



SCSJ2203: SOFTWARE ENGINEERING

SYSTEM DOCUMENTATION

REPAIR AND MAINTENANCE SYSTEM FOR KTDI STUDENTS

VERSION 1.0

20 MAY 2020

SCHOOL OF COMPUTING, FACULTY OF ENGINEERING

Prepared by: SAVVY TECH

Revision Page

a. Overview

Repair and Maintenance for KTDI Student version 1.0 describes the purpose and the scope of the system in the introduction section. The second section for the current system documentation is specific requirement. In this section, it explains external interface requirements, which includes user interfaces, hardware interfaces, software interfaces and communication interfaces. This section describes the system features such as use case diagram, domain model of the system and state machine diagram for Repair and Maintenance for KTDI Student system. This section also represents the performance requirements, design constraints, software system attributes and other requirements of the system that have been discussed together with group members.

b. Target Audience

Target audience for this system is mostly for student and college management.

c. Project Team Members

Team Members	Roles	Status
Muhammad Amirul Fahmi bin Noor Anim (B19EC0018)	2.1 External Interface 2.2 System Features 3. Detailed Description 4. Requirements Matrix	Completed
Mohd Anas bin Adnan (B19EC0043)	1. Introduction 2.2 System Features 3. Detailed Description	Completed
Nurul Najiha binti Hamdan (B19EC0047)	2.2 System Features 2.3 Performance Requirements 2.4 Design Constraints	Completed
Nur Hasanah bin Sariddon (B19EC0033)	2.2 System Features 2.5 Software System 2.6 Other Requirements 3. Detailed Requirements	Completed

d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
Version 1.0	Nurul Najiha binti Hamdan	Combines the part from <1. Introduction> until <Detailed Description of components: 3.1 Complete Package Diagram> Refines and compile the report.	20/05/2020
	Mohd Anas bin Adnan	Continue combining the part starting from <3.2 Detailed Description> until <4. Requirement Matrix>.	20/05/2020

Table of Contents

1	Introduction			
	1.1	Purpose		6
	1.2	Scope		6
	1.3	Definitions, Acronyms and Abbreviations		7
	1.4	References		8
	1.5	Overview		8
2	Specific Requirements			
	2.1	External Interface Requirements		9
		2.1.1	User Interfaces	9
		2.1.2	Hardware Interfaces	9
		2.1.3	Software Interfaces	9
		2.1.4	Communication Interfaces	10
	2.2	System Features		10 - 12
		2.2.1	UC001: Use Case <Login Requirement>	13 - 14
		2.2.2	UC002: Use Case <Manage Complaint>	14 - 18
		2.2.3	UC003: Use Case <Receive Respond>	18 - 19
		2.2.4	UC004: Use Case <Generate Report>	20 – 21
		2.2.5	UC005: Use Case <Manage Schedule>	21 – 22
	2.3	Performance Requirements		23
	2.4	Design Constraints		23
	2.5	Software System Attributes		24
	2.6	Other Requirements		24
3	Detailed Description of Components			
	3.1	Complete Package Diagram		25
	3.2	Detailed Descriptions		25
		3.2.1	P001: <Login> Subsystem	26
			3.2.1.1 Class Diagram	26

			3.2.1.2	Sequence Diagram	27
		3.2.2	P002: <Manage Complaint> Subsystem		28
			3.2.2.1	Class Diagram	29
			3.2.2.2	Sequence Diagram	30 - 31
		3.2.3	P003: <Manage Schedule> Subsystem		32
			3.2.3.1	Class Diagram	32 - 33
			3.2.3.2	Sequence Diagrams	34
		3.2.4	P004: <Respond> Subsystem		35
			3.2.4.1	Class Diagram	35
			3.2.4.2	Sequence Diagram	36
		3.2.5	P005: <Generate Report> Subsystem		37
			3.2.5.1	Class Diagram	37
			3.2.5.2	Sequence Diagram	38
4	Requirements Matrix				
	Appendices				39
	Appendix A: Traceability Matrix				39

1. Introduction.

1.1 Purpose

The System Documentation describes the purposed of Repair and Maintenance system which conceal the specific requirement which include external interface requirement, system features, performance requirement, design constraint, software system attributes and others requirement. The document represents the system architectural design that explain architectural style and rationale, and component model. It covers the description of component such as complete package diagram and detailed description of the subsystems. The data design will created including data description and data dictionary that keep information about of Kolej Tun Dr Ismail. Overview of student interface is complaint form of any failure of facilities of Kolej Tun Dr Ismail. The purpose of this documentation is to be reviewed in the purpose system by the stakeholders. Besides that, its purpose is also to let stakeholder understand more about the functionalities of the system so that they can decide either they approve the system or not and to provide the necessary information for showing and monitoring evidence of progress in the development process.

1.2 Scope

The software product is a Repair and Maintenance of Kolej Tun Dr Ismail which mainly has a complaint form of any failure facilities. The technical meaning of maintenance involves function checks, servicing, repairing or replacing of necessary building infrastructure.[1] The student need to log in to the system first before submit any form of complaint. The first time user for this system need to sign up to the system by email address. The system will automatic verified a valid email address for the student to proceed to the complaint form. The student need to complete the complaint form and the complaint form will be place in the database system. The student will receive the successful email of their complaint and get the estimated time for the technicians to repair the failure facilities. The student will also get notification by email when the technicians will come to do maintenance. There is a step by step guide that user can follow as a crucial process for them to use the system.

The scope of the system includes:

- Application platform
 - ❖ For student's registration.
 - ❖ For student's complaint form page.
 - ❖ For staff's complaint form page.
- Application management
 - ❖ The system automatically validates the first time student's registration.
 - ❖ The system classified complaint form with different facilities failure.
 - ❖ The system will send notification for estimated time by email.
- Monitoring application progress
 - ❖ Staff track how much failure facilities.
 - ❖ Staff track which block need to repair and maintain.
- Generating an overall complaint
 - ❖ For the student to view their complaint status.
 - ❖ For the staff to view overall complaint status.

This repair and maintenance system is an online application that can be accessed by the website which provides a world wide application for the student to access. The goal of the system is to create a platform for the student to do complaint. Meanwhile, the objective is to make student and staff jobs easier and saving a lot of time to do a complaint.

1.3 Definitions, Acronyms and Abbreviation

Definitions of all terms, acronyms and abbreviation used to be defined here

Definition/ Acronyms/ Abbreviation	Meaning
Mnemonic	A system such as a pattern of letters, ideas, or associations which assists in remembering something.
GUI User Interface	A user interface that allows users to interact with electronic devices through graphical icons and audio indicators such as keynotes, rather than text-based user interfaces, typed command labels, or text navigation.

Encrypted	A process of converting information or data into a code, especially to prevent unauthorized access.
Workload	The amount of work performed by an entity in a given period of time, or the average amount of work handled by an entity at a particular instant of time.
Throughput	Performance of tasks by a computing service or device over a specific period.

1.4 References

Specify complete list of references using a standardized reference format.

[1]"Maintenance (technical)," Wikipedia, [Online]. Available: [http://en.m.wikipedia.org/wiki/Maintenance_\(technical\)](http://en.m.wikipedia.org/wiki/Maintenance_(technical)) [Accessed 11 May 2020]

1.5 Overview

The remaining section of this document provide an overall description, including product's software, the product's hardware and the characteristic of the system, constraints user interfaces and the functional and data requirements of the product. The overall description of the system will be discussed in the second section. In the third section, this document will discuss the external interface requirements of the system, system features, performance requirements, design constraints, software system attributes and the other requirements of the proposed system. In the fifth section includes data design while the sixth section includes interface design and later section is requirements matrix and the last section is about test cases.

2. Specific Requirements

2.1 External Interface Requirements

2.1.1 User Interfaces

The main objective of this design is to create minimalist yet informative interface design. So, it is user – friendly and easy for students and college management to use the system.

- A user should see a log in page displayed when opening the system.
- A pop-up notification shall occur if the user enters incorrect ID or password.
- The system shall present each type of user (college staff, student) with respective GUI interface.
- The system shall collect the user inputs data through touchscreen, keyboard, or mouse.
- The icon for menu shall be located consistently on upper left of the screen that contains several different options based on each type of user (college staff, student).
- The student can view their complaint within 0.12 seconds.
- The college staff can generate report on complaint within 0.18 seconds.
- A pop-up ‘Undo’ notification shall be displayed when the user accidentally deleted a complaint within 10 seconds.

2.1.2 Hardware Interfaces

- The system supports multiple devices that are commonly used by user – Smartphone, Laptop, and PC.
- The system is supported in multiple web browsers that is Google Chrome, Microsoft Edge, and Mozilla Firefox.
- The system uses a database which is located on a network server. The user’s device transfers and receive data from the server using basic networking protocols. All system’s information is stored in server’s database.

2.1.3 Software Interfaces

The web-based system communicates with web server to execute its functionalities. The communication between the database and the web-based server consists of operation concerning both reading and modifying the data. The system required a web browser either on computer or mobile device such as Windows.

- Name: Windows 10
- Mnemonic: Threshold, Redstone, 19H1, YYHx
- Specification number: NT 10.0
- Version number: 19041 (version 2004), 18363 (version 1909)
- Source: https://en.wikipedia.org/wiki/List_of_Microsoft_Windows_versions (Wikipedia)

2.1.4 Communication Interfaces

This should specify the various interfaces to communications such as local network protocols, etc.

The communication in our system shall use Hypertext Transfer Protocol Secure (https) for receiving student's and college management information and encrypting all crucial information from and into our system for security of data transfer.

The system also uses TCP/IP (Transmission Control Protocol/ Internet Protocol) for interconnection of network devices on the internet. It is also used for data transmission over the Internet.

