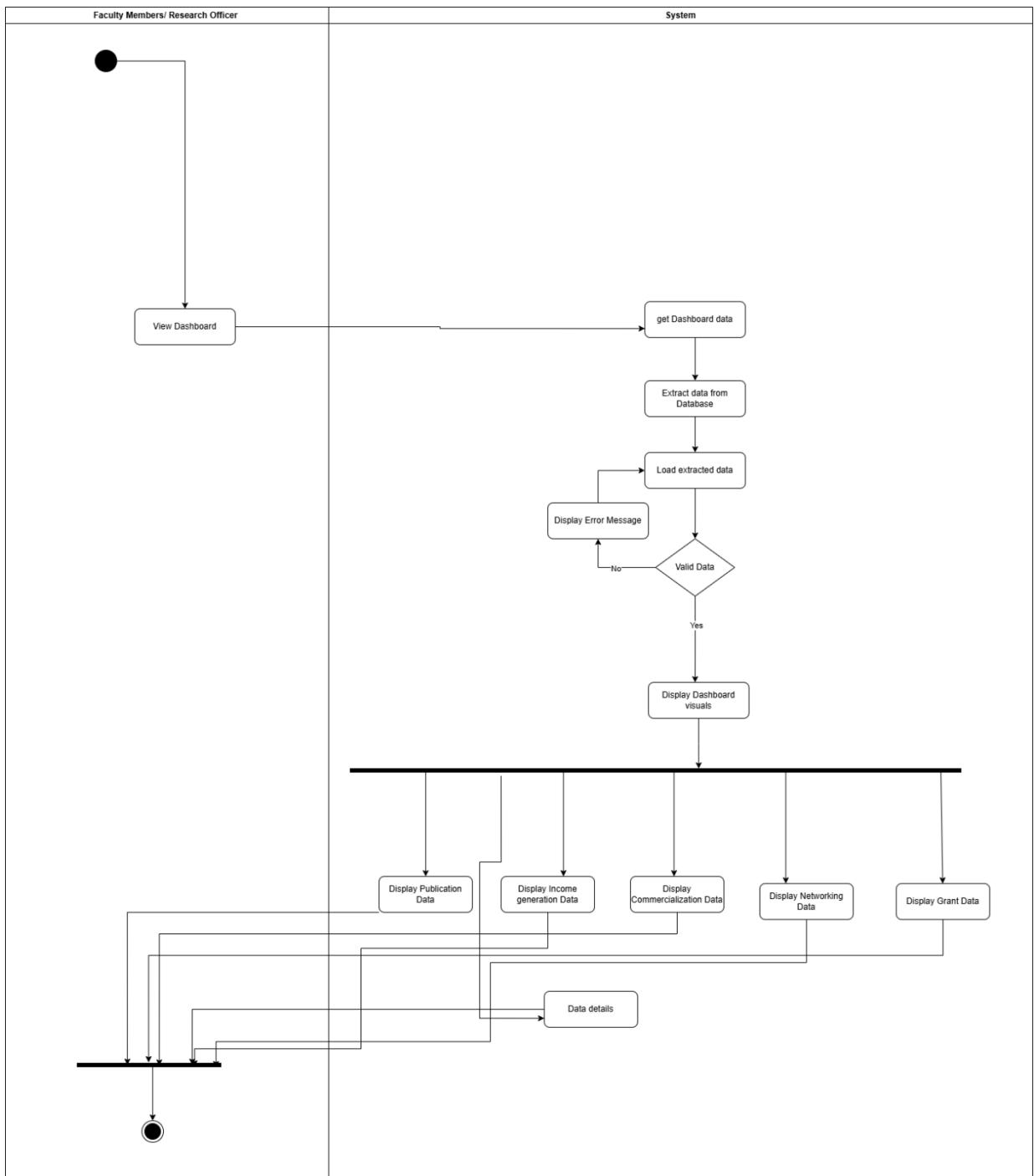
<b>Use case ID</b>	UC002
--------------------	-------

<b>Use case name</b>	View Dashboard Analytics
<b>Description</b>	This use case describes the process through which the users can view dashboard analytics to gain insights into various aspects of the faculty's research data.
<b>Actor</b>	Faculty Members, Research officer and Research Manager
<b>Pre-condition</b>	1. There is an active network connection to the platform
<b>Normal Flow(s)</b>	<p>1. Faculty member/ Research Staff enters the system</p> <p>2. User clicks on the dashboard statistics button.</p> <p>3. The system then retrieves data from the database. EF2 is performed if there's an error in retrieving data.</p> <p>4. The system will perform data preprocessing before displaying in the dashboard.</p> <p>    4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors</li> </ul> <p>    4.2. Data Transformation:</p> <ul style="list-style-type: none"> <li>• Convert data stored in database</li> </ul> <p>    4.3. Data Categorization:</p> <ul style="list-style-type: none"> <li>• Group data based on specific criteria</li> </ul> <p>5. User will be shown data based on the above methods.</p> <p>6. User can filter data and can interact with the dashboard.</p> <p>7. The system will display the pre-processed data in the form of graphs. If data is not available, AF1 will be performed. If there's an error in retrieving data from database, EF1 will be performed.</p>

<b>Alternative Flow(s)</b>	<p><b>1. Unavailable Data:</b></p> <ol style="list-style-type: none"> <li>1.1. The system displays an error message warning about the unavailability of the data</li> <li>1.2. The user is prompted to Continue from NF3</li> </ol>
<b>Exception Flow(s)</b>	<p><b>1. Connection error with the database:</b></p> <ol style="list-style-type: none"> <li>1.1. The system displays an error message.</li> <li>1.2. User is prompted to start from NF3.</li> </ol> <p><b>2. Error Retrieving data from Database:</b></p> <ol style="list-style-type: none"> <li>2.1. System displays an error message.</li> <li>2.2. User is prompted to retry.</li> </ol>
<b>Post-condition</b>	<ol style="list-style-type: none"> <li>1. User can view the dashboard analytics.</li> </ol>
<b>Related Requirement</b>	-

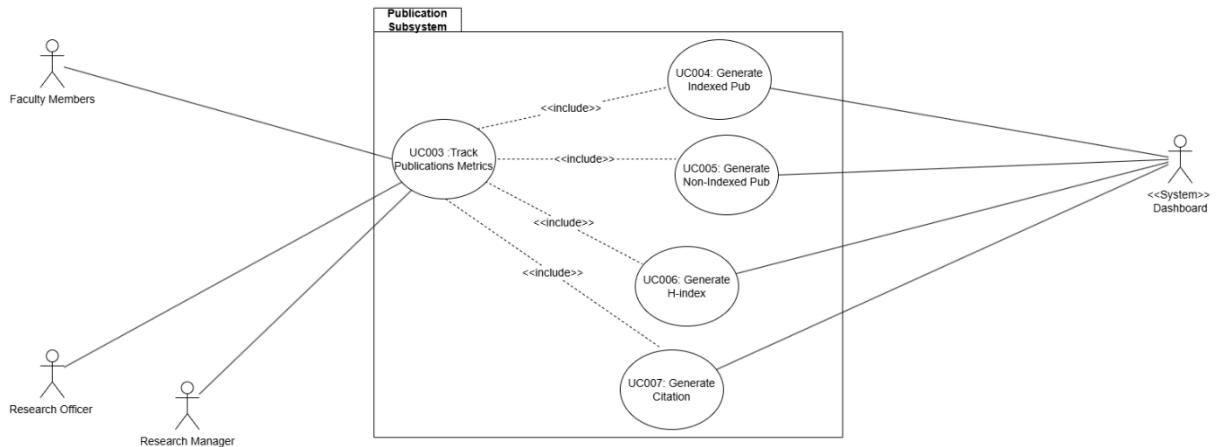


**Figure 3.6: Sequence diagram for View Dashboard Analytics**



**Figure 3.7: Activity Diagram for View Dashboard Analytics**

### 3.2.2 Module Publication Subsystem



**Figure 3.8: Module Publication Subsystem**

Functional Requirements for the following module are given below:

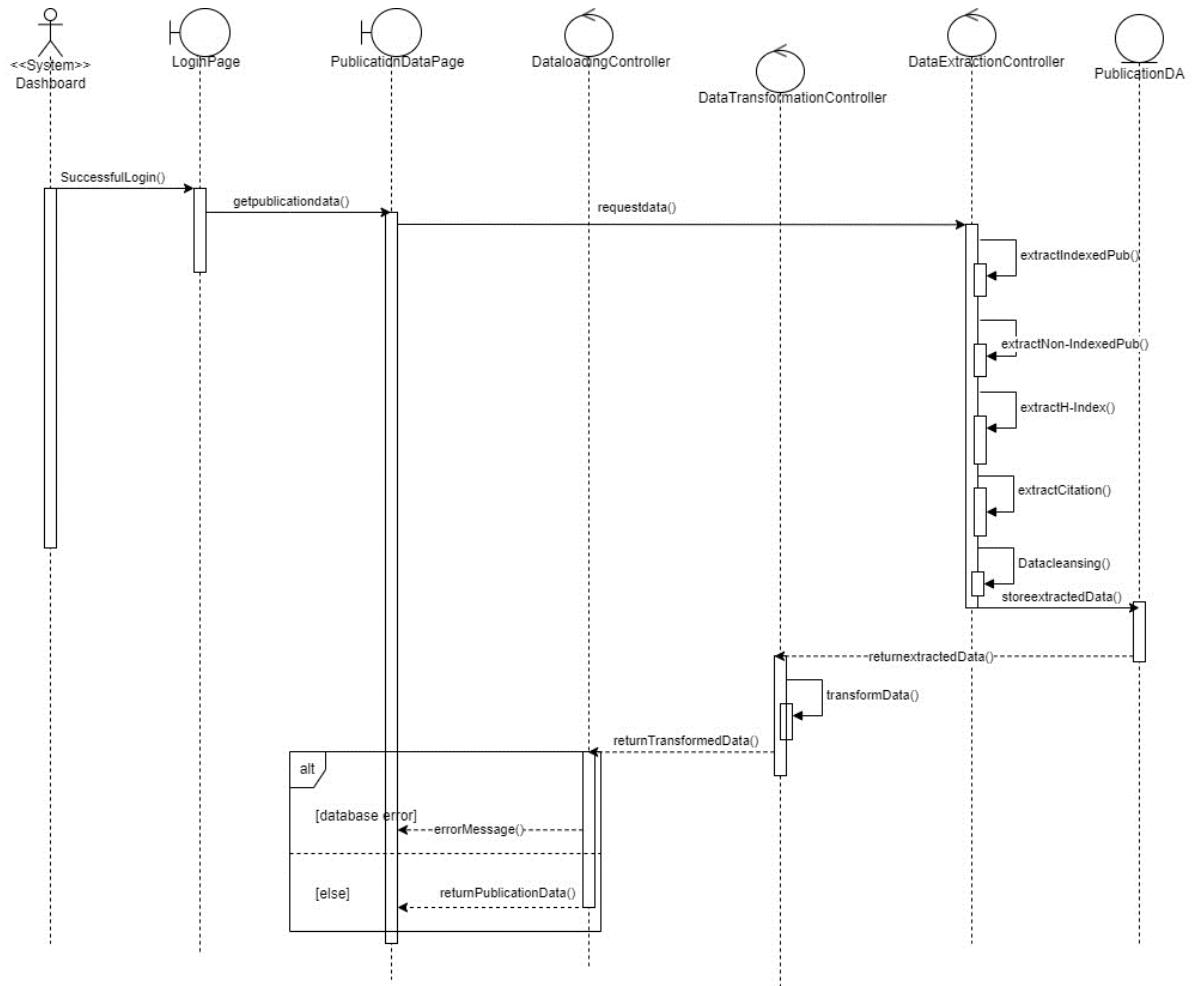
1. FR001: Track Publication Metrics- The user can view and track the publication metrics of the publications made.
2. FR002: Generate Indexed Publications- The system will generate indexed publications by adapting data science techniques like data cleansing, data filtering, data categorization and data loading.
3. FR003: Generate Non-Indexed Publications- The system will generate non-indexed publications by adapting data science techniques like data cleansing, data filtering, data categorization and data loading.
4. FR004: Generate H-Index Publications- The system will generate H-Index by adapting data science techniques like data cleansing, data filtering, data categorization and data loading.
5. FR005: Generate Citation- The system will generate Citation by adapting data science techniques like data cleansing, data filtering, data categorization and data loading.

### 3.2.2.1 UC003: Use case Track Publication Metrics

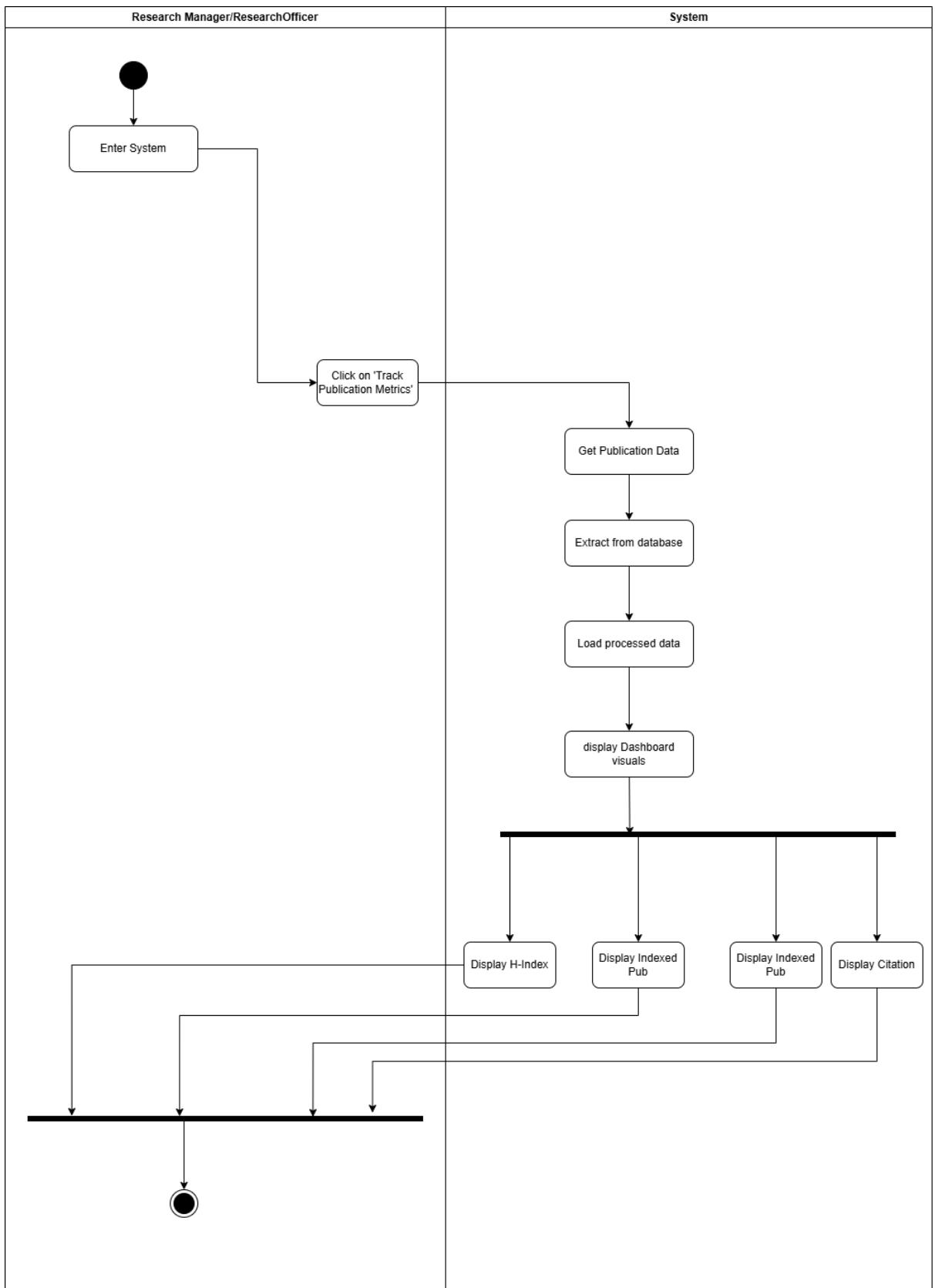
**Table 3.3: Track Publication Metrics**

<b>Use case ID</b>	UC003
<b>Use case name</b>	Track Publication Metrics
<b>Description</b>	This use case describes the process through which the users can view Publication metrics generated.
<b>Actor</b>	Faculty Members, Research officer and Research Manager
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User clicks on the dashboard.</p> <p>2. From the main dashboard page, user clicks on the “Track Publication metrics” option. If no data is available, AF 1 is performed.</p> <p>3. The system generates the scraping process to retrieve publication data from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors</li> <li>• Include Title, Authors, Publication date, Indexed Status (indexed or non-indexed)</li> </ul> <p>4.2. Data Transformation</p>

	<ul style="list-style-type: none"> <li>• Convert data stored in database to displaying format.</li> </ul> <ol style="list-style-type: none"> <li>5. System then stores the pre-processed publication data in the database.</li> <li>6. Publication metrics is updated based on the new publication data. If right data is not displayed, EF3 is performed.</li> <li>7. User can view the updated publication metrics in the form of graphs and charts allowing them to visualize the growth and development. If there's a connection error, EF 1 is performed. If there's an error in retrieving data from database, EF2 is performed.</li> <li>8. User can view the publication details by hovering over the respective data points in the graphs or charts.</li> </ol>
<b>Alternative Flow(s)</b>	<p><b>1. Unavailable Data:</b></p> <ol style="list-style-type: none"> <li>1.1. The system displays a message warning about the unavailability of the data</li> <li>1.2. The user is prompted to Continue from NF3</li> </ol>
<b>Exception Flow(s)</b>	<ol style="list-style-type: none"> <li><b>1. Connection error with the database:</b> <ol style="list-style-type: none"> <li>1.1. The system displays an error message.</li> <li>1.2. User is prompted to refresh and start from NF8.</li> </ol> </li> <li><b>2. Error Retrieving data from Database:</b> <ol style="list-style-type: none"> <li>2.1. System displays an error message.</li> <li>2.2. User is prompted to retry.</li> </ol> </li> <li><b>3. Wrong data Displayed:</b> <ol style="list-style-type: none"> <li>3.1. User will contact administrator.</li> <li>3.2. Data preprocessing will be done again.</li> </ol> </li> </ol>
<b>Post-condition</b>	User can view and track the publication data metrics.
<b>Related Requirement</b>	-



**Figure 3.9: Sequence Diagram for UC003**

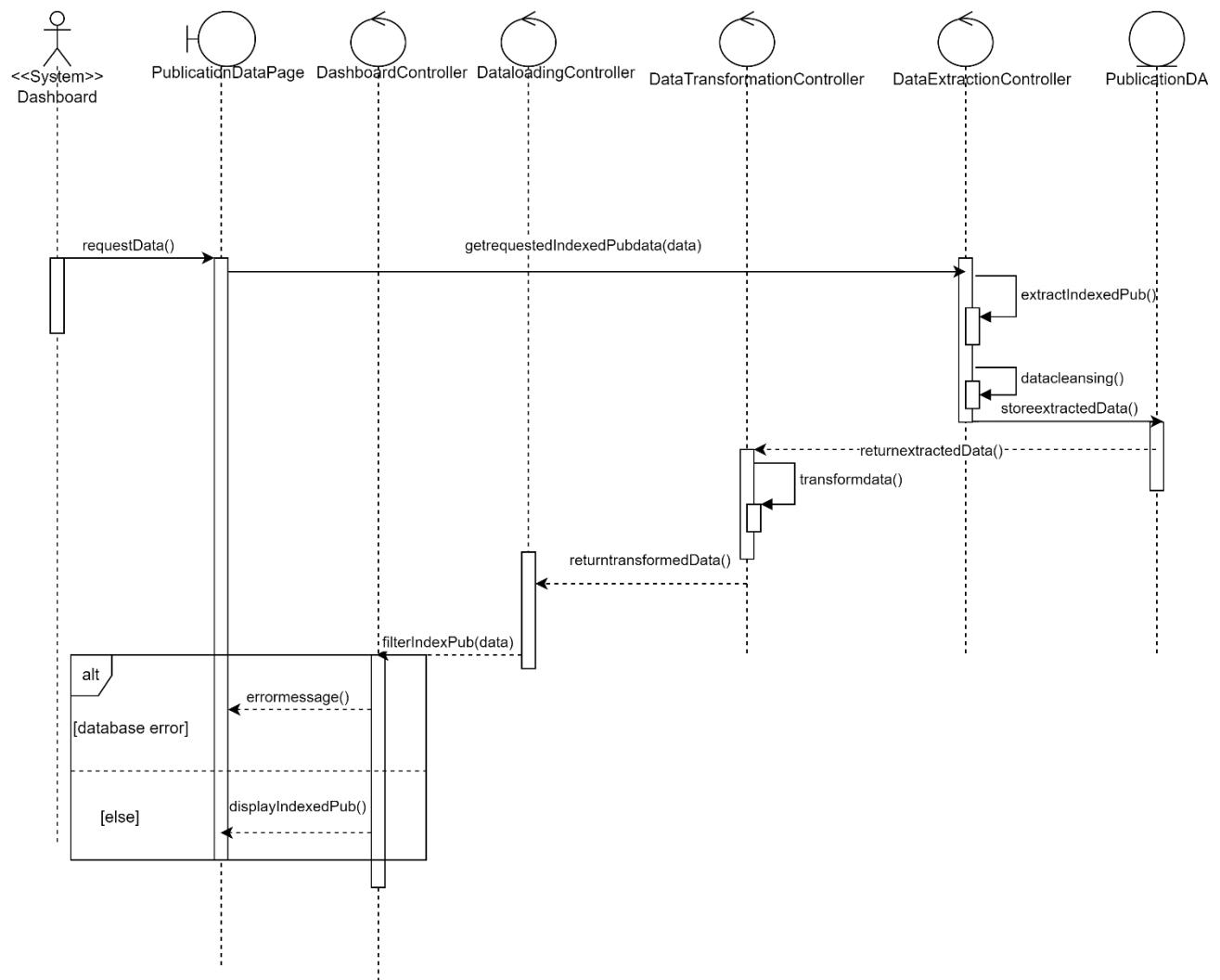


**Figure 3.10: Activity for UC003**

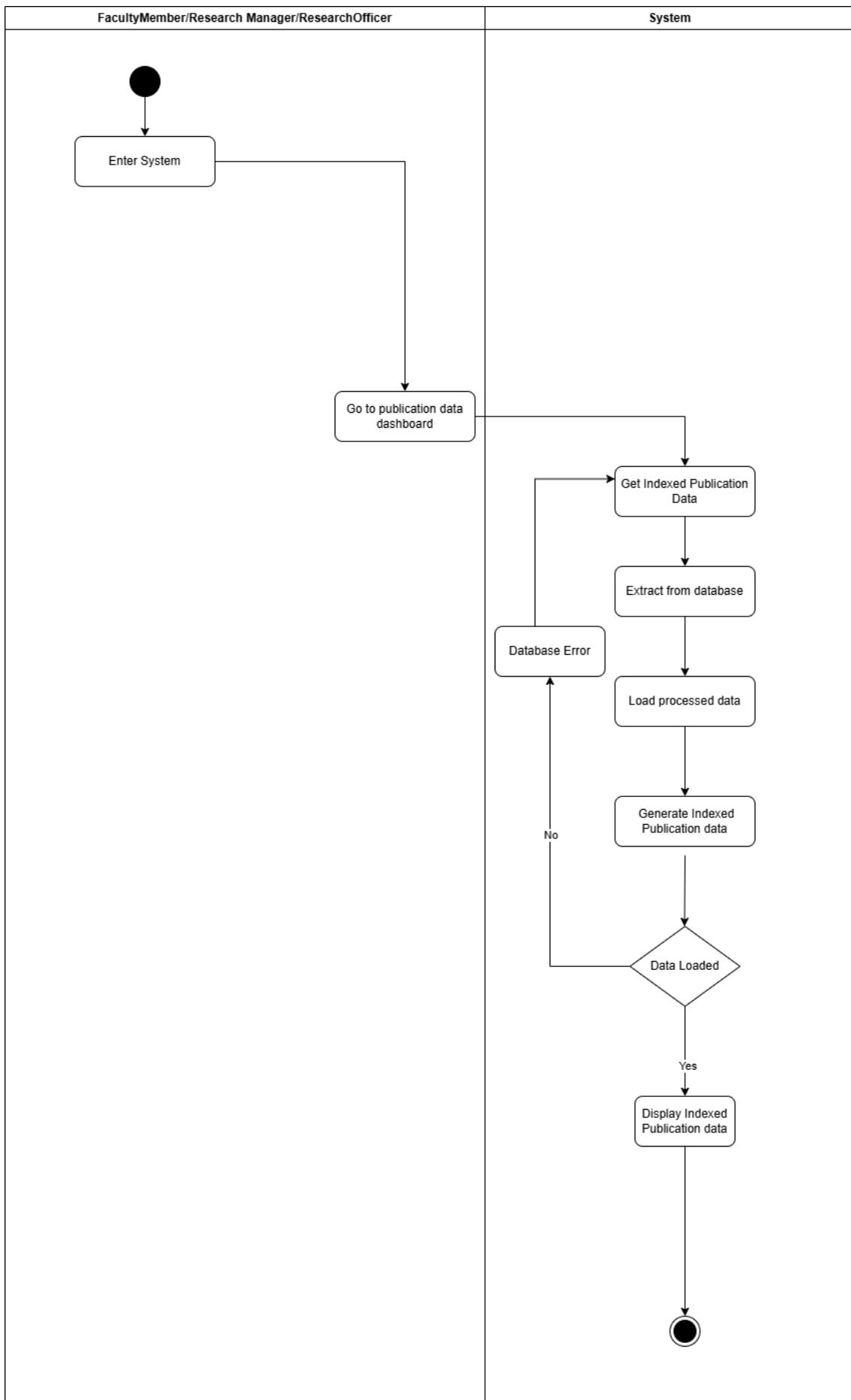
### 3.2.2.2 UC004: Use Case Generate Indexed Publication

<b>Use case ID</b>	UC004
<b>Use case name</b>	Generate Indexed Publications
<b>Description</b>	This use case describes the process through which system generates Indexed Publications.
<b>Actor</b>	System (Dashboard)
<b>Pre-condition</b>	<p>1. There is an active network connection to the platform.</p>
<b>Normal Flow(s)</b>	<p>1. User enters the system and goes to the publications section</p> <p>2. From the main dashboard page, user clicks on the “Track Publications” option.</p> <p>3. The system generates the data collection process to retrieve publication data that was collected through scraping including the Indexed Publications from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Include only the Indexed publication data.</li> </ul> <p>5. System then stores the pre-processed publication data in the database.</p> <p>6. System displays the Indexed publication data after filtering the publication data and retrieving data from the database. If there's an error while retrieving data, EF2 will be performed.</p> <p>7. System displays the data in the form of Bar/Pie chart.</p>

	8. User can view the publication details by hovering over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 3 will be performed.
<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <p>1.1. System displays an error message. 1.2. User is prompted to retry.</p> <p><b>2. Wrong data Displayed:</b></p> <p>2.1. User will contact administrator. 2.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view and track the publication data metrics.
<b>Related Requirement</b>	-



**Figure 3.11: Sequence for UC004**

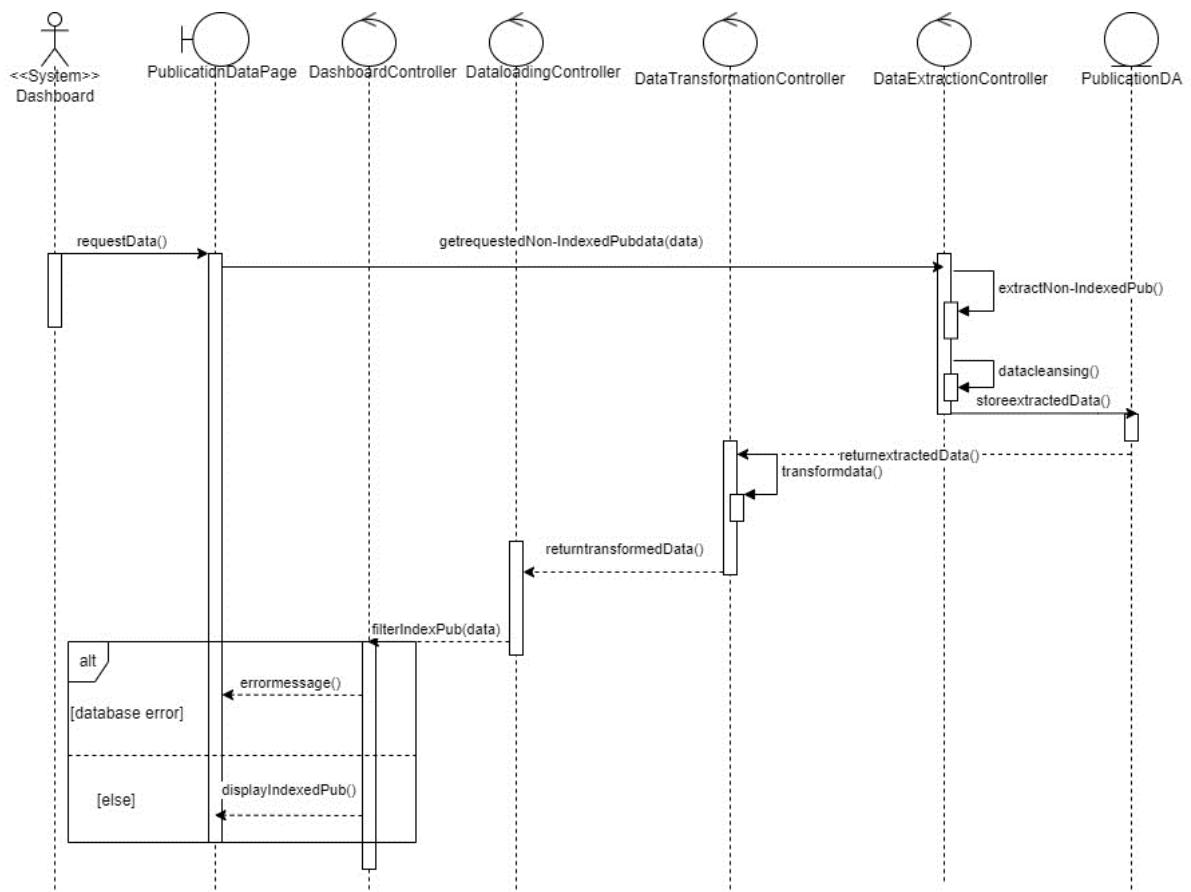


**Figure 3.12: Activity for UC004**

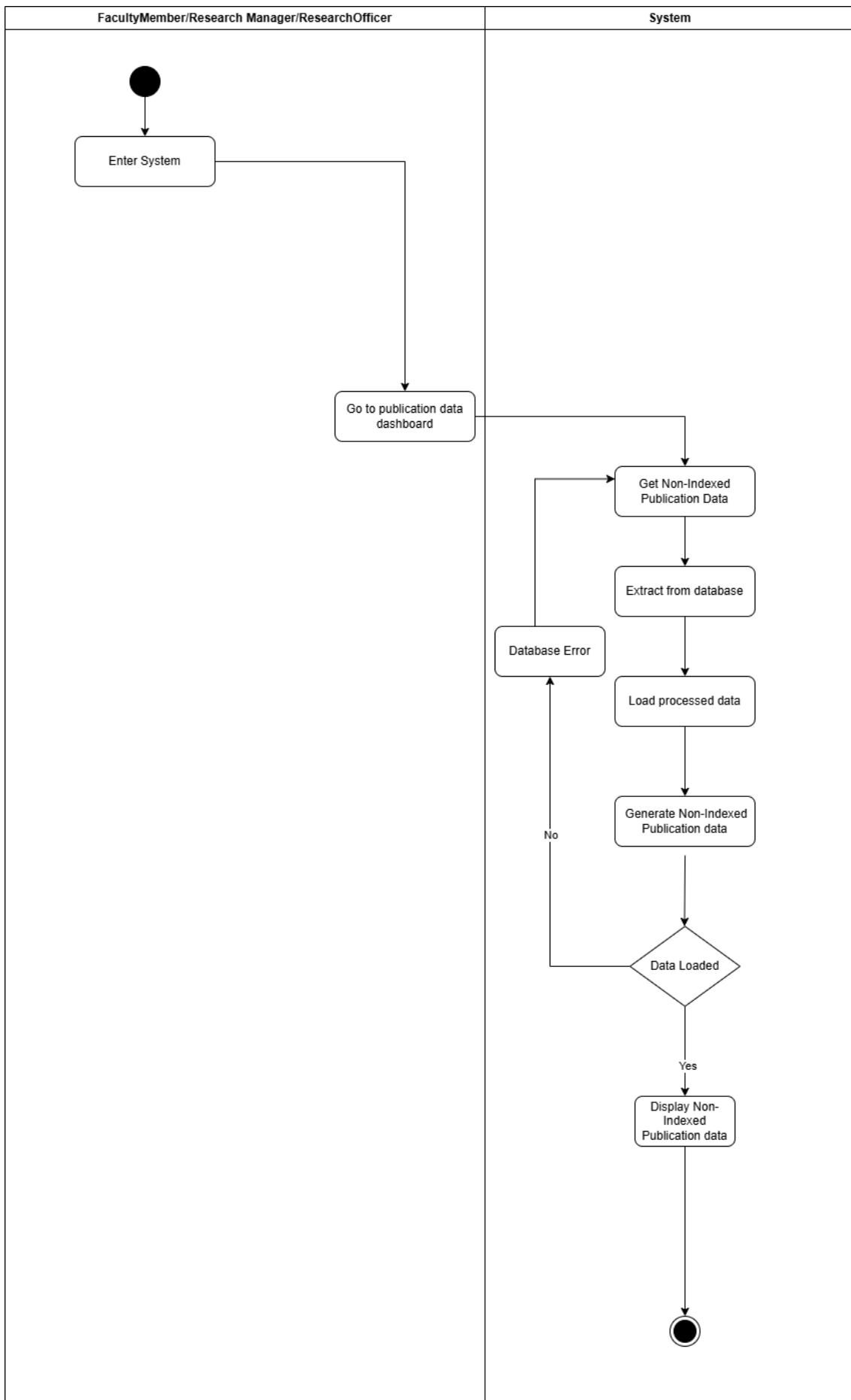
### 3.2.2.3 UC005: Use Case Generate Non-Indexed Publication

<b>Use case ID</b>	UC005
<b>Use case name</b>	Generate Non-Indexed Publications
<b>Description</b>	This use case describes the process through which system generates Non-Indexed Publications.
<b>Actor</b>	System (Dashboard)
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user goes to the “Track Publication metrics” option.</p> <p>3. The system generates the data collection process to retrieve publication data that was collected through scraping including the Non-Indexed Publications from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Include only the Non-Indexed publication data.</li> </ul> <p>5. System then stores the pre-processed publication data in the database.</p> <p>6. System displays the non-Indexed publication data after filtering the publication data and retrieving data from the database. If there’s an error while retrieving data, EF2 will be performed.</p>

	<p>7. System displays the data in the form of Bar/Pie chart.</p> <p>8. User can view the publication details by hovering over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 3 will be performed.</p>
<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <p>1.1. System displays an error message. 1.2. User is prompted to retry.</p> <p><b>2. Wrong data Displayed:</b></p> <p>2.1. User will contact administrator. 2.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view and track the non-indexed publication data metrics.
<b>Related Requirement</b>	-



