



UNIVERSITY ADMISSION SYSTEM (WEB APPLICATION)

Low Level Design

Revision Number: 1.0
Last date of revision: 20/08/2022

Document Control

Change Record

Version	Date	Author	Comments
0.1	20-Aug-2022	Atul Singh	Initial Version

Reviews

Version	Date	Reviewer	Comments

Approval Status

Version	Review Date	Review By	Approved By	Comments

Contents

Document Control	1
Change Record.....	1
Reviews	1
Approval Status	1
Contents	2
1 Introduction	3
1.1 What is Low-Level design document?.....	3
1.2 Scope.....	3
2 Architecture	4
3 Architecture Description	4
3.1 Register.....	4
3.2 Store to Database	5
3.3 Login	5
3.4 Validate User	5
3.5 Welcome Dashboard.....	5
3.6 Course Management.....	5
3.7 Application Submission	5
3.8 Application Review.....	5
3.9 Approve/Disapprove.....	6
3.10 Interview	6
3.11 Selected/Rejected	6
4 Use Case Diagram	7
5 User IO Workflow	8
6 Faculty I/O Flow	9
7 Student I/O Flow	10
8 Unit Test Cases.....	11

1 Introduction

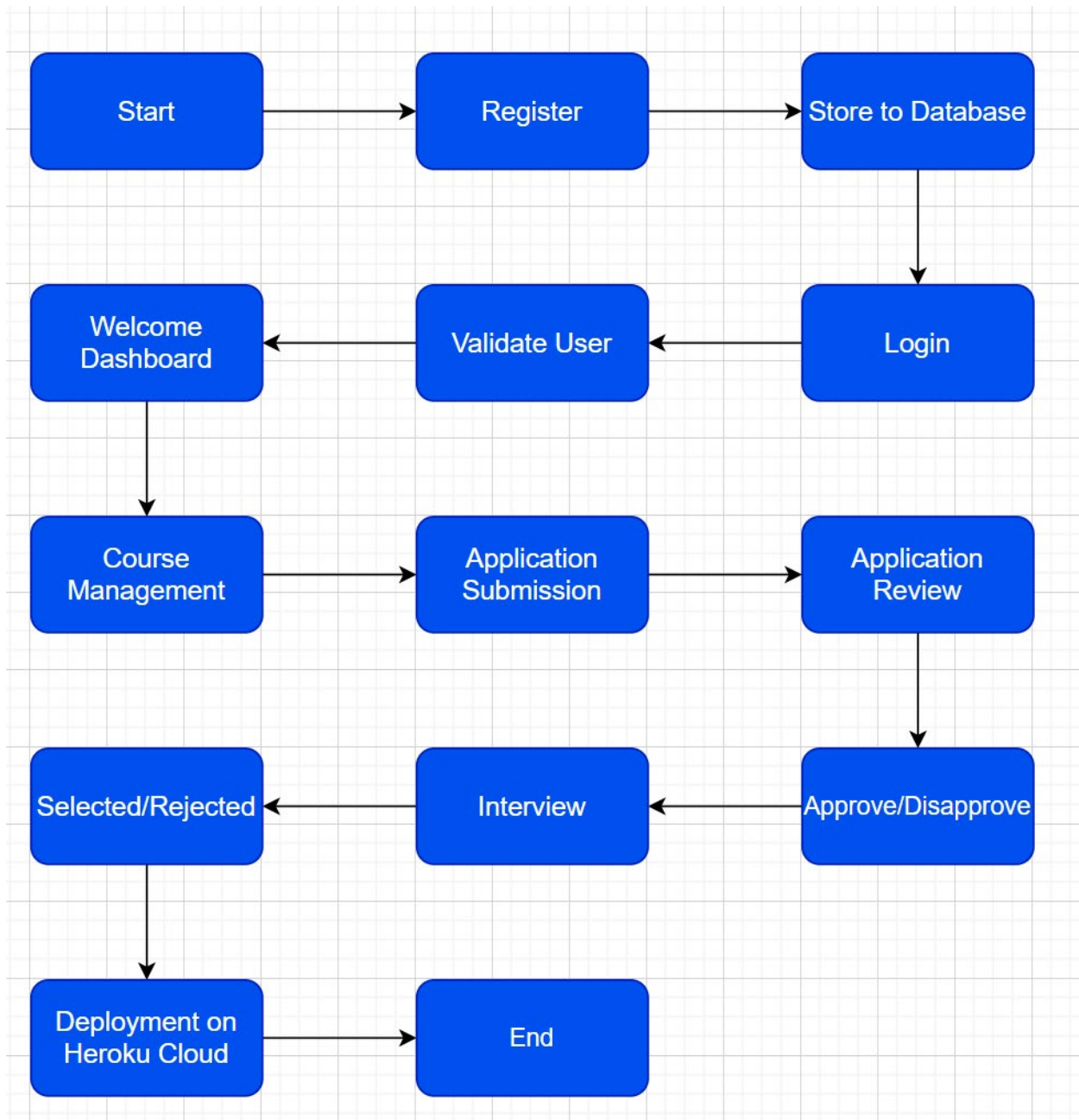
1.1 What is Low-Level design document?