2.2 System Features

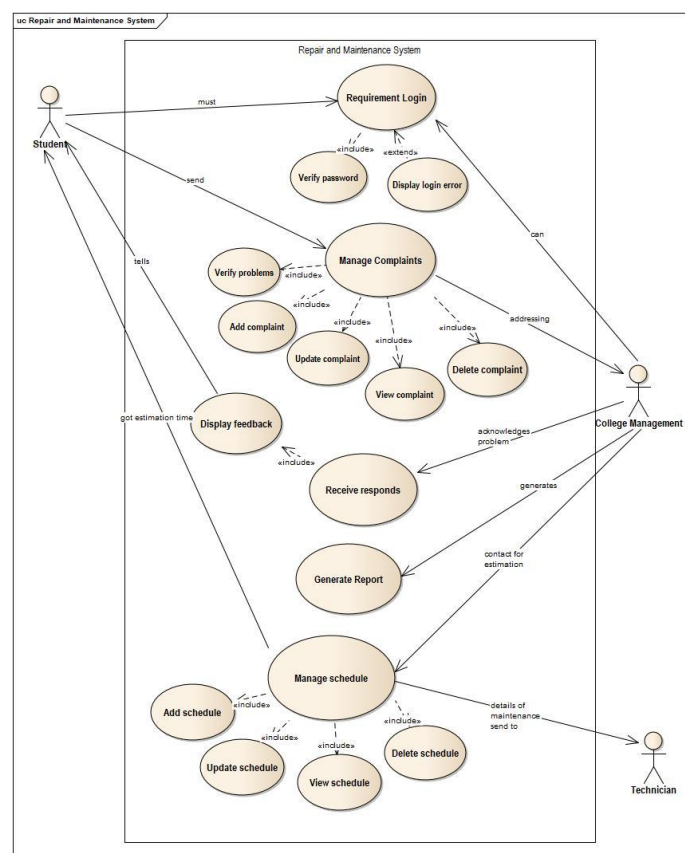


Figure 2.1: Use Case Diagram for <Repair and Maintenance for KTDI Student>

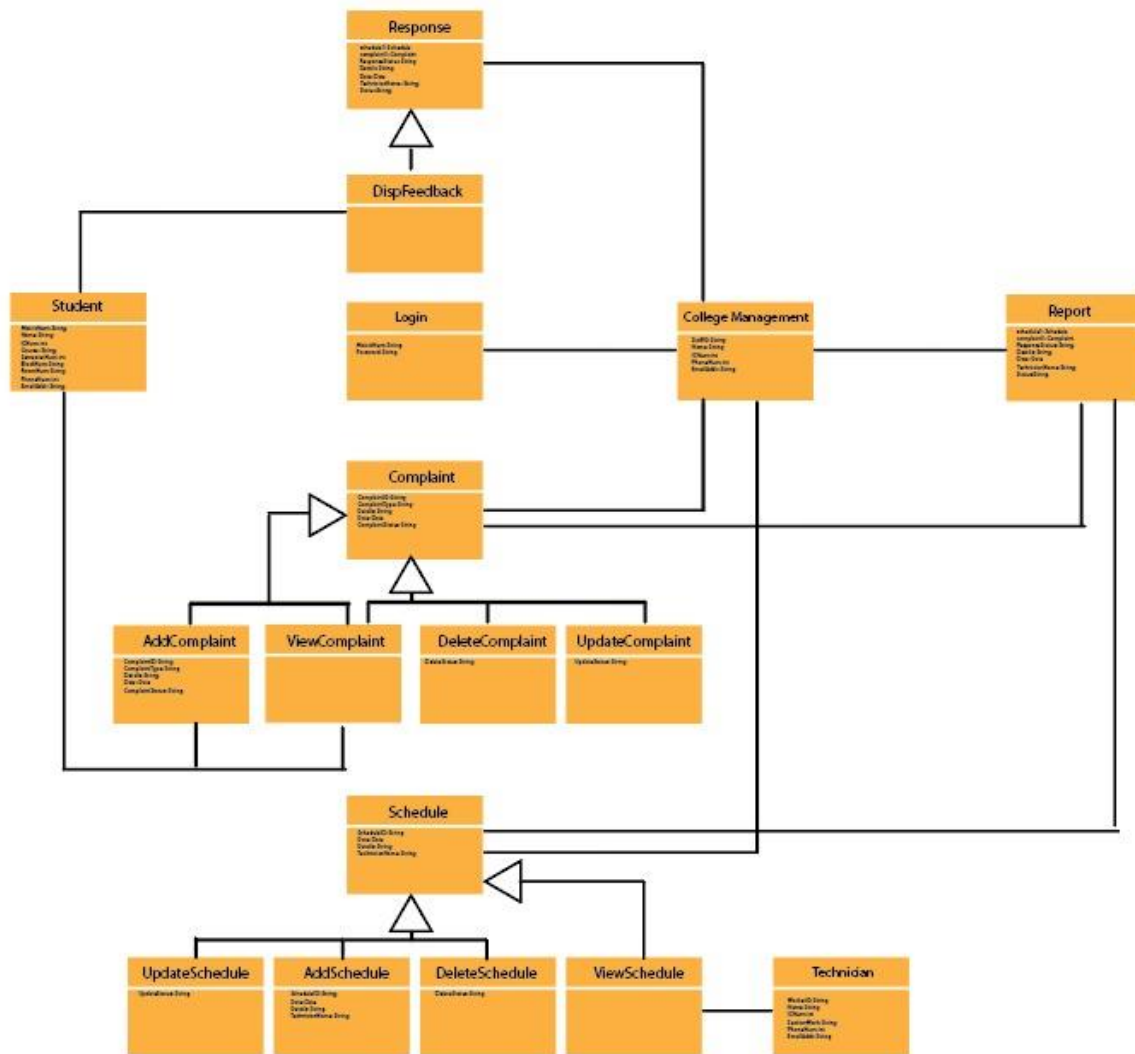


Figure 2.2: Domain Model for <Repair and Maintenance for KTDI Student>

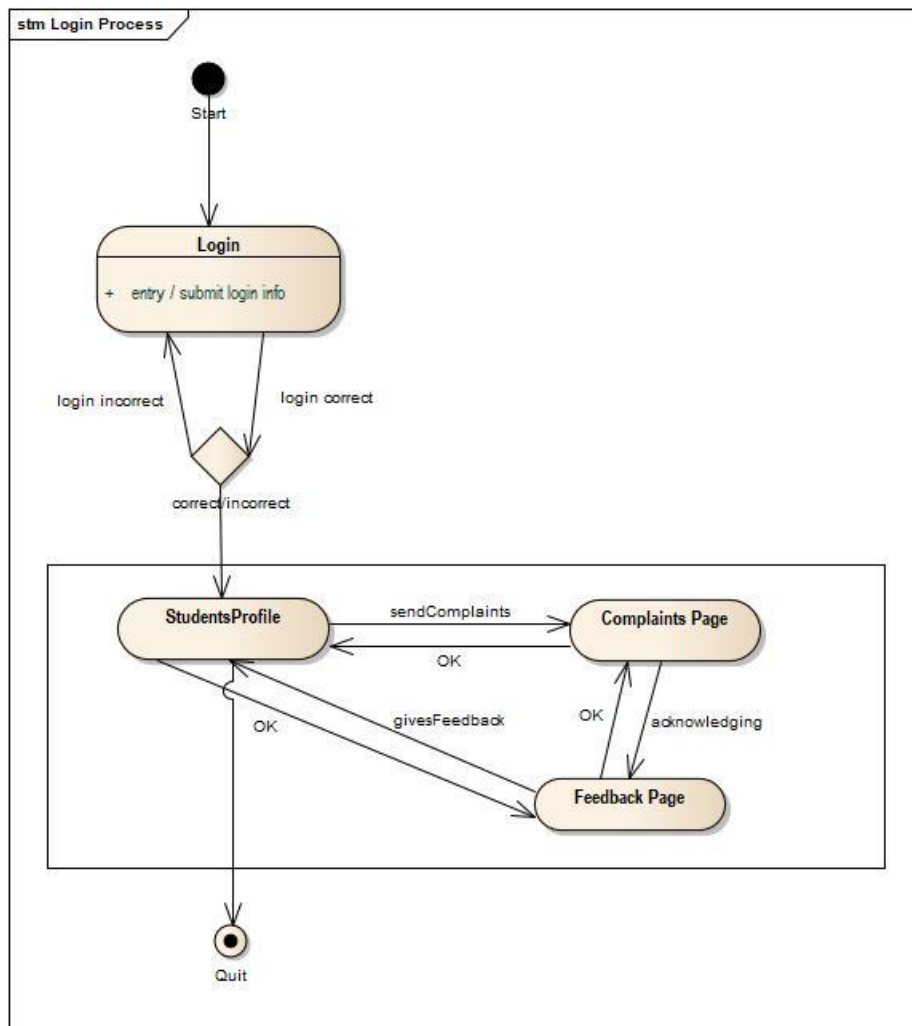


Figure 2.3: State Machine Diagram for <Login>

2.2.1 UC001: Use Case <Login Requirement>

Use Case: Requirement Login
ID: UC001
Actors: A1 Student A2 College Management
Includes: UC02 Verify Password
Extension points: UC03 Display Login Error
Preconditions: 1. Has Internet network for the user's device
Flow of events: 1. The Student/ College Staff log into using matric number/ staff ID and password into the system 2. include (Verify Password) 3. if matric number/ staff ID and password is correct 3.1 Log into the system successfully
Alternative Flow: 1. If matric number/ staff ID or password is incorrect 1.1 <u>extend</u> (Display Login Error) 1.2 Flow of events executed again
Postconditions: 1. Users can use the system. 2. The system able to save the matric number/ staff ID and password into the database
Exception Flow:

Table 2.1: Use Case Description for <Login>

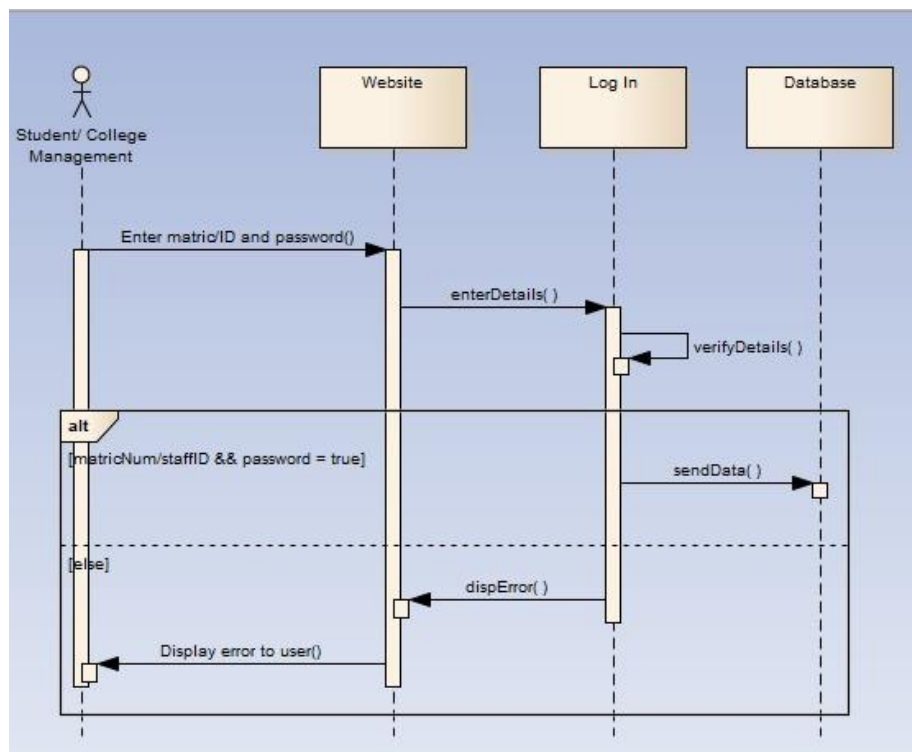


Figure 2.4: Sequence Diagram for <Login>

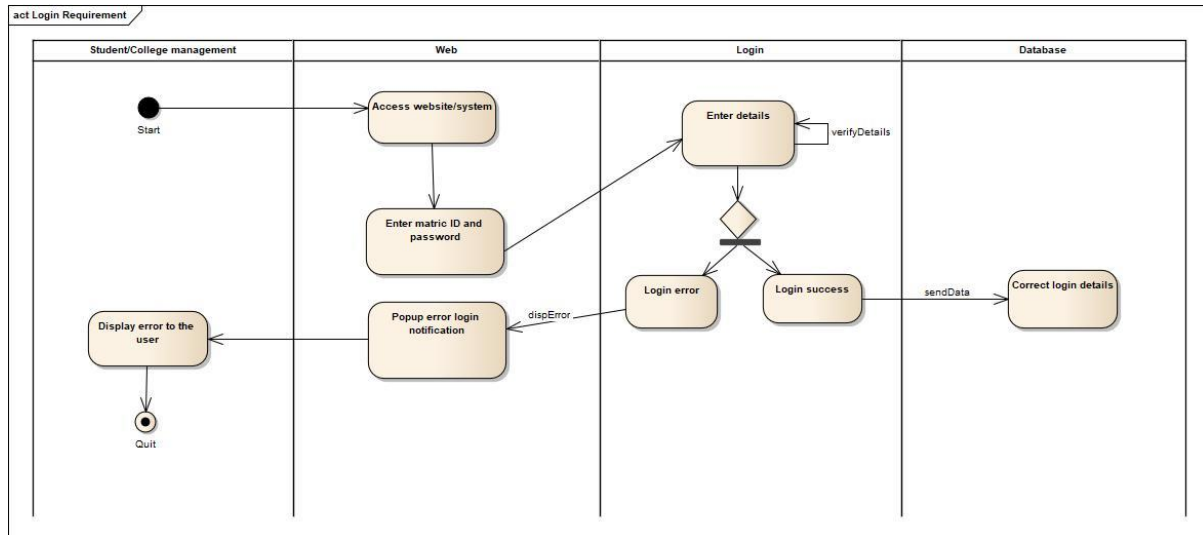


Figure 2.5: Activity Diagram for <Login>

2.2.2 UC001: Use Case <Manage Complaints>

Use Case: Manage Complaint
ID: UC004
Actors: A1 Student A2 College Management
Includes: UC005 Verify problems UC006 Add complaint UC007 Update complaint UC008 View complaint UC009 Delete complaint
Extension points:
Preconditions: 1. User has log into the system 2. There is stable network condition on user's device
Flow of events: 1. Include (Add complaint) 2. Student can send complaint into the system 3. Include (View complaint) 4. Student can view the complaint that they had submitted 3. if matric number/ staff ID and password is correct 3.1 Log into the system successfully

Alternative Flow 1: 1. If student did not complete the complaint form 1.1 System display error to student 2. Flow of events executed again
Alternative Flow 2: 1. Include (View complaint) 2. College management can view the complaint that the students had submitted 3. Include (Verify problems) 4. College management verify the problems of students in the complaint 5. Include (Update Complaint)
6. College management can update the status of the complaint in the system 7. Include (Delete Complaint) 8. College management can delete the complaint submitted by student
Postconditions: 1. Student can submit the complaint to system 2. Student can view their complaint in the system 3. College management can manage the student's complaint
Exception Flow:

Table 2.2: Use Case Description for <Manage Complaint>

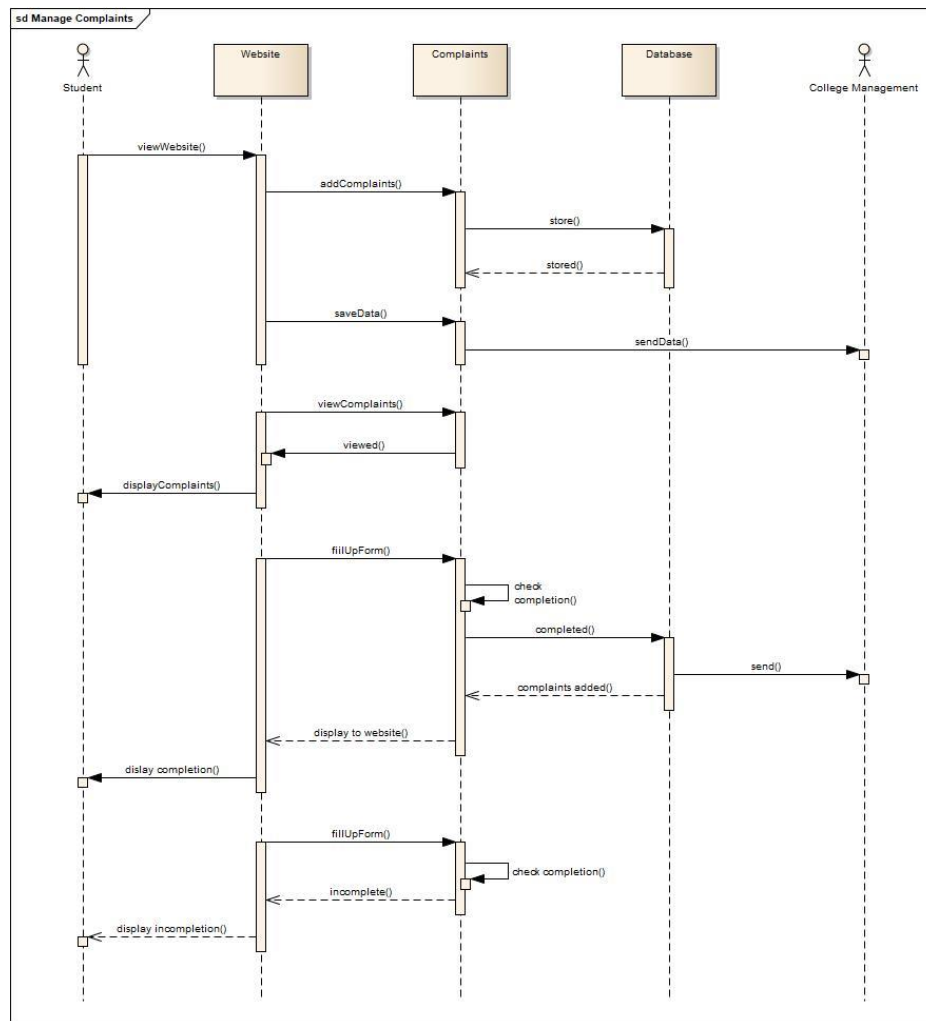


Figure 2.6: Sequence Diagram for <Manage Complaint from Student to College Management>

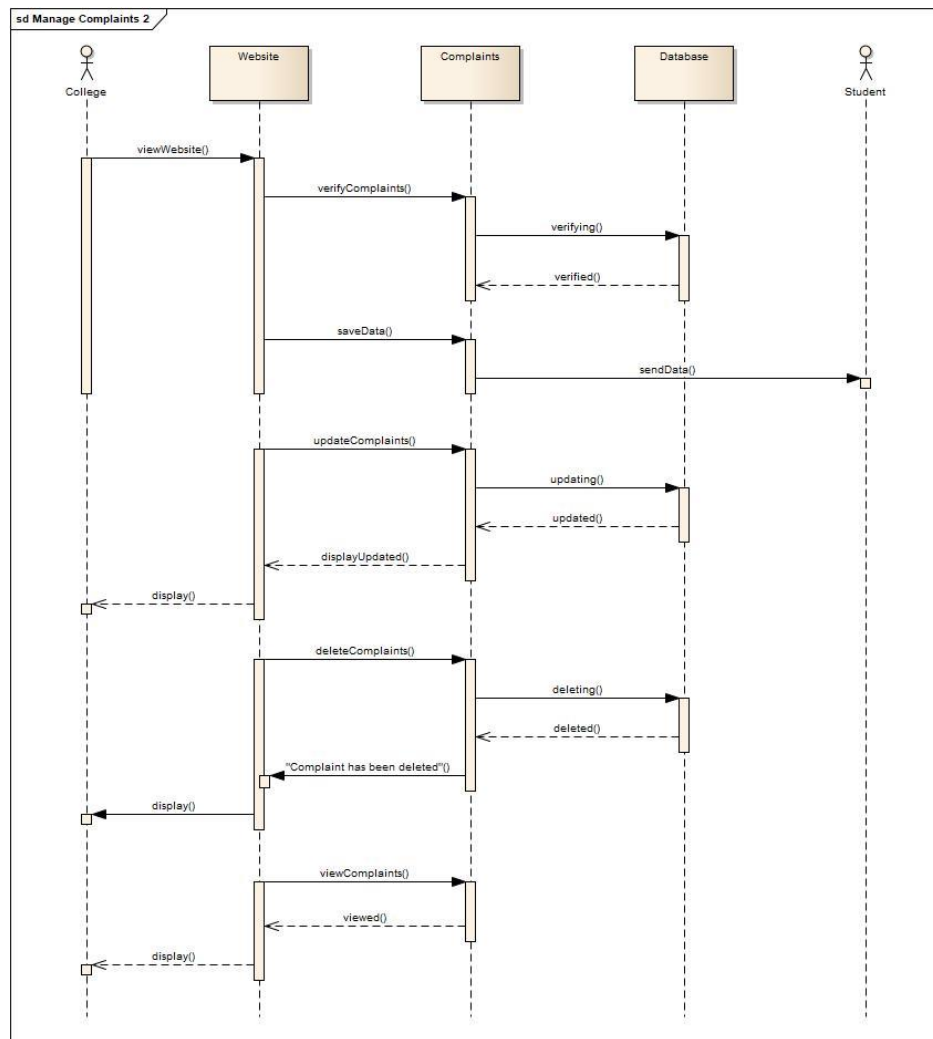


Figure 2.7: Sequence Diagram for <Manage Complaint from College Management to Student>

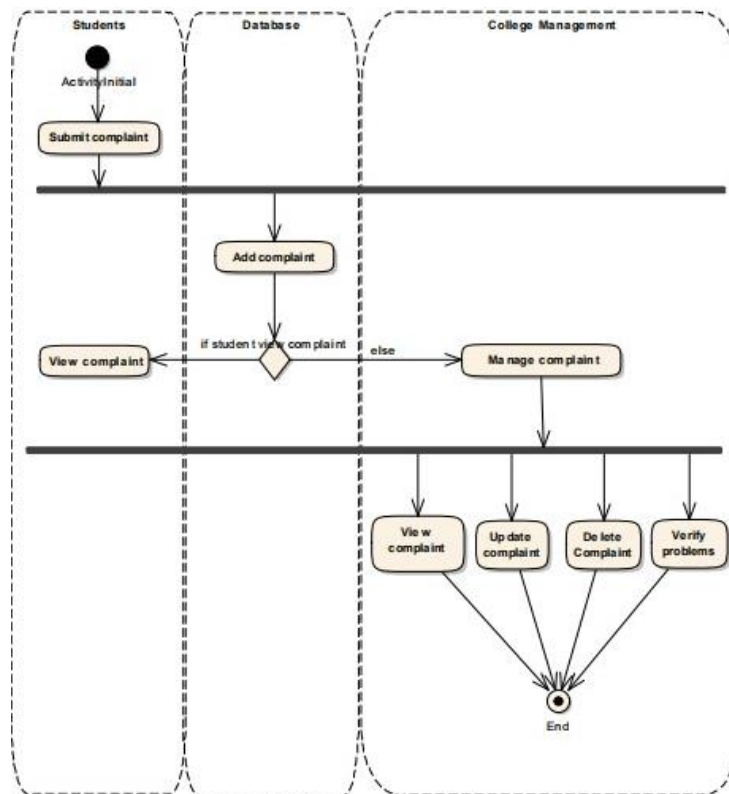


Figure 2.8: Activity Diagram for <Manage Complaint>

2.2.3 UC003: Use Case <Receive Respond>

Use Case: Receive Responds
ID: UC010
Actors: A1 Student A2 College Management
Includes: UC02 Display Feedback
Extension points:
Preconditions: 1. User has log into the system 2. Student has submitted the complaint into the system 3. College staff has verified the student's complaint
Flow of events: 1. College staff send feedback to student through the system 2. Include (Display Feedback) 3. Student view the feedback given by college staff when logging into the system
Alternative Flow:
Postconditions: 1. Student will notify about the status of their complaint 2. Student will know about feedback of their complaint
Exception Flow:

Table 2.3: Use Case Description for <Receive Respond>

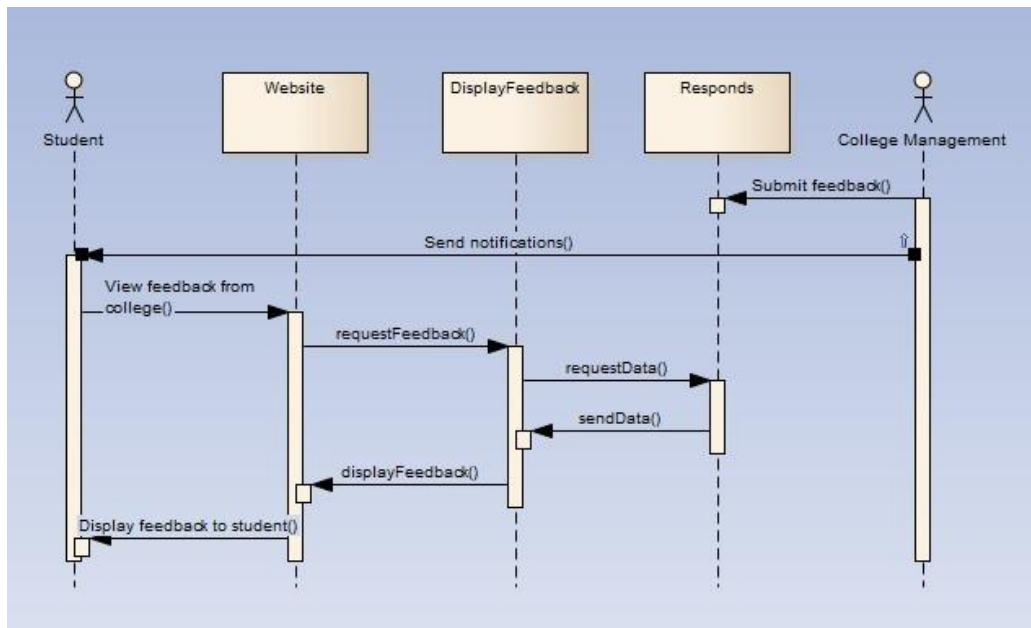


Figure 2.9: Sequence Diagram for <Receive Respond>

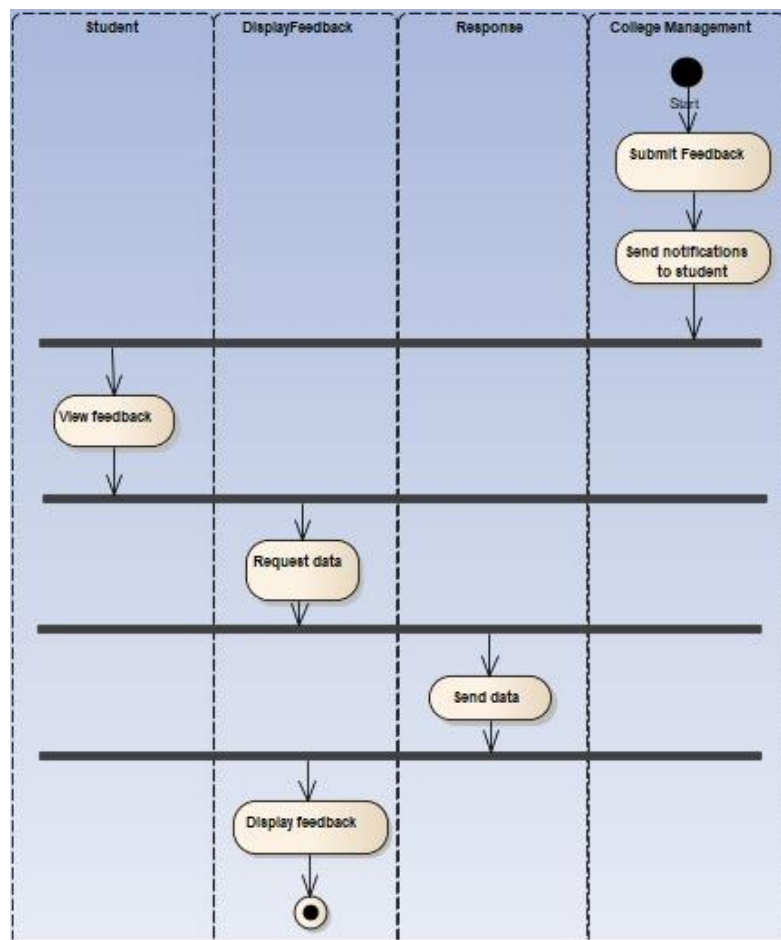


Figure 3.0: Activity Diagram for <Receive Respond>

2.2.4 UC004: Use Case <Generate Report>

Use Case: Generate Report
ID: UC004
Actors: College management
Preconditions: 1. College has receive the complaints 2. College has displayed the feedback to student
Flow of events: 1. College staff generate the report about breakdown 2. Report has been made based on complaints filed by student
Post-conditions: 1. College able to generate report following the complaints of facilities