**Figure 3.13: Sequence for UC005**

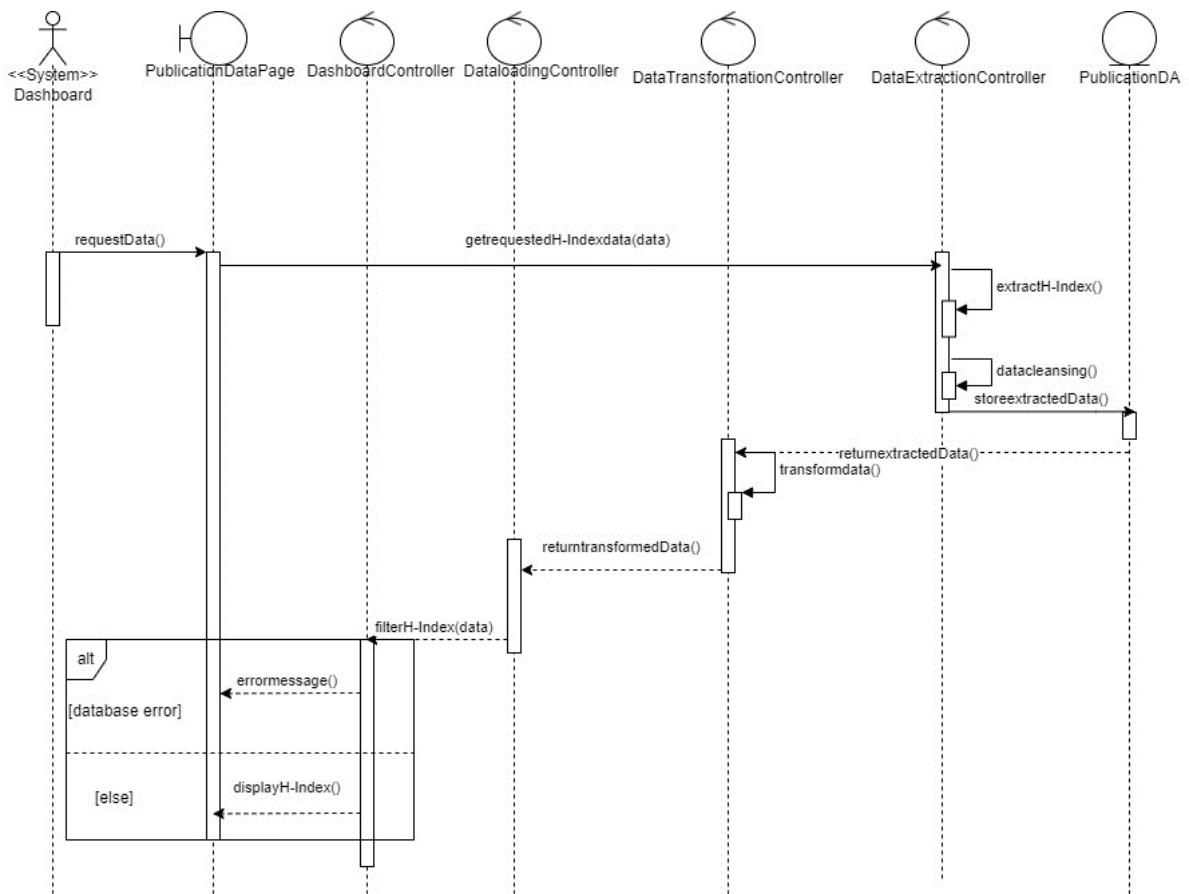


**Figure 3.14: Activity for UC005**

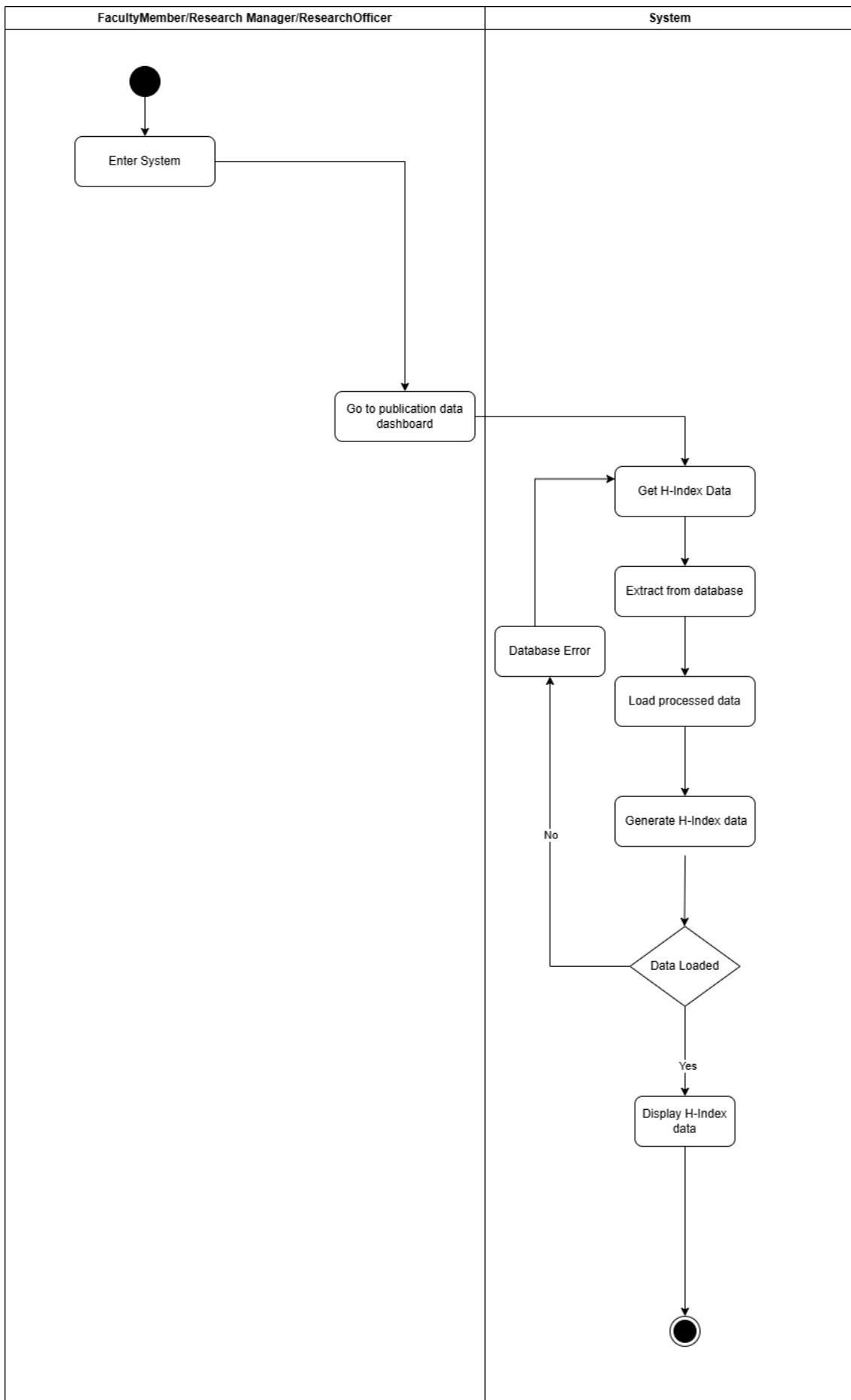
### 3.2.2.4 UC006: Use Case Generate H-Index

<b>Use case ID</b>	UC006
<b>Use case name</b>	Generate H-Index
<b>Description</b>	This use case describes the process through which system generates H-Index.
<b>Actor</b>	System (Dashboard)
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user clicks on the “H-Index(Scopus)” option.</p> <p>3. The system generates the data collection process to retrieve publication data that was collected through scraping including the H-Index Publications from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Include only the H-Index publication data.</li> </ul> <p>5. System then stores the pre-processed publication data in the database.</p>

	<p>6. System displays the H-Index publication data after filtering the publication data and retrieving data from the database. If there's an error while retrieving data, EF2 will be performed.</p> <p>7. System displays the data in the form of Bar/Pie chart.</p> <p>8. User can view the details by clicking over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 3 will be performed.</p>
<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <ul style="list-style-type: none"> <li>a. System displays an error message.</li> <li>b. User is prompted to retry.</li> </ul> <p><b>2. Wrong data Displayed:</b></p> <ul style="list-style-type: none"> <li>a. User will contact administrator.</li> <li>b. Data preprocessing will be done again.</li> </ul>
<b>Post-condition</b>	User can view and track the H-Index publication data metrics.
<b>Related Requirement</b>	-



**Figure 3.15: Sequence for UC006**

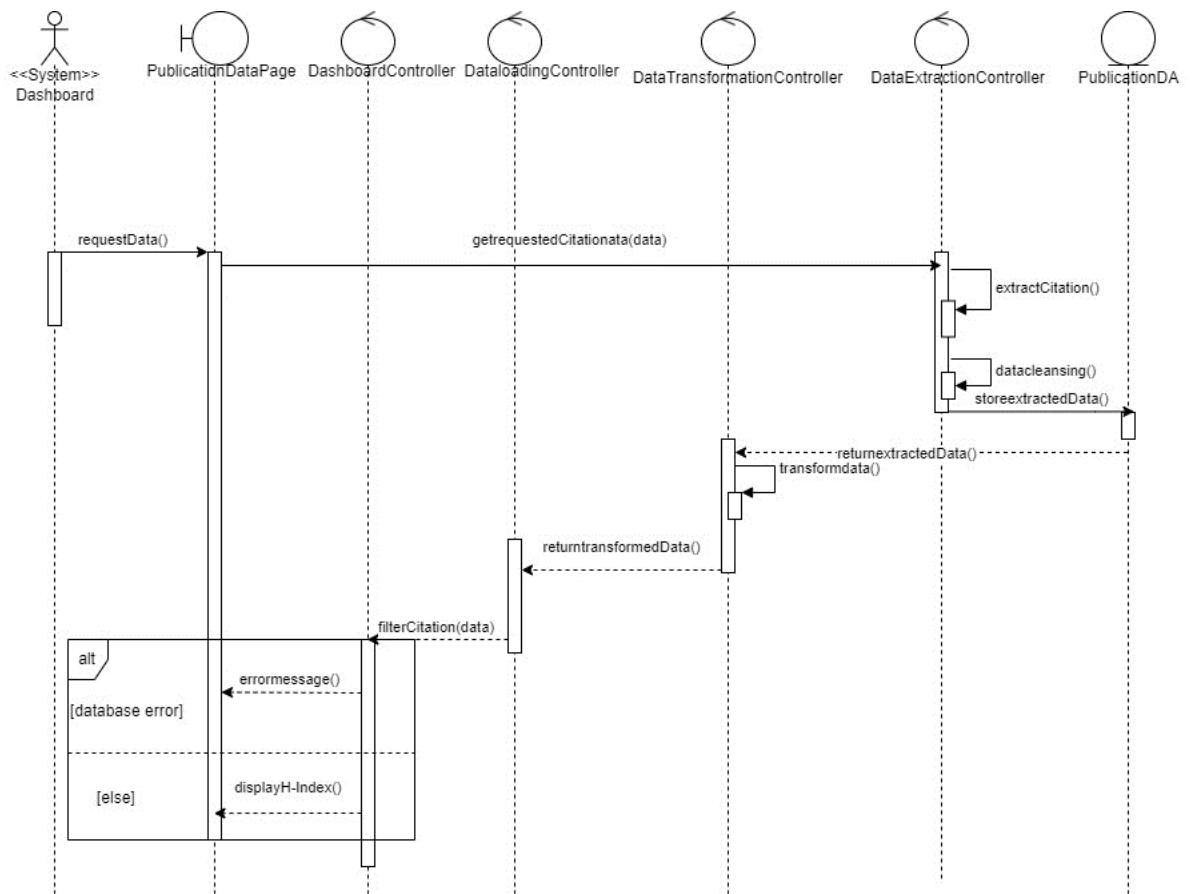


**Figure 3.16: Activity Diagram for UC006**

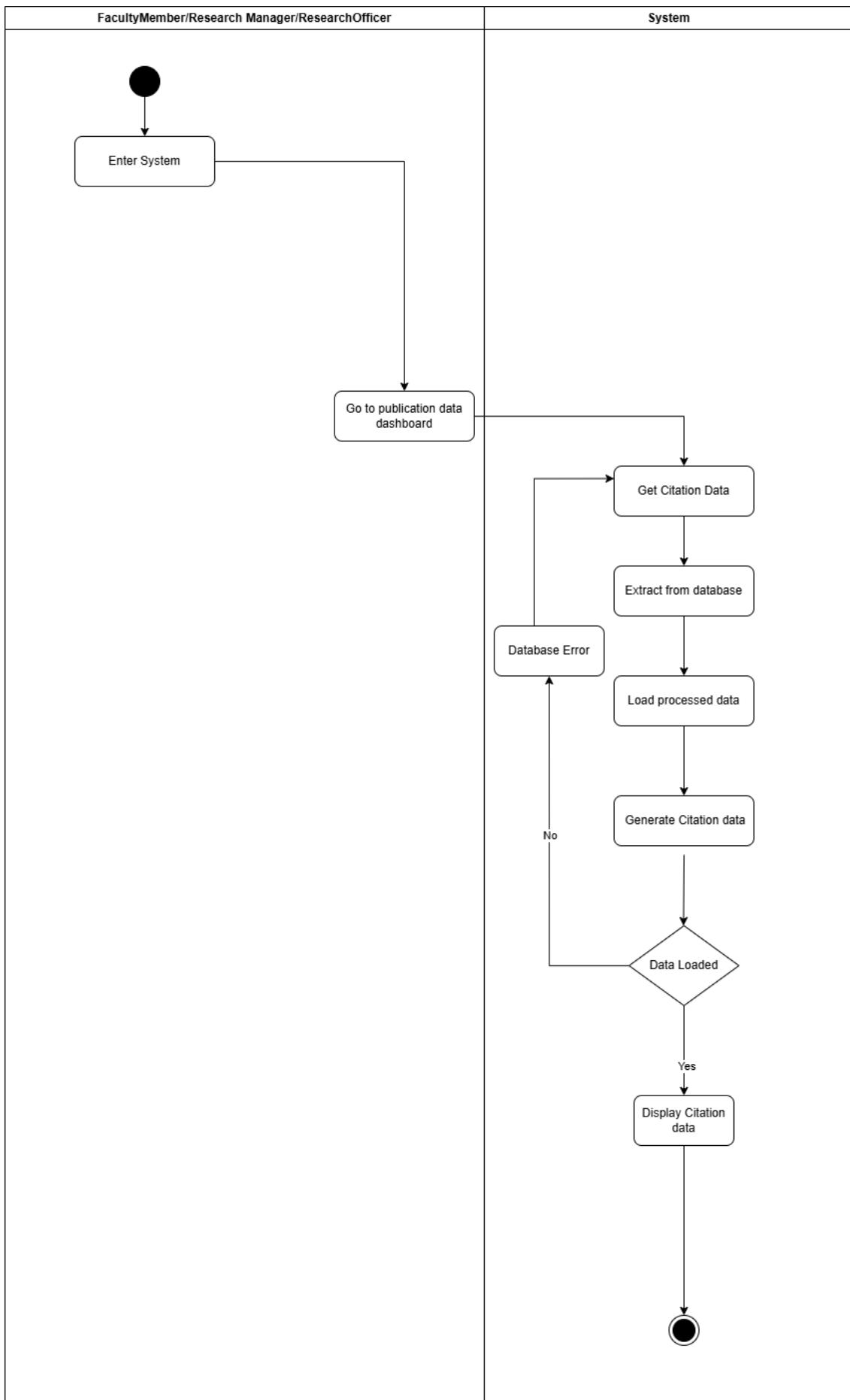
### 3.2.2.5 UC007: Use Case Generate Citation

<b>Use case ID</b>	UC007
<b>Use case name</b>	Generate Citation
<b>Description</b>	This use case describes the process through which system generates Citations.
<b>Actor</b>	System (Dashboard)
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user clicks on the “Citations (Scopus)” option.</p> <p>3. The system generates the data collection process to retrieve publication data that was collected through scraping including the citation Publications from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Include only the citation publication data.</li> </ul> <p>5. System then stores the pre-processed publication data in the database.</p>

	<p>6. System displays the citation publication data after filtering the publication data and retrieving data from the database. If there's an error while retrieving data, EF2 will be performed.</p> <p>7. System displays the data in the form of Bar/Pie chart.</p> <p>8. User can view the publication details by clicking over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 3 will be performed.</p>
<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <p>1.1. System displays an error message. 1.2. User is prompted to retry.</p> <p><b>2. Wrong data Displayed:</b></p> <p>2.1. User will contact administrator. 2.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view and track the citation data.
<b>Related Requirement</b>	-

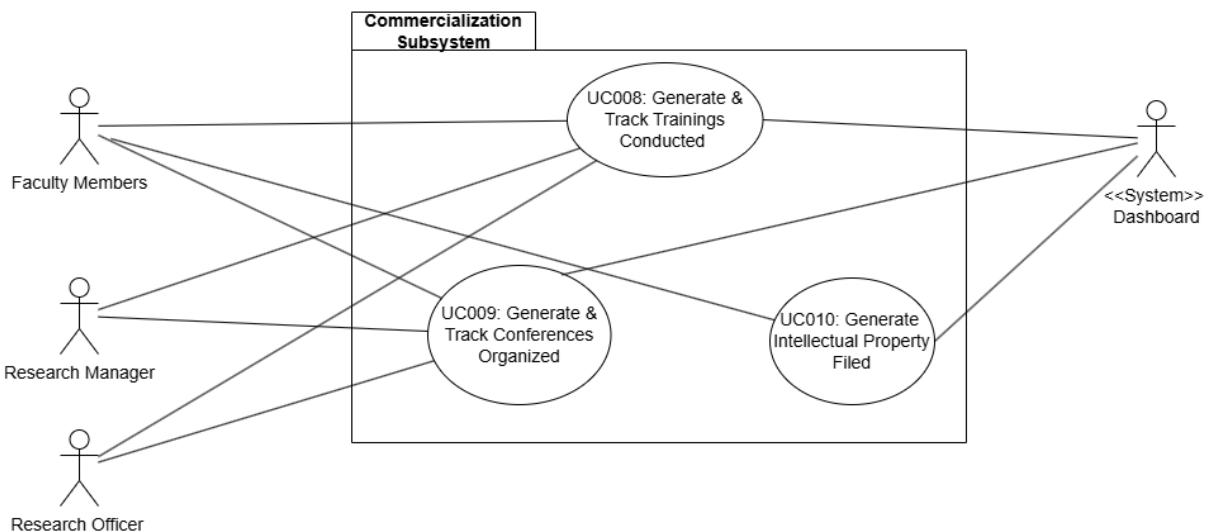


**Figure 3.17: Sequence Diagram for UC007**



**Figure 3.18: Activity Diagram for UC007**

### 3.2.3 Module Commercialization Subsystem



**Figure 3.19: Module 003 Commercialization Subsystem**

Functional Requirements are given below:

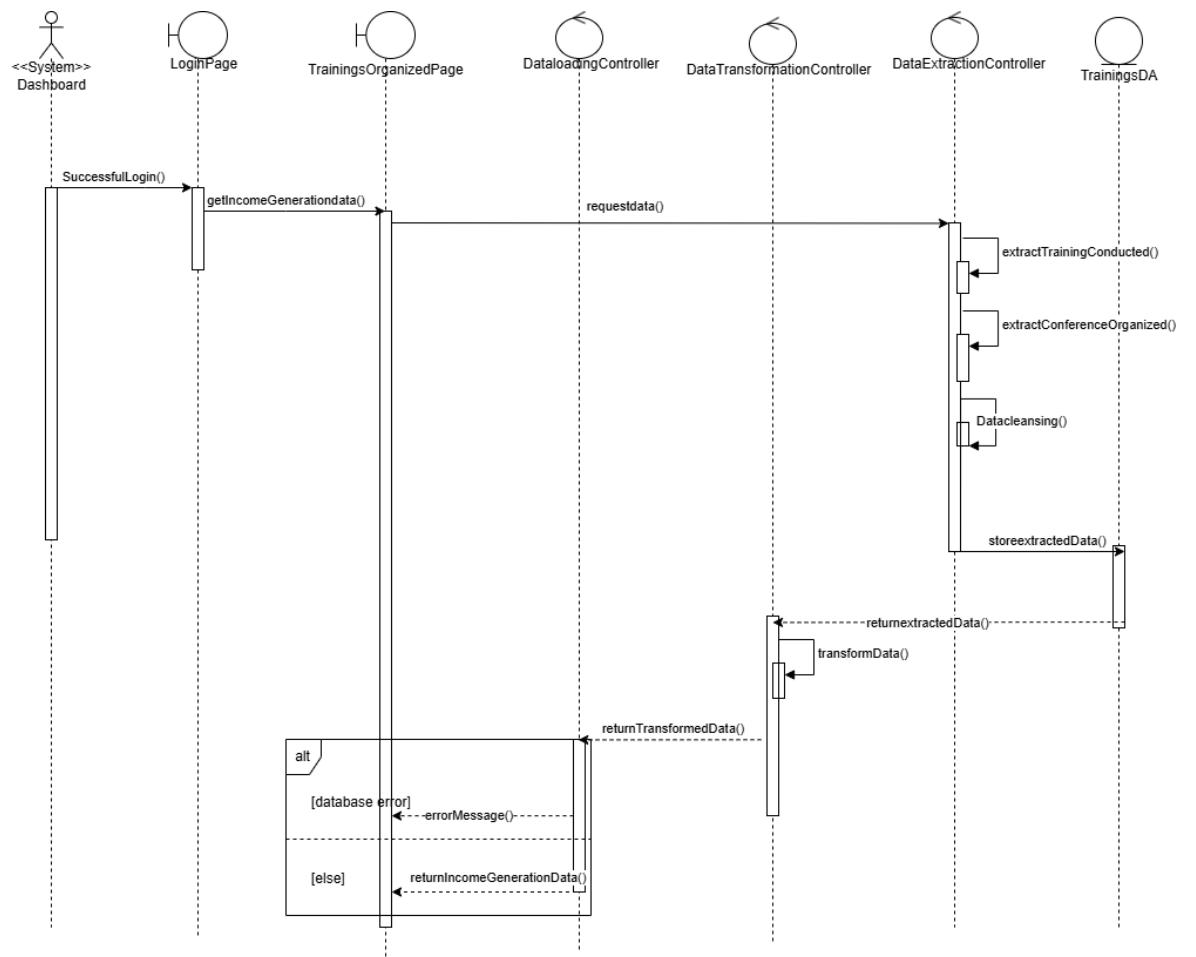
- FR001: Generate & Track Trainings Conducted: This will allow the Research staff to track the income generation of the faculty.
- FR002: Generate & Track Conferences Organized: This will allow the Research staff to track the commercialization data of the faculty of computing.
- FR003: Generate Intellectual Property Filed: This will allow the system dashboard to generate the data of Intellectual Property filed.

#### 3.2.3.1 UC008: Use case Generate & Track Trainings Conducted

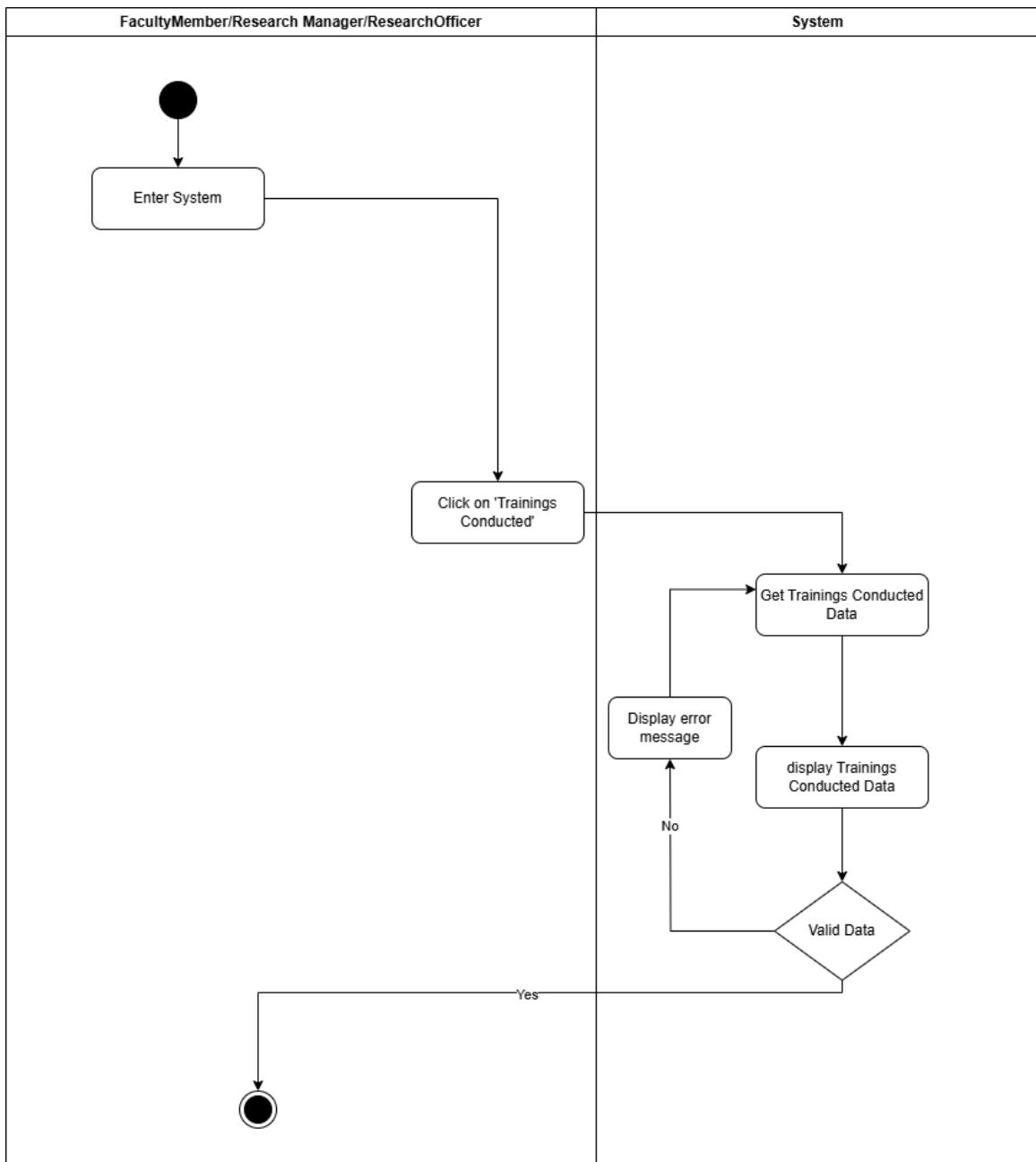
Use case ID	UC008
Use case name	Generate & Track Trainings Conducted

<b>Description</b>	This use case describes the process through which the users can view and track the Trainings Conducted.
<b>Actor</b>	Faculty Members, Research officer, Research Manager, Dashboard
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user clicks on the “Trainings conducted” option. If no data is available, AF 1 is performed.</p> <p>3. The system generates starts to retrieve Trainings Conducted data from database.</p> <p>4. The collected data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors</li> <li>• Include conference organized, training Conducted</li> </ul> <p>4.2. Data Transformation</p> <ul style="list-style-type: none"> <li>• Convert data stored in excel in different format.</li> </ul> <p>5. System then stores the pre-processed Trainings Conducted data in the database.</p> <p>6. Trainings Conducted is updated based on the new Trainings Conducted data. If right data is not displayed, EF3 is performed.</p> <p>7. User can view the updated Trainings Conducted metrics in the form of graphs and charts allowing them to visualize the growth and development. If there is a connection error, EF 1 is performed. If there is an error in retrieving data from database, EF2 is performed.</p>

	8. User can view Trainings Conducted details by clicking over the respective data points in the graphs or charts.
<b>Alternative Flow(s)</b>	<p><b>1. Unavailable Data:</b></p> 1.1. The system displays a message warning about the unavailability of the data. 1.2. The user is prompted to Continue from NF3
<b>Exception Flow(s)</b>	<p><b>1. Connection error with the database:</b></p> 1.1. The system displays an error message. 1.2. User is prompted to refresh and start from NF8. <p><b>2. Error Retrieving data from Database:</b></p> 2.1. System displays an error message. 2.2. User is prompted to retry. <p><b>3. Wrong data Displayed:</b></p> 3.1. User will contact administrator. 3.2. Data preprocessing will be done again.
<b>Post-condition</b>	User can view and track the Trainings Conducted data.
<b>Related Requirement</b>	-



**Figure 3.20: Sequence Diagram for UC008**



**Figure 3.21: Activity Diagram for UC008**

### 3.2.3.2 UC009: Use Case Generate & Track Conferences Organized

<b>Use case ID</b>	UC009
<b>Use case name</b>	Generate & Track Conferences Organized
<b>Description</b>	This use case describes the process through which the users can view and track the Conferences Organized Data.
<b>Actor</b>	Faculty Members, Research officer, Research Manager, Dashboard

<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user clicks on the “Conferences Organized” option. If no data is available, AF 1 is performed.</p> <p>3. The system generates the process to retrieve Conferences Organized data from database.</p> <p>4. The collected data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors</li> <li>• Include Intellectual Property filed.</li> </ul> <p>4.2. Data Transformation</p> <ul style="list-style-type: none"> <li>• Convert data stored in database.</li> </ul> <p>5. System then stores the pre-processed Conferences Organized data in the database.</p> <p>6. Conferences Organized is updated based on the new Conferences Organized data. If right data is not displayed, EF3 is performed.</p> <p>7. User can view the updated Conferences Organized in the form of graphs and charts allowing them to visualize the growth and development. If there's a connection error, EF 1 is performed. If there's an error in retrieving data from database, EF2 is performed.</p> <p>8. User can view the Conferences Organized details by clicking over the respective data points in the graphs or charts.</p>

<b>Alternative Flow(s)</b>	<p><b>1. Unavailable Data:</b></p> <p>1.1. The system displays a message warning about the unavailability of the data.</p> <p>1.2. The user is prompted to Continue from NF3</p>
<b>Exception Flow(s)</b>	<p><b>1. Connection error with the database:</b></p> <p>1.1. The system displays an error message.</p> <p>1.2. User is prompted to refresh and start from NF8.</p> <p><b>2. Error Retrieving data from Database:</b></p> <p>2.1. System displays an error message.</p> <p>2.2. User is prompted to retry.</p> <p><b>3. Wrong data Displayed:</b></p> <p>3.1. User will contact administrator.</p> <p>3.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view and track the Commercialization data.
<b>Related Requirement</b>	-

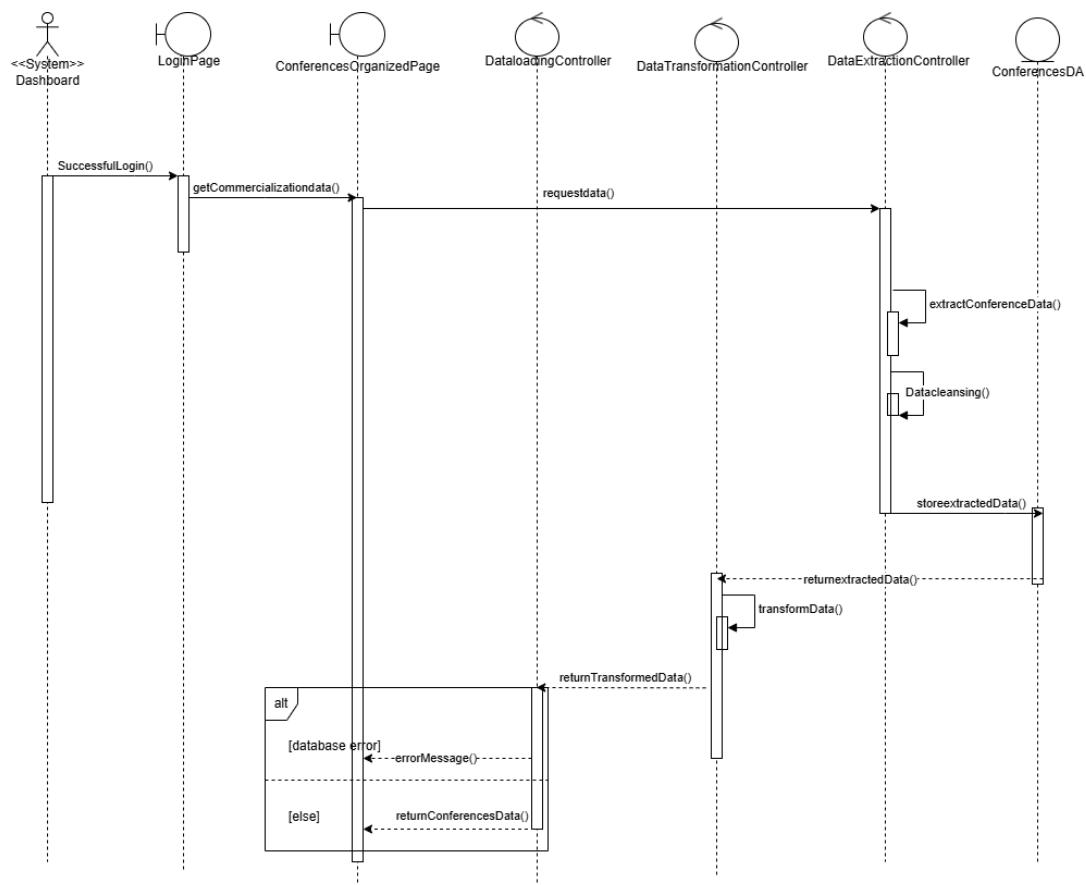
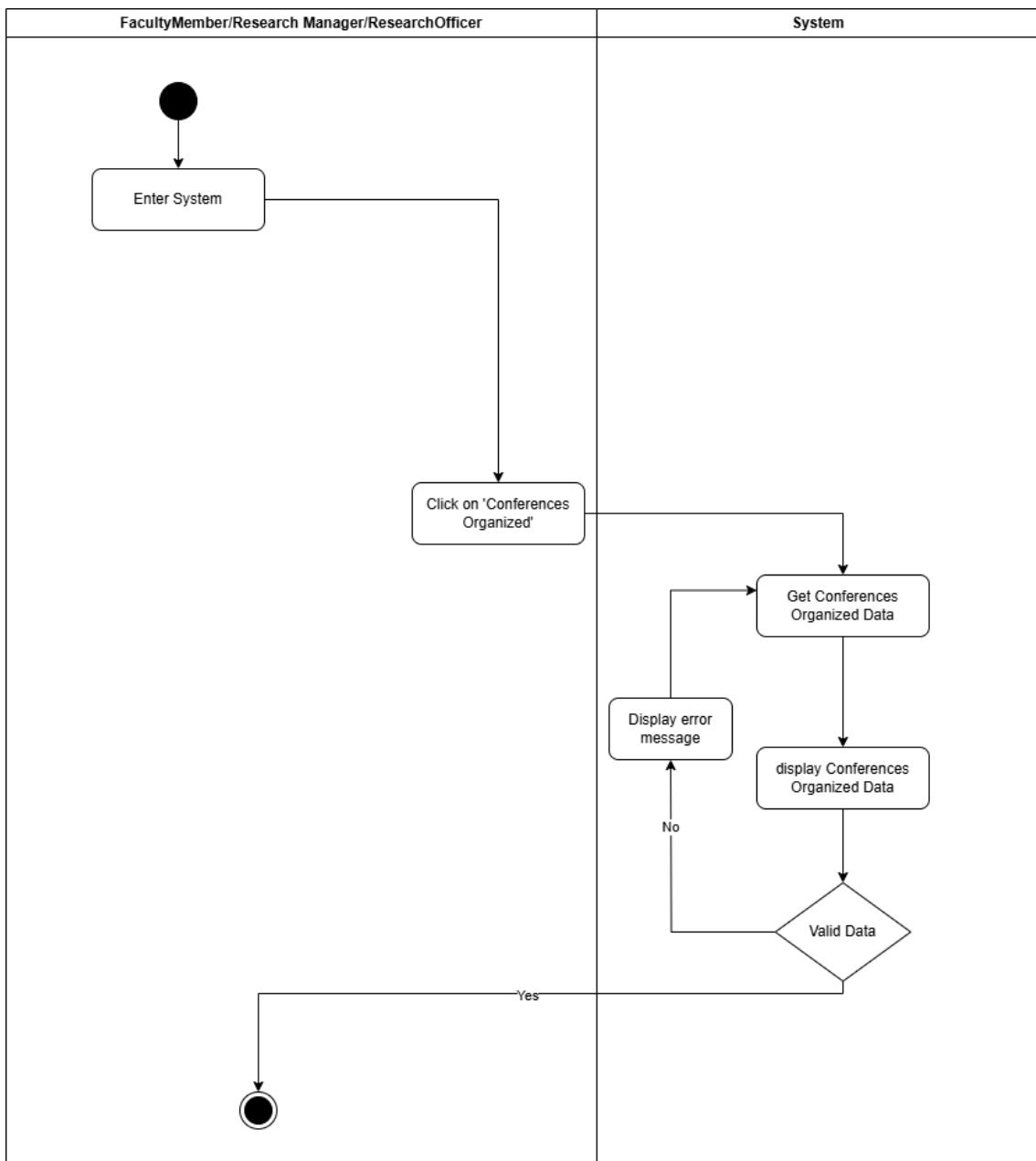


Figure 3.22: Sequence Diagram for UC009



**Figure 3.23: Activity Diagram for UC009**

### 3.2.3.3 UC010: Use Case Generate Intellectual Property Filed

<b>Use case ID</b>	UC010
<b>Use case name</b>	Generate Intellectual Property Filed
<b>Description</b>	This use case describes the process through which system generates Intellectual Property Filed.
<b>Actor</b>	System (Dashboard)
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user clicks on the “IP Filed” option.</p> <p>3. The system generates the scraping process to retrieve Intellectual Property Filed data from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Data will include Intellectual Property Filed.</li> </ul> <p>5. System then stores the pre-processed Intellectual property data in the database.</p> <p>6. System displays the Intellectual Property data after filtering the data and retrieving data from the database. If there's an error while retrieving data, EF1 will be performed.</p> <p>7. System displays the data in the form of Bar/Pie chart.</p> <p>8. User can view the IP filed data details by hovering over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 2 will be performed.</p>
<b>Alternative Flow(s)</b>	-

<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <ol style="list-style-type: none"> <li>1.1. System displays an error message.</li> <li>1.2. User is prompted to retry.</li> </ol> <p><b>2. Wrong data Displayed:</b></p> <ol style="list-style-type: none"> <li>2.1. User will contact administrator.</li> <li>2.2. Data preprocessing will be done again.</li> </ol>
<b>Post-condition</b>	User can view and track the Intellectual Property Filed data.
<b>Related Requirement</b>	-