The goal of this document is to give the internal logical design of the actual program code for University Admission System. It describes the class diagrams with methods and relations between classes and program specs. It will also describe the or use case diagram. It describes the modules so that the programmer can directly code the program

1.2 Scope

Low-level design (LLD) is a component-level document that will include the use case diagram, Class diagram and how all the module interact with each other.

2 Architecture



3 Architecture Description

3.1 Register

In order to use the system, faculty and student must register themselves first in the system. Only the registered user will be able to login to system. Already registered user will not be able to register again.

3.2 Store to Database

Once the user submits the registration form, all the details will get stored in the database.

3.3 Login

After registration process get completed, user will have to login with the same email and password to access the application.

3.4 Validate User

When the user will click login after filling the email and password, system will verify the user for its authenticity and then allow the user to login to the application.

3.5 Welcome Dashboard

After successful login user will be redirect to the correct dashboard.

If the user is a faculty then he/she will be redirected to the screen when he/she can manage the course or check the list of all the students who have applied for the course.

If the user is the student, then he/she will be redirected to the course selection and submission screen.

3.6 Course Management

Faculty will have access to add new course, modify existing course and remove the existing course in the system.

3.7 Application Submission

Once the student is registered in the system, he/she can apply for any active course by filling and submitting the application containing the basic detail and course name.

3.8 Application Review

Once the application is submitted by the student, the application will go to faculty for initial review of the data filled by the student.

3.9 Approve/Disapprove

After the initial review of the application. Faculty will either approve or disapprove the application for the interview process.

3.10 Interview

Once the Faculty will approve the application, interview will be scheduled with the student and student will get the email notification of the same.