Table 2.4: Use Case Description for <Generate Report>

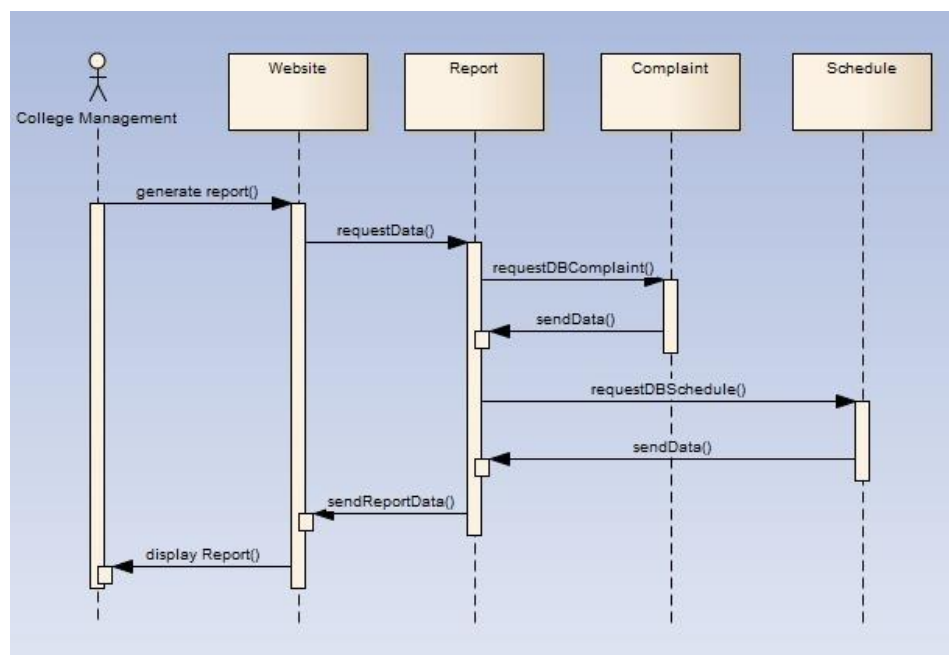


Figure 3.1: Sequence Diagram for <Generate Report>

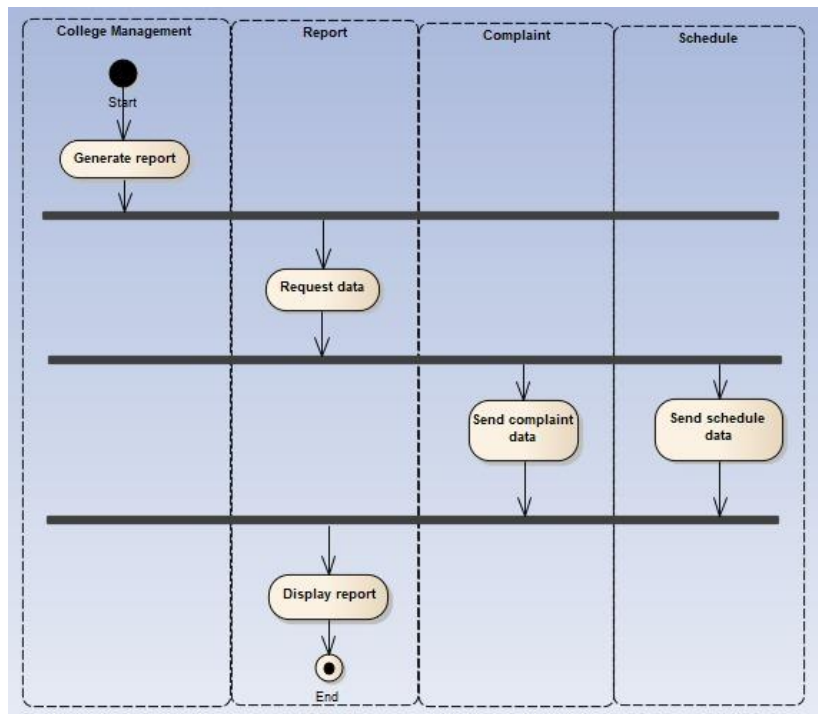


Figure 3.2: Activity Diagram for <Generate Report>

2.2.5 UC005: Use Case <Manage Schedule>

Use Case: Manage Schedule
ID: UC013
Actors: A1 Student A2 College Management A3 Technician
Includes: UC014 Add schedule UC015 Update schedule UC016 View schedule UC017 Delete schedule
Extension points:

Flow of events:

1. College can manage the schedule in the system
2. include (Add schedule)
3. College staff can add schedule to the system
4. include (Delete schedule)
5. College staff can delete schedule in the system
6. include (Update schedule)
7. College staff can update the schedule in the system
8. include (View schedule)
9. College staff can view the schedule in the system
10. Student will get estimated time of the maintenance from the system

Table 2.5: Use Case Description for <Manage Schedule>

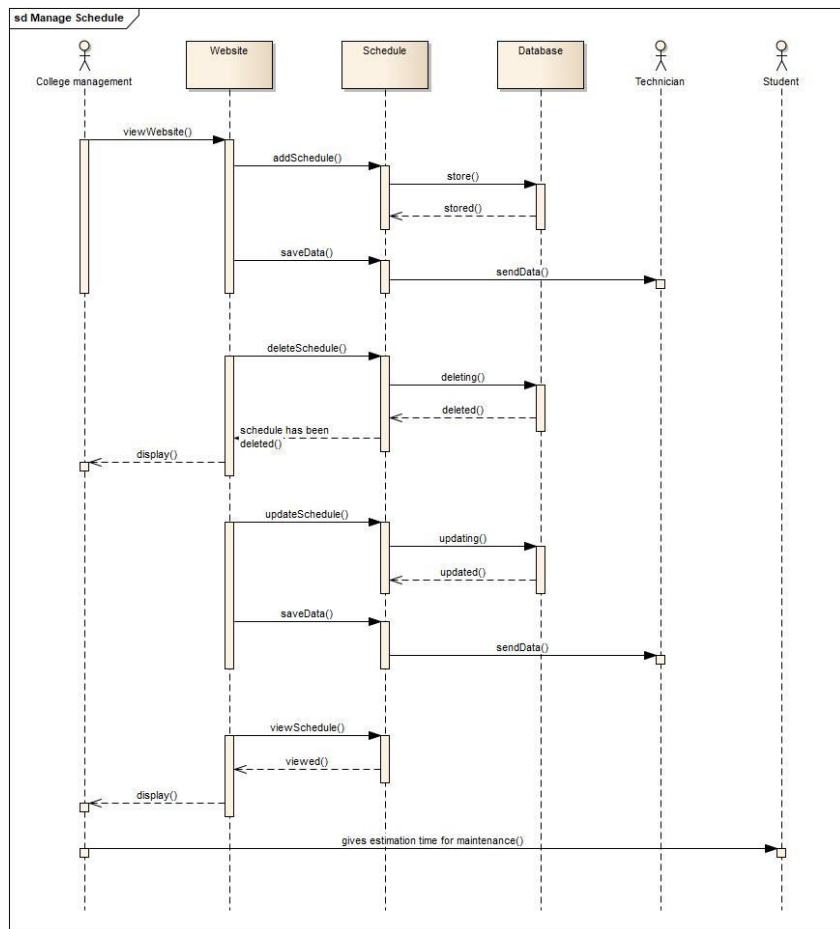


Figure 3.3: Sequence Diagram for <Manage Schedule>

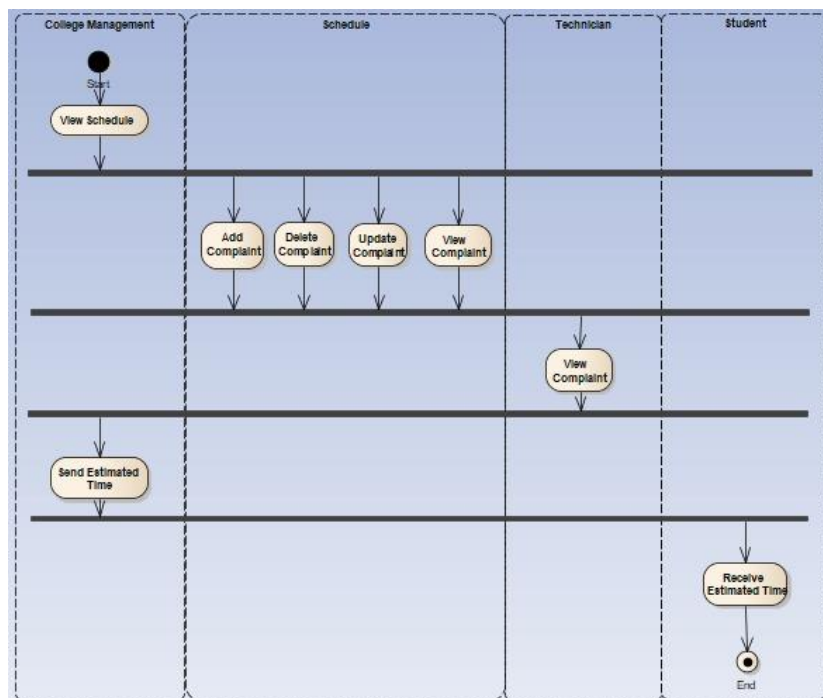


Figure 3.4: Activity Diagram for <Manage Schedule>

2.3 Performance Requirements

- i. Response time:
 - The system shall be able to show response after 1.0 second of student click on the login button to complete the login process.
 - Loading process of opening the complaint page shall be 5.0 seconds of waiting especially when the student uses the software for the first time.
- ii. Workload:
 - The system shall be able in supporting 500 KTDI students including the college staffs.
 - During in peak hours, at least 100 users can generate the interaction in the software because when the users are too much to handle, the responding time for the system would be slow.
- iii. Throughput:
 - The number of operations performed shall be 5 operations per second.
 - The system shall be able to support managing the operations performed by student such as update, add, view and delete the complaints at the same time.
- iv. Capacity:
 - The college management can view 10 complaints in one page document with scrolling up and down in 12 seconds.
 - Generating the report to the technician shall take 18 seconds with 5 complaints at one time.

2.4 Design Constraints

- i. Language requirement:
 - The system should be multilingual such Mandarin, Arabic, Hindi, Bengali, French, German, Spanish, Korean and Japanese but we can only provide English as our main language in the system.
- ii. Synchronization:
 - Browse the software using Android 7 and above, iOS 10 and above, and windows 10 and above only for more smooth and effective when using the system.
- iii. Operational:
 - Any unsuccessful login made by student or college management should display login error with popup notifications and “eye emoji” at the corner of password box to able see the password either it is in correct order or alphabets.
- iv. Emergency:
 - The system must be include fire station, ambulance, campus security, student affairs, counseling center and health care hotlines to avoid any unawareness emergency.

2.5 Software System Attributes

Usability	- The system should be user friendly or the layout or design should be easily understandable. - The system should satisfy a maximum number of users at the same times.
Maintainability	- The maintenance should maintain the system and update it at the time that users will not use it.
Availability	- The system should be available the specified time as many users want to use it.
Changeability	- The system should written in such way that can meet the growing demand of the stakeholder.

2.6 Other Requirements

Portability	- The system can be adapted to run on computers and any smartphone for which it was designed.
Efficiency	- The system shall take less time to log in to the system. - The system should not use the valuable resources of memory or the processor cycle.
Dependability	- The system should not use cause physical or economic damage.
Reliability	- Username and password enter by user shall be secured.

3. Detailed Description of Components

3.1 Complete Package Diagram

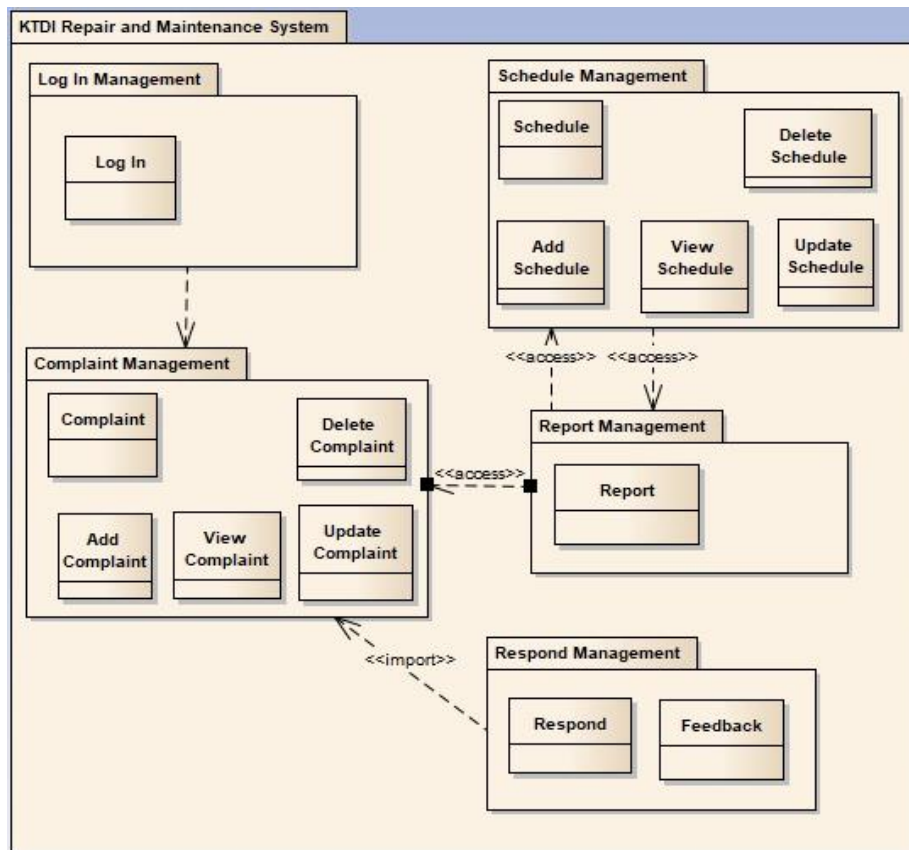


Figure 3.5: Package Diagram for <Repair and Maintenance for KTDI Student>

3.2 Detailed Description

For each subsystem/package there must be ONE class diagram and several sequence diagrams based on how many use cases you have in the subsystem/package. Use branching in sequence diagram to combine alternate flow in the same sequence diagram. If the diagram is cluttered, consider a new sequence diagram for respective scenario/alternate flow. The given example includes view, domain and data access layer in respective packages. Organise subsystem/package according to the chosen architectural style.