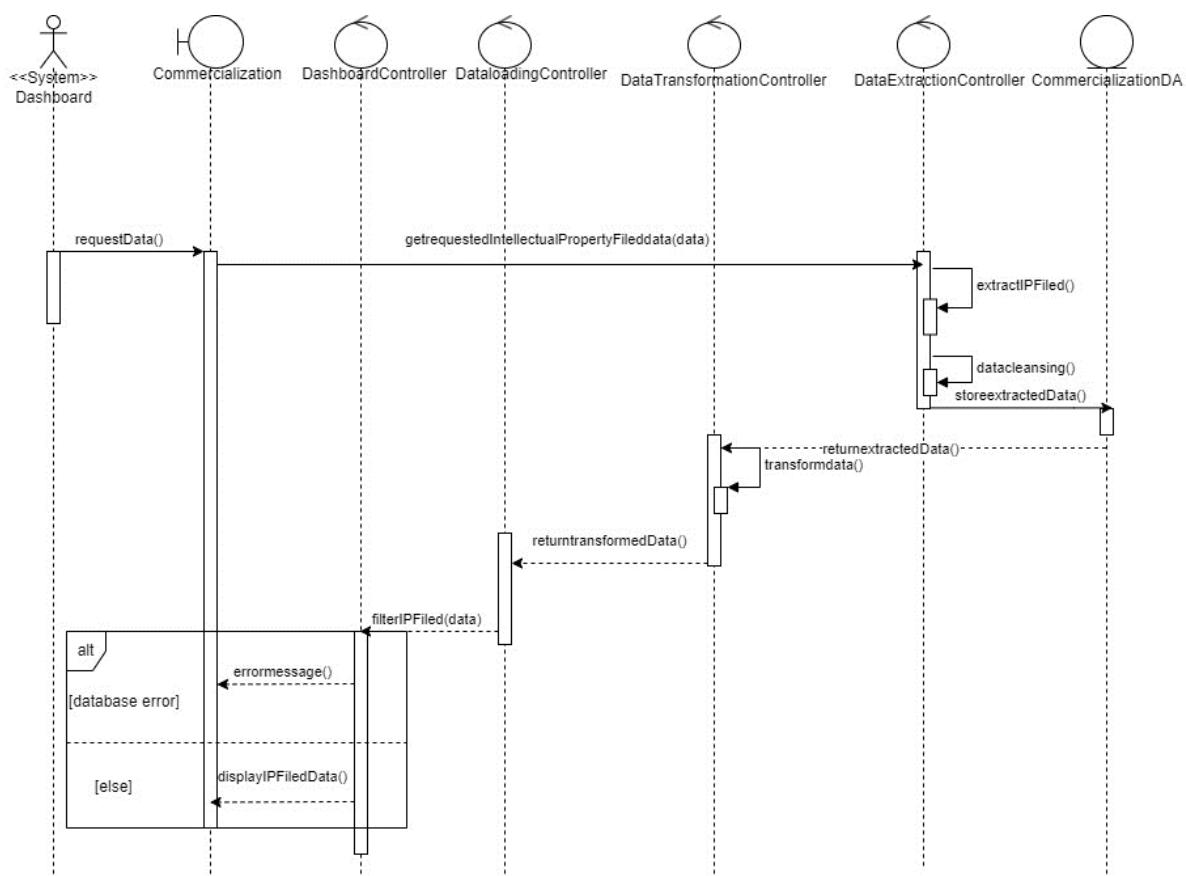
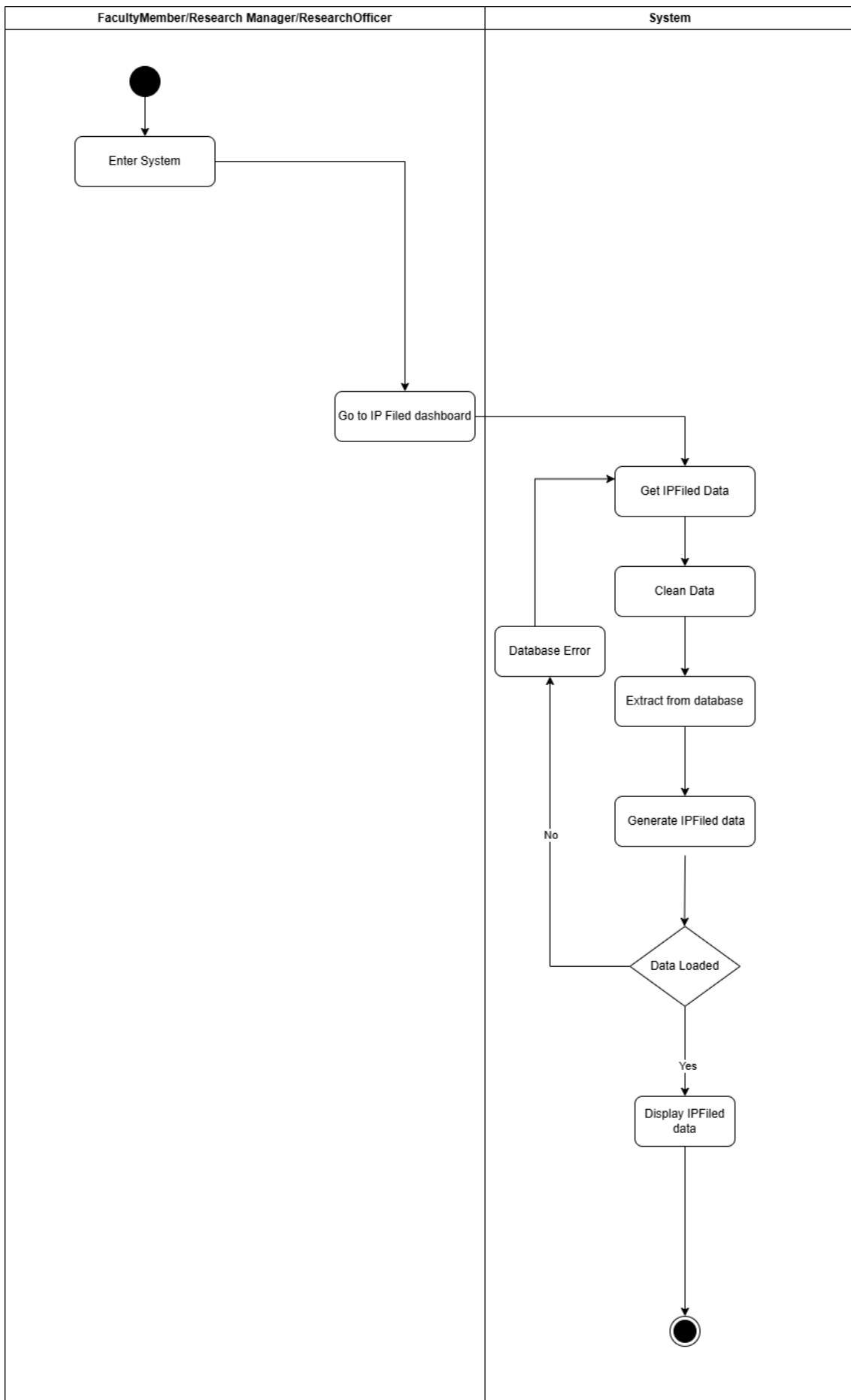
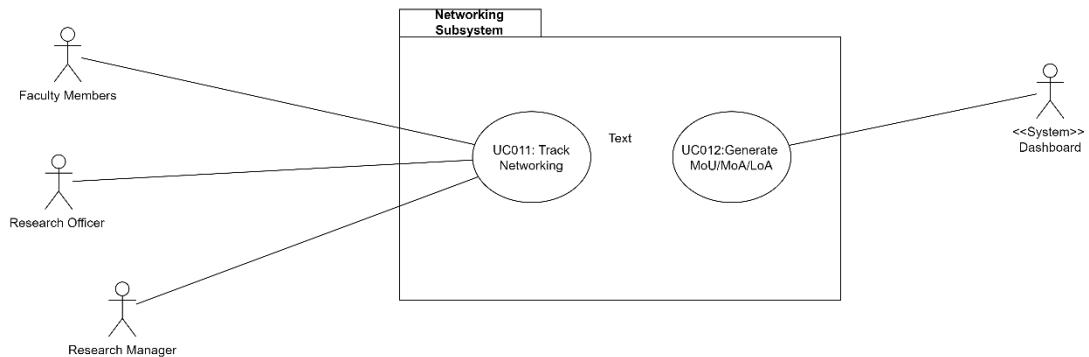


Figure 3.24: Sequence Diagram for UC010



**Figure 3.25: Sequence Diagram for UC010**

### 3.2.4 Module Networking Subsystem



**Figure 3.26: Module 004 Networking Subsystem**

Functional Requirements are given below:

- FR001: Track Networking: This will allow the Research staff to track the networking data of the faculty.
- FR002: Generate the agreement details: This will allow the system dashboard to generate the data of agreements.

#### 3.2.4.1 UC011: Use Case Track Networking

<b>Use case ID</b>	UC011
<b>Use case name</b>	Track Networking
<b>Description</b>	This use case describes the process through which the users can view and track the Networking Data.
<b>Actor</b>	Faculty Members, Research officer and Research Manager
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	1. User enters the system.

	<p>2. From the main dashboard page, user clicks on the “Networking” option. If no data is available, AF 1 is performed.</p> <p>3. The system generates the collection process to retrieve Networking data from external sources.</p> <p>4. The collected data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors</li> <li>• Include MoU/MoA/LoA.</li> </ul> <p>4.2. Data Transformation</p> <ul style="list-style-type: none"> <li>• Convert data stored in excel in different format.</li> </ul> <p>5. System then stores the pre-processed Networking data in the database.</p> <p>6. Networking metrics is updated based on the new Networking data. If right data is not displayed, EF3 is performed.</p> <p>7. User can view the updated Networking in the form of graphs and charts allowing them to visualize the growth and development. If there is a connection error, EF 1 is performed. If there is an error in retrieving data from database, EF2 is performed.</p> <p>8. User can view the Networking details by hovering over the respective data points in the graphs or charts.</p>
<b>Alternative Flow(s)</b>	<p><b>1. Unavailable Data:</b></p> <ol style="list-style-type: none"> <li>1.1. The system displays a message warning about the unavailability of the data.</li> <li>1.2. The user is prompted to Continue from NF3</li> </ol>

<b>Exception Flow(s)</b>	<p><b>1. Connection error with the database:</b></p> <p>1.1. The system displays an error message. 1.2. User is prompted to refresh and start from NF8.</p> <p><b>2. Error Retrieving data from Database:</b></p> <p>2.1. System displays an error message. 2.2. User is prompted to retry.</p> <p><b>3. Wrong data Displayed:</b></p> <p>3.1. User will contact administrator. 3.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view and track the Networking data.
<b>Related Requirement</b>	-

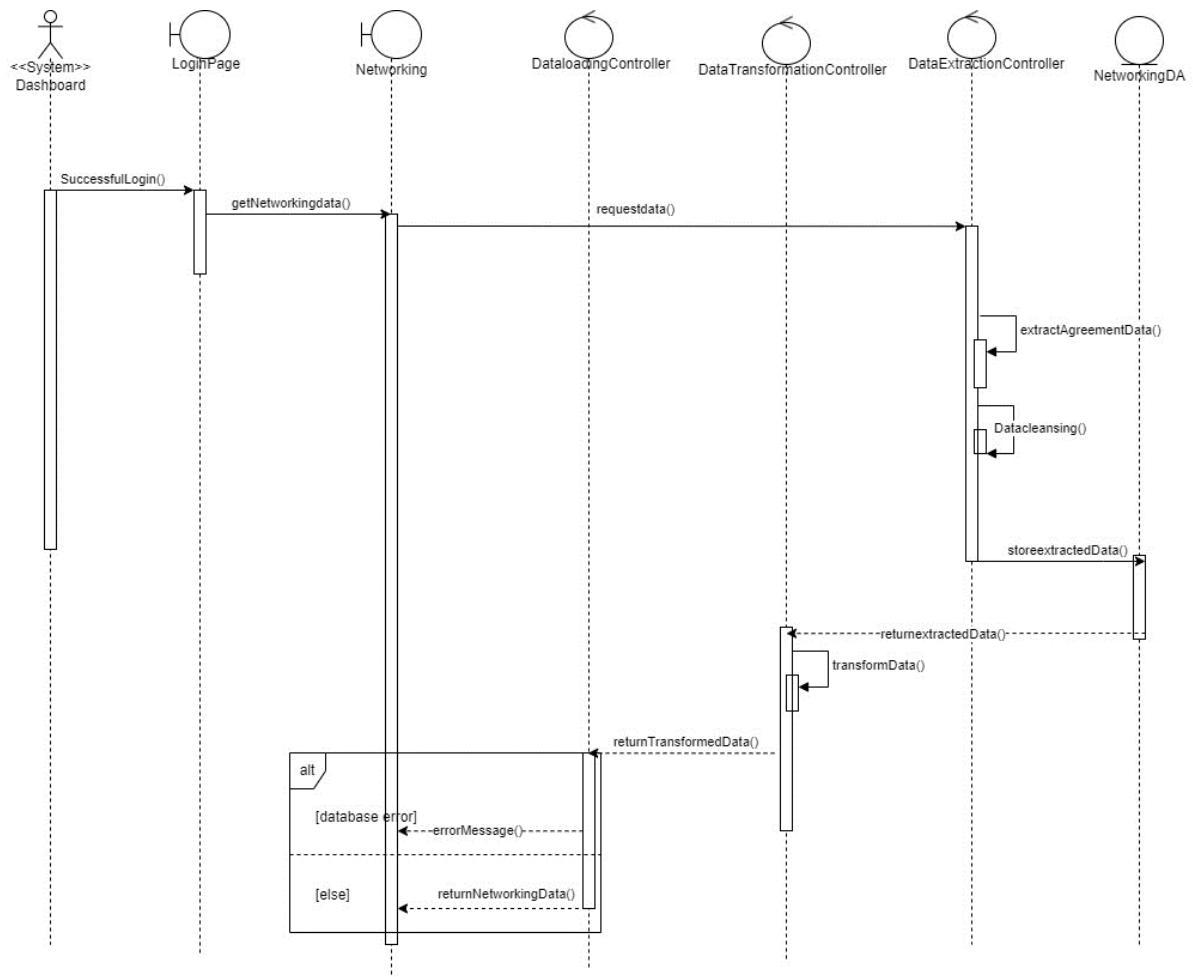
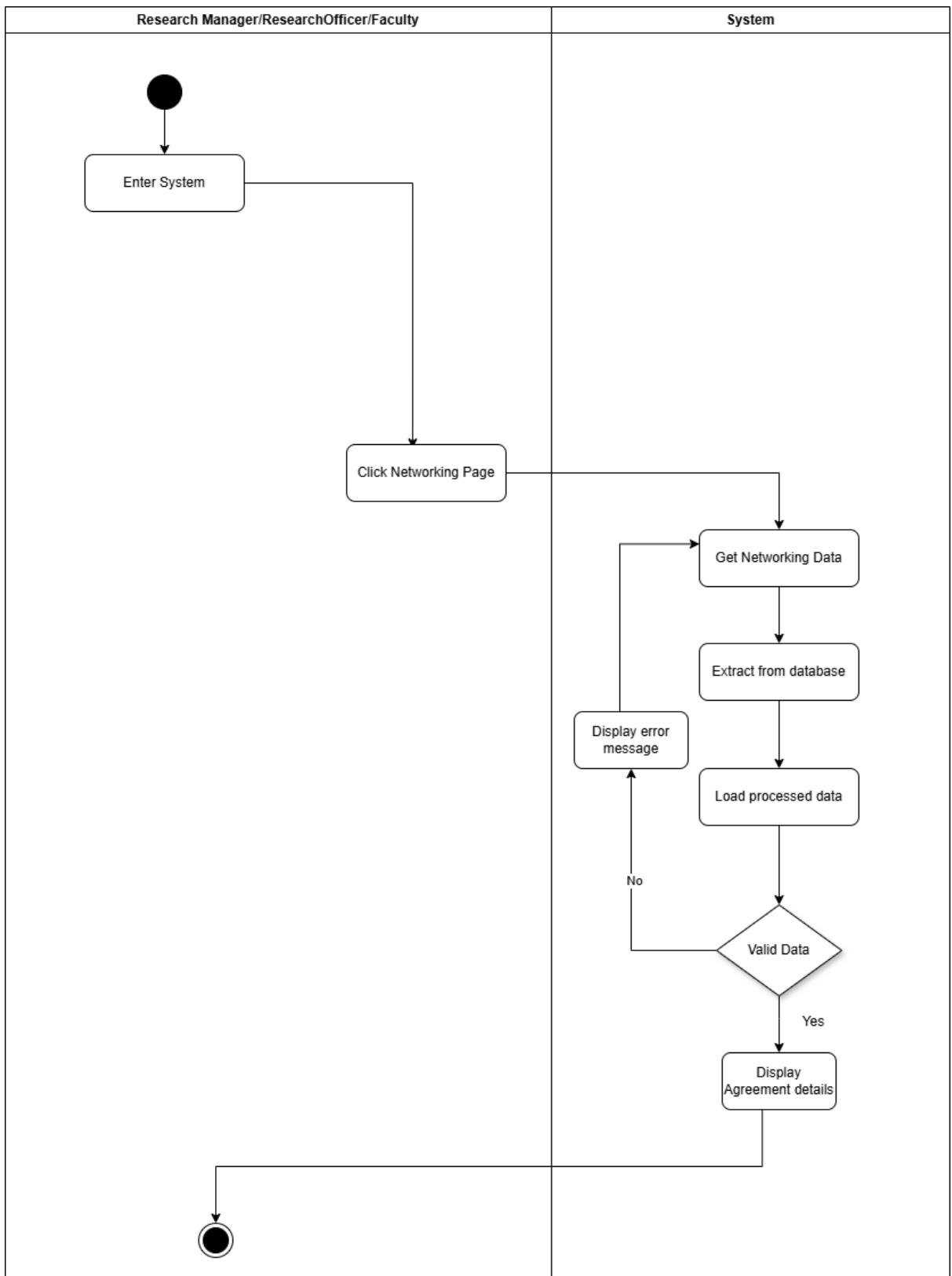


Figure 3.27: Sequence Diagram for UC011



**Figure 3.28: Activity Diagram for UC011**

#### 3.2.4.2 UC012: Use Case Generate MoU/LoA/MoA

<b>Use case ID</b>	UC012
<b>Use case name</b>	Generate Agreement Data
<b>Description</b>	This use case describes the process through which system generates Agreement Data.
<b>Actor</b>	Research Officer, Research Manager
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. User is taken to the home page after logging in successfully.</p> <p>3. From the main dashboard page, user clicks on the “Track Networking” option.</p> <p>4. The system generates the process to retrieve Agreement data from database.</p> <p>5. The collected data will undergo data preprocessing before being displayed in the dashboard.</p> <p>5.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Data will include MoU/MoA/LoA.</li> </ul> <p>6. System then stores the pre-processed Agreement data in the database.</p> <p>7. System displays the Agreement data after filtering the data and retrieving data from the database. If there's an error while retrieving data, EF1 will be performed.</p> <p>8. System displays the data in the form of Bar/Pie chart.</p> <p>9. User can view the Agreement Data details by hovering over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 2 will be performed.</p>

<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <p>1.1. System displays an error message. 1.2. User is prompted to retry.</p> <p><b>2. Wrong data Displayed:</b></p> <p>2.1. User will contact administrator. 2.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view and track the Intellectual Property Filed data.
<b>Related Requirement</b>	-

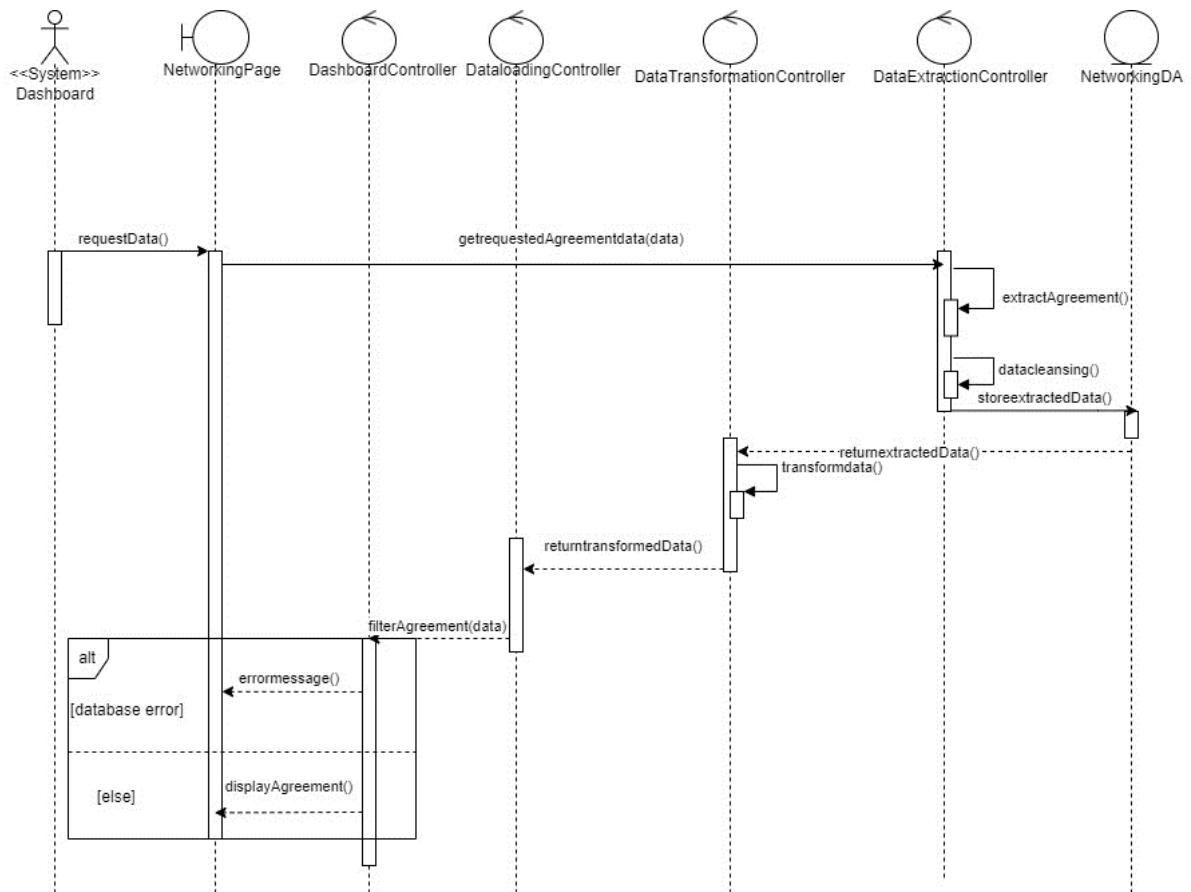
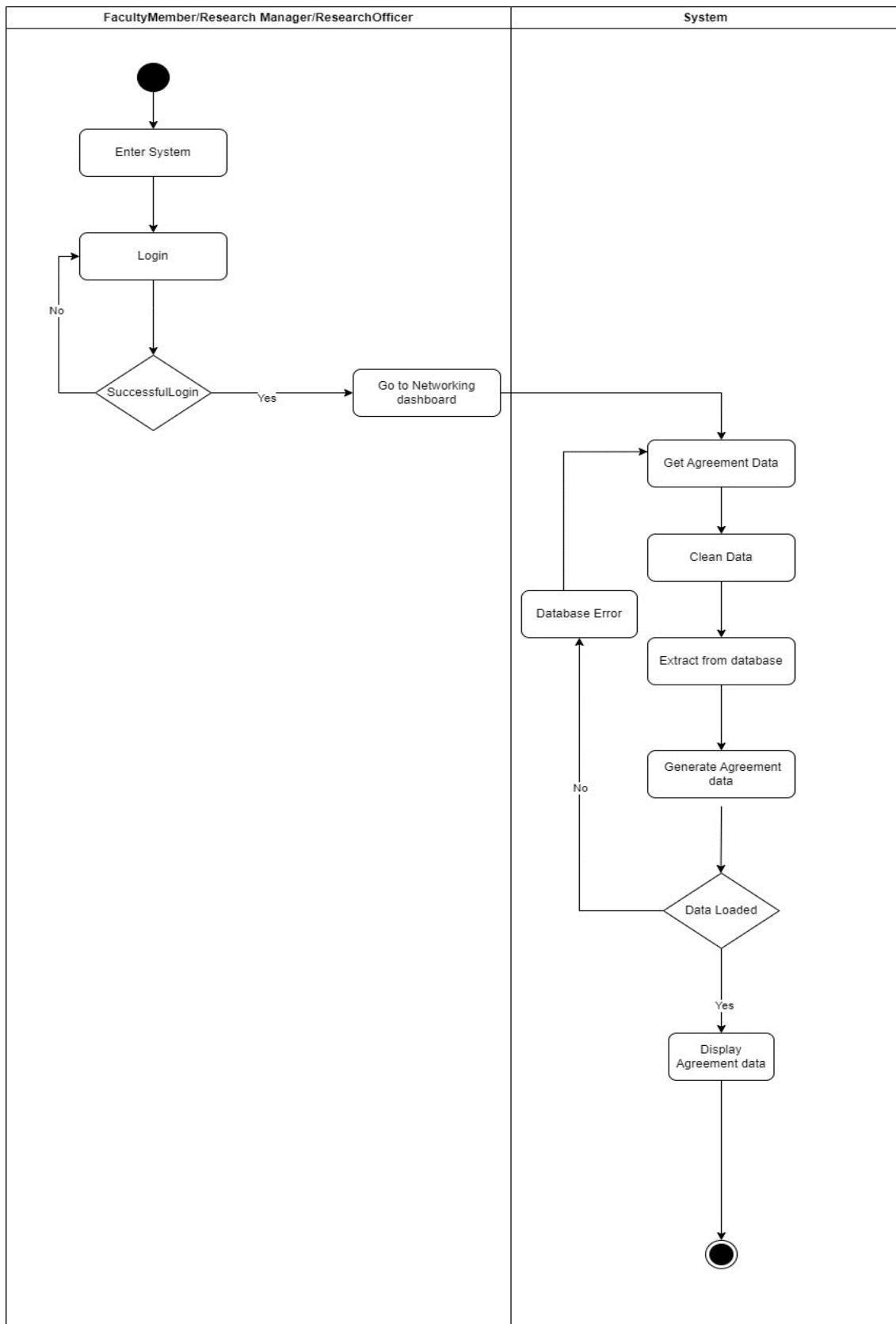
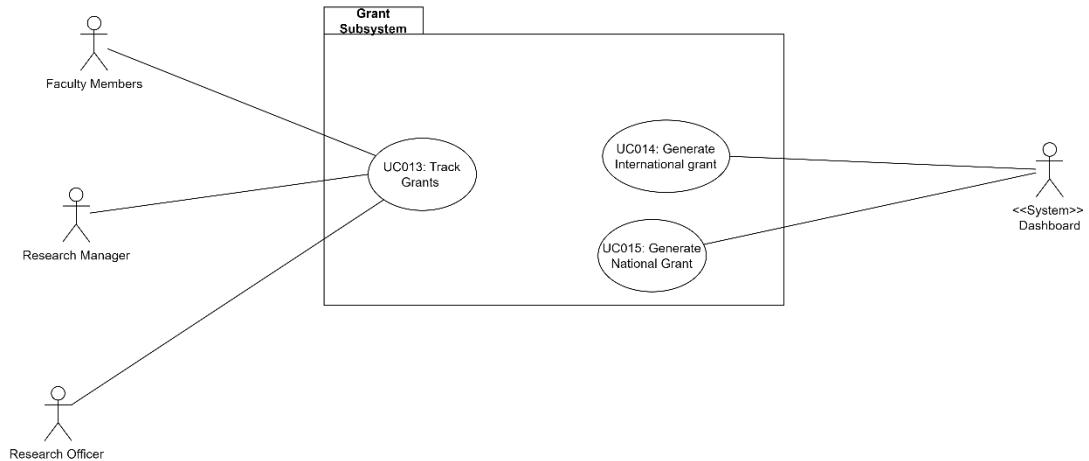


Figure 3.29: Sequence Diagram for UC012



**Figure 3.30: Activity Diagram for UC012**

### 3.2.5 Module Grant Subsystem



**Figure 3.31: Module 005**

Functional Requirements are given below:

- FR001: Track Grant: This will allow the Research staff to track the Grant of the faculty.
- FR002: Generate International Grant: This will allow the System to generate International Grant.
- FR003: Generate National Grant: This will allow the System to generate National Grant.

#### 3.2.5.1 Use Case Track Grant

<b>Use case ID</b>	UC013
<b>Use case name</b>	Track Grant
<b>Description</b>	This use case describes the process through which the users can view and track the Grant Data.
<b>Actor</b>	Faculty Members, Research officer and Research Manager
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<ol style="list-style-type: none"> <li>1. User enters the system.</li> <li>2. From the main dashboard page, user clicks on the “Grant” option. If no data is available, AF 1 is performed.</li> </ol>

	<p>3. The system generates the collection of data through scraping to retrieve Grant data from database.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors</li> <li>• Include International and national grant.</li> </ul> <p>4.2. Data Transformation</p> <ul style="list-style-type: none"> <li>• Convert data stored in excel in different format.</li> </ul> <p>5. System then stores the pre-processed Grant data in the database.</p> <p>6. Grant data metrics is updated based on the new Grant data. If right data is not displayed, EF3 is performed.</p> <p>7. User can view the updated Grant in the form of graphs and charts allowing them to visualize the growth and development. If there's a connection error, EF 1 is performed. If there's an error in retrieving data from database, EF2 is performed.</p> <p>8. User can view the Grant details by hovering over the respective data points in the graphs or charts.</p>
<b>Alternative Flow(s)</b>	<p><b>1. Unavailable Data:</b></p> <ol style="list-style-type: none"> <li>1.1. The system displays a message warning about the unavailability of the data.</li> <li>1.2. The user is prompted to Continue from NF3</li> </ol>

<b>Exception Flow(s)</b>	<p><b>1. Connection error with the database:</b></p> <p>1.1. The system displays an error message. 1.2. User is prompted to refresh and start from NF8.</p> <p><b>2. Error Retrieving data from Database:</b></p> <p>2.1. System displays an error message. 2.2. User is prompted to retry.</p> <p><b>3. Wrong data Displayed:</b></p> <p>3.1. User will contact administrator. 3.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view and track the Networking data.
<b>Related Requirement</b>	-

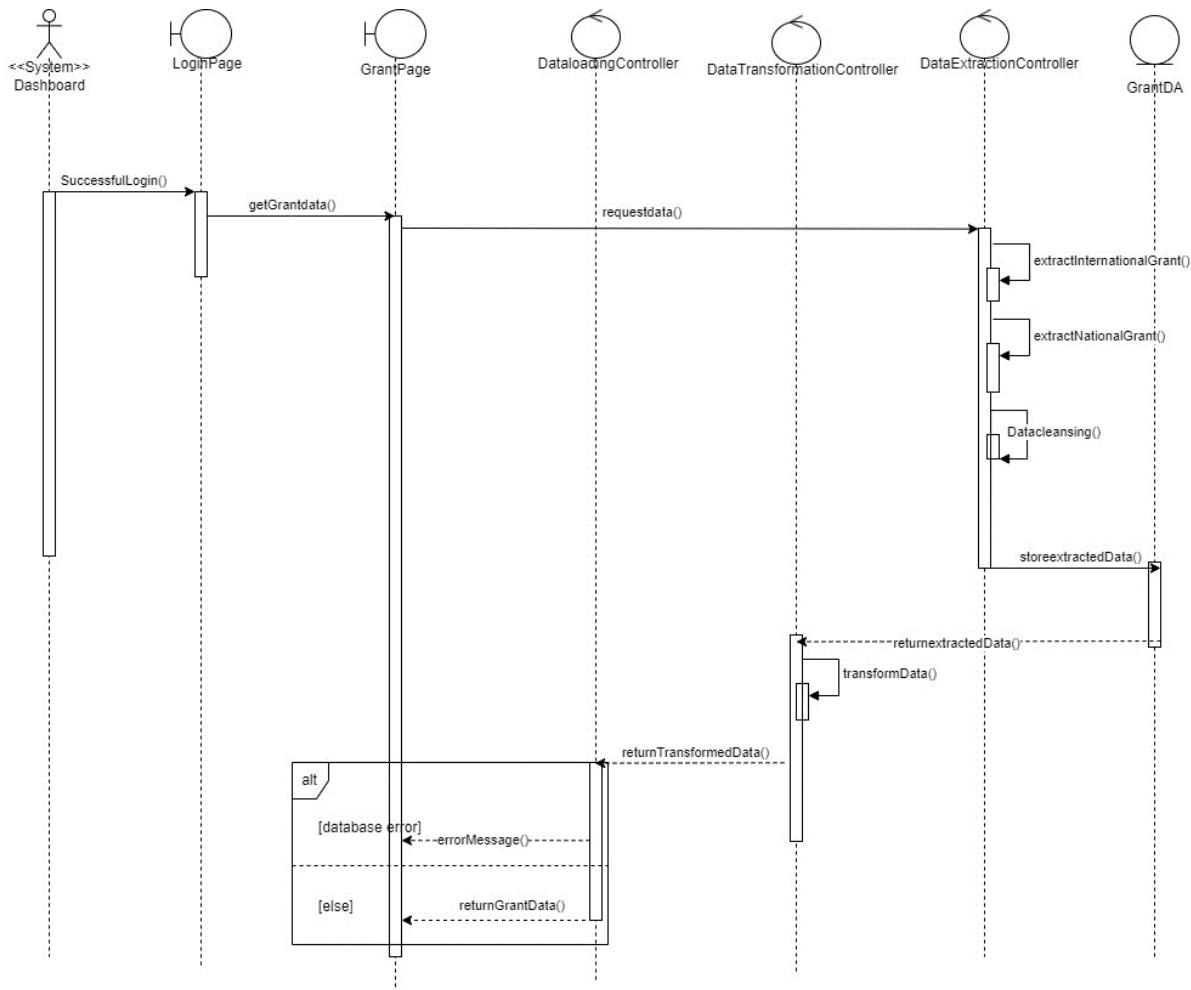
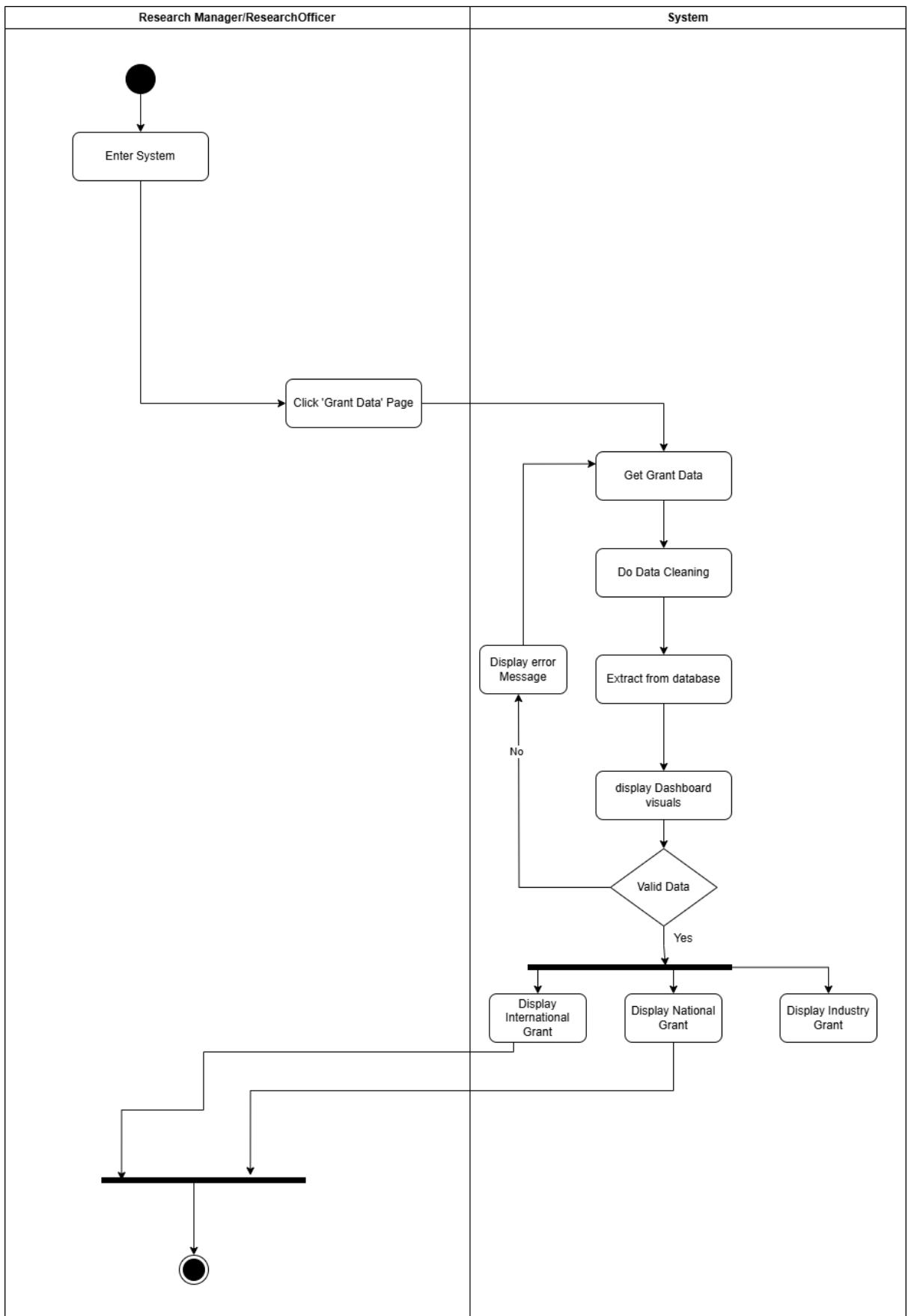


Figure 3.31: Sequence Diagram for UC013



**Figure 3.32: Activity Diagram for UC013**

### 3.2.5.2 UC014: Use Case Generate International Grant Data

<b>Use case ID</b>	UC014
<b>Use case name</b>	Generate International Grant Data
<b>Description</b>	This use case describes the process through which system generates International Grant Data.
<b>Actor</b>	System (Dashboard)
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user clicks on the “Track Grant” option.</p> <p>3. The system generates the data collection through scraping to retrieve International Grant data from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Data is related to International Grant</li> </ul> <p>5. System then stores the pre-processed international data in the database.</p> <p>6. System displays the international data after filtering the data and retrieving data from the database. If there's an error while retrieving data, EF1 will be performed.</p>

	<p>7. System displays the data in the form of Bar/Pie chart.</p> <p>8. User can view the international Data details by hovering over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 2 will be performed.</p>
<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <p>1.1. System displays an error message. 1.2. User is prompted to retry.</p> <p><b>2. Wrong data Displayed:</b></p> <p>2.1. User will contact administrator. 2.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view details of the International Grants data.
<b>Related Requirement</b>	-