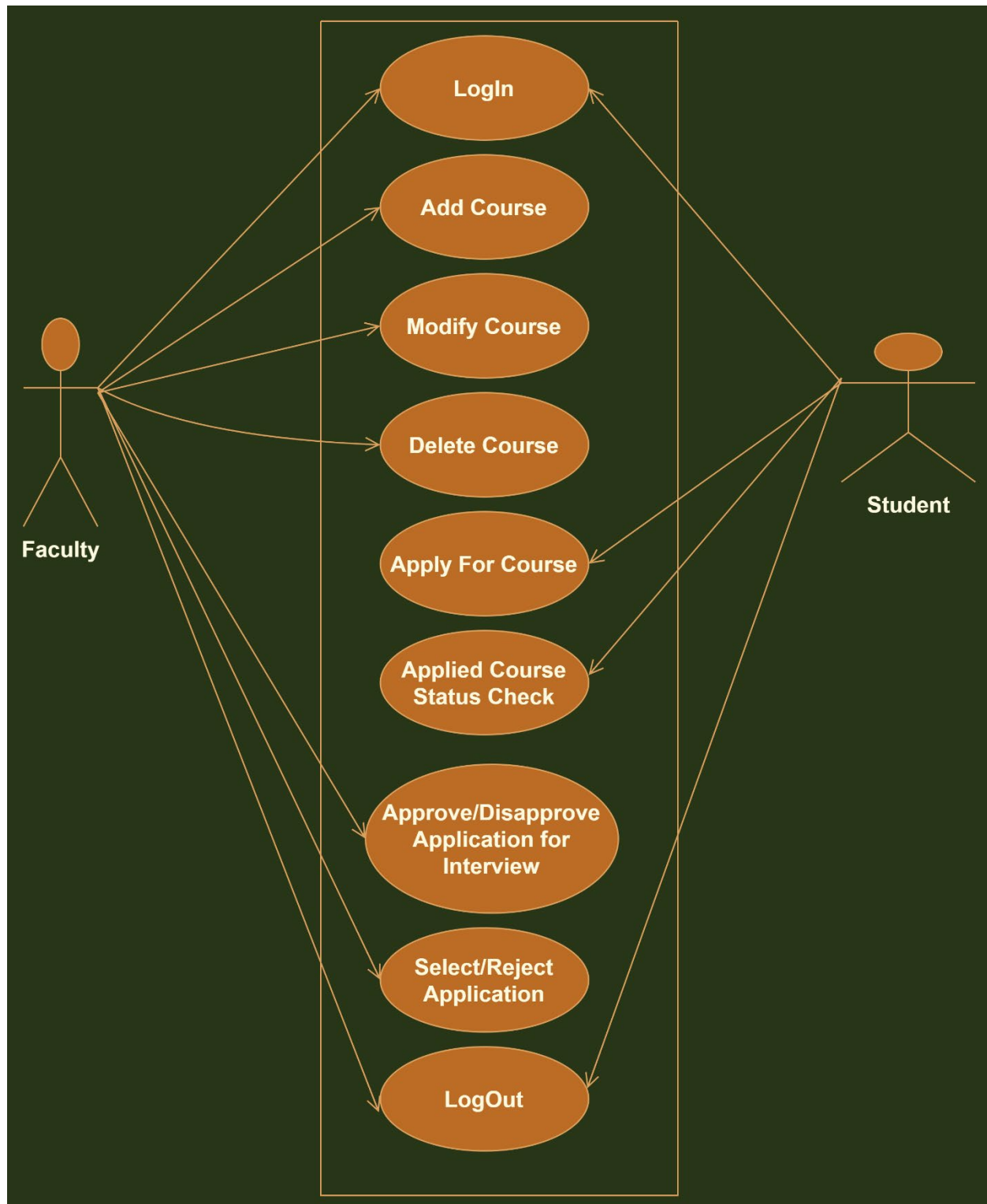
3.11 Selected/Rejected

Based on the interview with student, faculty will either select or reject the applicant for the course.