3.2.1 P001: <Log in> Subsystem

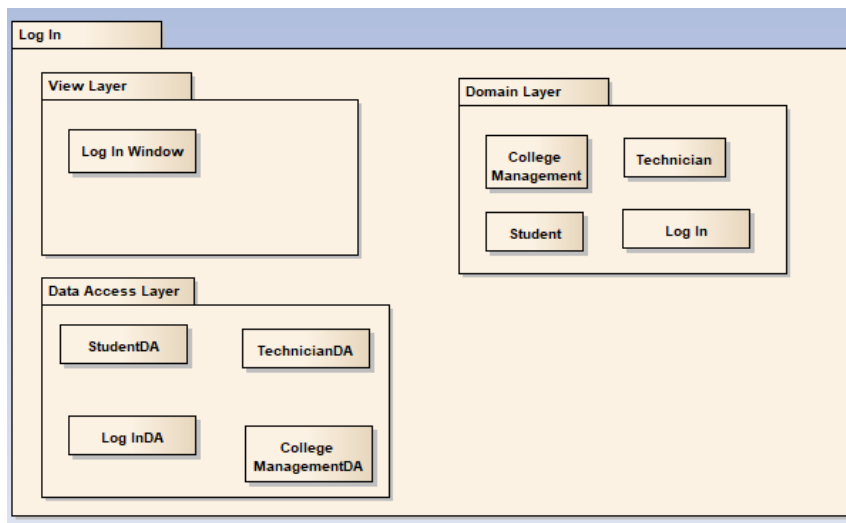


Figure 3.6: Package diagram for <Log in> Subsystem

3.2.1.1 Class Diagram

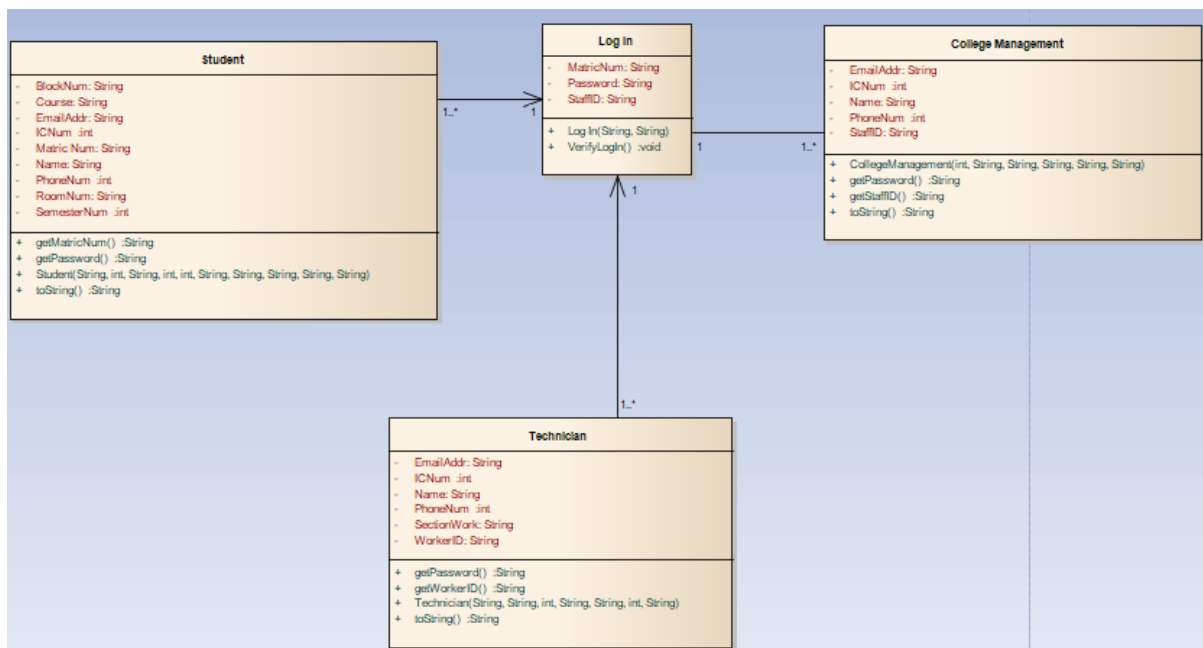


Figure 3.7: Class diagram for <Log In> Subsystem

Entity Name	Log In, College Management, Technician, Student
Method Name	VerifyLogin(), getStaffID(), getWorkerID(), getMatricNum(), getPassword()
Input	MatricNum, StaffID, WorkerID, Password
Output	Login
Algorithm	1. Start 2. If user is Student 2.1 Student enters matric number and password 3. Else if user is College Staff 3.1 College Staff enters staff id and password 4. Else if user is Technician 4.1 Technician enters worker ID and password 5. System verify user id and password using verifyLogin() method 6. If verify is passed 6.1 User log into the system 7. Else 7.1 Repeat steps 1 8. End