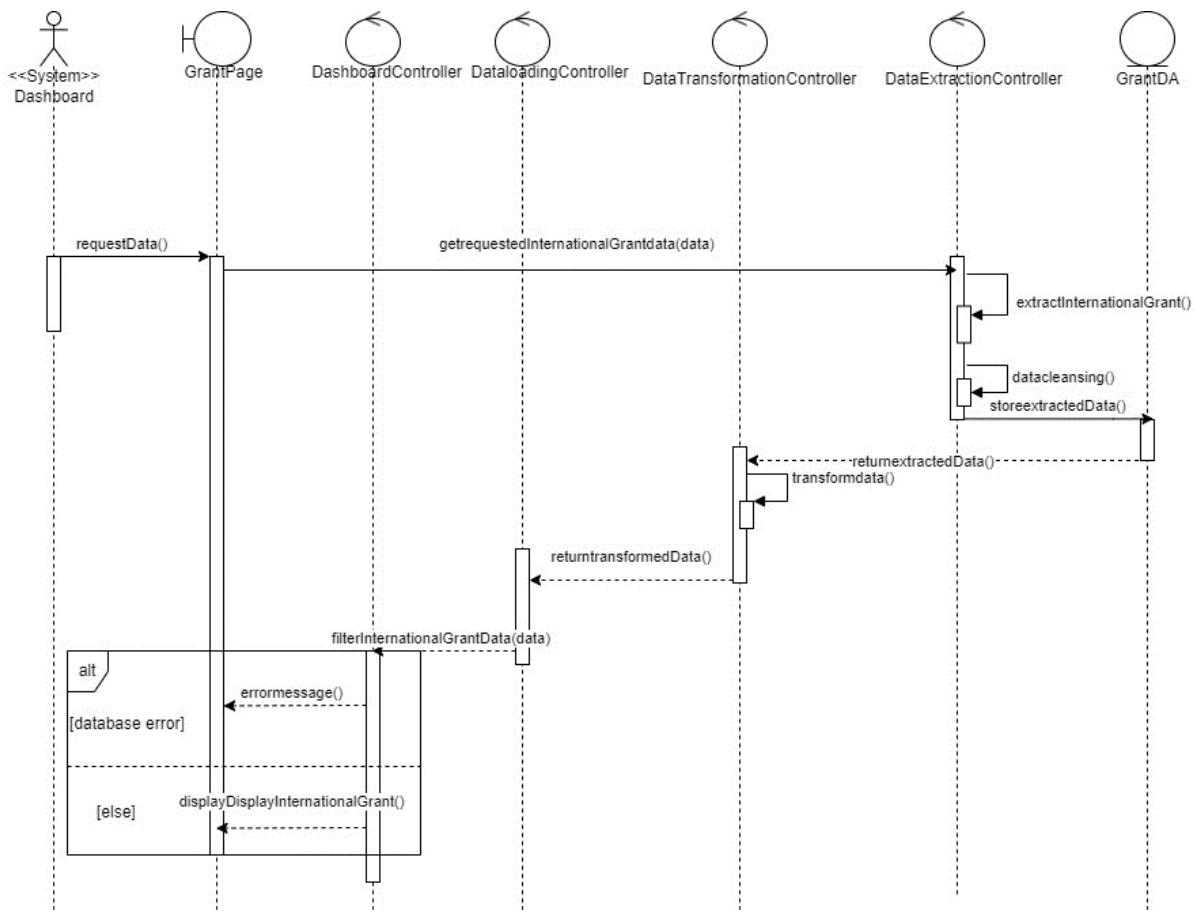
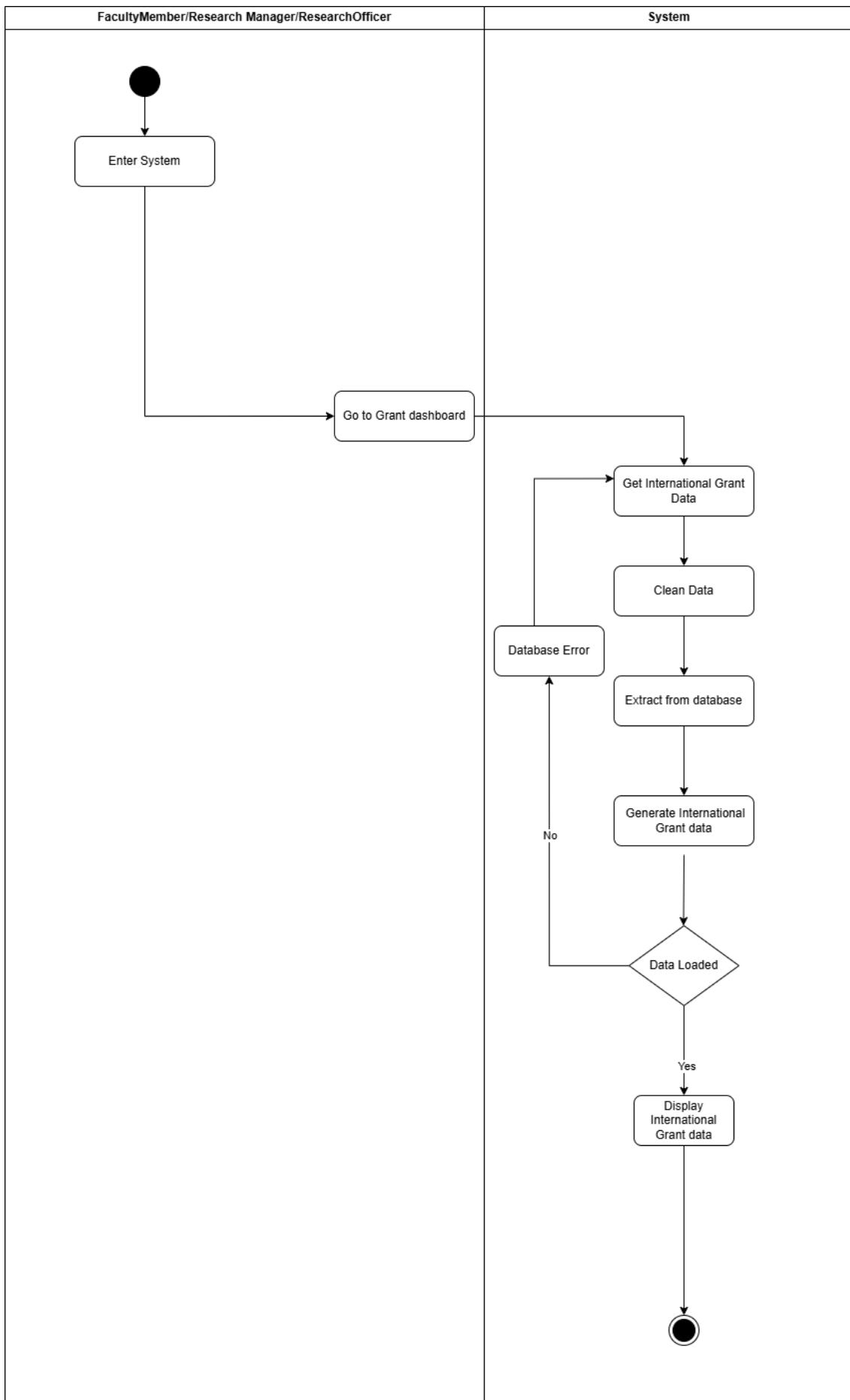


Figure 3.33: Sequence Diagram for UC014



**Figure 3.34: Activity Diagram for UC014**

### 3.2.5.3 UC015: Use Case Generate National Grant Data

<b>Use case ID</b>	UC015
<b>Use case name</b>	Generate National Grant Data
<b>Description</b>	This use case describes the process through which system generates National Grant Data.
<b>Actor</b>	System (Dashboard)
<b>Pre-condition</b>	1. There is an active network connection to the platform.
<b>Normal Flow(s)</b>	<p>1. User enters the system.</p> <p>2. From the main dashboard page, user clicks on the “Track Grant” option.</p> <p>3. The system generates the data that was collected through scraping to retrieve National Grant data from external sources.</p> <p>4. The scraped data will undergo data preprocessing before being displayed in the dashboard.</p> <p>4.1. Data Cleaning:</p> <ul style="list-style-type: none"> <li>• Data cleaning by removing duplicate records or entries.</li> <li>• Excluding incomplete data</li> <li>• Handling missing values by imputing them</li> <li>• Check data errors.</li> <li>• Data is related to National Grant</li> </ul> <p>5. System then stores the pre-processed National data in the database.</p> <p>6. System displays the National data after filtering the data and retrieving data from the database. If there's an error while retrieving data, EF1 will be performed.</p>

	<p>7. System displays the data in the form of Bar/Pie chart.</p> <p>8. User can view the National Data details by hovering over the respective data points in the graphs or charts. If the data displayed isn't correct, EF 2 will be performed.</p>
<b>Alternative Flow(s)</b>	-
<b>Exception Flow(s)</b>	<p><b>1. Error Retrieving data from Database:</b></p> <p>1.1. System displays an error message. 1.2. User is prompted to retry.</p> <p><b>2. Wrong data Displayed:</b></p> <p>2.1. User will contact administrator. 2.2. Data preprocessing will be done again.</p>
<b>Post-condition</b>	User can view details of the National Grants data.
<b>Related Requirement</b>	-

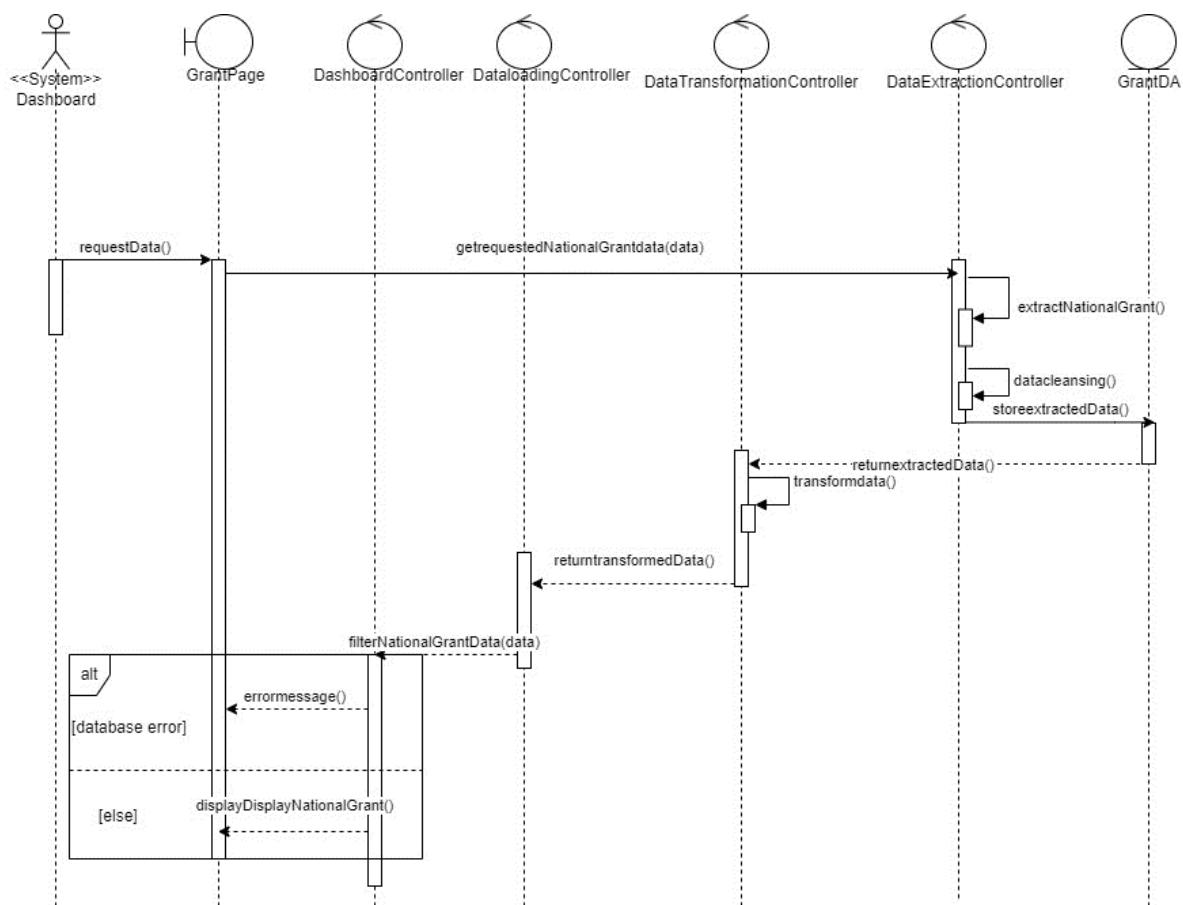
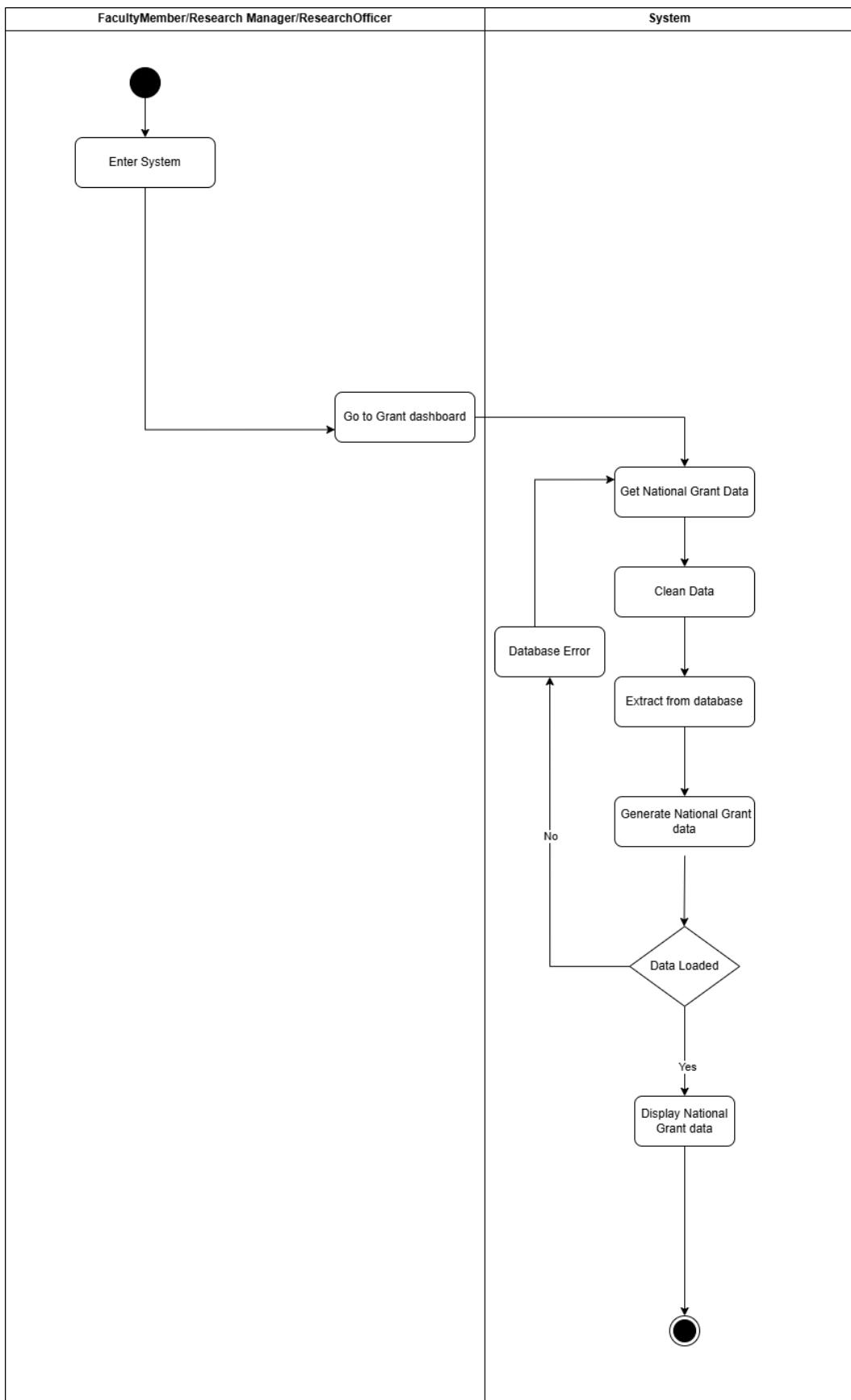


Figure 3.35: Sequence Diagram for UC015



**Figure 3.36: Activity Diagram for UC015**

### **3.3 Performance Requirements**

- 1) Response Time: The response time of the system should be quick to guarantee an efficient user experience. To avoid user frustration, the response time for generating and exhibiting dashboard analytics, including graphs and charts, should be minimal. Response time play a key role in maintaining user satisfaction while using the system.
- 2) Scalability: System must be designed in a way to maintain high scale data. The approach should scale well to accommodate a larger dataset as the number of researchers, and scholarly works grows every day.
- 3) Database Efficiency: The publication data should be stored in a database that allows for fast retrieval of the data. This will help in indexing and displaying the data accurately in the dashboard.
- 4) Safety- All the data that will be used must be kept safe and must be delivered to the server in a way that guarantees integrity.

### **3.4 Design Constraints**

The design must be done in a way to ensure that it is operable on all desktop computers and laptops. It should be accessible through any web browsers with the help of an internet connection.

### **3.5 Software System Attributes**

- 1) Reliability: The system should be dependable, with consistent and accurate operation. It should elegantly manage mistakes and exceptions, recover from failures, and minimise data loss or corruption. The system should also have backup and recovery techniques to prevent data loss and assure system availability.
- 2) Security: The system must prioritise data and user information security. It should put in place proper authentication and authorisation methods to ensure that only authorised people may access the data. In addition, the system should use secure data transmission methods to protect data during transit and comply with applicable data protection and privacy standards.

- 3) Usability: The system should be simple to use and intuitive, allowing faculty and research staff to explore and engage with the dashboard and its features. The user interface should be developed in way that will ensure users can simply obtain needed information, perform actions, and comprehend visualisations without confusion or excessive complexity.
- 4) Portability: The system should be built to be portable, allowing it to work on multiple operating systems or environments. It should make use of technologies and frameworks that are cross-platform compatible, ensuring flexibility in deployment options.

### **3.6 Other Requirements**

No other requirements are required for this project.

## **Appendix B System Design Documentation**



SCSJ3323: Software Design and Architecture

---

## **Software Design Document**

Faculty of Computing Staff Publication Dashboard

Version 1.0

25/06/2023

Faculty of Computing, Software Engineering

Prepared by: Adib Bin Morshed

# Revision Page

## a. Overview

This document is the first draft of the Software Design and Architecture (SDD) for the Faculty of Computing Staff Publication Dashboard.

## b. Target Audience

- Prof Madya Dr. Siti Zaiton Bt. Mohd Hashim, Faculty of Computing, Universiti Teknologi Malaysia (UTM).
- Faculty of computing Staff, UTM.

## c. Project Team Members

- Adib Bin Morshed

## d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
Version 1.0	Adib Bin Morshed	SDD of Faculty of Computing Staff publication Dashboard	25 June,2023

### Note:

This template is an annotated outline for a software design document adapted from the IEEE Recommended Practice for Software Design Descriptions. The IEEE Recommended Practice for Software Design Descriptions have been reduced

in order to simplify this assignment while still retaining the main components and providing a general idea of a project definition report. Please refer to IEEE Std 1016-1998 1 for the full IEEE Recommended Practice for Software Design Descriptions. Examples of models are from Satzinger (2011). Compiled by Shahliza Abdul Halim, PhD and checked by Shahida Sulaiman, PhD on 2 May 2016.

# Table of Contents

## **1 Introduction**

- 1.1 Purpose
- 1.2 Scope
- 1.3 Definitions, Acronyms and Abbreviations
- 1.4 Reference Materials
- 1.5 System Overview

## **2 System Architectural Design**

- 2.1 Architectural Style and Rationale
- 2.2 Component Model
- 2.3 Use Case Diagram

## **3 Detailed Description of Modules**

- 3.1 Complete Package Diagram
- 3.2 Modules Detailed Descriptions

## **4 Data Design**

- 4.1 Data Description
- 4.2 Data Dictionary

## **5 User Interface Design**

- 5.1 Overview of User Interface
- 5.2 Screen Images

## **6 Requirements Matrix**

## **7 Appendices**

Appendices (if any)

# **1. INTRODUCTION**

---

## **1.1 Purpose**

The main purpose of this document is to describe the necessary methods and designs adapted for this system. This document will include the database design, class diagram design and the architectural model required to fulfil the requirements set by the stakeholder. The requirements were gathered from our stakeholders. This SDD is important for the development of the system since it contains all the diagrams required to design the system according to the requirements that have been discussed with the stakeholders. This document is developed for the ease of understanding of the system by the developer and stakeholders.

## **1.2 Scope**

The proposed system is named the research publication dashboard for faculty of computing staff. This system is specifically designed for the stakeholders who are researchers and research staff of the faculty of computing at Universiti Teknologi Malaysia (UTM). This system will help to recognize patterns and performance measures to evaluate the current situation of the research department. Although it will have all the necessary data required to make the evaluation, there will be a few limitations as well. This system will not include some data related to researchers which is a limitation of the system. Nevertheless, it will be used for the performance measure of the publications by the researchers and research staff. This document is needed for the better understanding of the system by the stakeholders and developer.

## **1.3 Definitions, Acronyms and Abbreviation**

<b>Acronyms</b>	<b>Definition</b>
SDD	Software Design Document
MVC	Model-View-Controller
UTM	Universiti Teknologi Malaysia

## **1.4 References**

- 1) Software Design Document: What is it & How to Create it! (Template Included).  
(2022, October 31). Bit Blog. <https://blog.bit.ai/software-design-document/>

## 1.5 Overview

This document is divided into three main sections that detail the specifications for the Faculty of Computing Staff Publication Dashboard. A thorough explanation of the chosen system architecture design will be covered in section 2. After that, the database design of the system is mentioned in section 3. A prototype and user interface design will be provided in section 3.

## **2. SYSTEM ARCHITECTURAL DESIGN**

---

This section will provide an overview of the chosen architecture style and rationale for this system. It will also include the architecture model and use case diagram of the whole system for a better understanding of the chosen architecture design.

### **2.1 Architecture Style and Rationale**

The chosen architectural style for this system is the MVC architecture. The MVC architecture refers to the Model-View-Controller architecture. The MVC architecture brings numerous benefits to the system. The system component is divided into three major parts. The three parts are model, view and controller. This architecture pattern makes it easier to manage code because of its structure.

The description of its three different parts is given below:

#### **Model:**

The business logic of the application is represented by the model layer. It focuses on the data structures and data operations which is one of the crucial factors for this system. Its sustainability is one of the big factors in choosing this architectural style.

- Represents the business logic of the application.
- Data storage and retrieval is handled in this layer.
- If there is any change in data, this layer notifies the observer.

#### **View:**

The presentation layer is represented by the view layer. It displayed the interface and data to the user. It receives data from the model and works as a presentation layer of the collected data in the form of tables, charts and so on.

- The data is displayed by this layer.
- Works as a presentation layer.
- User interface elements are rendered in this layer.

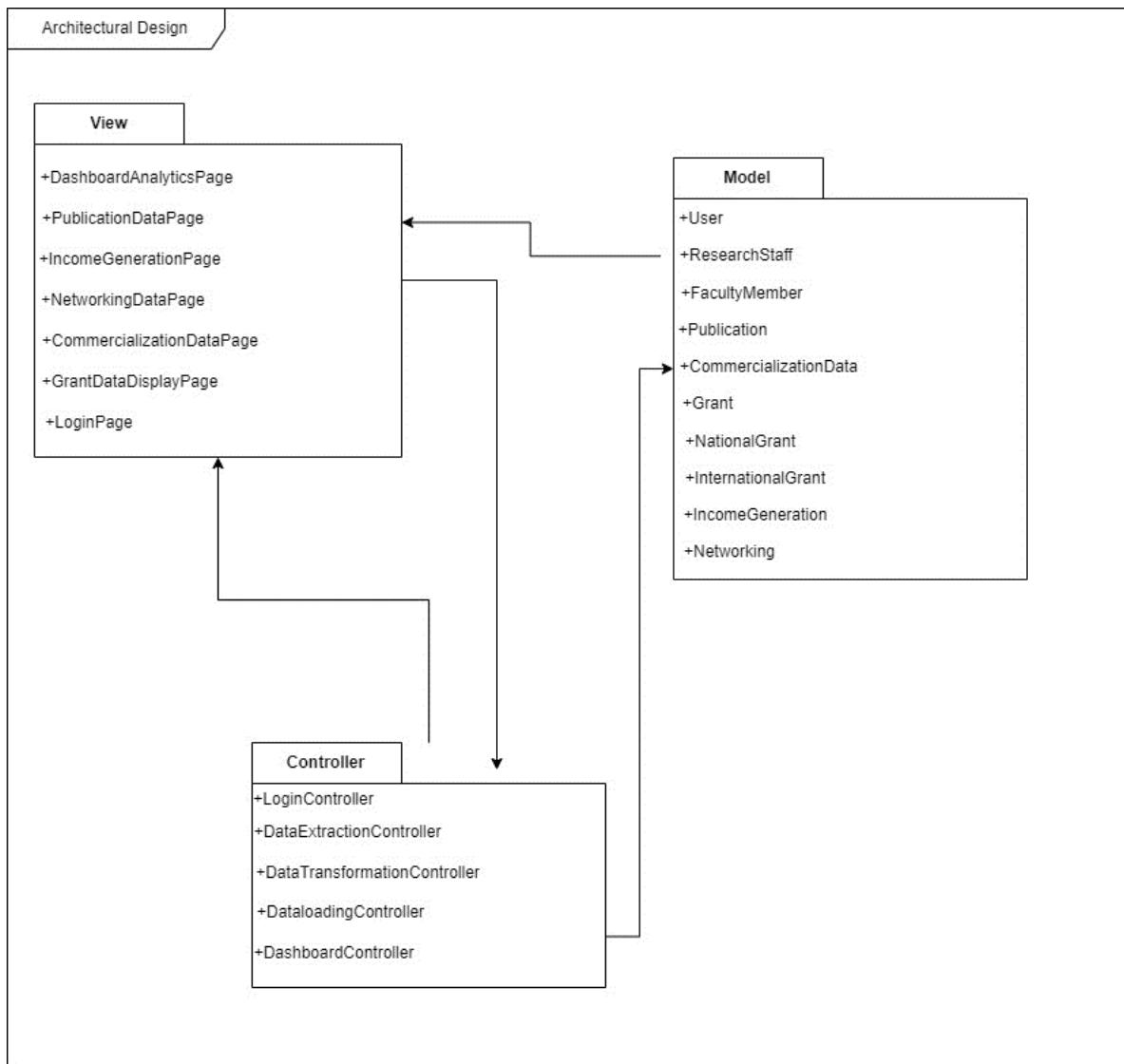
- It passes user input to the controller.

### **Controller:**

The controller acts as a bridge between the model and view layer. It interacts with the model layer to update or retrieve data and interacts with the view layer to reflect the changes made.

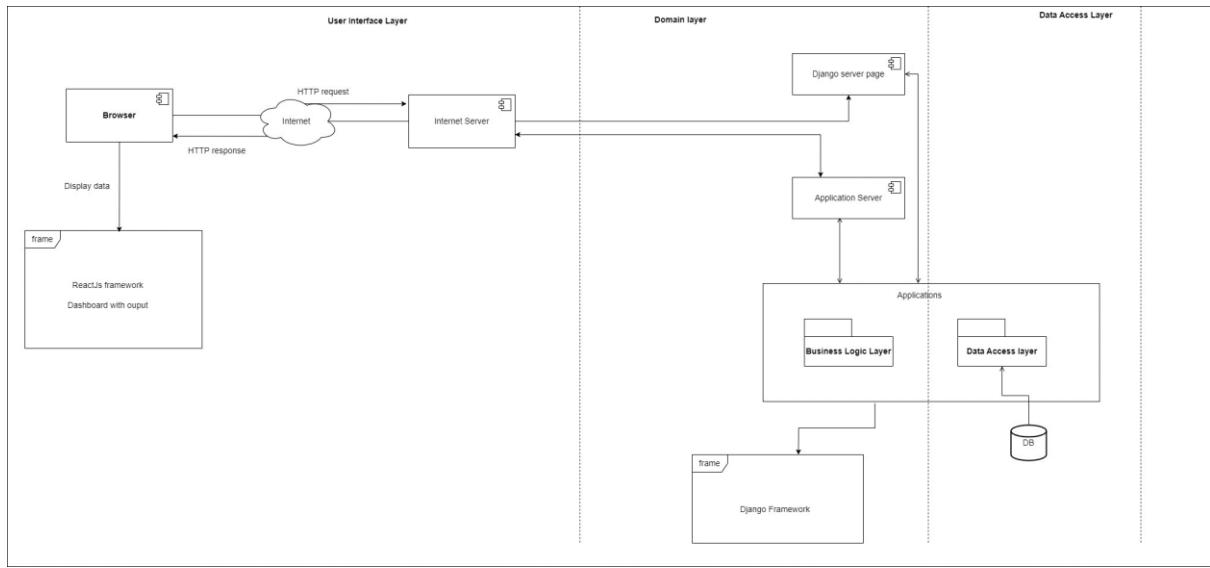
- It receives user input, and it performs actions based on the user input.
- Interacts with the model to collect data.
- Interacts with the view layer to reflect changes made in the model layer.
- It controls the data flow between the model and the view layer.

The reason for choosing this architectural model is due to its durability, testability, reusability, and flexibility.



**Figure 2.1: Architecture Model of Dashboard**

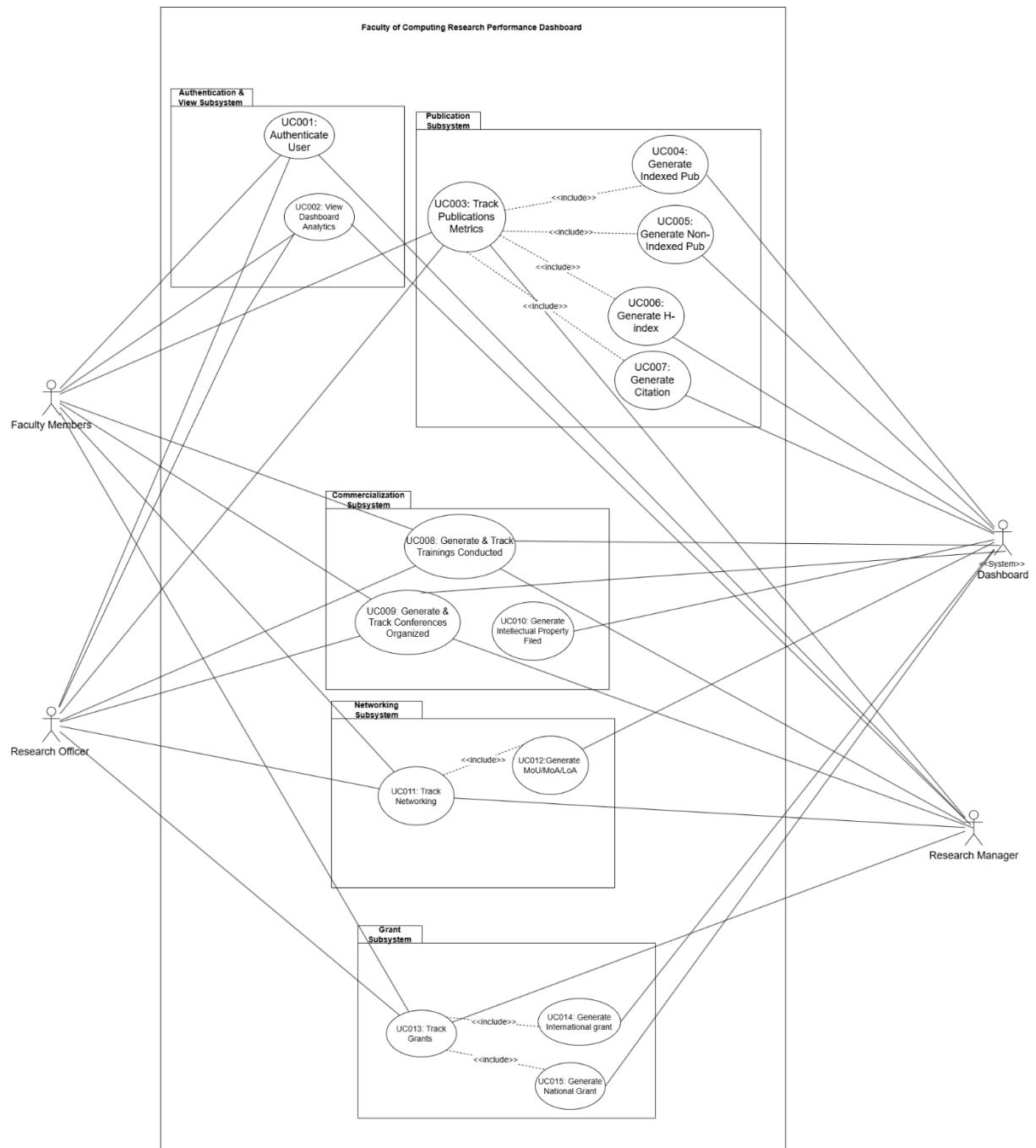
## 2.2 Component Model



**Figure 2.2: Component Model of Dashboard**

The component model of the system is depicted above. It is divided into three layers. The user interface layer, Domain layer and Data access layer. ReactJS framework will be used for the frontend and Nodejs will be used for the backend of the system. Our system will collect data from the data access layer with the help of an api and will display it in the dashboard which is in the user interface layer. This is the overall component structure of the system.

## 2.3 Use Case Diagram

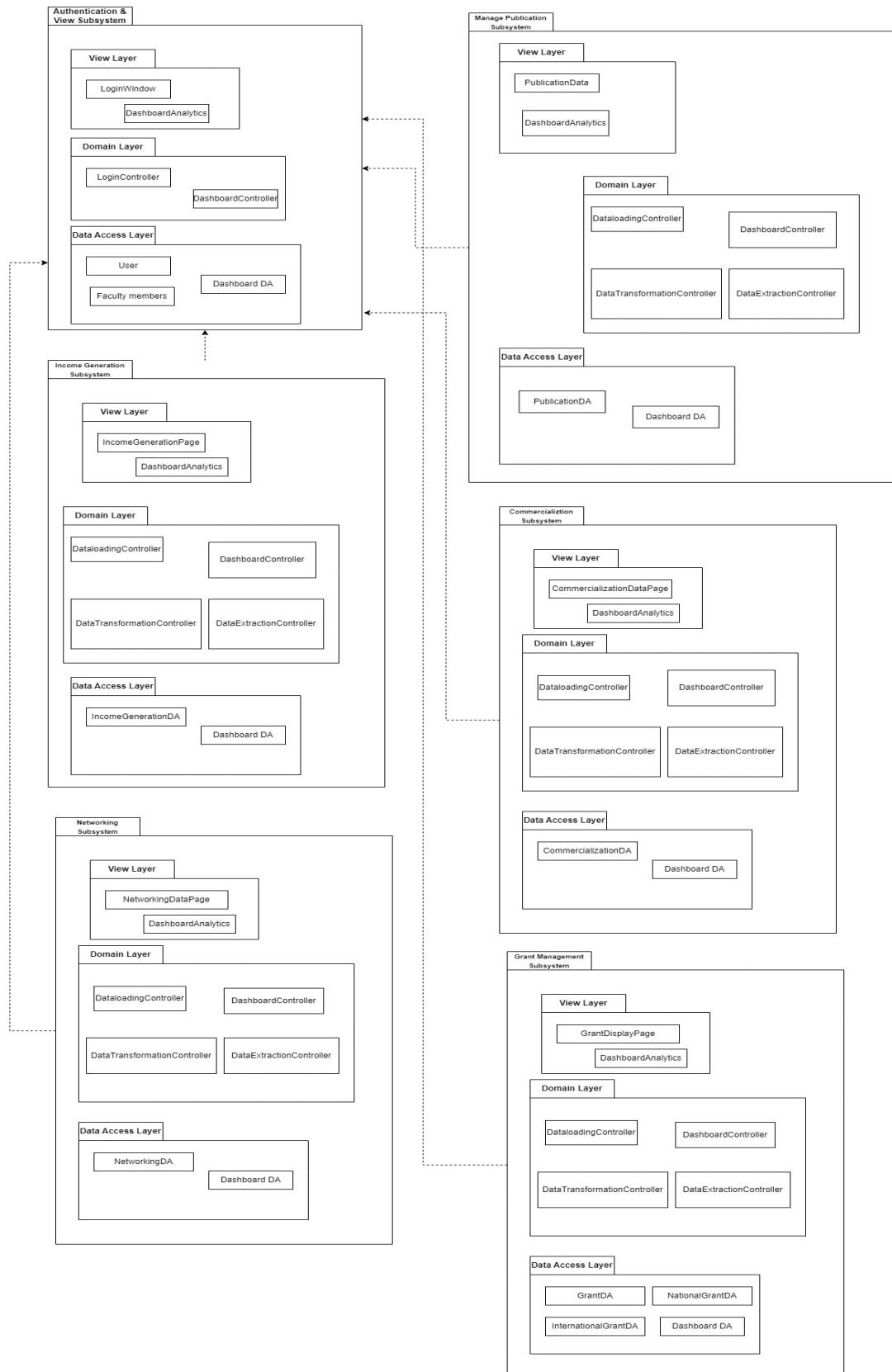


**Figure 2.3: Use Case Diagram of Faculty of Computing Staff Publication Dashboard**

The use case diagram above consists of the key features of the system and depicts a clear picture of what the feature will look like. It also gives an idea of the users of the system.