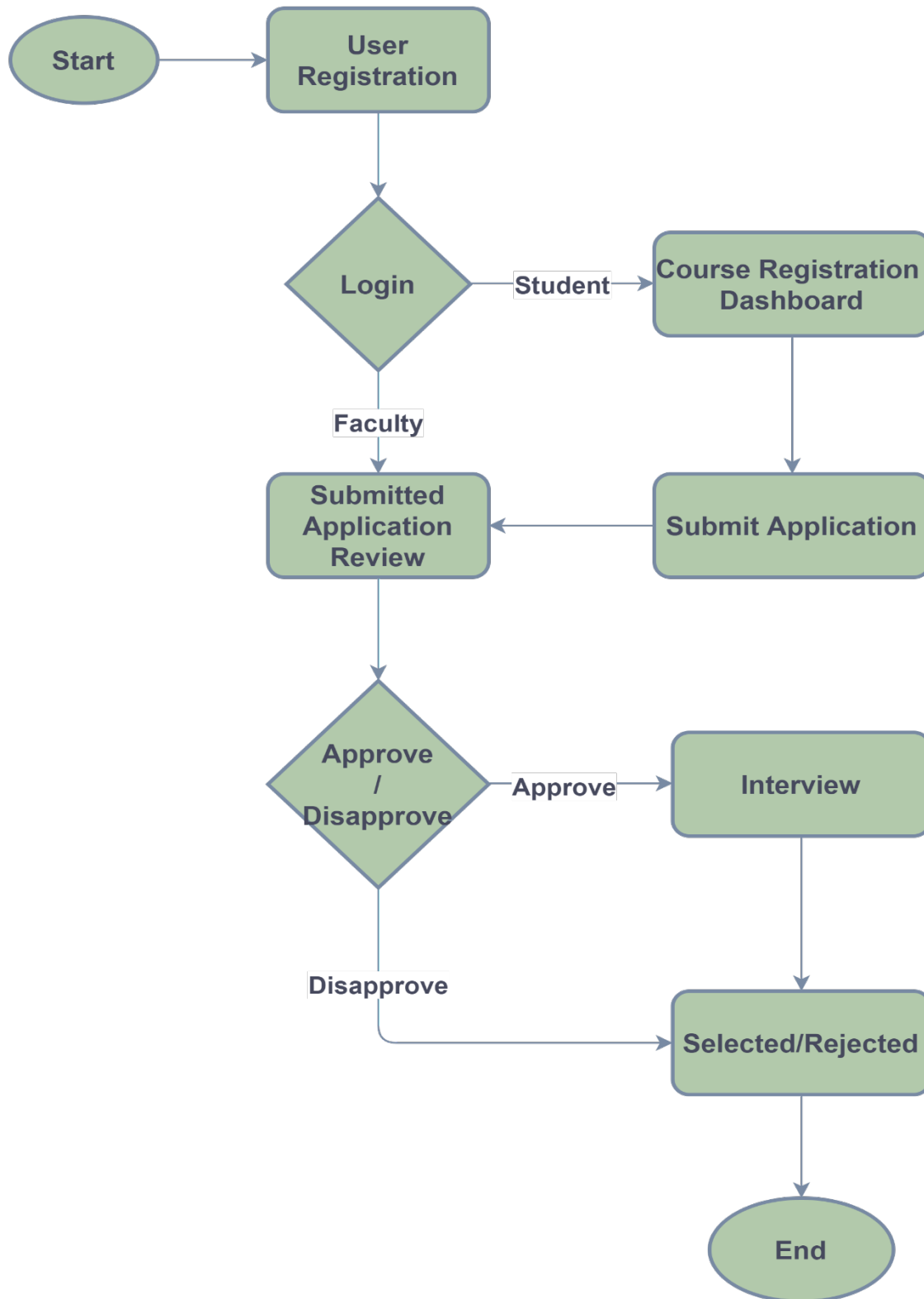
3.12 Deployment on Heroku.

Once the development is completed the code will be deployed to Heroku cloud for it to be available for everyone.