3.2.1.2 Sequence Diagram

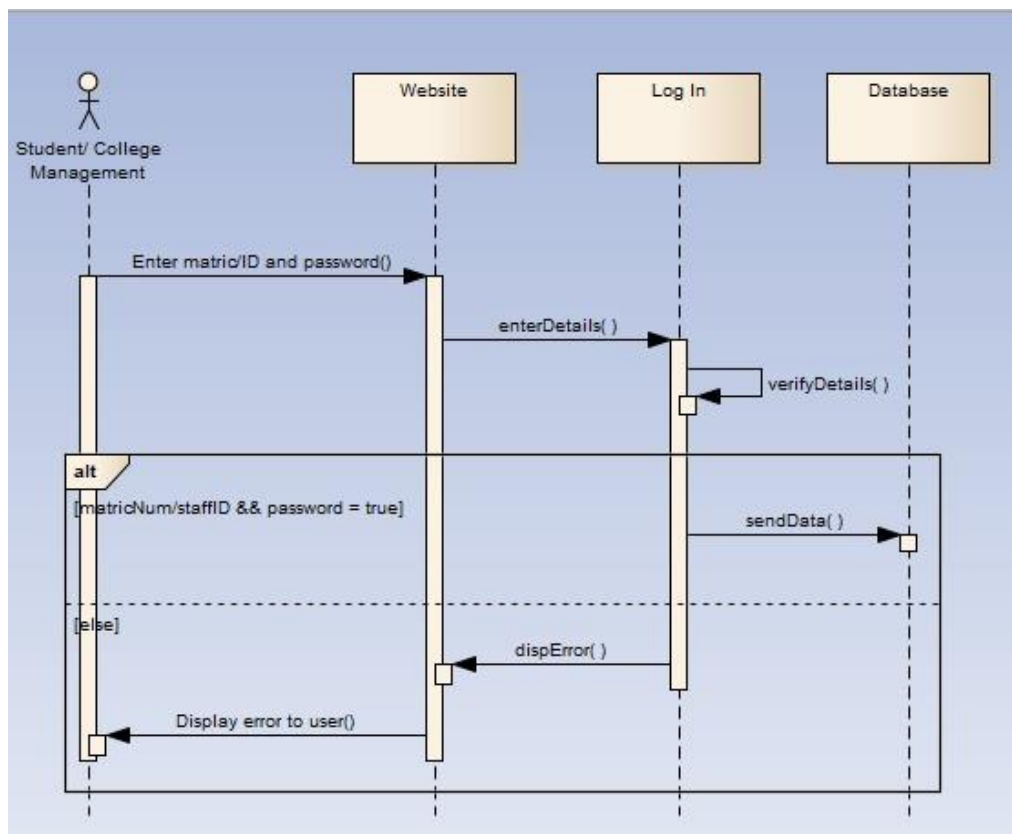


Figure 3.8: SD001: Sequence Diagram for <Log In> Subsystem

3.2.2 P002: <Manage Complaint> Subsystem

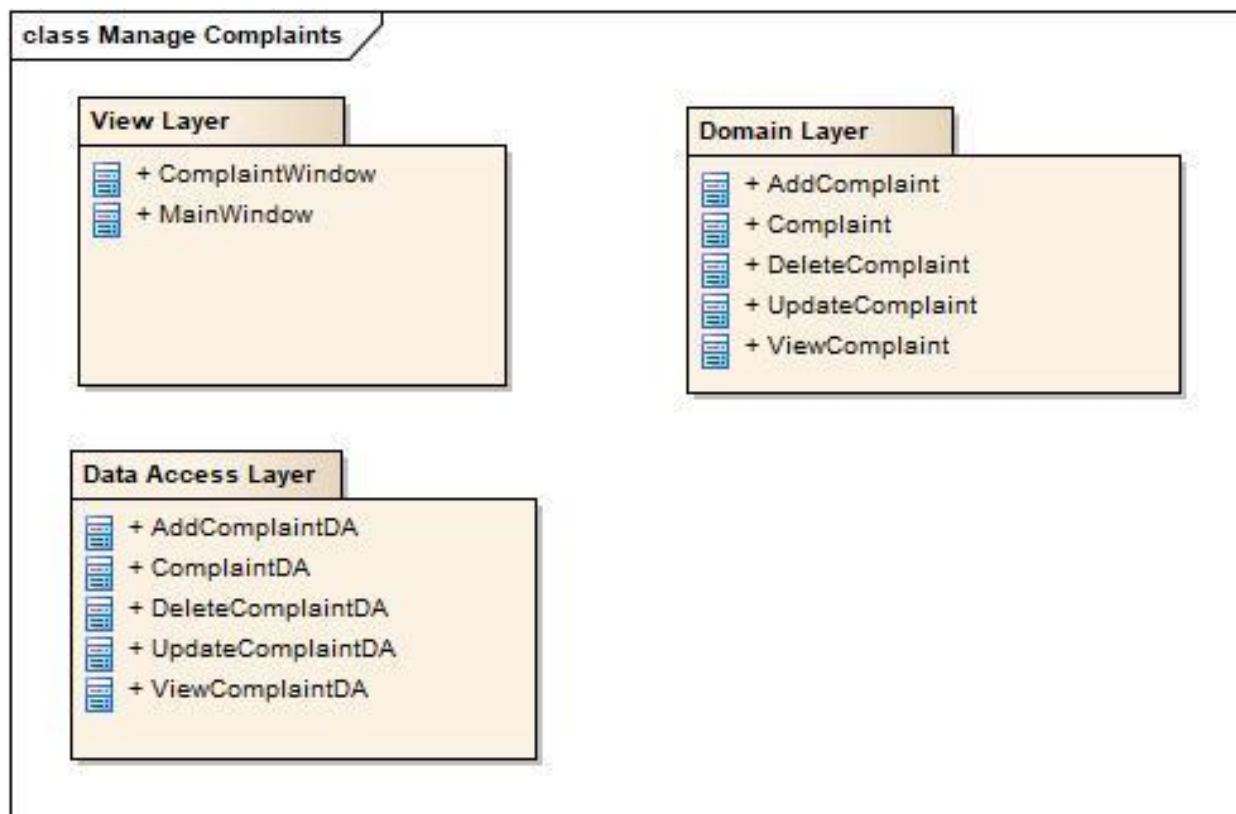


Figure 3.9 Package diagram for <Manage Complaint> Subsystem

3.2.2.1 Class Diagram

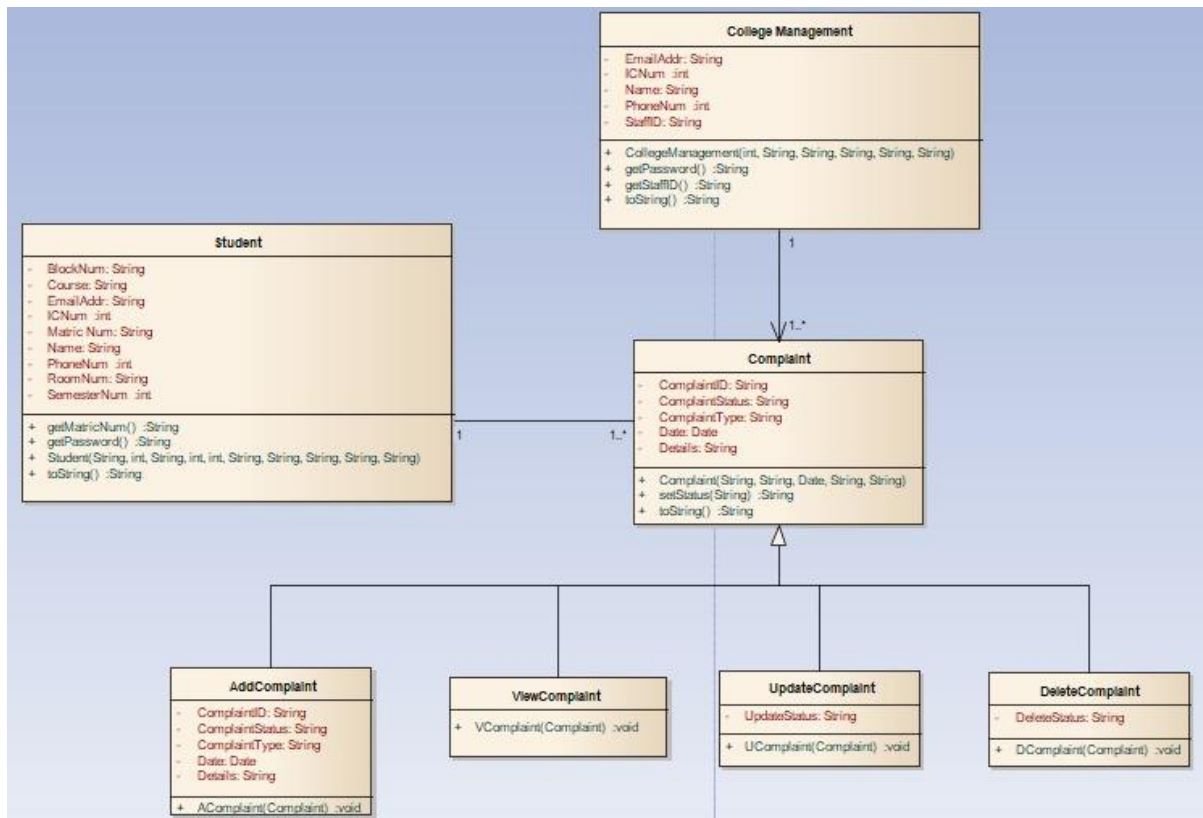


Figure 4.0 Class diagram for <Manage Complaint> Subsystem

3.2.2.2 Sequence Diagram

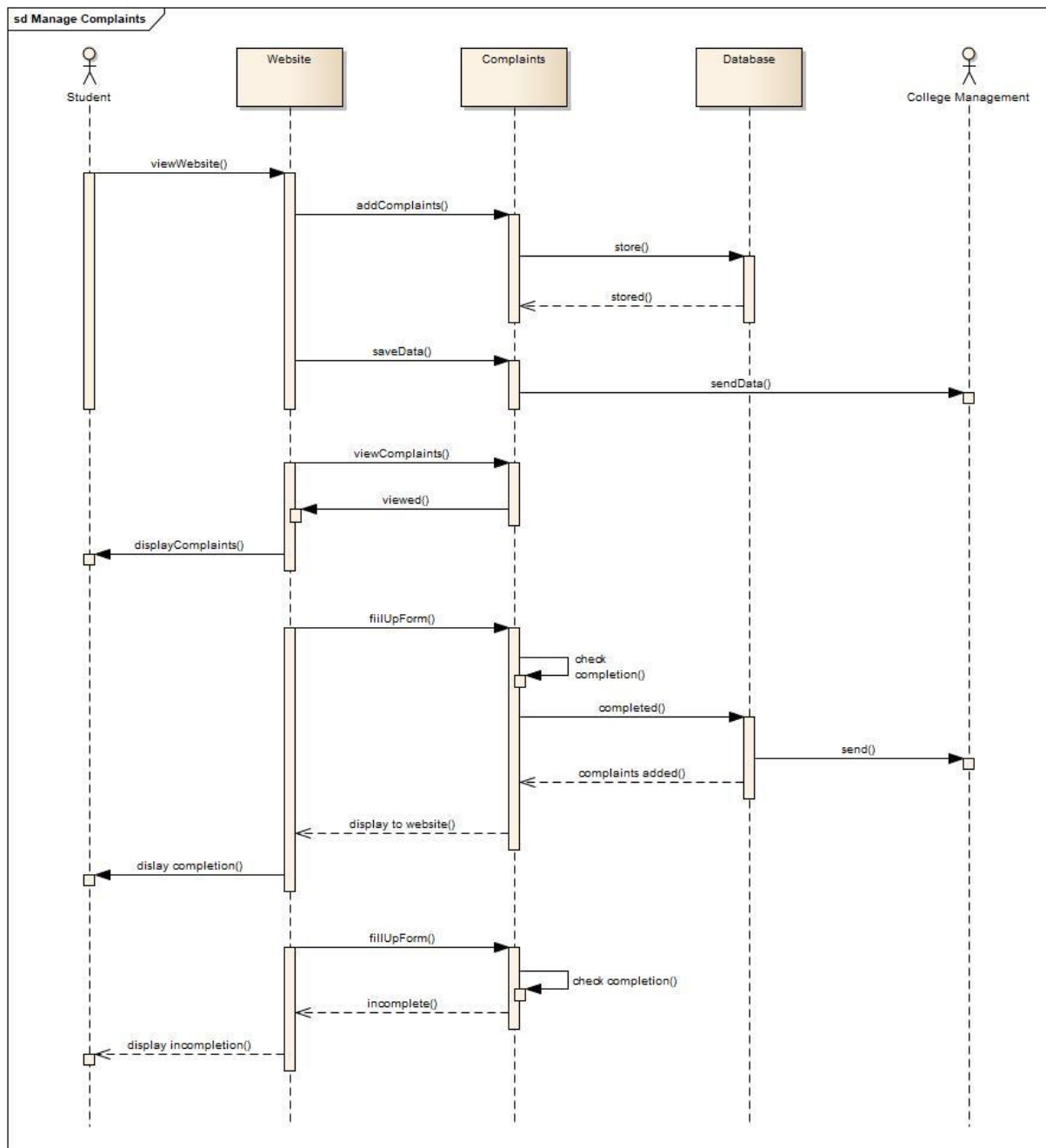


Figure 4.1 SD002: Sequence diagram for <Manage Complaint> for student

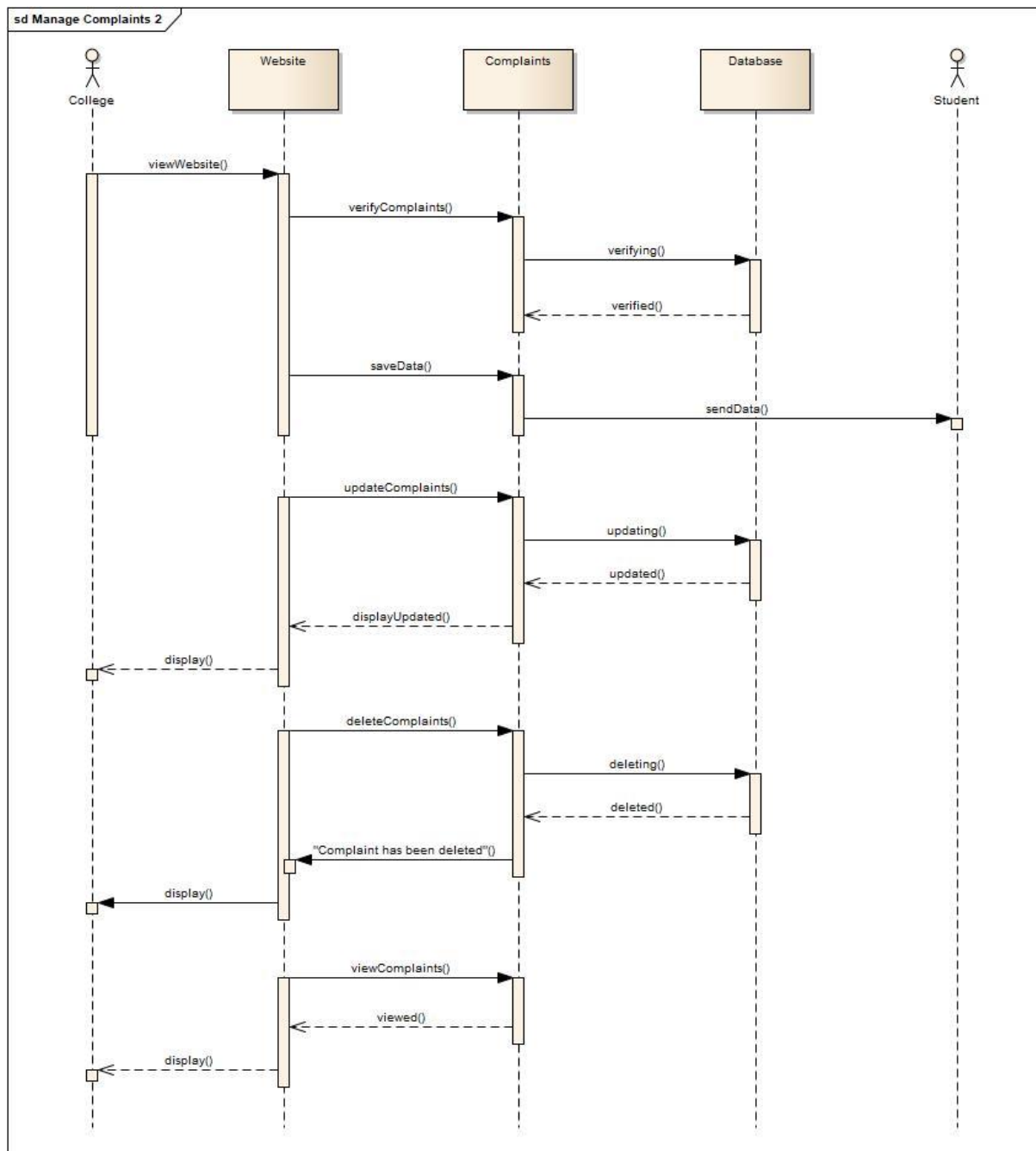


Figure 4.2 SD003: Sequence diagram for <Manage Complaint> for Collage Management

3.2.3 P003: Schedule Subsystem

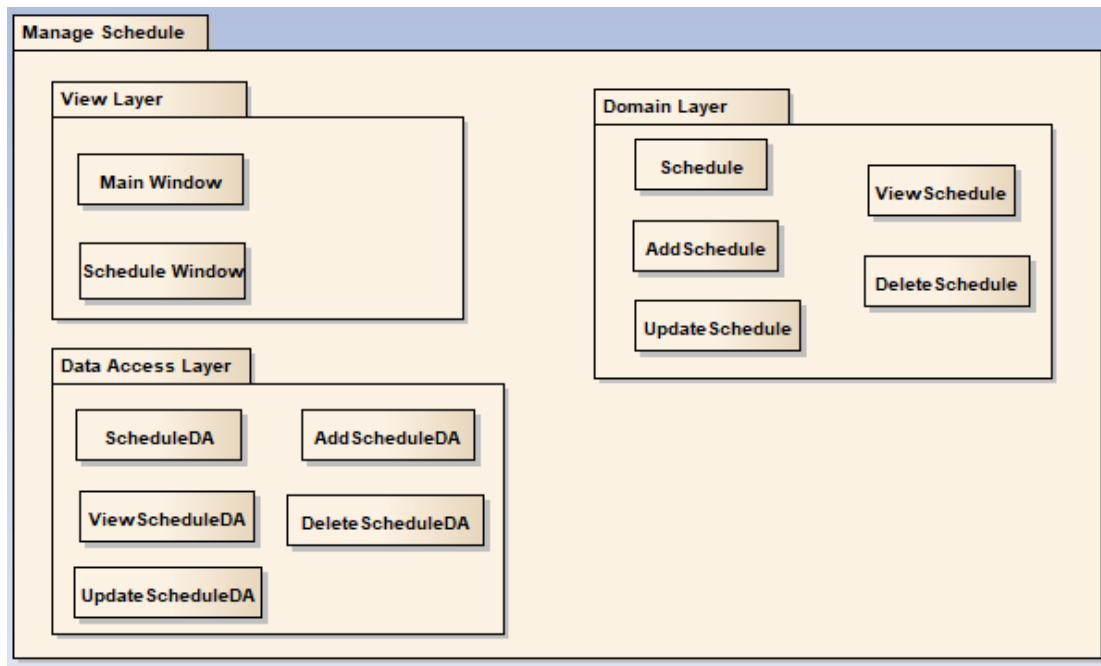


Figure 4.3 Package diagram for <Manage Schedule> Subsystem Class diagram

3.2.3.1 Class Diagram

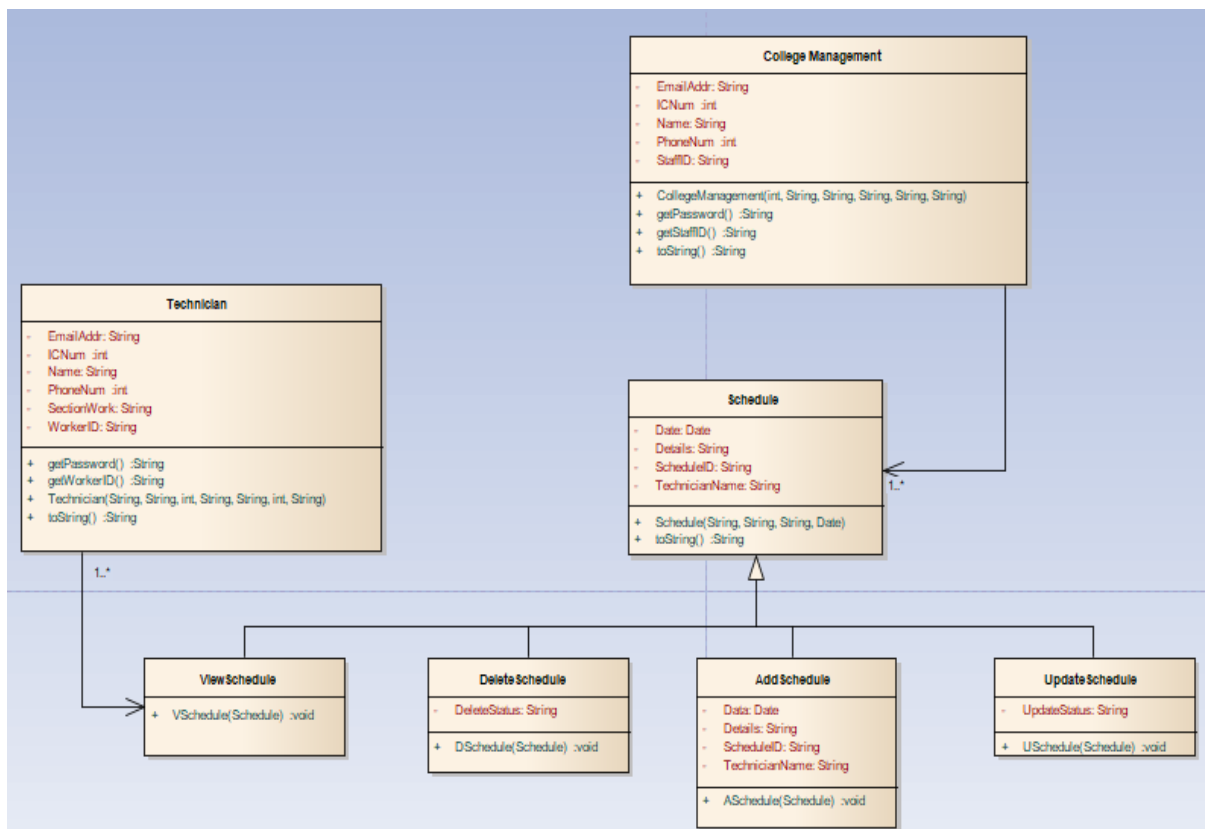


Figure 4.4 Class diagram for <Manage schedule> Subsystem

Entity Name	Schedule, AddSchedule, ViewSchedule, DeleteSchedule, UpdateSchedule
Method Name	ASchedule(), VSchedule(), DSchedule(), USchedule(), Schedule()
Input	Date, ScheduleID, Details, TechnicianName
Output	Date, ScheduleID, Details, TechnicianName