### 3. DETAILED DESCRIPTION OF COMPONENTS

#### 3.1 Complete Package Diagram

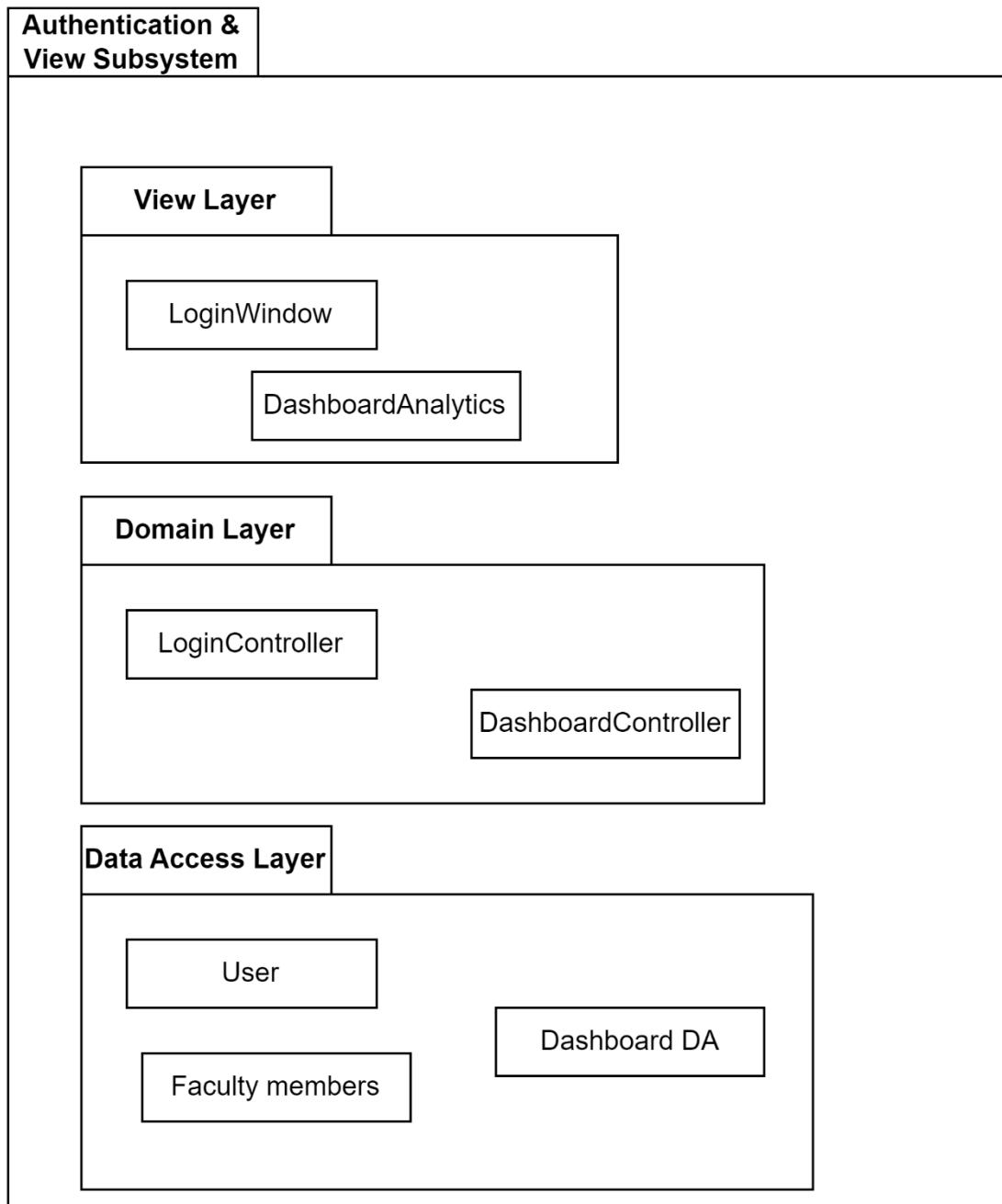


**Figure 3.1: Package Diagram of Staff Publication Dashboard**

### **3.2 Detailed Description**

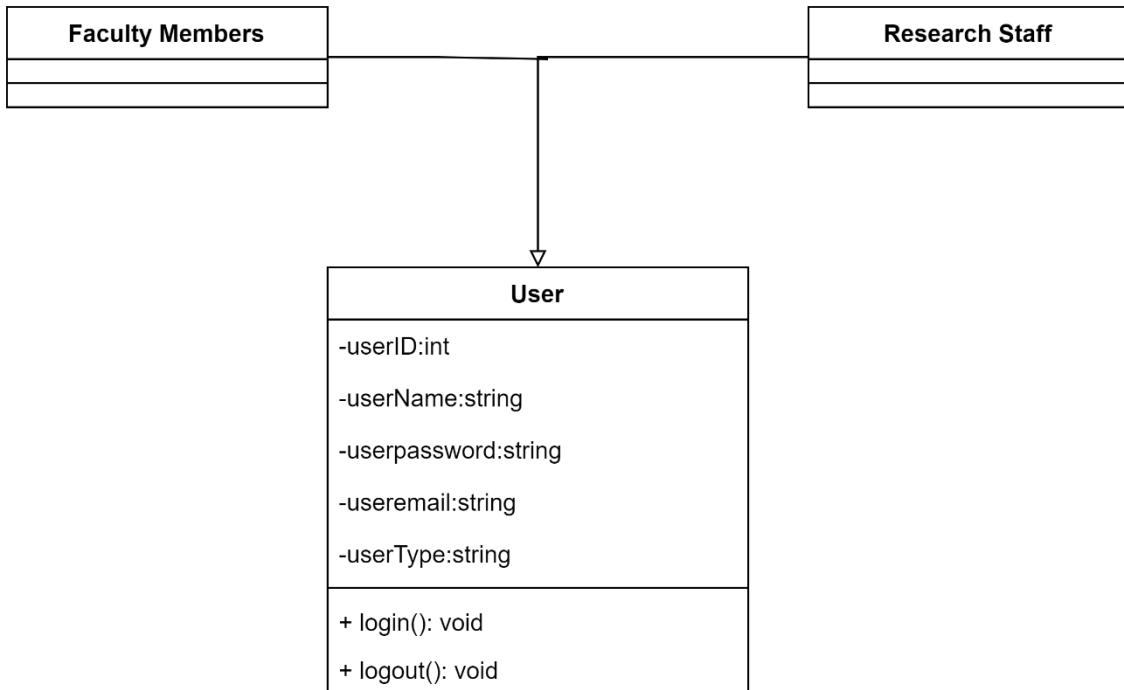
#### **3.2.1 Subsystem Authentication & View**

##### **3.2.1.1 P001: Package Authentication & View**

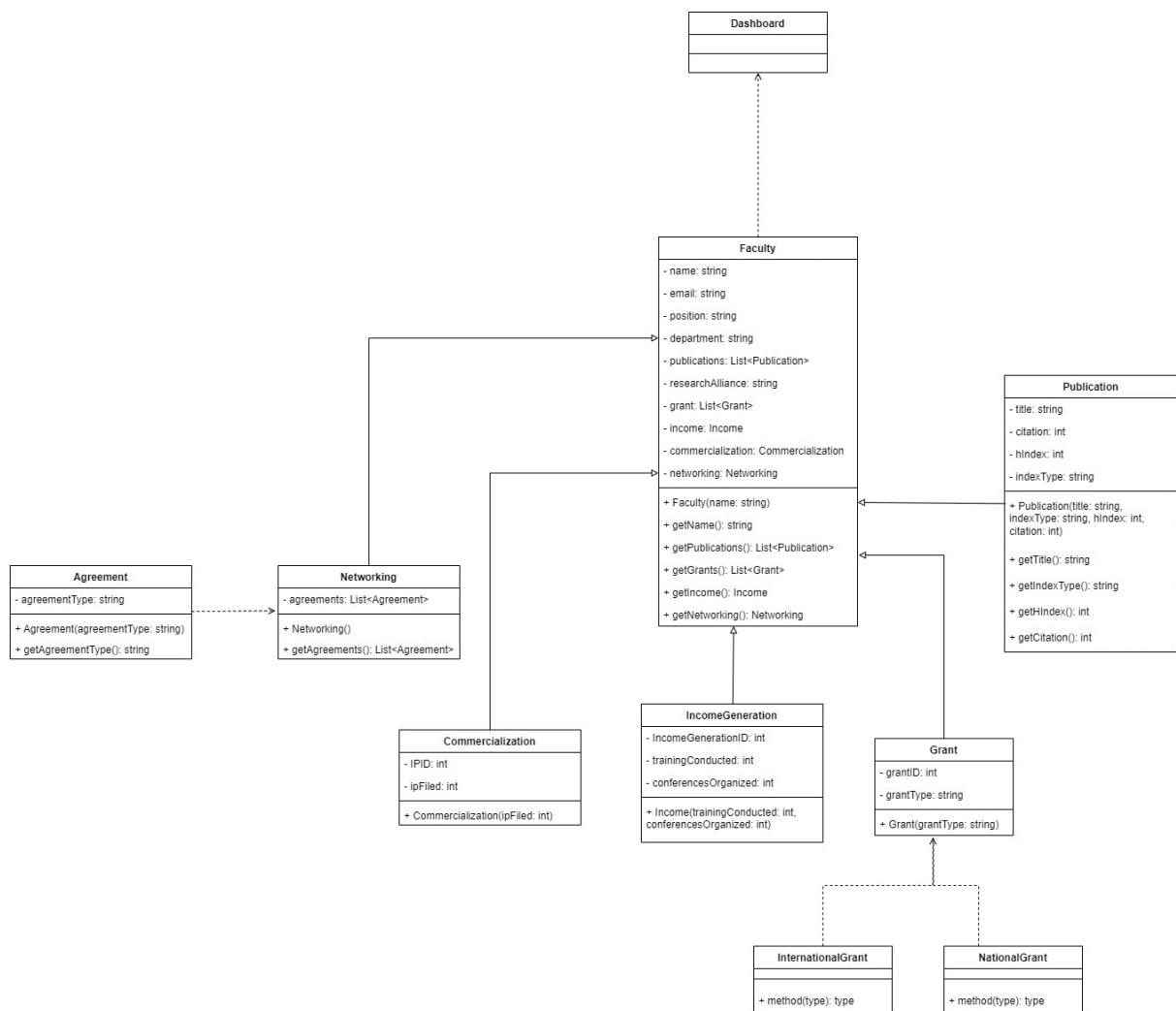


**Figure 3.2: P001 Authentication & View**

##### **3.2.1.2 Class Diagram**



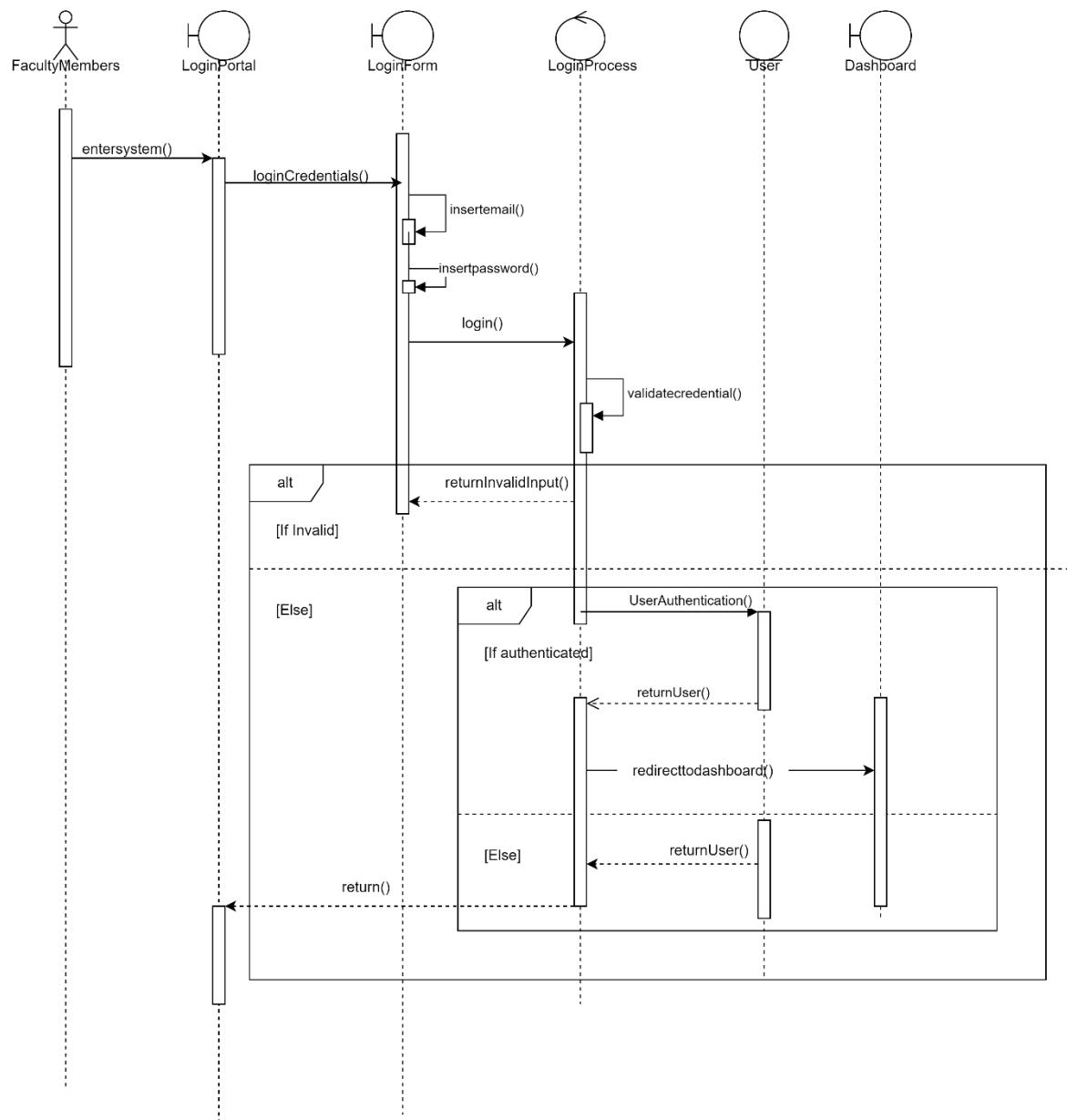
**Figure 3.3: C001 Class Diagram for Authentication**



**Figure 3.4: C002 Class Diagram for View Dashboard Analytics**

### 3.2.1.3 Sequence Diagrams

#### a) SD001: Sequence diagram for Login



**Figure 3.5: SD001 Sequence Diagram of Login**

b) SD002: Sequence diagram for View Dashboard Analytics

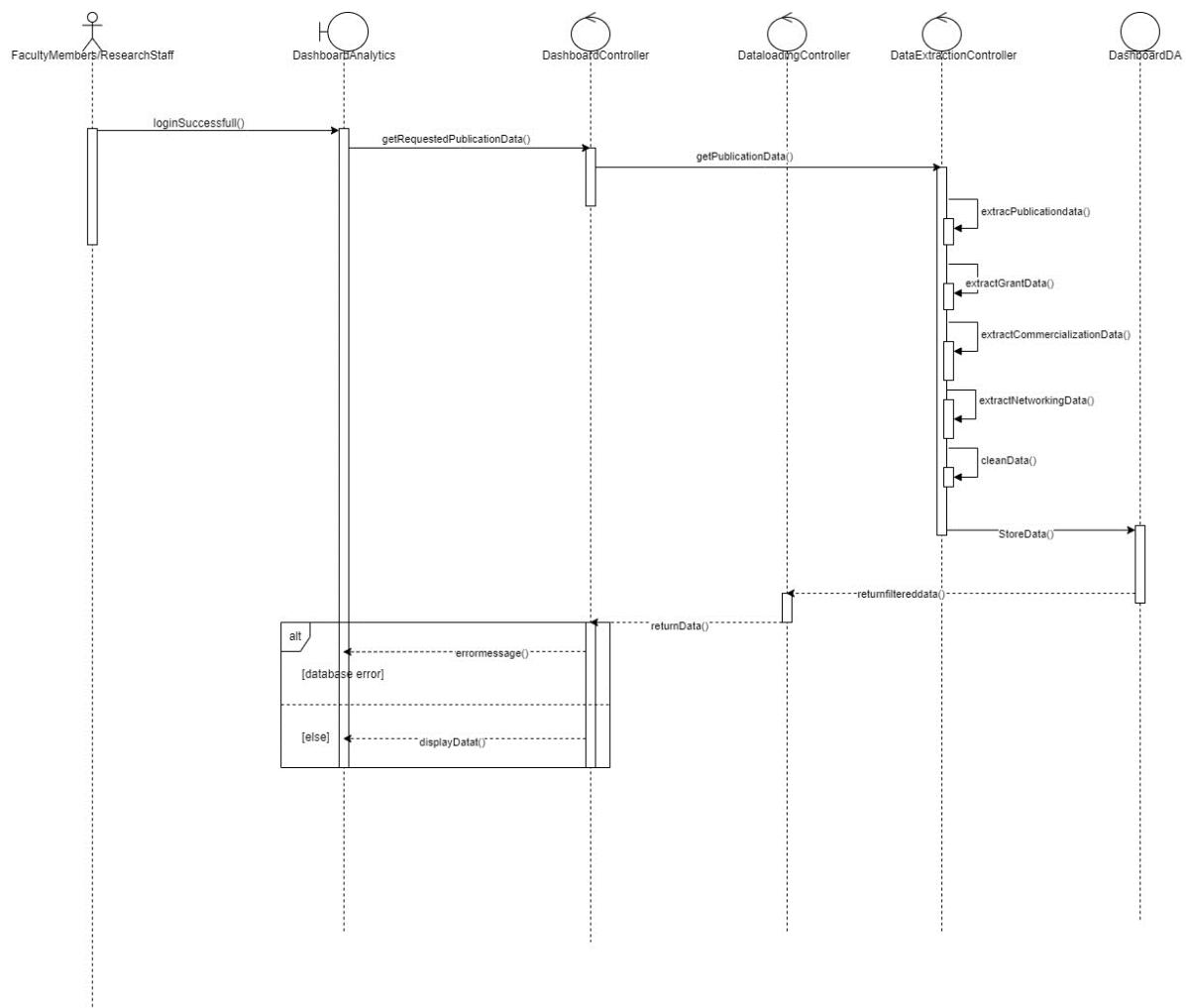
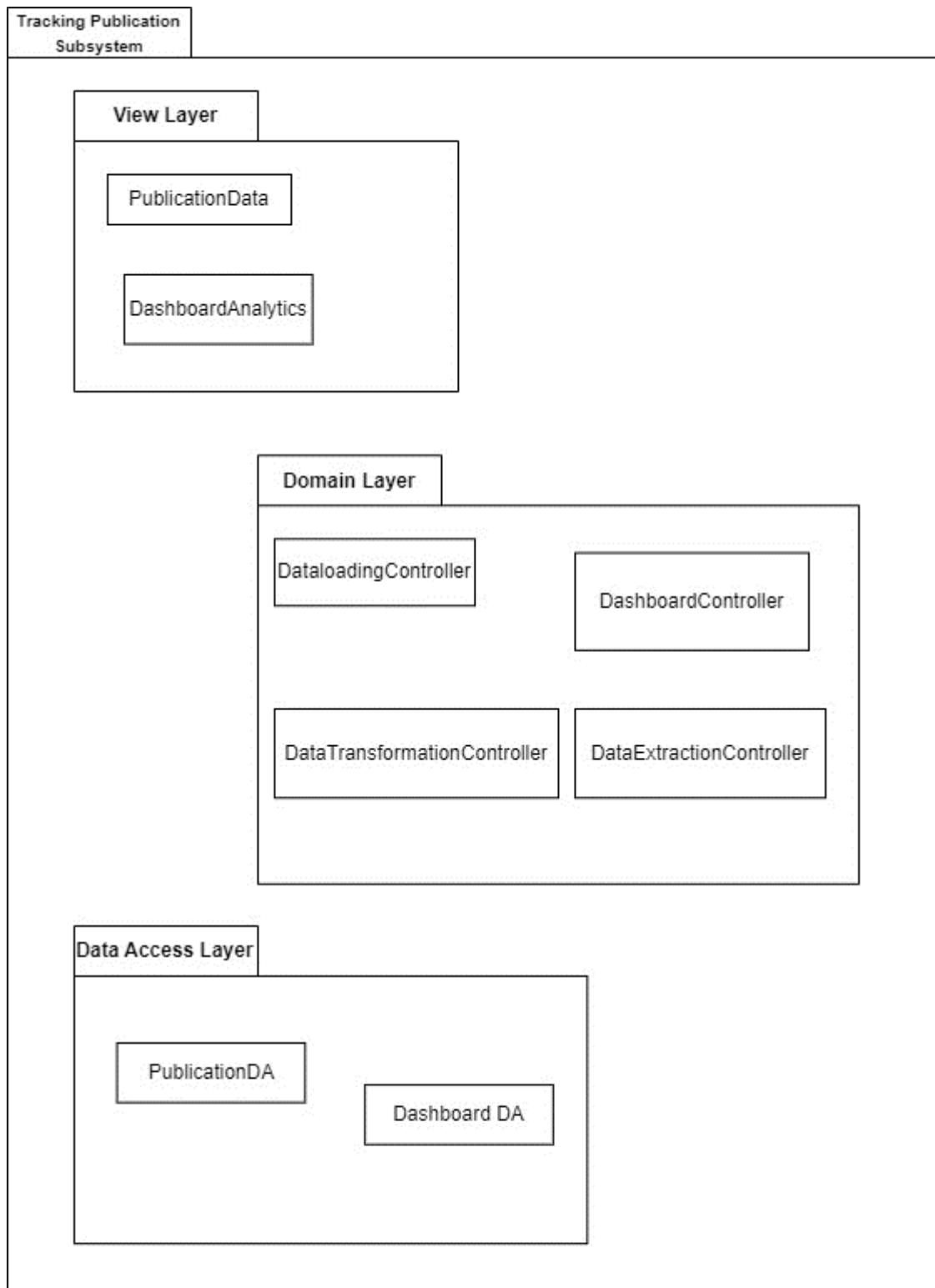


Figure 3.6: SD002 Sequence diagram for View Dashboard Analytics

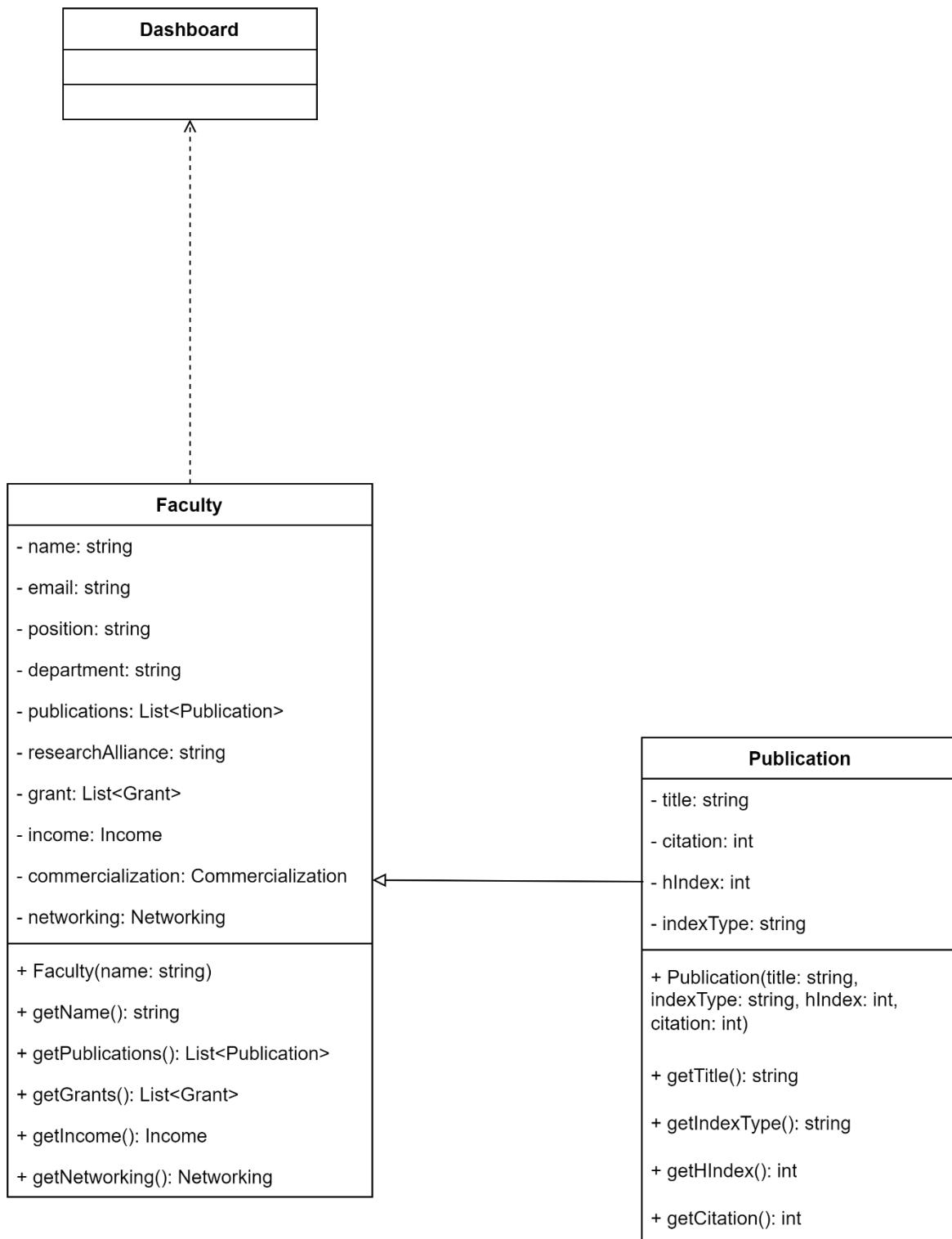
### **3.2.2 Subsystem Publication**

#### **3.2.2.1 P002: Package Publication**



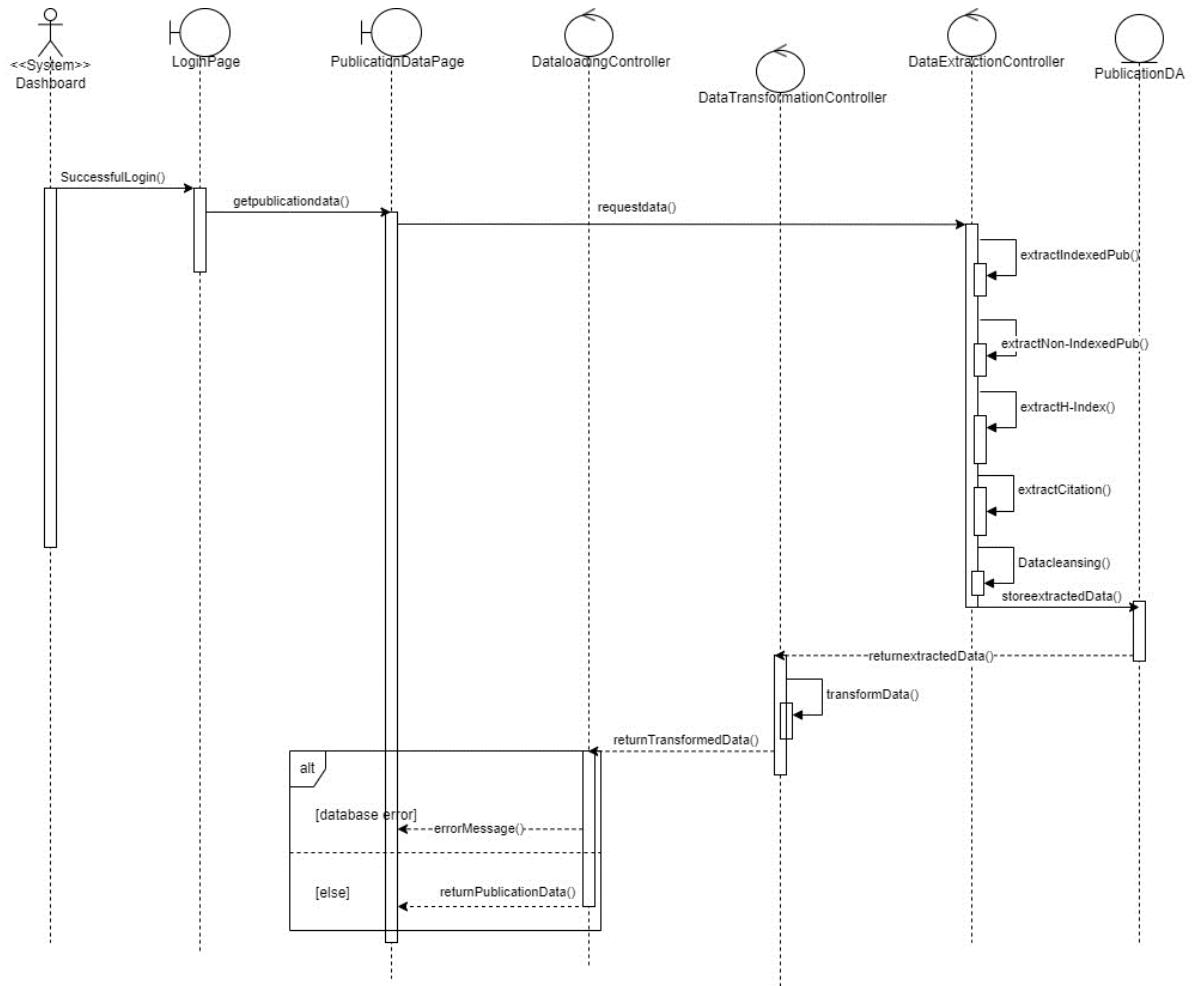
**Figure 3.7: P002 Packge Diagram for Publication**

### 3.2.2.2 Class Diagram

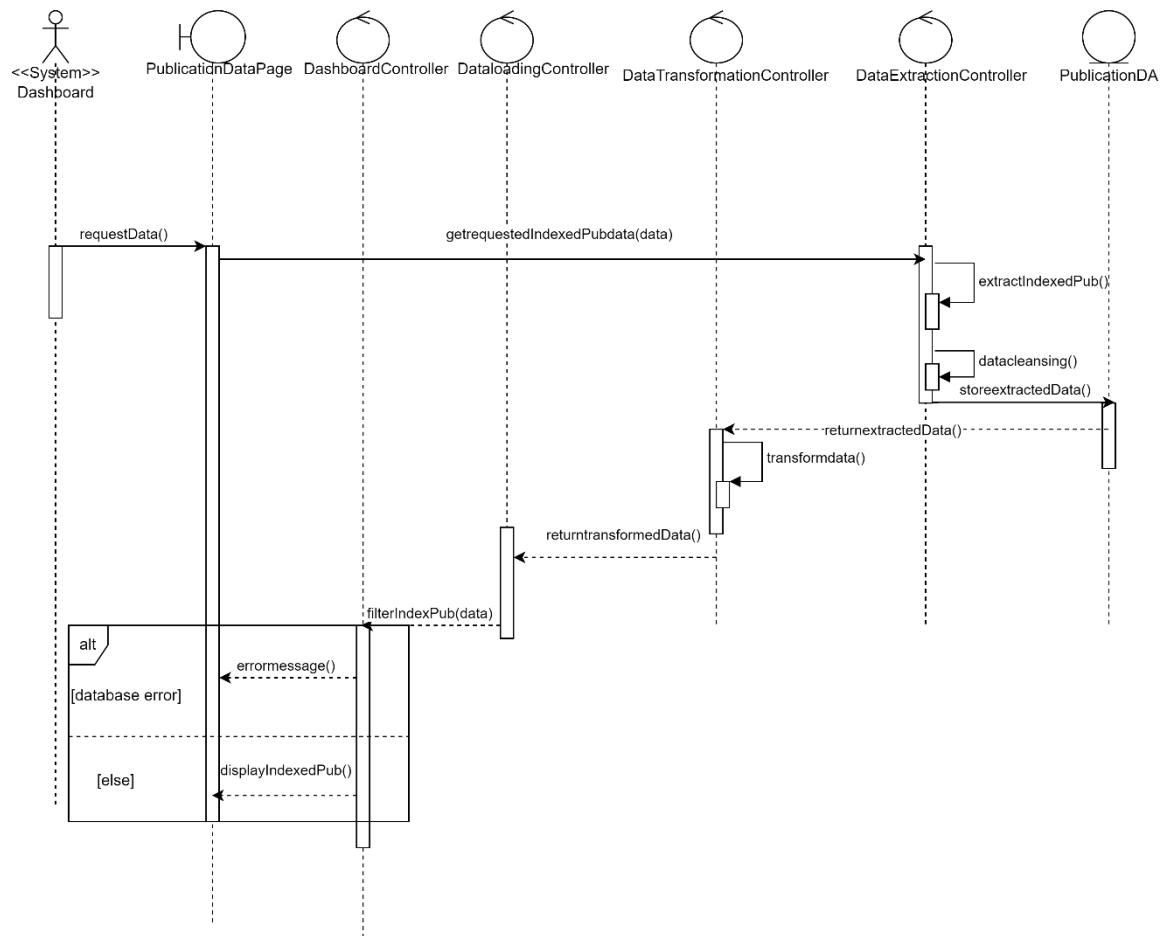


**Figure 3.8: C003 Class Diagram for Publication**

### 3.2.2.3 Sequence Diagrams



**Figure 3.9: SD003 Sequence Diagram**



**Figure 3.10: SD004 Sequence Diagram**

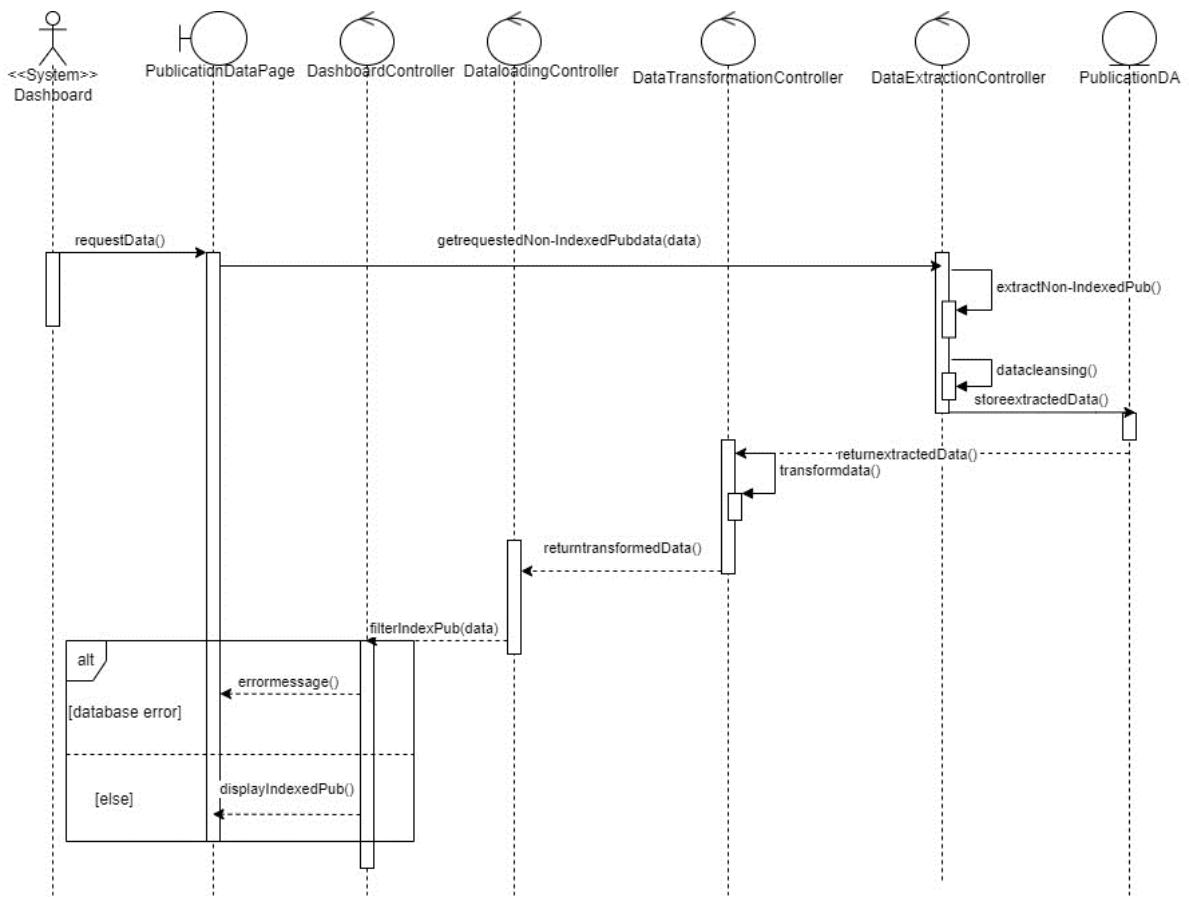
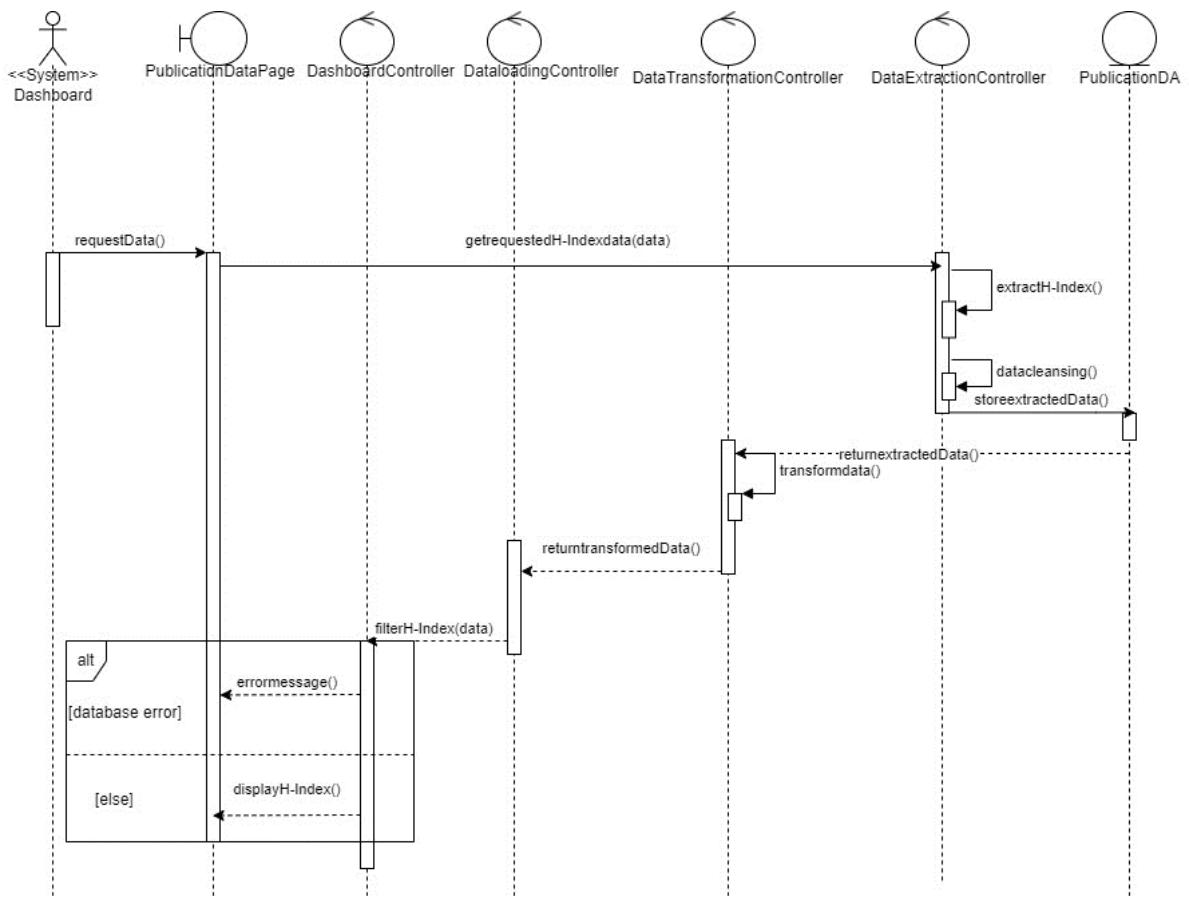
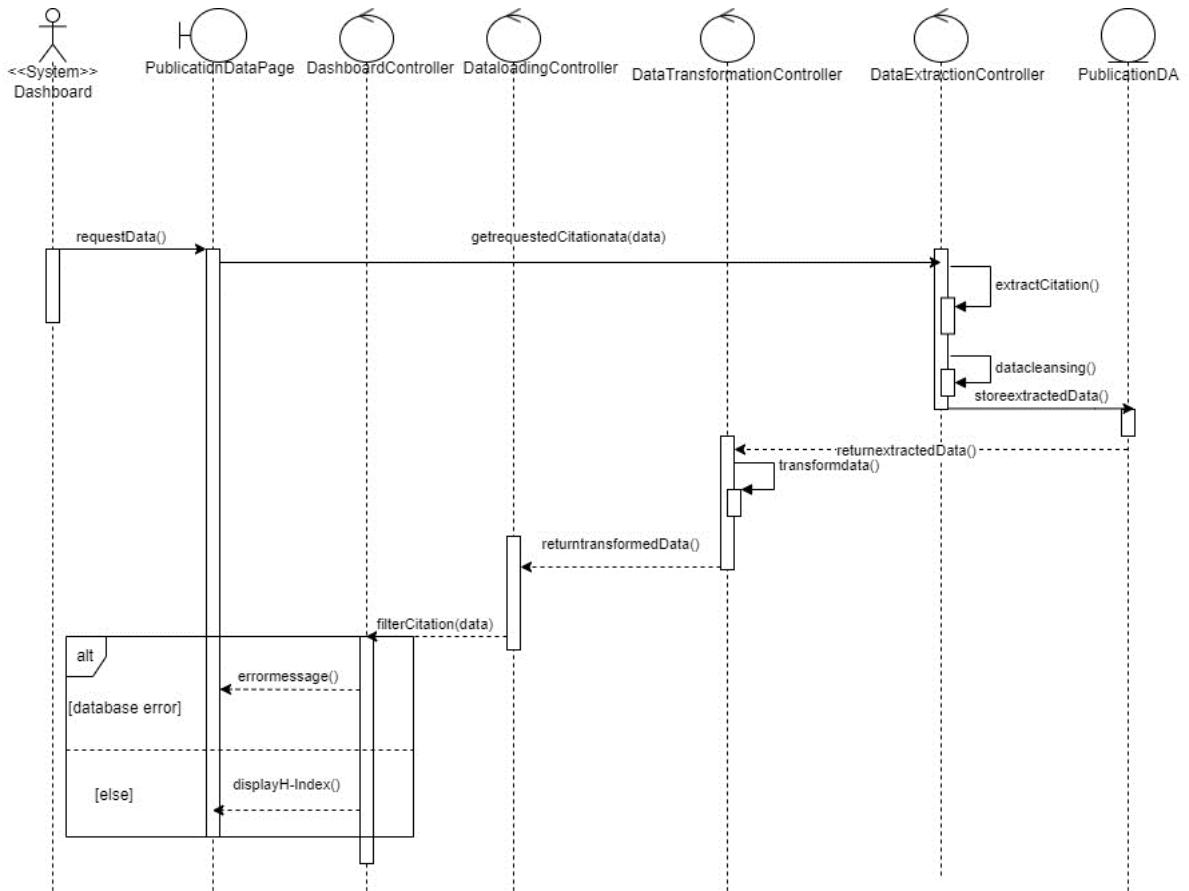


Figure 3.11: SD005 Sequence Diagram



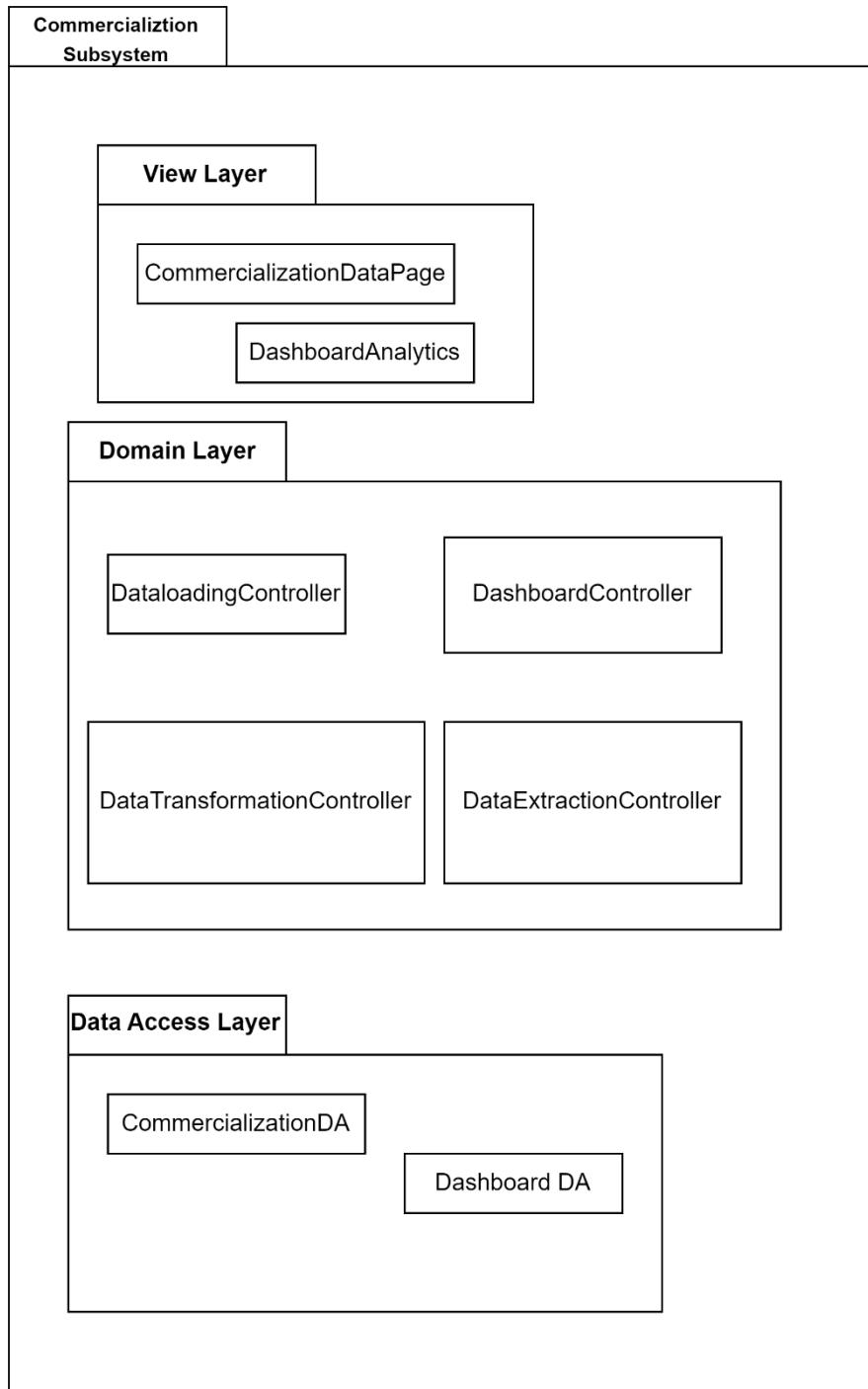
**Figure 3.12: SD006 Sequence Diagram**



**Figure 3.13: SD007 Sequence Diagram**

### 3.2.3 Subsystem Commercialization

#### 3.2.3.1 P003: Package Commercialization



**Figure 3.14: P003 Packge Diagram for Commercialization**

### 3.2.3.2 Class Diagram

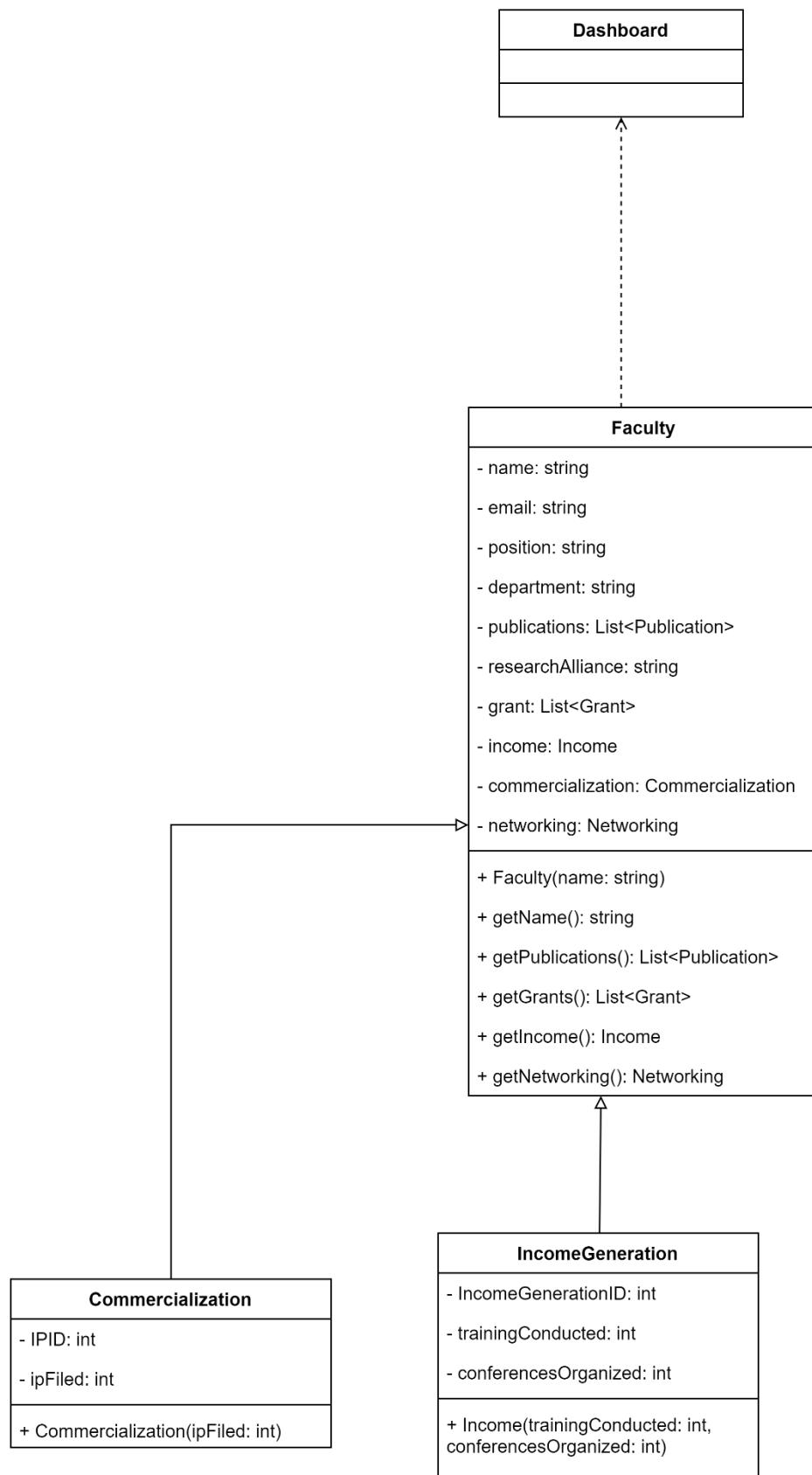


Figure 3.15: CD004

### 3.2.3.3

### Sequence Diagrams

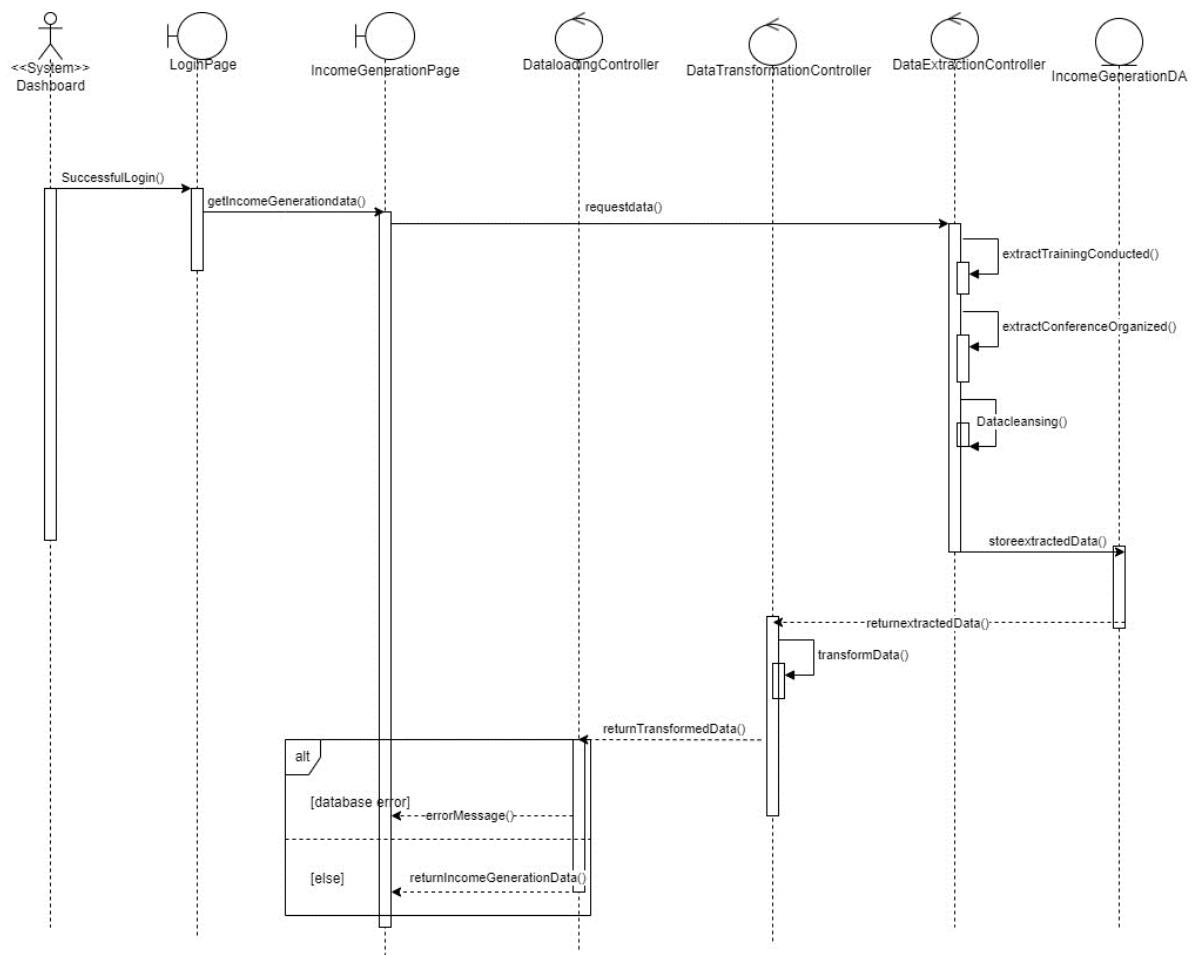
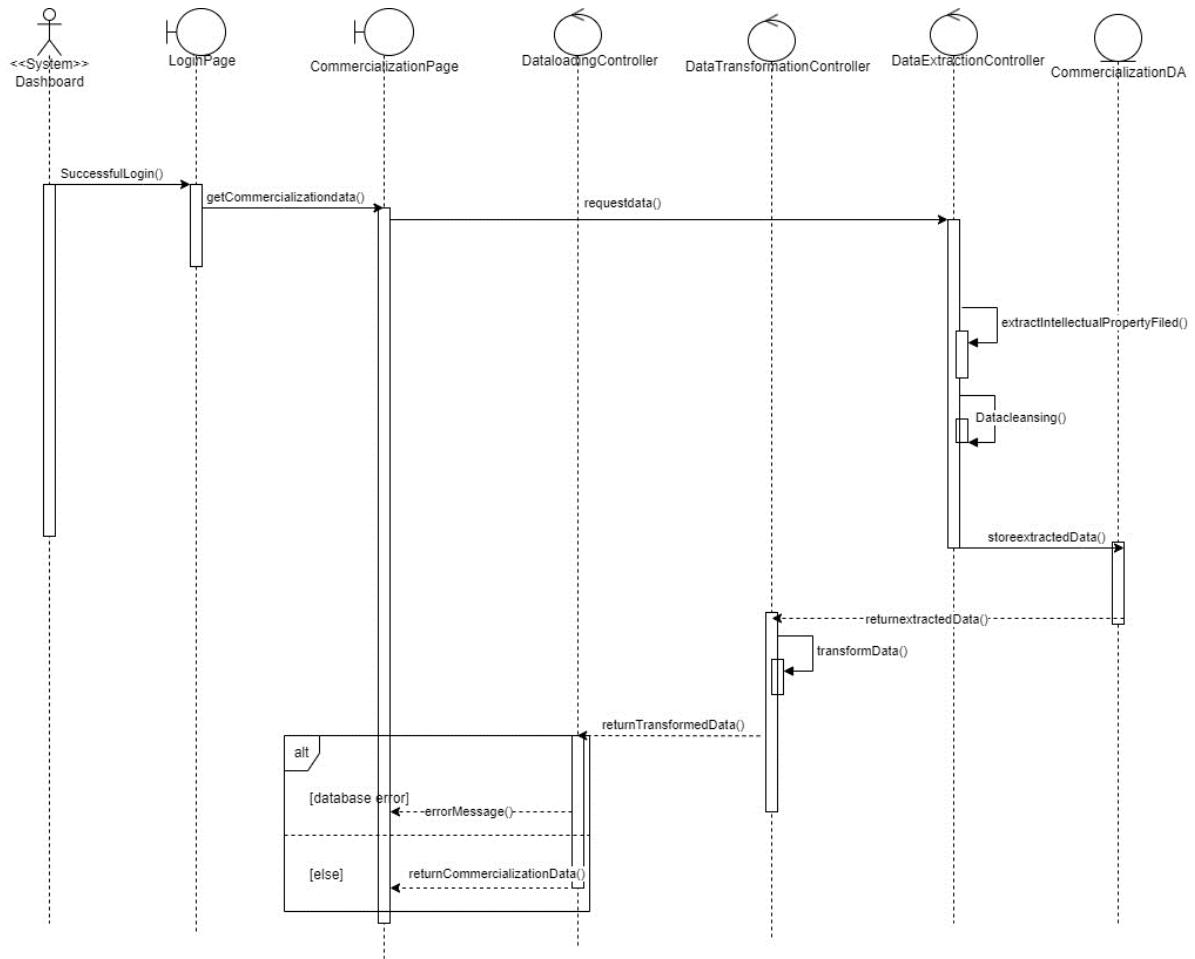


Figure 3.16: SD008



**Figure 3.17: SD009**

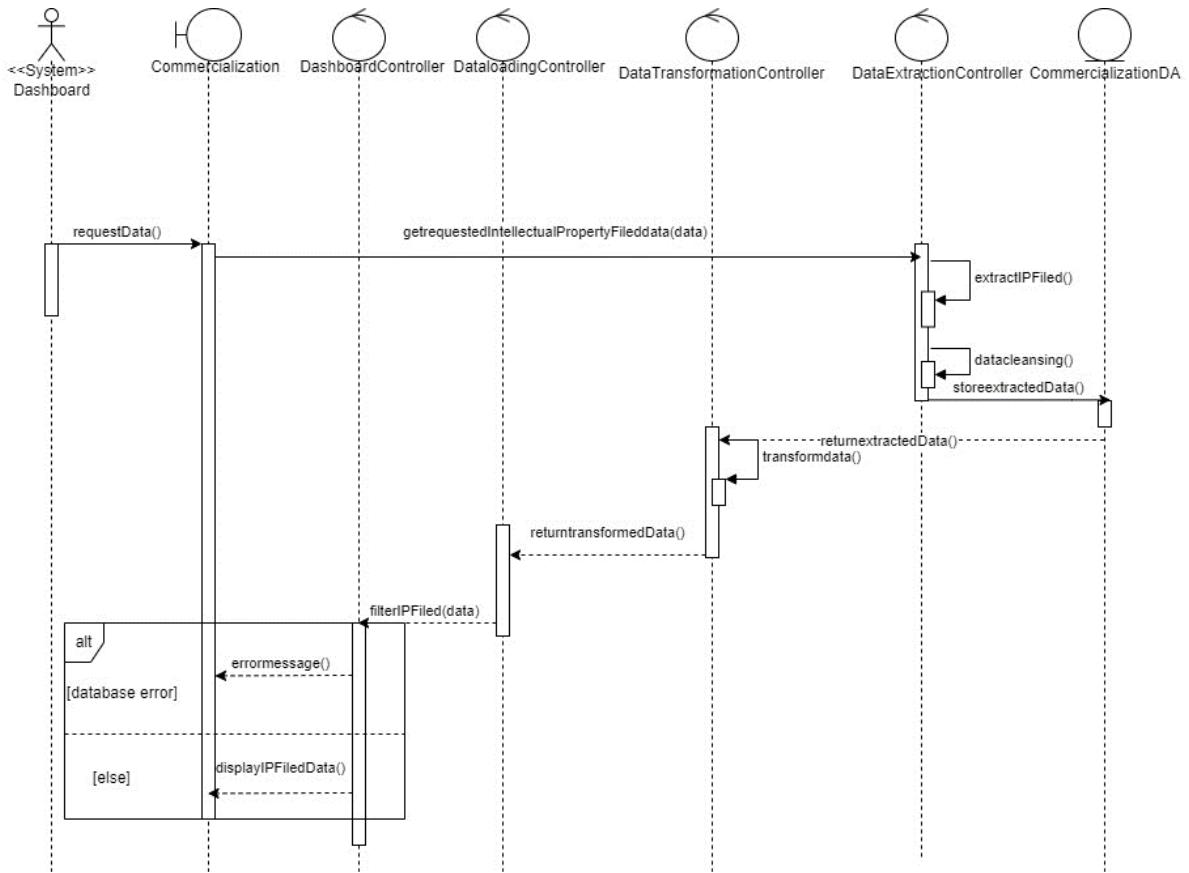
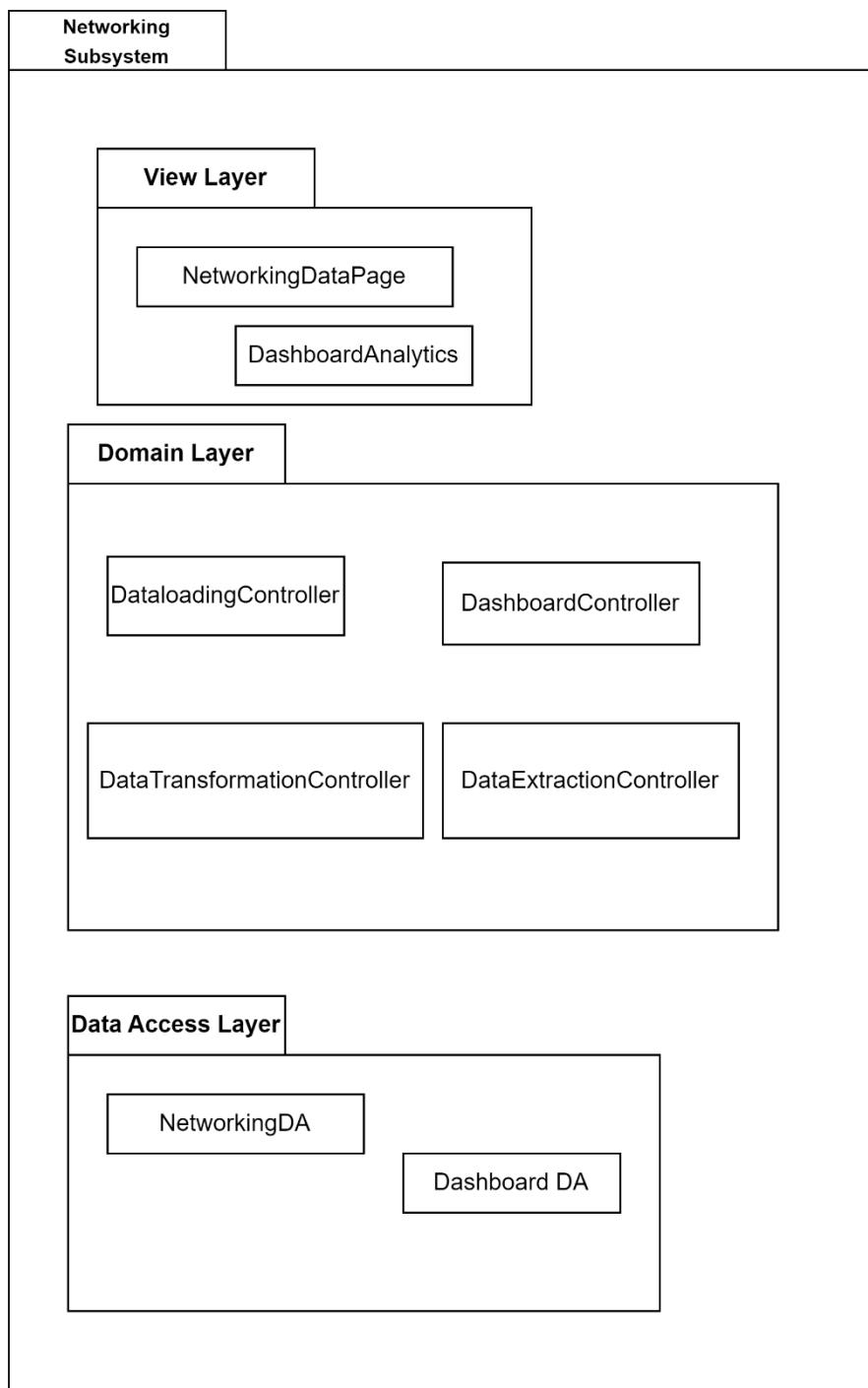


Figure 3.18: SD010

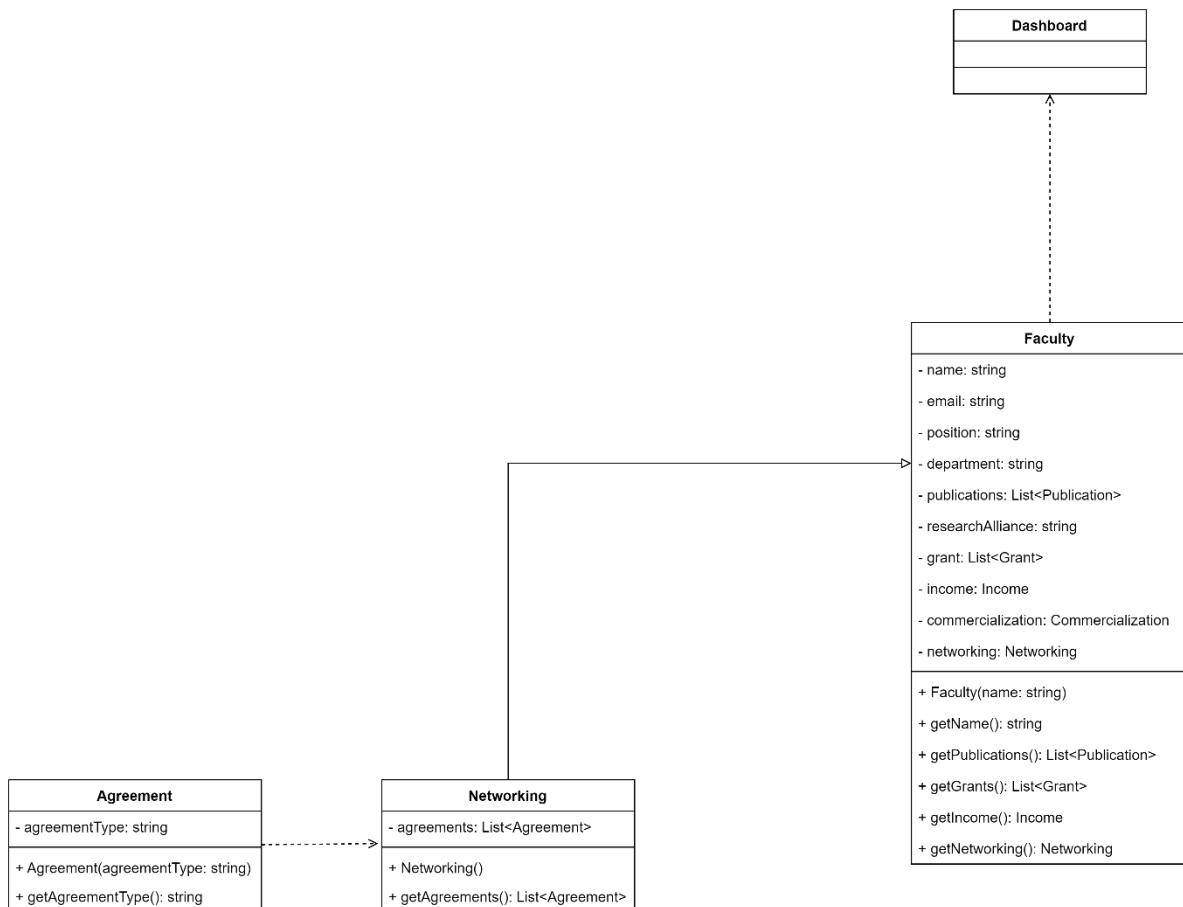
### 3.2.4 Subsystem Networking

#### 3.2.4.1 P004: Package Networking



**Figure 3.19: P004**

#### 3.2.4.2 Class Diagram



**Figure 3.20: CD004**

### 3.2.4.3 Sequence Diagram

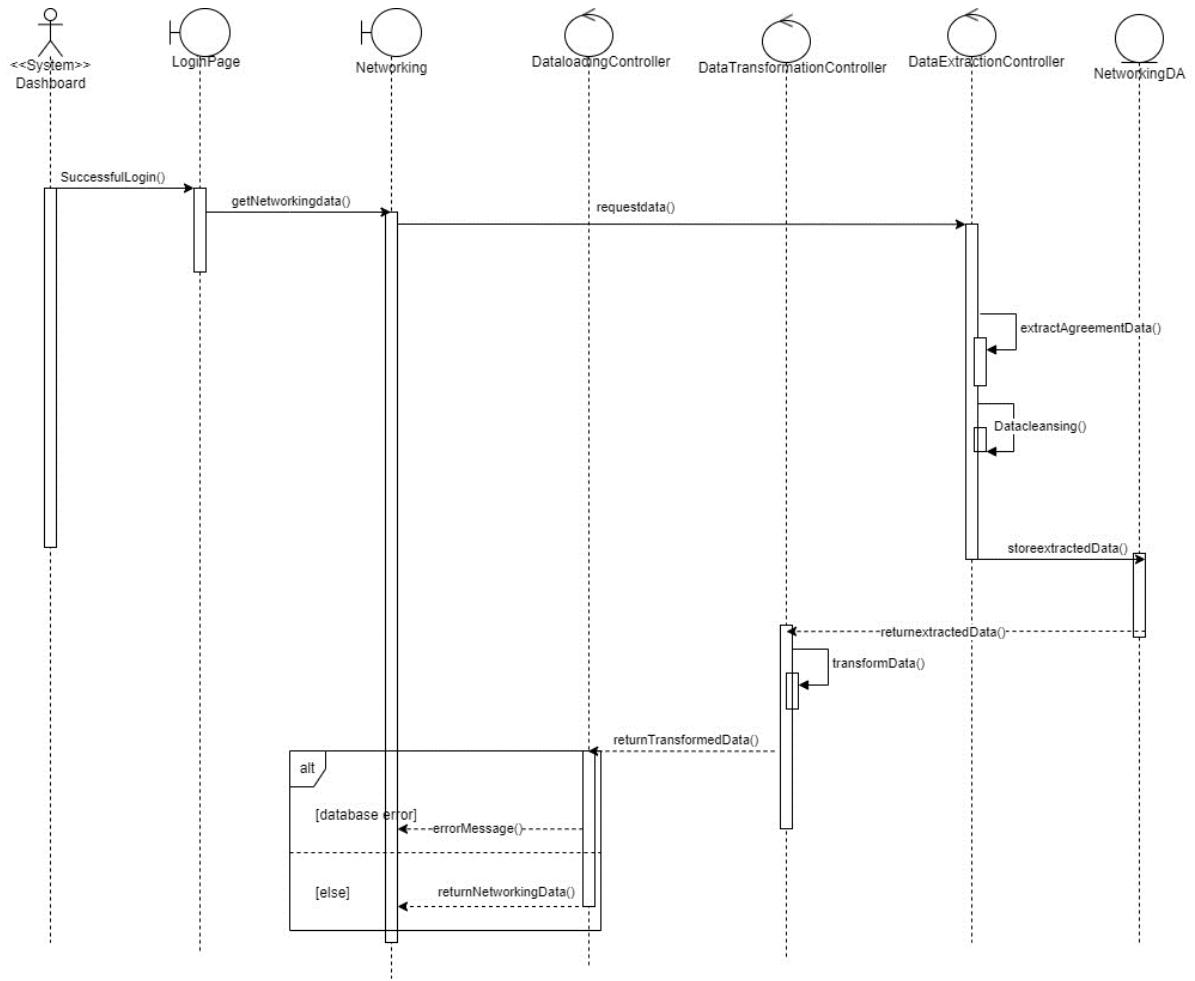
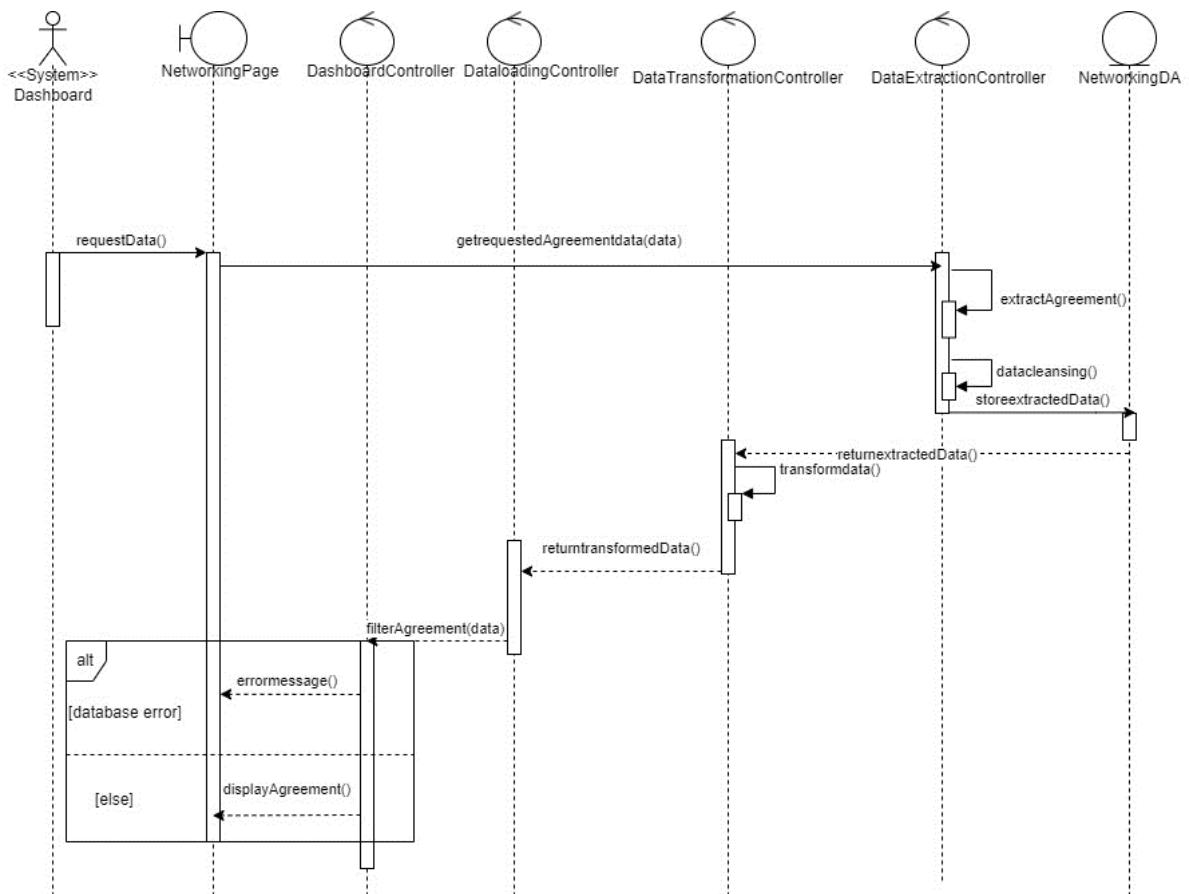