Algorithm	1. Start 2. College Staff views the schedule using VSchedule() method 3. If college staff wants to update schedule 3.1 College staff update schedule using USchedule() method 4. If college staff wants to delete schedule 4.1 College staff delete schedule using DSchedule() method 5. If college staff wants to add schedule 5.1 College staff insert schedule using ASchedule() method 6. Technician views the schedule using VSchedule() method 7. End

3.2.3.2 Sequence Diagram

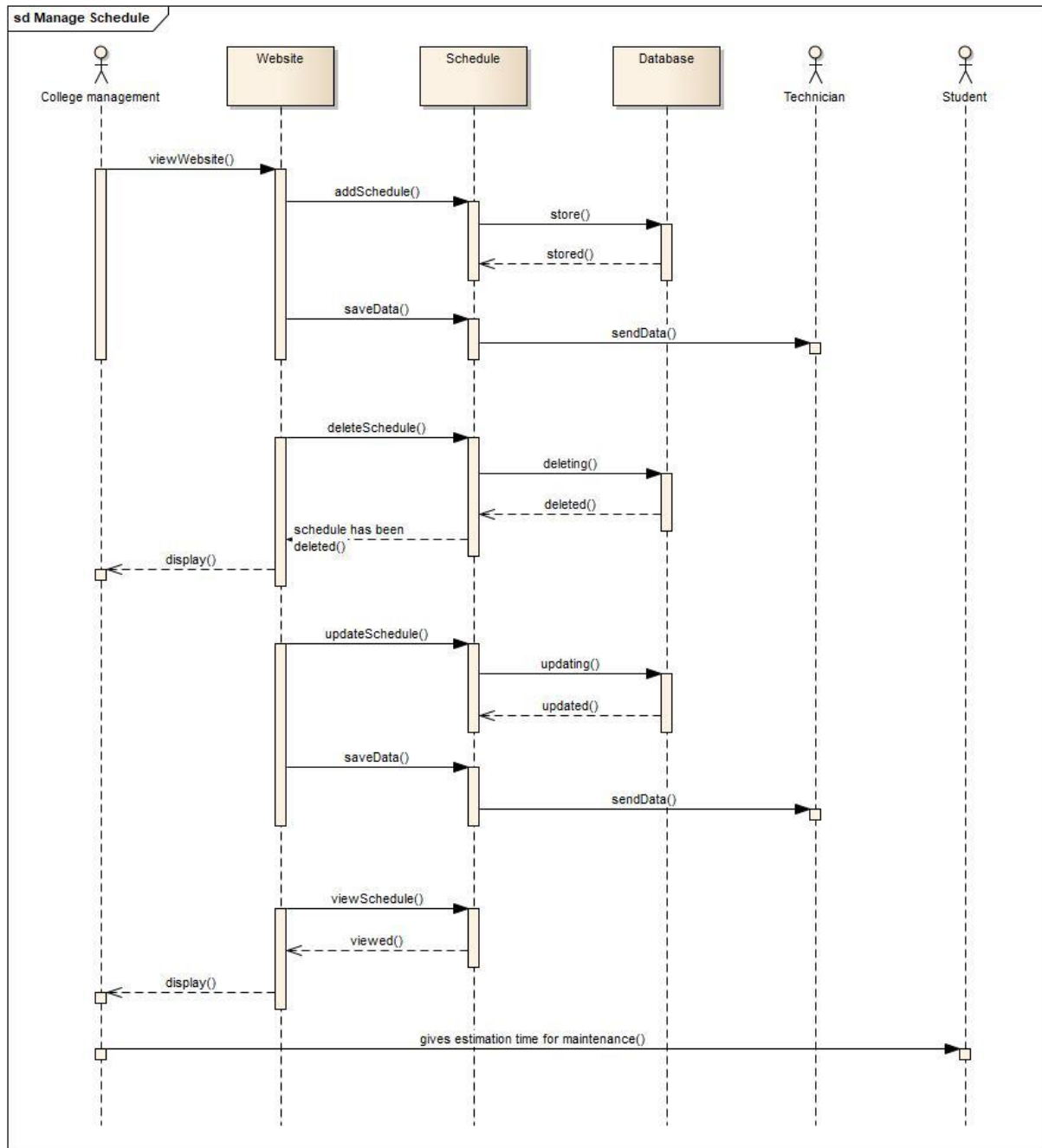


Figure 4.5 SD004: Sequence diagram for <Manage Schedule>

3.2.4 P004: Respond Subsystem

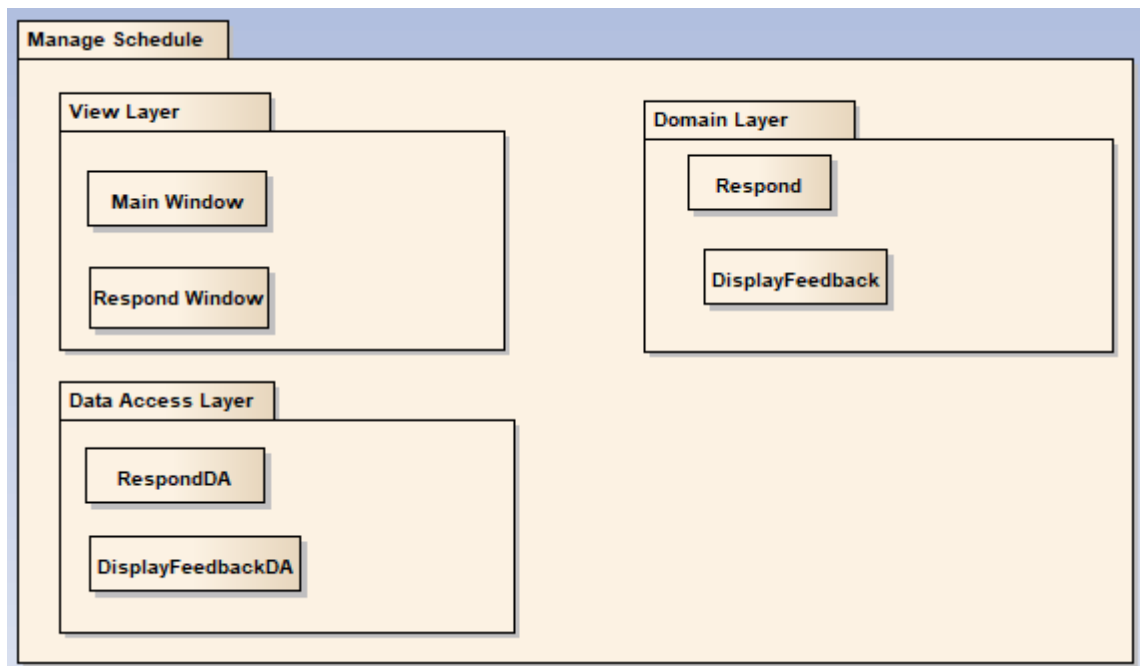


Figure 4.6 Package diagram for <Respond> Subsystem

3.2.4.1 Class Diagram

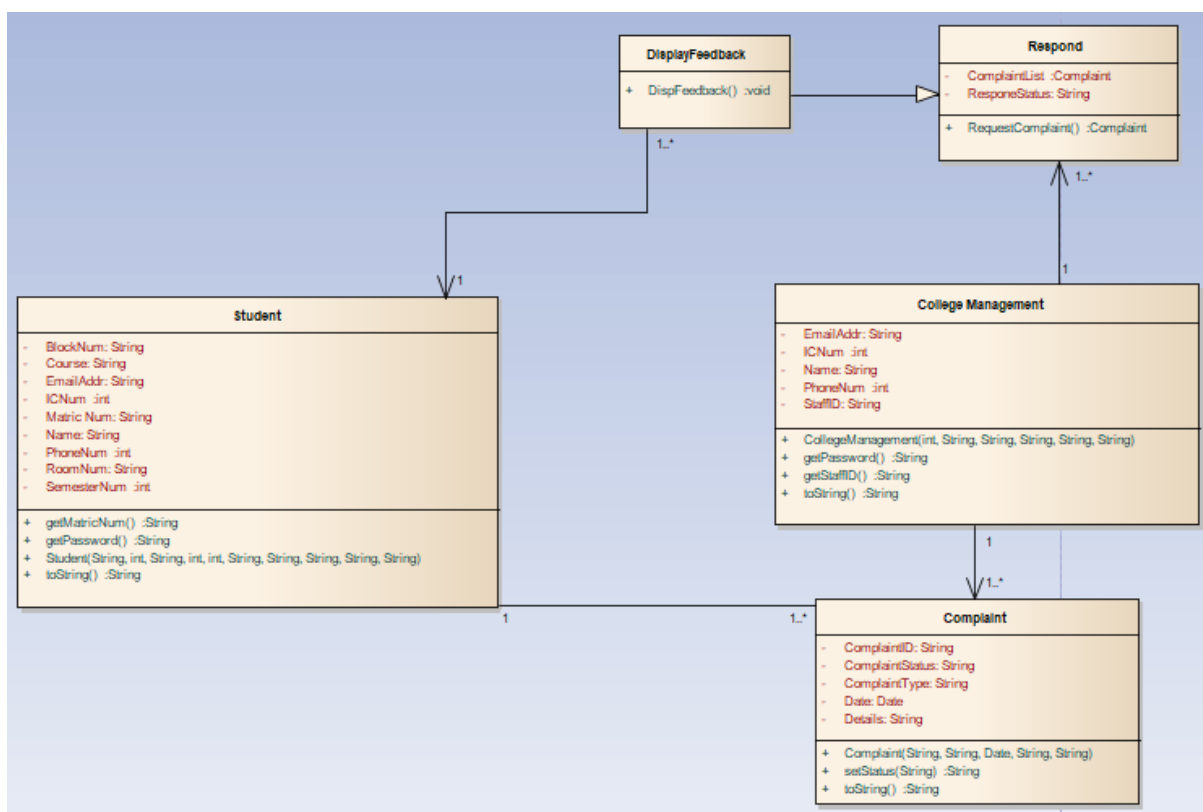


Figure 4.7 Class diagram for <Respond> Subsystem

Entity Name	Respond, DisplayFeedback
Method Name	RequestComplaint(), DispFeedback()
Input	requestFeedback, requestData, sendData
Output	ComplaintID, ComplaintStatus, ComplaintType, Date, Details
Algorithm	1. Start 2. College staff submit the feedback 3. College send notifications to the student 4. Student wants to view the feedback by clicking the button 5. System will request feedback and data to the DisplayFeedback and send it back to the website 6. Students now can display the feedback page 7. End