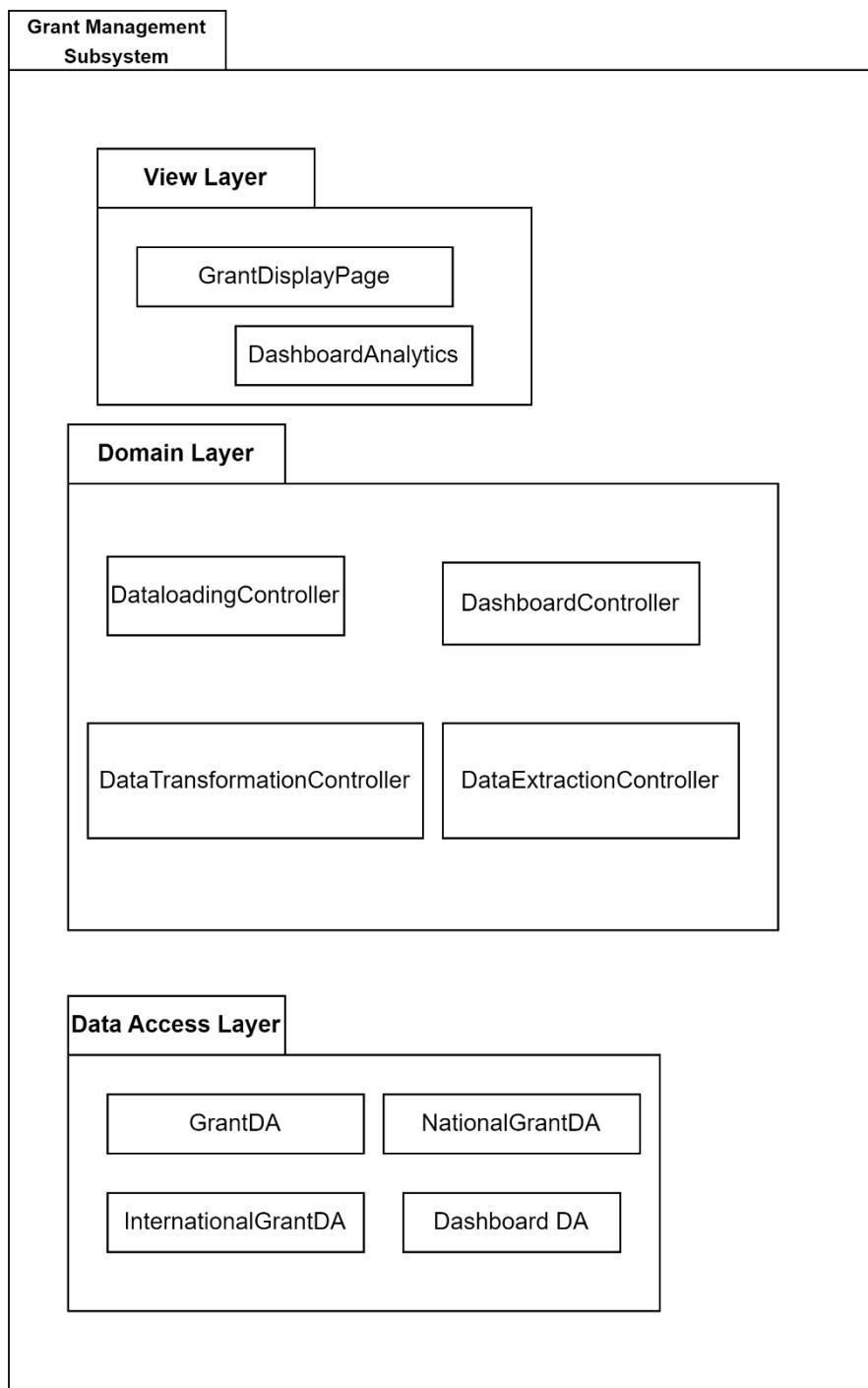
Figure 3.21 :S011



**Figure 3.22: SD012**

### 3.2.5 Subsystem Grant

#### 3.2.5.1 P005: Package Grant



**Figure 3.23: PD005**

### 3.2.5.2 Class Diagram

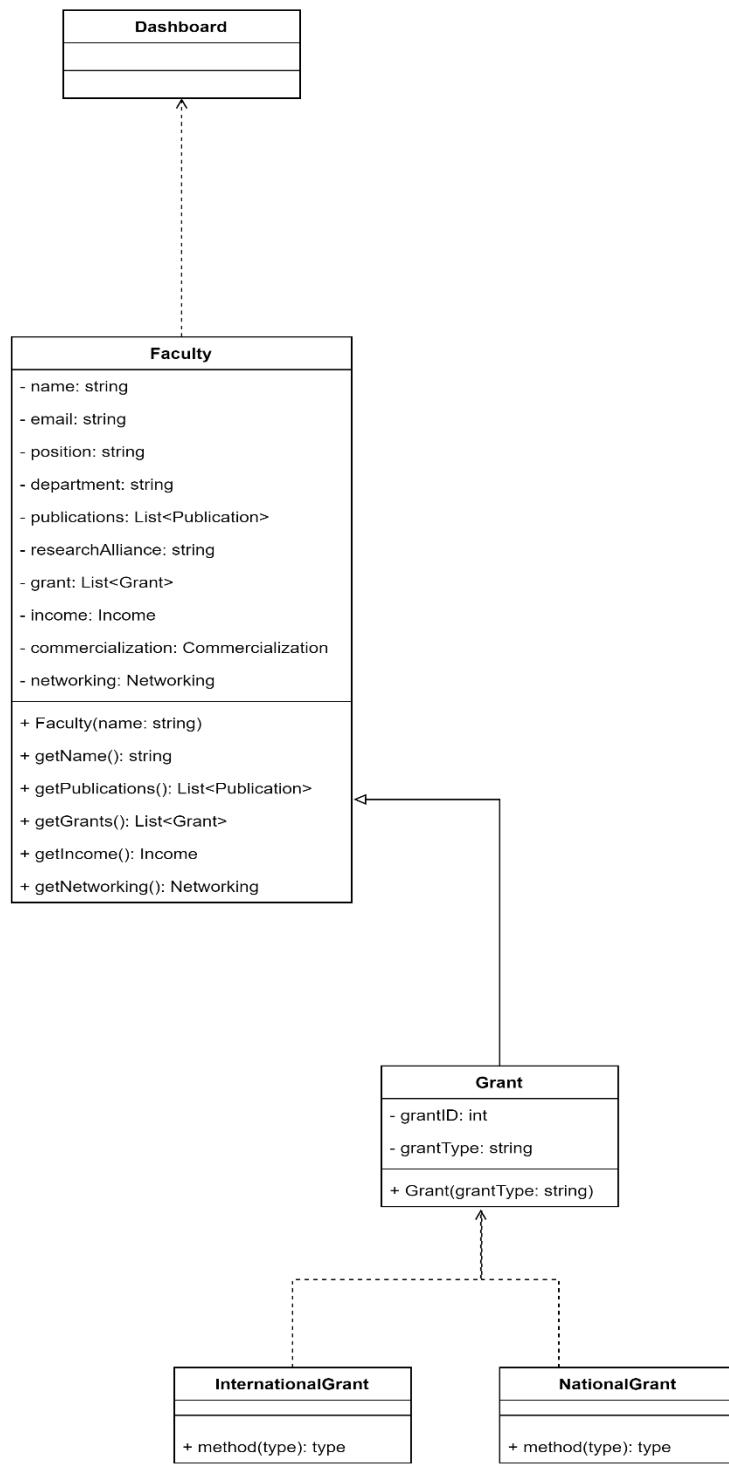


Figure 3.24: CD006

### 3.2.5.3 Sequence Diagram

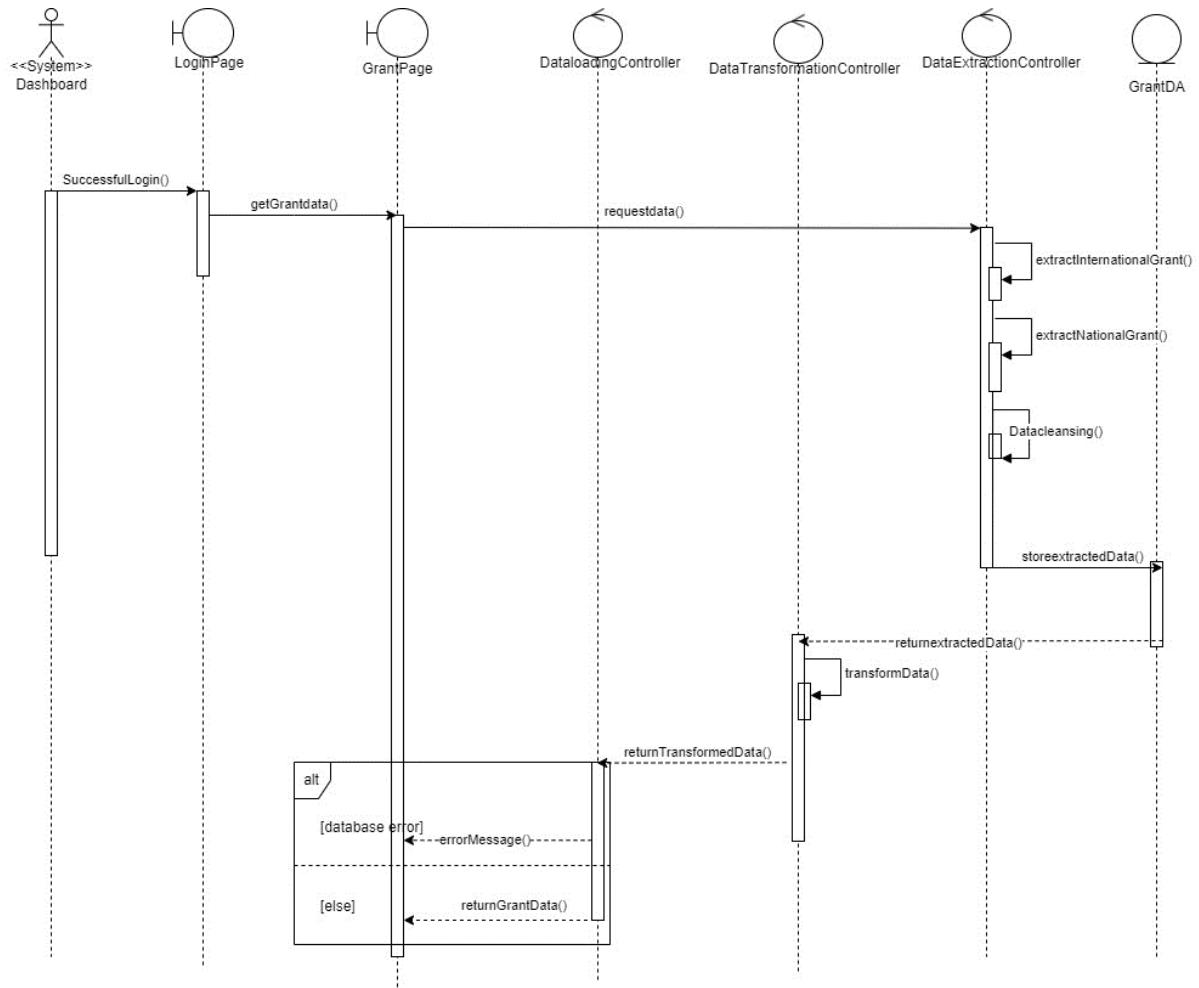
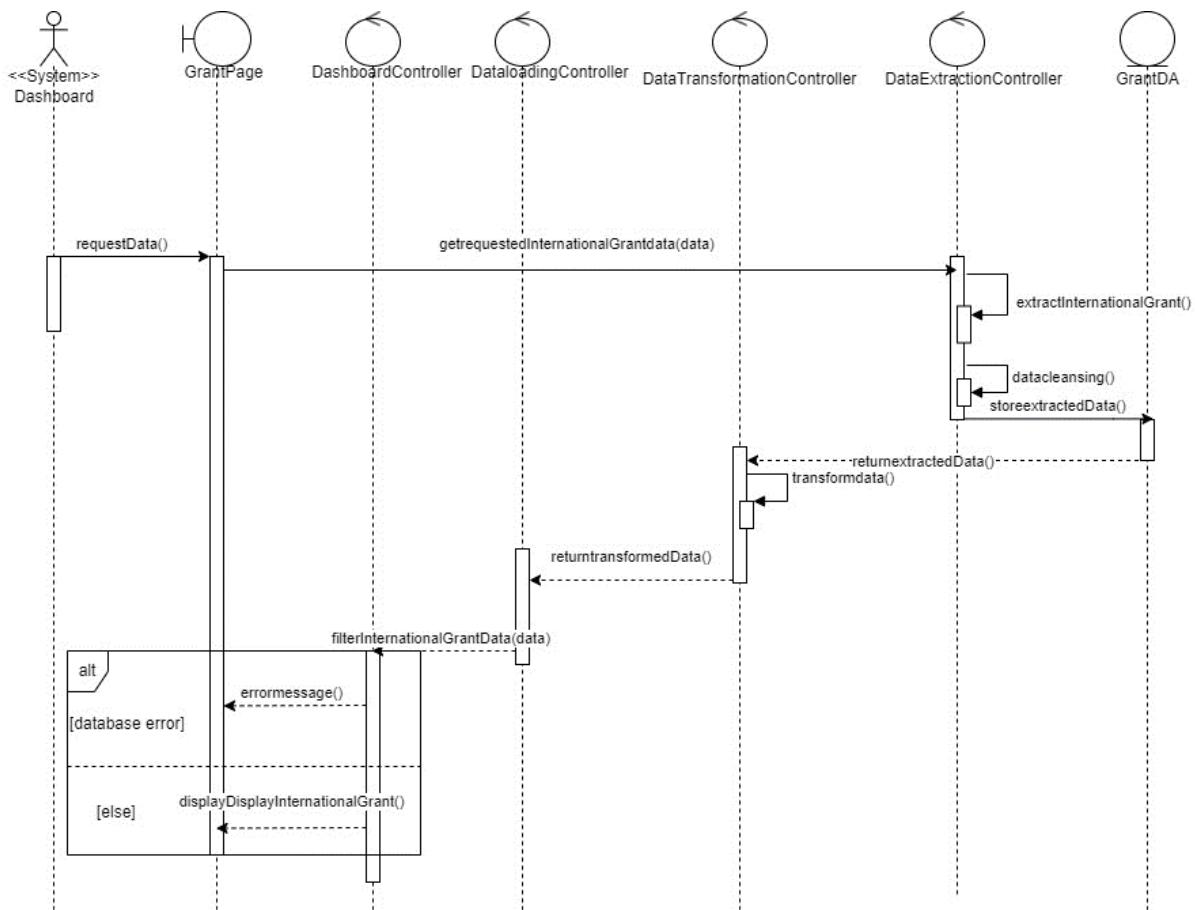


Figure 3.25: SD013



**Figure 3.26: SD014**

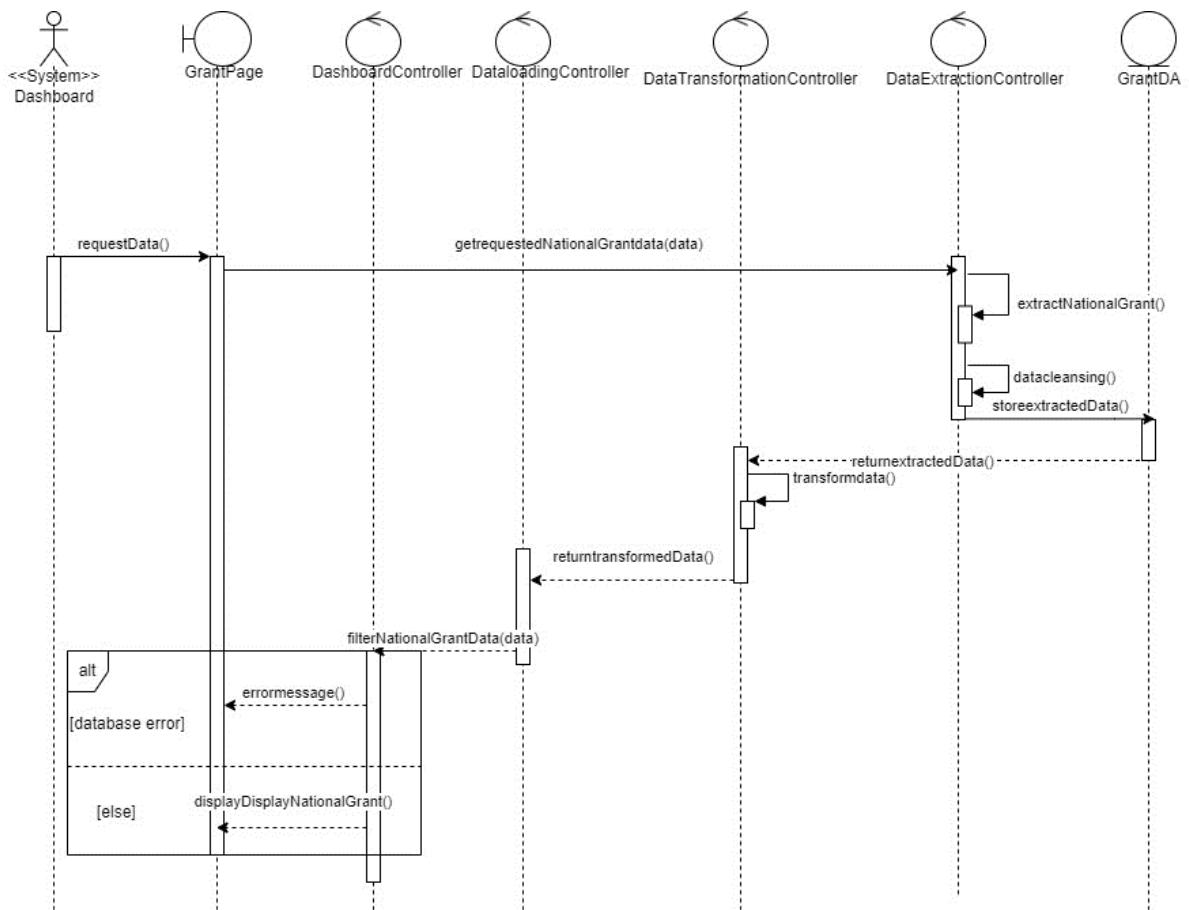


Figure 3.27: SD015

## 4. DATA DESIGN

### 4.1 Data Description

In this section, the database design will be documented. The description of the database will be done in this section as well. Data Dictionary is also listed in this section.

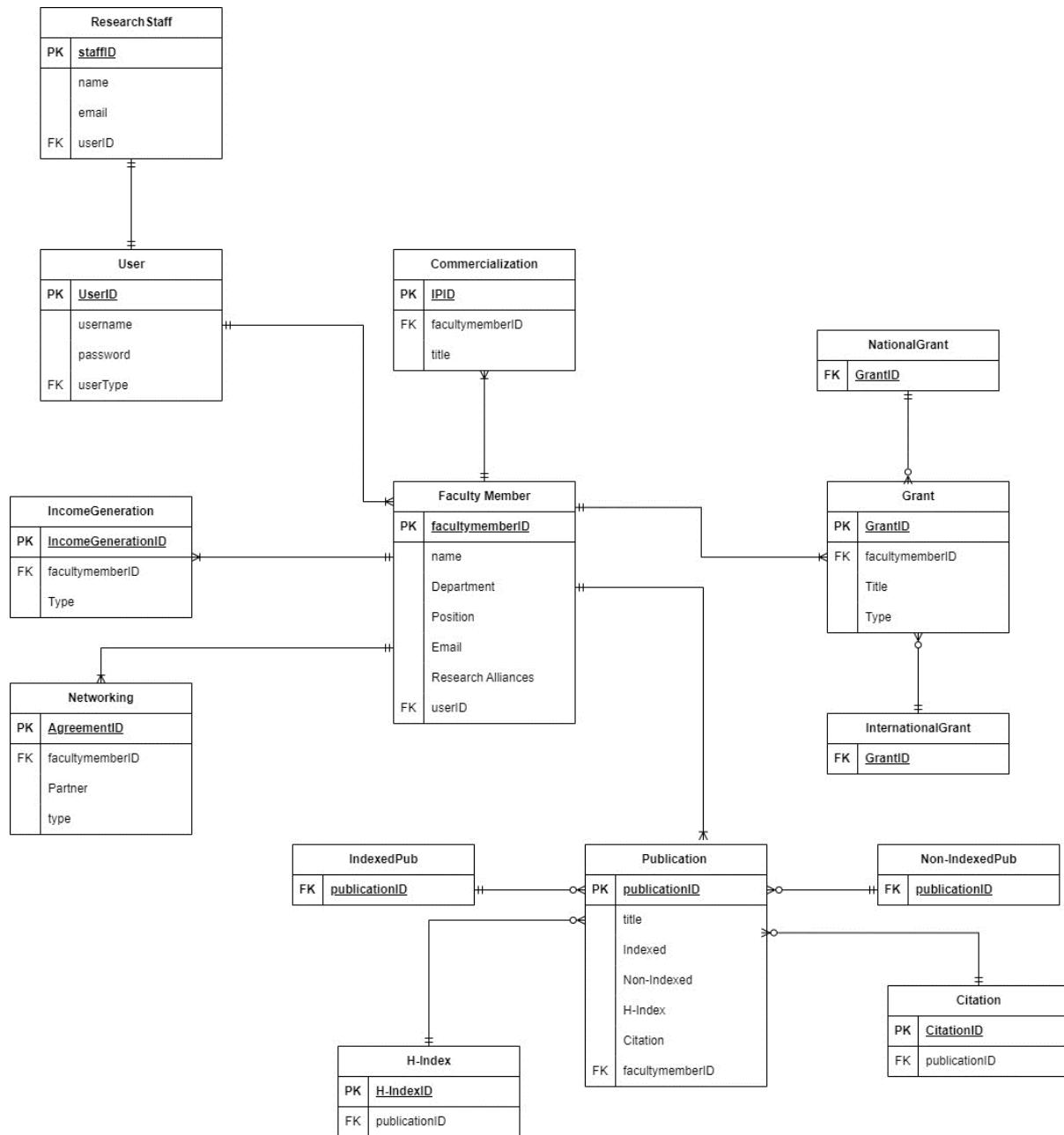


Figure 3.28: ERD001 DataBase

Entity Name	Description
Publication	It stores data related to the publication made by the researchers.

Citation	It stores citation data of the publications made by the researchers.
Indexed-Pub	It stores Indexed Publication data of the publications made by the researchers.
Non-Indexed Pub	It stores Non-Indexed Publication data of the publications made by the researchers.
H-Index	It stores number of H-Index Publication data of the publications made by the researchers.
Grant	It stores the number of grants received by the faculty.
International Grant	It stores all the number of international grants received by the faculty.
National Grant	It stores all the number of National grants received by the faculty.
Networking	It stores all the data related to Networking.
Income Generation	It stores all the data related to trainings conducted and conferences organized.
User	It contains the information of different users in one place.
Faculty Members	It contains all the data and information of the faculty members that'll be displayed.
Research Staff	It contains all the data and information of the Research staff that are required for our system.
Commercialization	It stores all the data of number of Intellectual Property filed.

## 4.2 Data Dictionary

## DD001

<b>Field Name</b>	<b>Datatype</b>	<b>Constraint</b>	<b>Description</b>
<b>User</b>			
UserId	INT	Primary Key	Unique ID for user
username	VARCHAR	Not NULL	Username of the user
password	VARCHAR	Not NULL	Password of the user
userType	VARCHAR	Foreign Key	Unique type of user
<b>Research Staff</b>			
staffID	INT	Primary Key	Unique ID for user
name	VARCHAR	Not NULL	Name of the user
email	VARCHAR	Not NULL	Email of the user
userID	INT	Foreign Key	Unique ID for user
<b>Income Generation</b>			
IncomeGenerationID	INT	Primary Key	Unique ID for Income Generation field
facultymemeberID	INT	Foreign Key	Unique ID of the faculty member
Type	VARCHAR	Not NULL	Type of Income Generation
<b>Networking</b>			
AgreementID	INT	Primary Key	Unique ID for Agreements made
facultymemeberID	INT	Foreign Key	Unique ID of the faculty member
Partner	VARCHAR	NOT NULL	Describes the partner for networking
type	VARCHAR	Not NULL	Type of Agreement made
<b>Commercialization</b>			

<b>IPID</b>	INT	Primary Key	Unique ID for Intellectual Properties Filed
facultymemeberID	INT	Foreign Key	Unique ID of the faculty member
title	VARCHAR	<b>NOT NULL</b>	Title for the intellectual property filed
<b>Faculty Member</b>			
facultymemeberID	INT	Primary Key	Unique ID of the faculty member
name	VARCHAR	<b>NOT NULL</b>	Name of the user
Department	VARCHAR	<b>NOT NULL</b>	Department of the faculty member
Position	VARCHAR	<b>NOT NULL</b>	Position of the user
Email	VARCHAR	<b>NOT NULL</b>	Email of the user
Research Alliances	VARCHAR	<b>NOT NULL</b>	Research Alliances of the user
userID	INT	Foreign Key	Unique ID of the user
<b>Grant</b>			
GrantID	INT	Primary Key	Unique ID of the Grant
facultymemeberID	INT	Foreign Key	Unique ID of the faculty member
Title	VARCHAR	<b>NOT NULL</b>	Title of the grant
Type	CHAR	<b>NOT NULL</b>	Type of the grant
<b>NationalGrant</b>			
GrantID	INT	Foreign Key	Unique ID of the Grant
<b>InternationalGrant</b>			
GrantID	INT	Foreign Key	Unique ID of the Grant

Publication			
PublicationID	INT	Primary Key	Unique ID of the Publication
title	VARCHAR	NOT NULL	Title of the publication
Indexed	BOOLEAN	NOT NULL	Indicates Indexed Publication
Non-Indexed	BOOLEAN	NOT NULL	Indicates non-Indexed Publication
H-Index	INT	NOT NULL	Indicates number of H-Index
Citation	INT	NOT NULL	Indicates count of citation
facultymemberID	INT	Foreign Key	Unique ID of the faculty member
Non-IndexedPub			
PublicationID	INT	Foreign Key	Unique ID of the Publication
IndexedPub			
PublicationID	INT	Foreign Key	Unique ID of the Publication
H-Index			
H-IndexID	INT	Primary Key	Unique ID for the H-Index
PublicationID	INT	Foreign Key	Unique ID of the Publication
Citation			
CitationID	INT	Primary Key	Unique ID for the Citation
publicationID	INT	Foreign Key	Unique ID of the Publication

## 5. USER INTERFACE DESIGN

---

### 5.1 Overview of User Interface

The overview of all the interfaces designed is discussed below for a better understanding of the user interfaces.

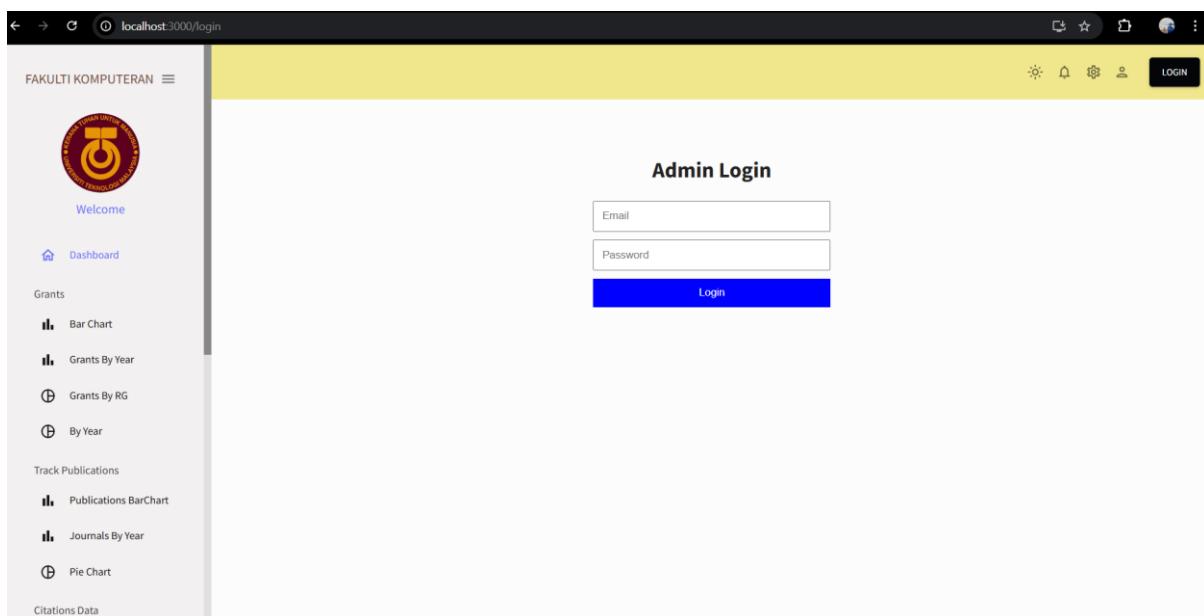
#### 5.1.1 Login Page

Login Page is the basic page for entering the system. Login is required for any system for better maintenance and security of the system. User has to login using his valid credentials.

#### 5.1.2 Dashboard Analytics Page

Dashboard Analytics consist of all the data to be displayed for the project. The data will be displayed in the form of graphs and charts. Data will include publication, networking, income generation, commercialization, and grant pages.

### 5.2 Screen Images



SN001 Login Page

localhost:3000

FAKULTI KOMPUTERAN

**FC Scholar Dashboard**  
Faculty of Computing

**Total Staff** 103

**Research Groups** 7

**Publications** 6,078

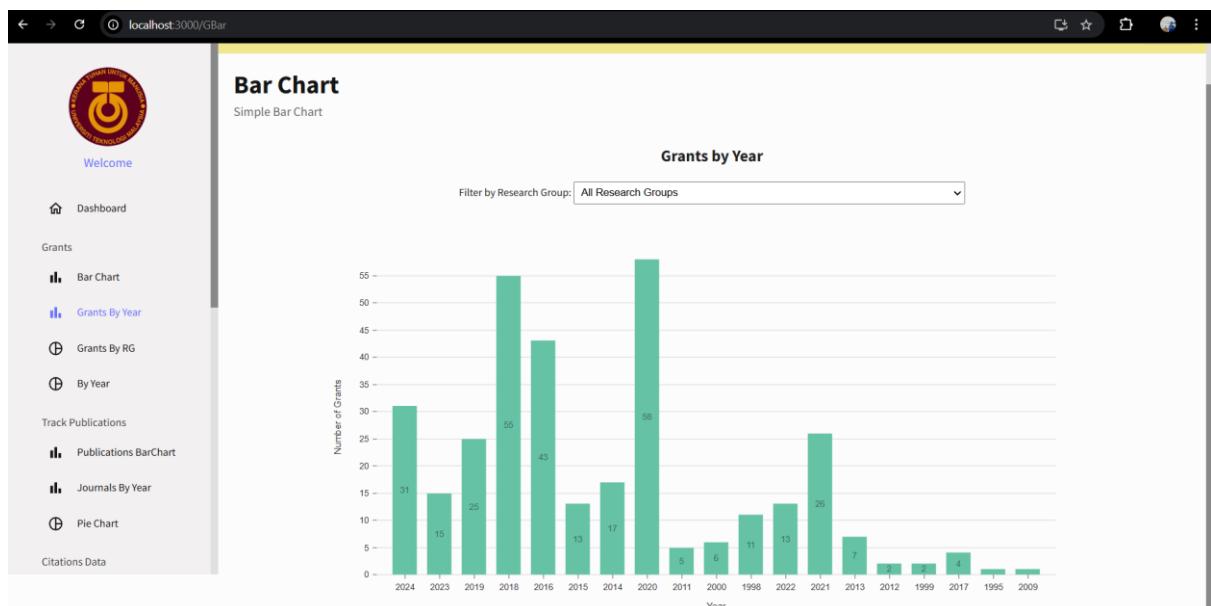
**Grants** 402

**H-Indexed (Scopus)** 902

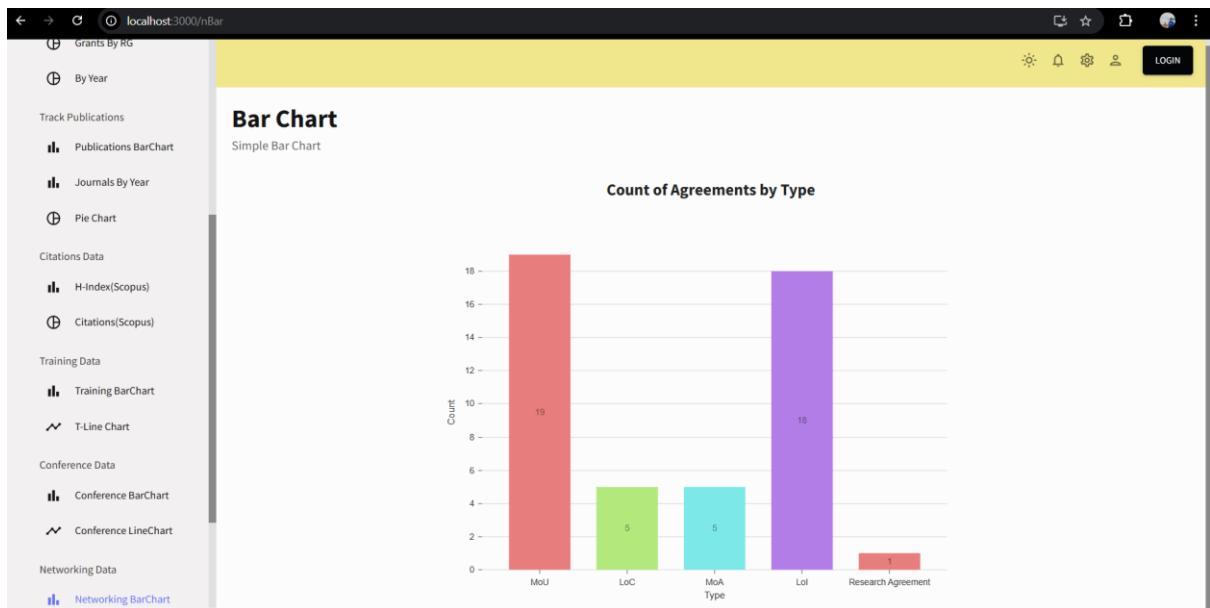
**Citations (Scopus)** 62,108

**Grants (PI & Members)** 3,523

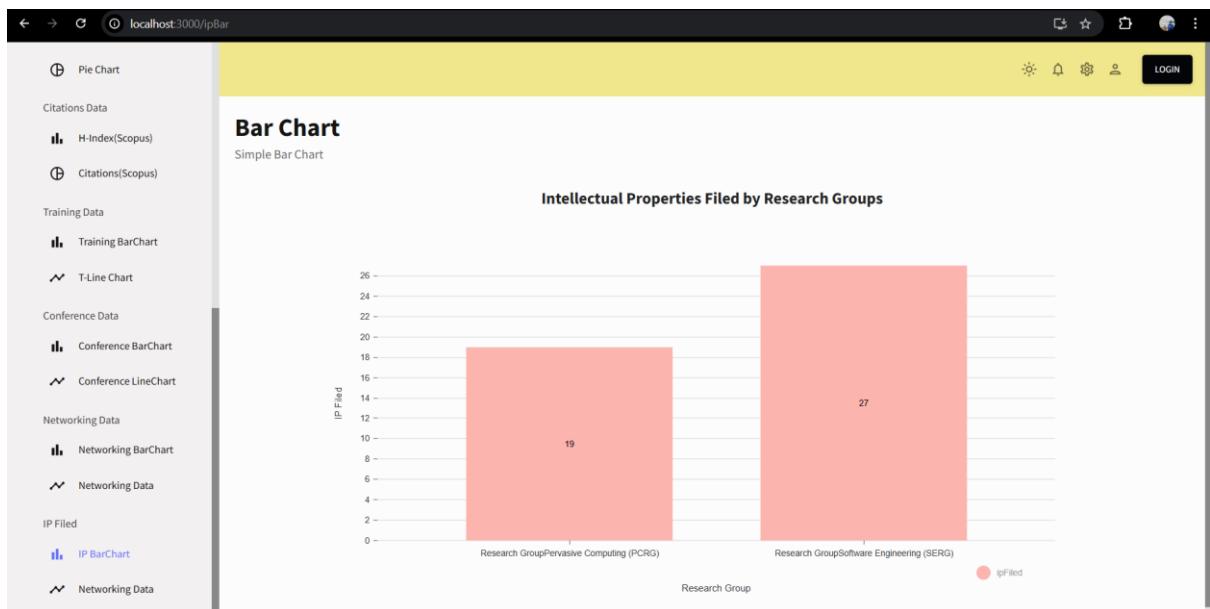
SN002 Dashboard Page



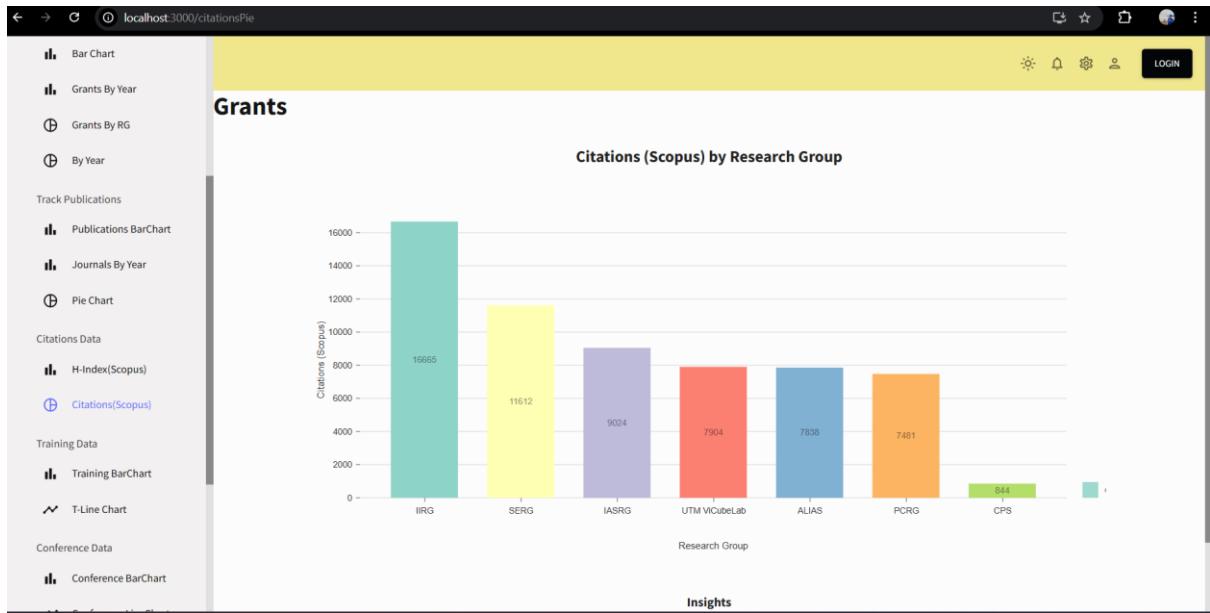
SN003 Grants by year Page



SN004 Networking Page



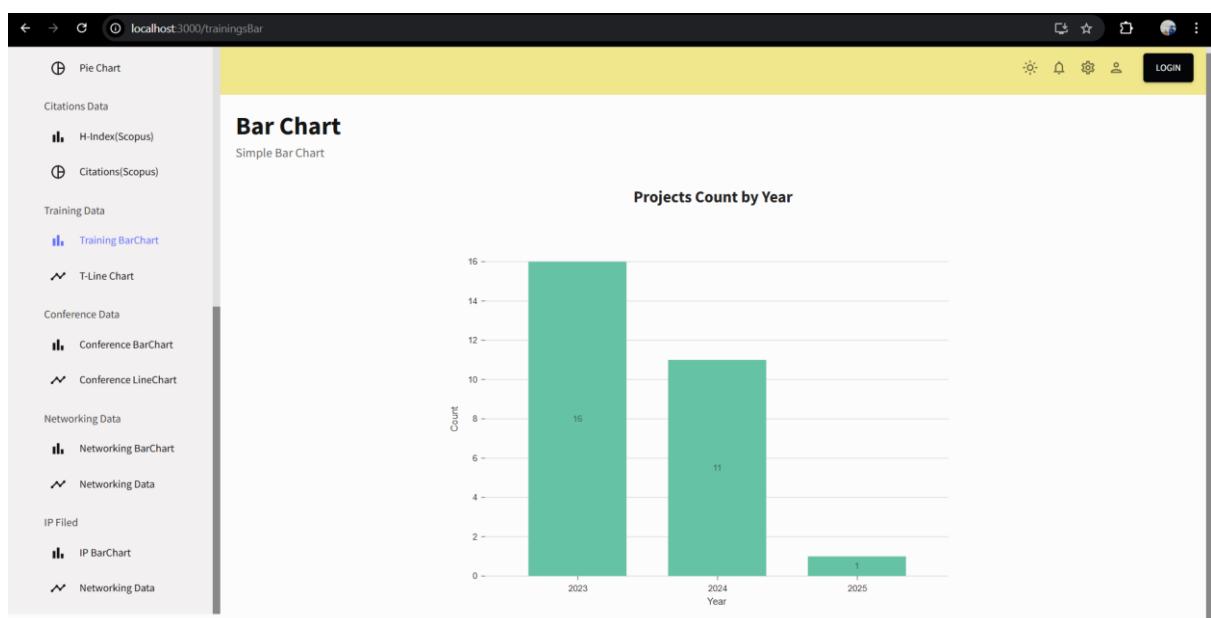
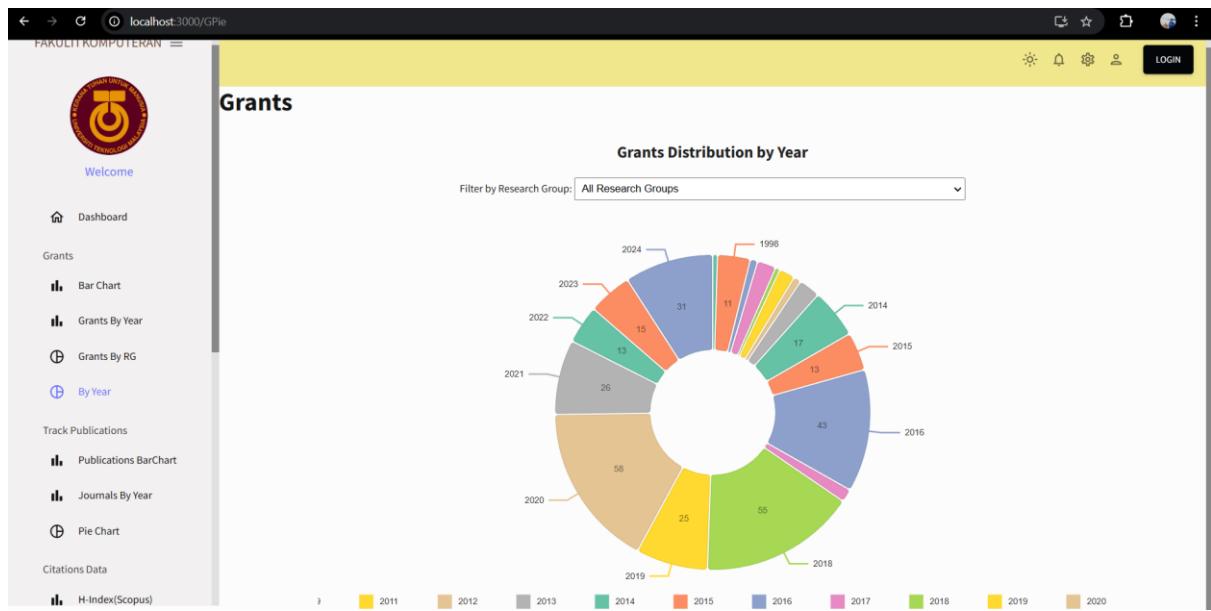
SN005 Networking Page



## SN006 Citation(Scopus) Page



## SN007 H-Index(scopus) Page



SN009 Income Generation Page

## 6. REQUIREMENTS MATRIX

---

	P001	P002	P003	P004	P005
UC001	X				
UC002	X				
UC003		X			
UC004		X			
UC005		X			
UC006		X			
UC007		X			
UC008			X		
UC009			X		
UC010			X		
UC011				X	
UC012				X	
UC013					X
UC014					X
UC015					X

## **7. APPENDICES**

---

No appendices

## **Appendix C System Testing Documentation**



---

# **Software Testing Documentation**

Faculty of Computing Staff Publication Dashboard

Version 1.0

23/01/2025

School of Computing, Software engineering

Prepared by: Adib Bin Morshed

# Revision Page

## a. Overview

This document is the first draft of the Software Test Document (STD) for the Faculty of Computing Staff Publication Dashboard.

## b. Target Audience

- **Prof Madya Dr. Siti Zaiton Bt. Mohd Hashim, Faculty of Computing, Universiti Teknologi Malaysia (UTM).**
- **Faculty of computing Staff, UTM.**

## c. Project Team Members

Adib Bin Morshed

## d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
Version 1.0	Adib Bin Morshed	SRS of Faculty of Computing Staff publication Dashboard	25 June,2023

### Note:

This template is an annotated outline for a software testing document. It is based on IEEE standards 829, 1008, 1012 and 1012a. This document covers: unit testing (the verification of individual sub-systems or components of the system against their specifications), integration testing (the testing of inter-operating sub-systems or components against their specifications)

and system testing (both verification against the system specification, and validation against the user requirements). This template has been simplified and customized to meet the need of SCSJ2203 course at Faculty of Computing, UTM. Compiled by Ruhaidah Samsudin, PhD and checked by Masitah Ghazali, PhD and Shahida Sulaiman, PhD (revised on 14 May 2016).

## Table of Contents

<b>1</b>	<b>Introduction</b>	1
1.1	Purpose	1
1.2	Scope	
1.3	Definitions, Acronyms and Abbreviations	
1.4	Reference Materials	
1.5	System Overview	
<b>2</b>	<b>Test Cases</b>	

## **2. INTRODUCTION**

---

This section describes about the test cases developed for our system.

### **2.1 Purpose**

The main purpose of this document is to describe the necessary test cases that are required to test the system. This document will include the test cases required to fulfil the requirements of the system. The requirements were gathered from our stakeholders. This STD is important for the development of the system since it contains all the necessary test data required to test the system according to the requirements that have been discussed with the stakeholders. This document is developed for the ease of understanding of the system by the developer and stakeholders.

### **2.2 Scope**

The proposed system is named the research publication dashboard for faculty of computing staff. This system is specifically designed for the stakeholders who are researchers and research staff of the faculty of computing at Universiti Teknologi Malaysia (UTM). This system will help to recognize patterns and performance measures to evaluate the current situation of the research department. Although it'll have all the necessary data required to make the evaluation, there will be a few limitations as well. This system will not include some data related to researchers which is a limitation of the system. Nevertheless, it'll be used for the performance measure of the publications by the researchers and research staff. This document is needed for the better understanding of the system by the stakeholders and developer.

### **2.3 Definitions, Acronyms and Abbreviation**

<b>Acronyms</b>	<b>Definition</b>
STD	Software Testing Documentation
TC	Test Cases

## **2.4 System Overview**

This STD will contain only the test cases for each use cases and module.

### 3. TEST CASES, DATA AND EXPECTED RESULTS

---

#### 3.1 Test TC001 for Module Authentication & View: Login(UC001)

<b>Test Case ID</b>	TC001_1		
<b>Test Case Description</b>	Able to login Successfully	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	There is an active internet connection		

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass
2	User clicks on the email field and enters his registered email	Email: example@gmail.com	Email field works correctly entered by the user	Email Field is Working	Pass
3.	User clicks on the password field and enters the password	Password: 1234admin	Password field is taking inputs	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Successful	Pass
5.	User is taken to their respective dashboard by the system	-	System verifies the user successfully by checking with the database	Page loaded successfully	Pass

<b>Test Case ID</b>	TC001_2		
<b>Test Case Description</b>	Display error message for	<b>Test Priority</b>	High

	Unsuccessful Login		
<b>Pre-Requisite</b>	There is an active internet connection		

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass
2	User clicks on the email field and enters his registered email	Email: example@gmail.com	Email field works correctly and will display the username entered by the user	Email Field is Working	Pass
3.	User clicks on the password field and enters the password	Password: 1234admin	Password field is taking inputs and user can view the entered password	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Unsuccessful	Pass
5.	User is displayed an error message for unsuccessful login.	-	System displays error message for unsuccessful login.	Invalid email or Password is displayed	Pass

<b>Test Case ID</b>	TC001_3			
<b>Test Case Description</b>	Display error message for invalid input	<b>Test Priority</b>	High	
<b>Pre-Requisite</b>	There is an active internet connection			

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass

2	User clicks on the email field and enters his registered email	Email: abc	Email field works correctly and will display to write @ in email	Email Field is Working	Pass
3.	User clicks on the password field and enters the password	Password: 1234	Password field is taking inputs and user can view the entered password	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Unsuccessful	Pass
5.	User is displayed an error message for unsuccessful login.	-	System displays error message for Invalid input.	Message Displayed	Pass

<b>Test Case ID</b>	TC001_4			
<b>Test Case Description</b>	Display error message for Empty Input	<b>Test Priority</b>		High
<b>Pre-Requisite</b>	There is an active internet connection			

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	The user opens the login page	-	System displays the login page	Page is Displayed	Pass
2	User clicks on the email field and leaves it blank	Email:	Email field works correctly	Email Field is Working	Pass
3.	User clicks on the password field and enters the password	Password: 1234	Password field is taking inputs and user can view the entered password	Password Field is Working	Pass
4.	User clicks on the 'Login' Button to confirm login	-	Login button is visible and working	Login Unsuccessful	Pass

5.	User is displayed an error message for unsuccessful login.	-	System displays error message for empty input.	Message Displayed	Pass
----	------------------------------------------------------------	---	------------------------------------------------	-------------------	------

### 3.2 Test TC002 for Module1: View Dashboard Analytics (UC002)

<b>Test Case ID</b>	TC002_1			
<b>Test Case Description</b>	Able to view dashboard analytics clearly	<b>Test Priority</b>		High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. User Logs in to the system			

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	User enters homepage	-	System Displays the homepage	Page is Displayed	Pass
2	User clicks on the Dashboard button.	-	Button works properly and user is taken to Dashboard metrics page.	Button and page are working	Pass
3.	User can view the dashboard analytics	-	All the data are displayed properly.	Data are visible	Pass

<b>Test Case ID</b>	TC002_2			
<b>Test Case Description</b>	Able to view error message	<b>Test Priority</b>		Medium
<b>Pre-Requisite</b>	1. There is an active internet connection			

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the homepage	-	Home Page is displayed	Page is Displayed	Pass
2	User clicks on the Dashboard button.	-	Button works properly and user is taken to	Button and page are working	Pass

			publication metrics page.		
3.	User cannot view the dashboard analytics due to connection error	-	Error message is shown.	Message displayed “failed to load data”	Pass

<b>Test Case ID</b>	TC002_3			
<b>Test Case Description</b>	Able to view zero data because of failure to fetch from database	<b>Test Priority</b>	Medium	
<b>Pre-Requisite</b>	1. There is an active internet connection			

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the homepage	-	System displays the page	Page is Displayed	Pass
2	User clicks on the Dashboard button.	-	Button works properly and user is taken to metrics page.	Button and page are working	Pass
3.	User cannot view the dashboard analytics due to connection error	Example: Count of total Publications	Statbox displays zero meaning data cannot be fetched from the database	Null data is shown	Pass

### 3.3 Test TC003 for Module Publication: Track Publication Metrics (UC003)

<b>Test Case ID</b>	TC003_1			
<b>Test Case Description</b>	Able to view accurate data	<b>Test Priority</b>	Medium	
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured			

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1	User clicks on the Publication button.	-	Button works properly and user is taken to publication metrics page.	Button and page is displayed properly	Pass
2	User clicks on the indexed pub graphical representation of the data.	Indexed Pub	Indexed Pub data are viewable and accurate.	Data is visible and onclick a table is displayed	Pass
3.	User clicks on the non-indexed pub graphical representation of the data.	Non-Indexed Pub	Non-Indexed Pub data are viewable and accurate.	Data is visible and onclick a table is displayed	Pass
4.	User clicks on the no. of other publications in the form of graphical representation of the data.	Other publications	H-Index Data are viewable and accurate.	Data is visible and onclick a table is displayed	Pass
5.	Users can view the accurate results of the data collected from the database.	Publication related Data	Accurate results from database.	Data is visible and accurate results displayed	Pass

<b>Test Case ID</b>	TC003_2		
<b>Test Case Description</b>	Verify that user is unable to view accurate results.	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured		

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the homepage	-	Browser displays the system	Page is Displayed	Pass
2	User clicks on the Publication Data button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	Users cannot view the results of the data collected from database.	Publication Data	System displays an error message	Error Message “failed to load data” Is displayed	Pass

<b>Test Case ID</b>	TC003_3			
<b>Test Case Description</b>	Verify that system is displaying invalid table of data onclick	<b>Test Priority</b>		Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the homepage	-	System displays homepage	Page is Displayed	Pass
2	User clicks on the Publication metrics button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	Clicking on any Bar Graph related to publication data	Publication Data	Bar Graph displays a table onclick	Table is displayed	Pass
4.	Bargraph is displaying invalid data.	Publication Data	Invalid data are displayed.	Data doesn't match with database and is thus invalid	Pass

<b>Test Case ID</b>	TC003_4			
<b>Test Case Description</b>	Verify that system is displaying valid table of data onclick	<b>Test Priority</b>		Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the homepage	-	System displays homepage	Page is Displayed	Pass
2	User clicks on the Publication metrics button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	Clicking on any Bar Graph related to publication data	Publication Data	Bar Graph displays a table onclick	Table is displayed	Pass

4.	Bargraph is displaying valid data.	Publication Data	Valid data are displayed.	Data matches with database and is thus valid	Pass
----	------------------------------------	------------------	---------------------------	----------------------------------------------	------

### 3.4 Test TC004 for Module Publication: Generate Indexed Publication (UC004)

<b>Test Case ID</b>	TC004_1		
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the publication section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Publication Data button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	User can view the generated data from the database	Indexed Publication Data	Data has been successfully retrieved from database.	Data displayed in the form of a bargraph	Pass

<b>Test Case ID</b>	TC004_2		
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is Secured		

**Test execution steps:**

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the publication section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Publication metrics button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	Onlick function is working	Pass
4.	User can view the accurate data from the database	Indexed Publication Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC004_3		
<b>Test Case Description</b>	Verify that system is not displaying table of data	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured		

**Test execution steps:**

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the publications section	-	Publication Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Publication metrics button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is not displaying a table onclick	Onlick function is not working, and error message is shown	Pass

**3.5 Test TC005 for Module Publication: Generate Non-Indexed Publication (UC005)**

<b>Test Case ID</b>	TC005_1		
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the publication section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Publication Data button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	User can view the generated data from the database	Non-Indexed Publication Data	Data has been successfully retrieved from database.	Data displayed in the form of a bargraph	Pass

<b>Test Case ID</b>	TC005_2		
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the publication section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Publication metrics button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass

3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	On click function is working	Pass
4.	User can view the accurate data from the database	Non-Indexed Publication Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC005_3			
<b>Test Case Description</b>	Verify that system is not displaying accurate data	<b>Test Priority</b>		High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is Secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the publications section	-	Publication Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Publication metrics button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	Non-Indexed Publications	Bar Graph is not displaying a table onclick	On click function is not working, and error message is shown	Pass

### 3.6 Test TC006 for Module Publication: Generate H-Indexed Publication (UC006)

<b>Test Case ID</b>	TC006_1			
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>		Medium
<b>Pre-Requisite</b>	1. There is an active internet connection			

	2. Database Connection is secured
--	-----------------------------------

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Citations section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the H-Index (Scopus) Data button.	-	Button works properly and user is taken to H-Index metrics page.	Button and page are working	Pass
3.	User can view the generated data from the database	Non-Indexed Publication Data	Data has been successfully retrieved from database.	Data displayed in the form of a bargraph	Pass

<b>Test Case ID</b>	TC006_2		
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Citations section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the H-Index (Scopus) button.	-	Button works properly and user is taken to H-Index metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	Onlick function is working	Pass
4.	User can view the accurate data from the database	H-Indexed Publication Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC006_3		
<b>Test Case Description</b>	Verify that system is not displaying accurate data	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Citations section	-	Citations Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the H-Index (Scopus) metrics button.	-	Button works properly and user is taken to H-Index metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	H-Indexed (Scopus)	Bar Graph is not displaying a table onclick	On click function is not working, and error message is shown	Pass

### 3.7 Test TC007 for Module publication: Generate Citation Publication (UC007)

<b>Test Case ID</b>	TC007_1		
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Citations section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass

2	User clicks on the Citations (Scopus) Data button.	-	Button works properly and user is taken to Citations page.	Button and page are working	Pass
3.	User can view the generated data from the database	Citations Data	Data has been successfully retrieved from database.	Data displayed in the form of a bar graph	Pass

<b>Test Case ID</b>	TC007_2		
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Citations section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Citations (Scopus) button.	-	Button works properly and user is taken to Citations metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	On click function is working	Pass
4.	User can view the accurate data from the database	Citations Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC007_3		
<b>Test Case Description</b>	Verify that system is not displaying accurate data	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Citations section	-	Citations Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Citations (Scopus) metrics button.	-	Button works properly and user is taken to Citations metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	Citations (Scopus) Data	Bar Graph is not displaying a table onclick	On click function is not working, and error message is shown	Pass

### 3.8 Test TC008 for Module Commercialization: Generate & Track Trainings Conducted (UC008)

Test Case ID	TC008_1		
Test Case Description	Able to view accurate data	Test Priority	Medium
Pre-Requisite	1. There is an active internet connection 2. Database connection is Secured		

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Trainings Conducted Section	-	Home Page is displayed	Section is visible on sidebar	Pass
2	User clicks on the Trainings Conducted button.	-	Button works properly and user is taken to Trainings Conducted metrics page.	Button and page are working	Pass
3.	User clicks over the training conducted representation of the data.	Training Conducted	Training conducted data are viewable and accurate.	On click function is working	Pass

4.	Users can view the accurate results of the data collected from the database.	Trainings Conducted Data	Accurate results are displayed.	Results matches with database	Pass
----	------------------------------------------------------------------------------	--------------------------	---------------------------------	-------------------------------	------

<b>Test Case ID</b>	TC008_2		
<b>Test Case Description</b>	Verify that user is unable to view accurate results.	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Trainings Conducted section	-	Trainings Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Trainings Conducted button.	-	Button works properly and user is taken to Trainings Conducted metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	Trainings Conducted Data	Bar Graph is not displaying a table onclick	On click function is not working, and error message is shown	Pass

<b>Test Case ID</b>	TC008_3		
<b>Test Case Description</b>	Verify that admin can do CRUD Operations on Trainings Conducted	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Admin Logs in to the system		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Managing Section after logging in	-	Management Section is displayed	Section is visible on sidebar	Pass

2	User clicks on the Trainings Management button.	-	Button works properly and user is taken to Management page.	Button and page are working	Pass
3.	User Adds a new data of the project	Project Leader: Dr. Adib, Budget Code: 1122, Title: "asdad", Client: "Cyber", Project_Cost: 2001042, Year: 2025	Data Added Successfully	Data is added and displayed in the table	Pass
4.	User Updates data of a project	Project Leader: "Dr. Nihal"	Data Updated Successfully	Data is updated and displayed in the table	Pass
5.	User Deletes a row	-	Data Deleted Successfully	Data is deleted successfully	Pass
6.	Data is saved in the database successfully after every operation	Trainings Conducted Data	Database updated Successfully	Database is updated according to the table	Pass

### 3.9 Test TC009 for Module Commercialization: Generate & Track Conferences Organized (UC009)

<b>Test Case ID</b>	TC009_1		
<b>Test Case Description</b>	Able to view accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is Secured		

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Conferences Organized Section	-	Conferences Organized is displayed	Section is visible on sidebar	Pass

2	User clicks on the Conferences Organized button.	-	Button works properly and user is taken to Conferences Organized metrics page.	Button and page are working	Pass
3.	User clicks over the Conferences Organized representation of the data filtered on yearly basis	Conferences Organized	Conferences Organized data are viewable and accurate.	Onclick function is working	Pass
4.	Users can view the accurate results of the data collected from the database.	Conferences Organized Data	Accurate results are displayed.	Results matches with database	Pass

<b>Test Case ID</b>	TC009_2			
<b>Test Case Description</b>	Verify that user is unable to view results.	<b>Test Priority</b>	Medium	
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Conferences Organized section	-	Conferences Organized is displayed	Section is visible on sidebar	Pass
2	User clicks on the Conferences Organized button.	-	Button works properly and user is taken to Conferences Organized metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	Conferences Organized Data	Bar Graph is not displaying a table onclick	Onclick function is not working, and error message is shown	Pass

<b>Test Case ID</b>	TC008_3		
<b>Test Case Description</b>	Verify that admin can do CRUD Operations on Conferences Organized	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Admin Logs in to the system		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Managing Section after logging in	-	Management Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Conference Management button.	-	Button works properly and user is taken to Management page.	Button and page are working	Pass
3.	User Adds a new data of the project	Title: "asdad", Data: "22 <sup>nd</sup> October,2022", Year: 2025, Venue: "Johor Bahru", Note: "Conference"	Data Added Successfully	Data is added and displayed in the table	Pass
4.	User Updates data of a project	Year: "2021"	Data Updated Successfully	Data is updated and displayed in the table	Pass
5.	User Deletes a row	-	Data Deleted Successfully	Data is deleted successfully	Pass
6.	Data is saved in the database successfully after every operation	Conferences Organized Data	Database updated Successfully	Database is updated according to the table	Pass

### 3.10 Test TC010 for Module Commercialization: Generate IP Filed Data (UC010)

<b>Test Case ID</b>	TC010_1		
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Commercialization Section	-	Commercialization Section is displayed on sidebar	Section is visible on sidebar	Pass
2	User clicks on the IP Filed Info button.	-	Button works properly and user is taken to Ip Filed metrics page.	Button and page are working	Pass
3.	User can view the generated IP Filed data from the database	IP Filed Data	Data has been successfully retrieved from database.	Bar Graph is displayed with data from database	Pass

<b>Test Case ID</b>	TC010_2		
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. User Logs in to the system		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Commercialization section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the IP Filed Info button.	-	Button works properly and user is taken to IP Filed Info page.	Button and page are working	Pass

3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	On click function is working	Pass
4.	User can view the accurate data from the database	IP Filed Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC010_3			
<b>Test Case Description</b>	Verify that system is not displaying accurate data		<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Commercialization section	-	Commercialization Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the IP Filed metrics button.	-	Button works properly and user is taken to IP Filed metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	IP Filed Data	Bar Graph is not displaying a table onclick	On click function is not working, and error message is shown	Pass

### 3.11 Test TC011 for Module Networking: Generate & Track Networking (UC011)

<b>Test Case ID</b>	TC011_1			
<b>Test Case Description</b>	Able to view accurate data		<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is secured			

**Test execution steps:**

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1	User clicks on the Networking button.	-	Button works properly and user is taken to Networking metrics page.	Button and page are displayed properly	Pass
2	User clicks on the networking graphical representation of the data.	Networking Data	Networking data are viewable and accurate.	Data is visible and onclick a table is displayed	Pass
3.	Users can view the accurate results of the data collected from the database.	Networking related Data	Accurate results from database.	Data is visible and accurate results displayed	Pass

<b>Test Case ID</b>	TC011_2			
<b>Test Case Description</b>	Verify that user is unable to view accurate results.	<b>Test Priority</b>		Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured			

**Test execution steps:**

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Networking section	-	Networking is displayed	Section is visible on sidebar	Pass
2	User clicks on the Networking Organized button.	-	Button works properly and user is taken to Networking metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	Networking Data	Bar Graph is not displaying a table onclick	Onlick function is not working, and error message is shown	Pass

<b>Test Case ID</b>	TC008_3		
<b>Test Case Description</b>	Verify that admin can do CRUD Operations on Conferences Organized	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Admin Logs in to the system		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Managing Section after logging in	-	Management Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Networking Management button.	-	Button works properly and user is taken to Management page.	Button and page are working	Pass
3.	User Adds a new data of the project	Title: "asdad", Data: "22 <sup>nd</sup> October,2022", StartYear: "16 march 2025", EndYear:"15 march 2027", PIC: "Dr. Adib", Type: "MoU"	Data Added Successfully	Data is added and displayed in the table	Pass
4.	User Updates data of a project	Year: "2021"	Data Updated Successfully	Data is updated and displayed in the table	Pass
5.	User Deletes a row	-	Data Deleted Successfully	Data is deleted successfully	Pass
6.	Data is saved in the database successfully after every operation	Networking Data	Database updated Successfully	Database is updated according to the table	Pass

**3.12 Test TC012 for Module Networking: Generate MoU/LoA/MoA  
(UC012)**

<b>Test Case ID</b>	TC012_1		
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Networking Section	-	Networking Section is displayed on sidebar	Section is visible on sidebar	Pass
2	User clicks on the Networking button.	-	Button works properly and user is taken to Networking metrics page.	Button and page are working	Pass
3.	User can view the generated Networking data from the database	Networking Data	Data has been successfully retrieved from database.	Bar Graph is displayed with data from database	Pass

<b>Test Case ID</b>	TC012_2		
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail

1.	User goes to the Networking section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Networking Info button.	-	Button works properly and user is taken to Networking Info page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	Onclick function is working	Pass
4.	User can view the accurate data from the database	Networking Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC012_3			
<b>Test Case Description</b>	Verify that system is not displaying accurate data	<b>Test Priority</b>		High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Networking section	-	Networking Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Networking metrics button.	-	Button works properly and user is taken to Networking metrics page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	Networking Data	Bar Graph is not displaying a table onclick	Onlick function is not working, and error message is shown	Pass

### 3.13 Test TC013 for Module Grant: Track Grant (UC013)

<b>Test Case ID</b>	TC013_1		
<b>Test Case Description</b>	Able to view accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured		

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1	User clicks on the Grants button.	-	Button works properly and user is taken to Grants metrics page.	Button and page is displayed properly	Pass
2	User clicks on the National Grant graphical representation of the data.	National Grant	National Grant data are viewable and accurate.	Data is visible and onclick a table is displayed	Pass
3.	User clicks on the International Grant graphical representation of the data.	International Grant	International Grant data are viewable and accurate.	Data is visible and onclick a table is displayed	Pass
4.	User clicks on the no. of industry Grants in the form of graphical representation of the data.	Industry Grants	Industry Grants Data are viewable and accurate.	Data is visible and onclick a table is displayed	Pass
5.	Users can view the accurate results of the data collected from the database.	Grants related Data	Accurate results from database.	Data is visible and accurate results displayed	Pass

<b>Test Case ID</b>	TC013_2		
<b>Test Case Description</b>	Verify that user is unable to view accurate results.	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. User Logs in to the system		

Test execution steps:

No.	Action	Test data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the homepage	-	Browser displays the system	Page is Displayed	Pass
2	User clicks on the Grants Data button.	-	Button works properly and user is taken to grants metrics page.	Button and page are working	Pass
3.	Users can view the results of the data collected from database.	Grants Data	Database connection is working	Connection completed	Pass

<b>Test Case ID</b>	TC013_3		
<b>Test Case Description</b>	Verify that system is displaying valid table of data onclick	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection Secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the homepage	-	Systems displays homepage	Page is Displayed	Pass
2	User clicks on the Grants metrics button.	-	Button works properly and user is taken to Grants metrics page.	Button and page are working	Pass
3.	Clicking on any Bar Graph related to Grants data	Grants Data	Bar Graph displays a table onclick	Table is displayed	Pass
4.	Bar graph is displaying valid data.	Grants Data	Valid data are displayed.	Data matches with database and is thus valid	Pass

**3.14 Test TC014 for Module Grant: Generate International Grant (UC014)**

<b>Test Case ID</b>	TC014_1		
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Grants section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Grants Data button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	User can view the generated data from the database	International Grants Data	Data has been successfully retrieved from database.	Data displayed in the form of a bargraph	Pass

<b>Test Case ID</b>	TC014_2		
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>	Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail

1.	User goes to the Grants section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Grants button.	-	Button works properly and user is taken to Grants page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	Onclk function is working	Pass
4.	User can view the accurate data from the database	International Grants Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC014_3		
<b>Test Case Description</b>	Verify that system is not displaying accurate data	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database Connection is secured		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Grants section	-	Grants Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Grants button.	-	Button works properly and user is taken to Grants page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is not displaying a table onclick	Onclk function is not working, and error message is shown	Pass

### 3.15 Test TC015 for Module Grant: Generate National Grant (UC015)

<b>Test Case ID</b>	TC015_1			
<b>Test Case Description</b>	Verify that system is successfully retrieving data from database	<b>Test Priority</b>		Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User goes to the Grants section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Grants Data button.	-	Button works properly and user is taken to publication metrics page.	Button and page are working	Pass
3.	User can view the generated data from the database	National Grants Data	Data has been successfully retrieved from database.	Data displayed in the form of a bargraph	Pass

<b>Test Case ID</b>	TC015_2			
<b>Test Case Description</b>	Verify that system is displaying accurate data	<b>Test Priority</b>		Medium
<b>Pre-Requisite</b>	1. There is an active internet connection 2. Database connection is secured			

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
-----	--------	------------	-----------------	---------------	-----------

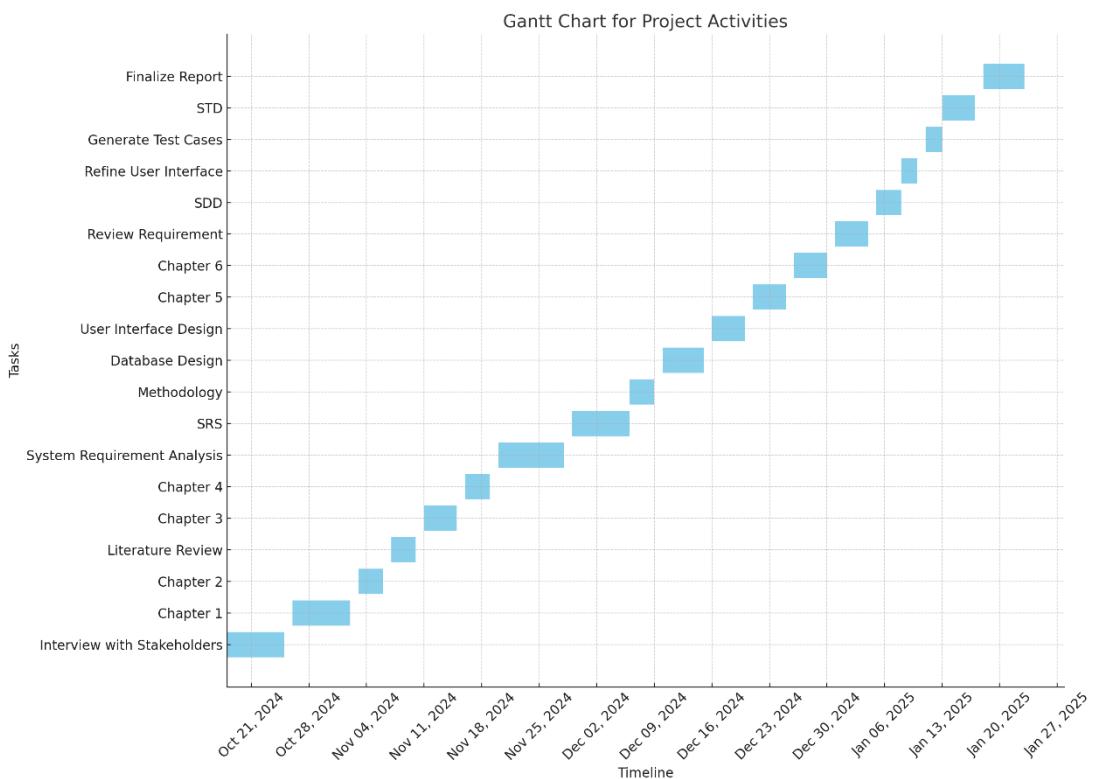
1.	User goes to the Grants section	-	Section is on the sidebar and visible	Section is visible on sidebar	Pass
2	User clicks on the Grants button.	-	Button works properly and user is taken to Grants page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is displaying a table onclick	Onclick function is working	Pass
4.	User can view the accurate data from the database	National Grants Data	Accurate data has been displayed.	Accurate Data displayed in the form of a table	Pass

<b>Test Case ID</b>	TC015_3		
<b>Test Case Description</b>	Verify that system is not displaying accurate data	<b>Test Priority</b>	High
<b>Pre-Requisite</b>	1. There is an active internet connection		

Test execution steps:

No.	Action	Input data	Expected result	Actual Result	Pass/Fail
1.	User is taken to the Grants section	-	Grants Section is displayed	Section is visible on sidebar	Pass
2	User clicks on the Grants button.	-	Button works properly and user is taken to Grants page.	Button and page are working	Pass
3.	User clicks on the bar graph for displaying the data in the form of a table	-	Bar Graph is not displaying a table onclick	Onclick function is not working, and error message is shown	Pass

## Appendix D Gantt Chart



## **Appendix:**

### **List of Prompts Used**

## List of Prompts:

1. <https://chatgpt.com/c/6791d86a-8774-8012-b263-2df45d5541df>

The screenshot shows a dark-themed interface for ChatGPT 4.0. At the top, it says "ChatGPT 4.0". A user message bubble contains the question: "What is web scraping and how can it be utilized for my web-based dashboard system?". Below the message, an AI response starts with "Web scraping is the automated process of extracting data from websites. It involves using scripts or tools to fetch and parse web pages, then extract specific information for analysis, storage, or further processing." This is followed by a section titled "How Web Scraping Works:" which lists four steps: 1. Sending Requests, 2. Parsing the HTML, 3. Data Extraction, and 4. Storing the Data. Another section titled "Utilization in Your Web-Based Dashboard System:" follows. At the bottom of the interface, there's a message input field, a footer note "ChatGPT can make mistakes. Check important info.", and a help icon.

2. <https://claude.ai/chat/409247c9-c69f-4385-bd35-cbe4fd96a802>

The screenshot shows a dark-themed interface for Claude. At the top, it says "Claude". A user message bubble contains the question: "I'm good in javascript, which library can help me to scrap data?". Below the message, an AI response starts with "For web scraping in JavaScript, three popular libraries are:". It then lists three libraries: 1. Puppeteer, 2. Axios, and 3. Cheerio. The response concludes with "Each library serves slightly different scraping needs, so choose based on your specific requirements.". At the bottom of the interface, there's a message input field, a footer note "Claude can make mistakes. Please double-check responses.", and a help icon.

3. <https://chatgpt.com/c/6791d86a-8774-8012-b263-2df45d5541df>

ChatGPT 4o ▾

I'm planning to use puppeteer for my project, can you guide me for the environment setup of puppeteer?

Sure! Setting up Puppeteer for your project is straightforward. Here's a step-by-step guide to get your environment ready:

---

**1. Install Node.js**

Puppeteer requires Node.js. Make sure you have it installed:

- Download Node.js: <https://nodejs.org/>
- Install the latest LTS version (recommended for stability).

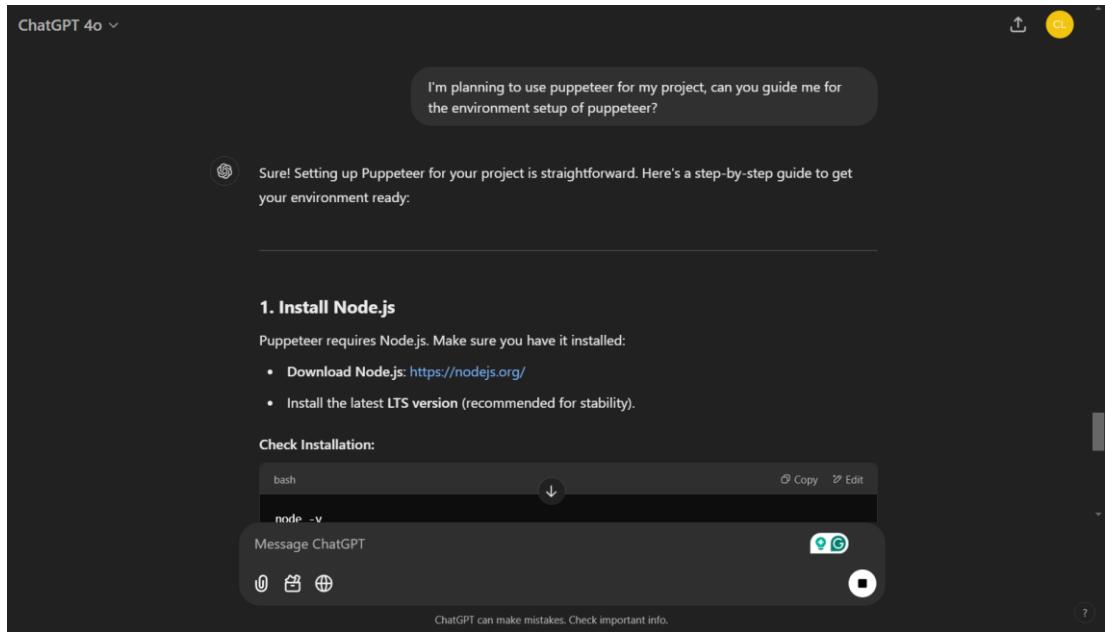
Check Installation:

```
bash
↓
node -v
Message ChatGPT
```

Copy Edit

ChatGPT can make mistakes. Check important info.

?



No.	Function	Description	Performed By
1	Scraping the data from the website	This function ensures the scraping of the desired data	Admin

**Table 1: User Acceptance Testing**

No.	Actions	Expected Results	Pass/Fail	Comment
1.	Admin logs in	Login is successful	Pass	
2.	Admin clicks on the start scraping button	Button works fine	Pass	
3.	Admin clicks on the start scraping button	Scraping is starting and ends with a successfully completed message	Pass	

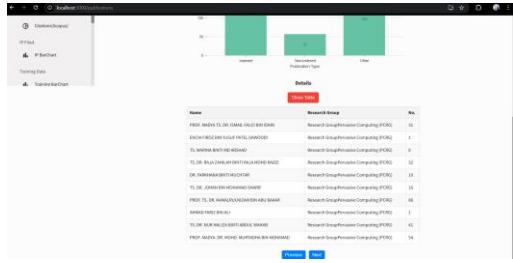
No.	Function	Description	Performed By
1	Manage Networking data	This function ensures the successful management of networking related data	Admin

**Table 2: User Acceptance Testing**

No.	Actions	Expected Results	Pass/Fail	Comment
1.	Admin logs in	Login is successful	Pass	
2.	Admin clicks on the networking data button	Button works fine	Pass	
3.	Admin tries to create/update/delete data	Data can be added, updated and deleted successfully	Pass	

No.	Function	Description	Performed By
1	Dasj	This function ensures the successful management of networking related data	Admin

**Table 3: User Acceptance Testing**

No.	Actions	Expected Results	Pass/Fail	Comment
1.	User Enters the system	Dashboard page is displayed	Pass	
2.	User clicks on any bar chart from sidebar to verify if it's analytical or not.	Table will be generated 	Pass	
3.	User can close the table	Button works perfectly	Pass	

**Signature:**



**ASSOC. PROF. DR. SITI ZAITON BINTI MOHD HASHIM**  
Deputy Dean (Research, Innovation & Development)  
Faculty of Computing  
UTM Johor Bahru