3.2.4.2 Sequence Diagram

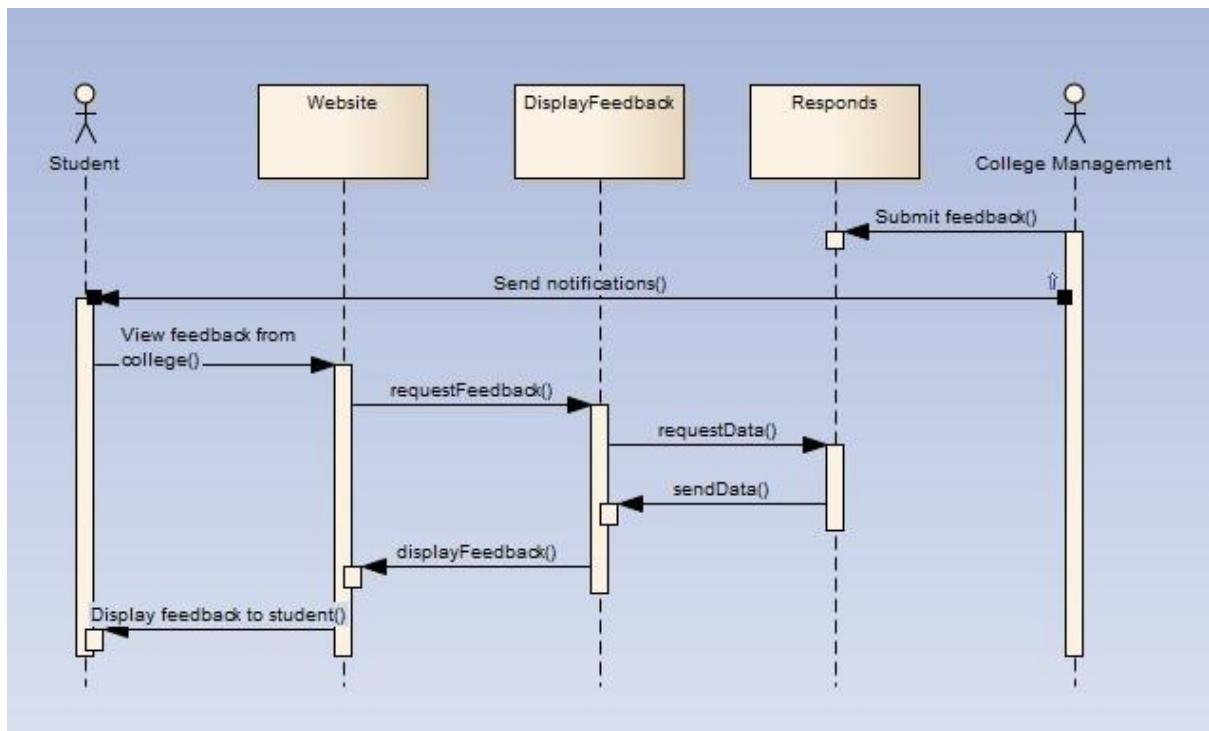


Figure 4.8 SD005: Sequence diagram for <Display Feedback>

3.2.5 P005: Report Subsystem

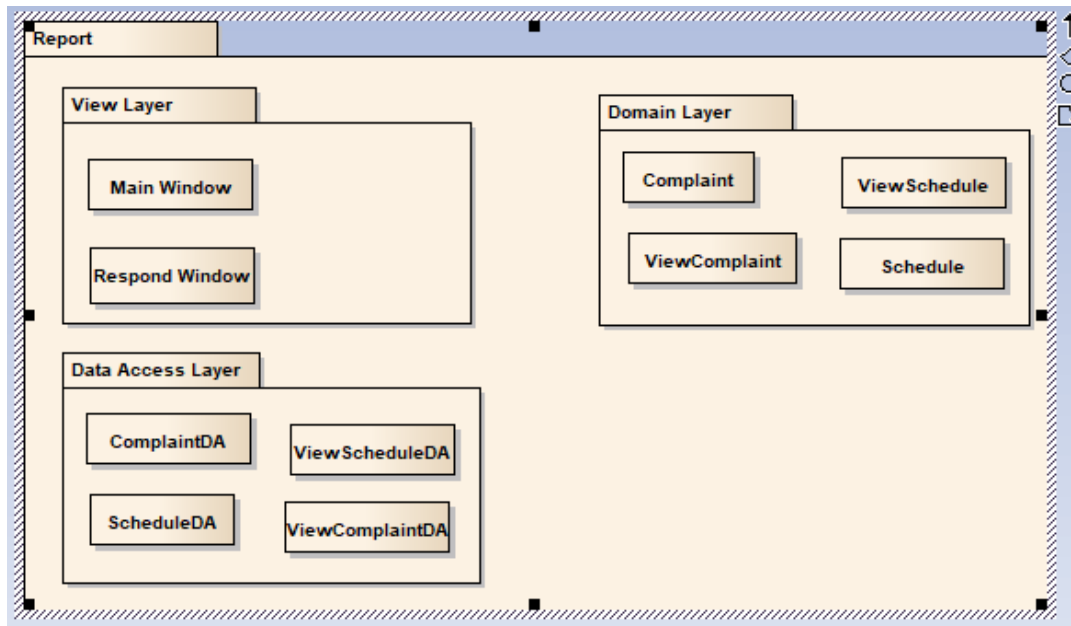


Figure 4.9 Package diagram for <Respond> Subsystem

3.2.5.1 Class Diagram

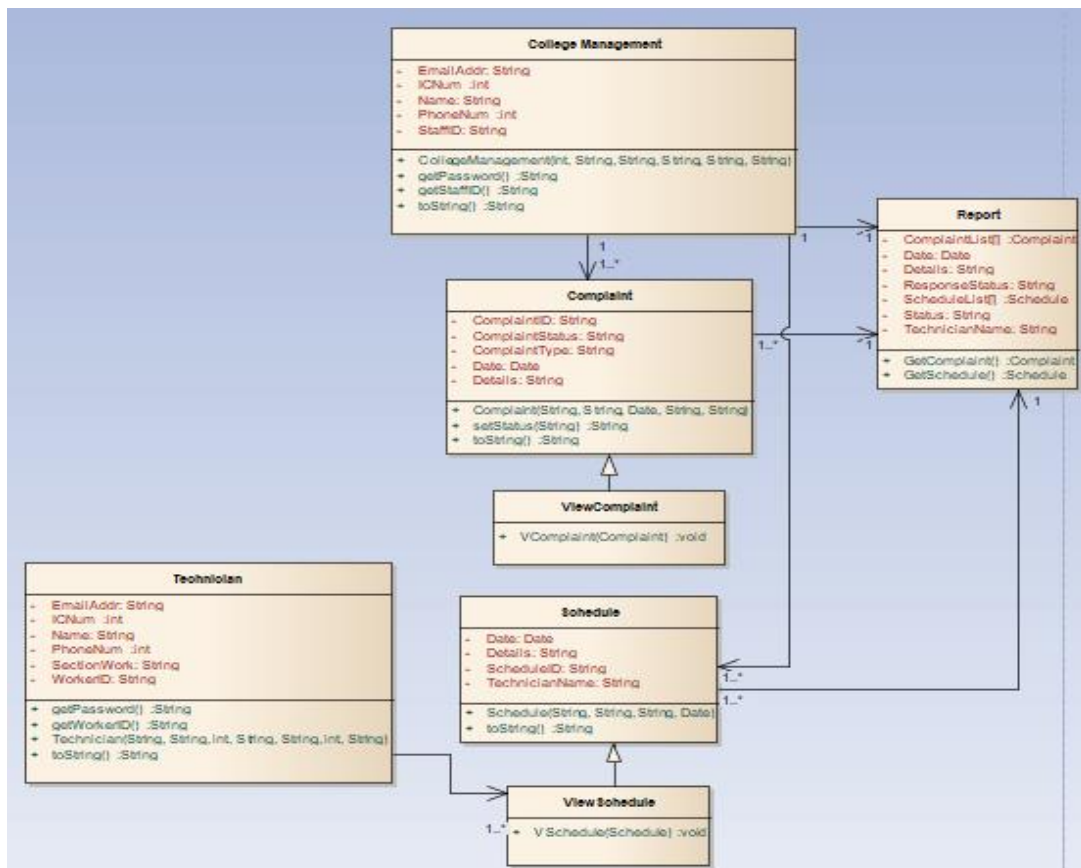


Figure 5.0 Class diagram for <Respond> Subsystem

Entity Name	Complaint, ViewComplaint, Schedule, ViewSchedule, Report
Method Name	VSchedule(), VComplaint(), GetComplaint(), GetSchedule()
Input	Report
Output	ComplaintID, ScheduleID, ResponseStatus, Date, Details, TechnicianName
Algorithm	1. Start 2. College Staff view report 3. System collect data from complaint using GetComplaint() method 4. System collect data from Schedule using GetSchedule() method 5. System display the report of complaint and schedule 1. End

3.2.5.2 Sequence Diagram

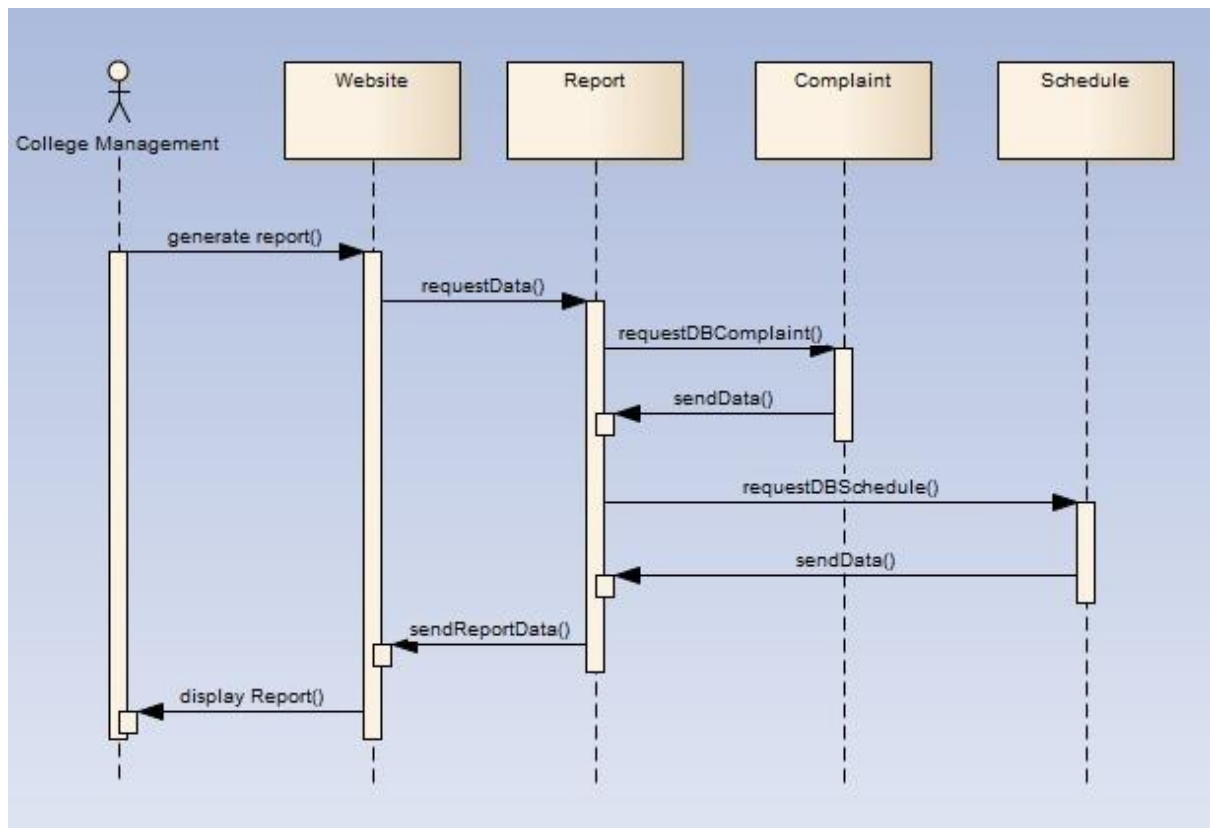


Figure 5.1 SD006: Sequence diagram for <Display Report>

4. Requirements Matrix

	Student	Technician	College Management	Log In	Complaint	Schedule	Respond	Report
P001, UC001, SD001	X	X	X	X				
P002, UC002, SD002	X		X		X			
P002, UC002, SD003	X		X		X			
P003, UC005, SD004		X	X			X		
P004, UC003, SD005	X		X				X	
P005, UC004, SD006			X					X

Appendix A: Traceability Matrix

Test Case ID	Use Case ID/ Sequence Diagram ID	Package ID
TC001 for Log in Subsystem • TC001_01	UC001 • SD001	P001
TC002 for Manage Complaint Subsystem • TC002_01 • TC002_02	UC002 • SD002 • SD003	P002
TC002 for Manage Schedule Subsystem • TC003_01	UC005 • SD004	P003
TC003 for Respond Subsystem • TC004_01	UC003 • SD005	P004
TC003 for Report Subsystem • TC005_01	UC004 SD005	P005