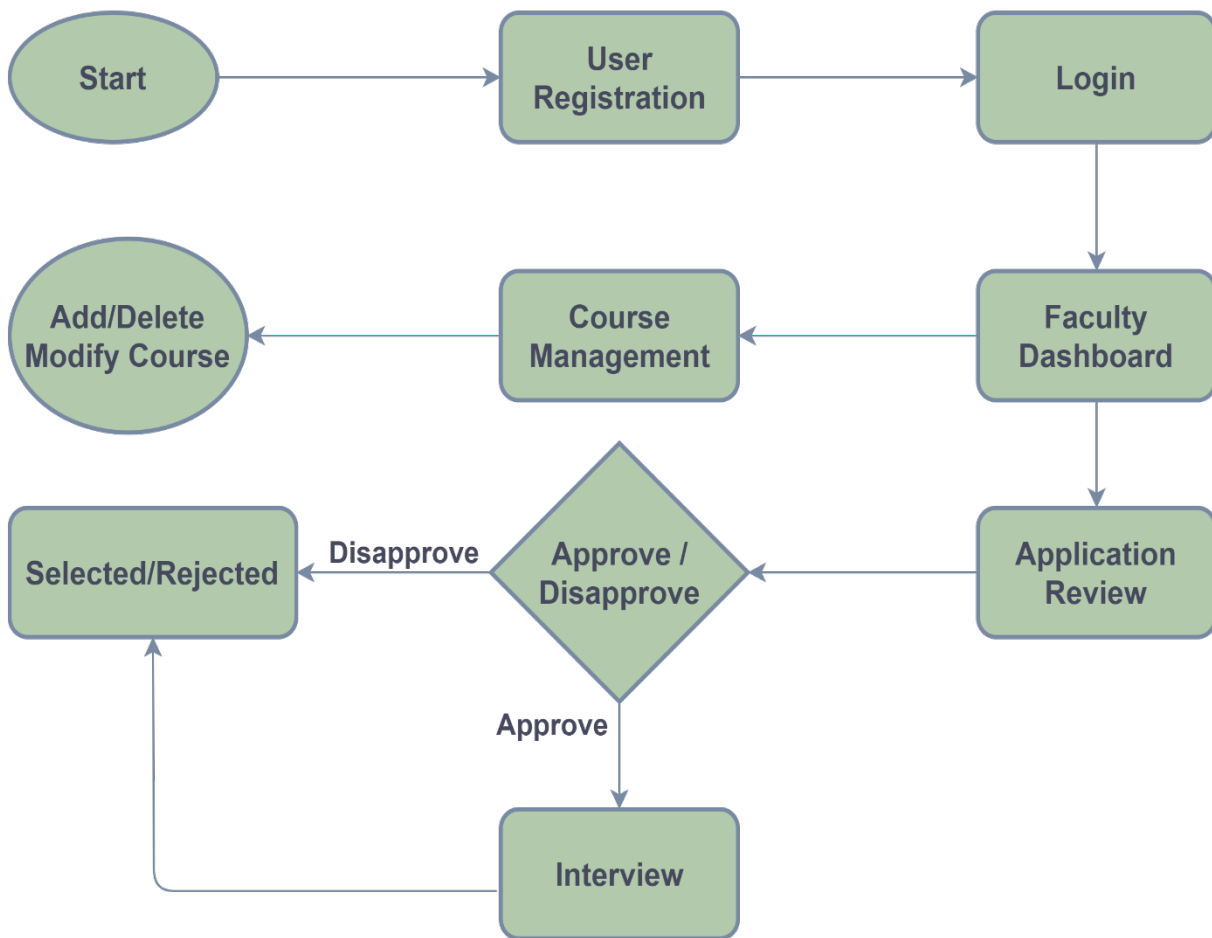
4 Use Case Diagram



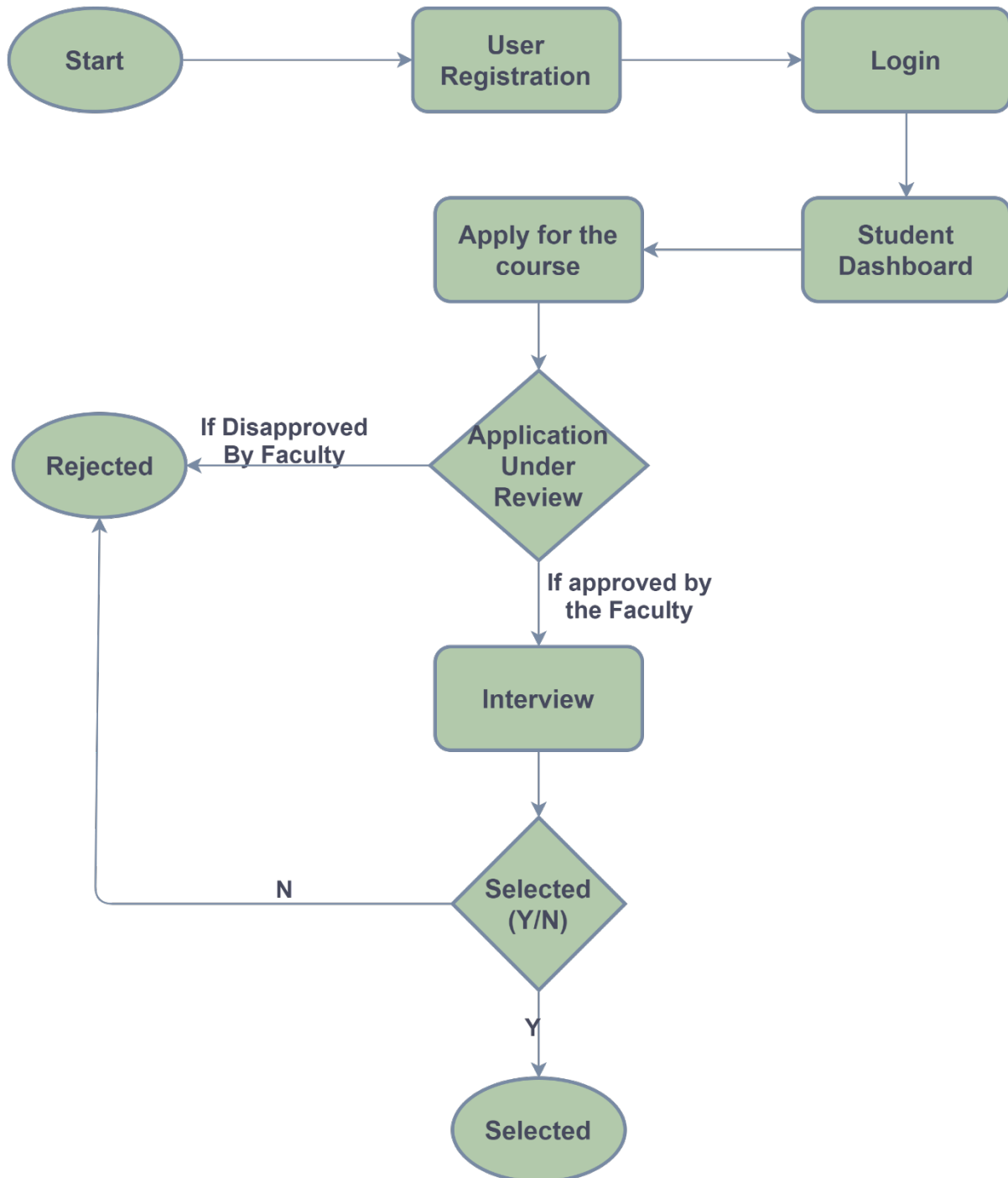
5 User IO Workflow



6 Faculty I/O Flow



7 Student I/O Flow



8 Unit Test Cases

Description	Pre-Requisite	Expected Result
Faculty registration	1. Faculty should have faculty Id.	Faculty should be able to do sign up without any error.
Student registration	1. No	Student should be able to do sign up without any error.
Faculty Login	1. Already registered	Faculty should be able to do login successfully without any error.
Student Login	1. Already registered	Student should be able to do the login successfully without any error.
Student apply for the new course	1. Student should be logged into the system.	Student should be able to apply for the new course without any error.
Student check the status of the applied course	1. Student should be logged in to the system. 2. Student should have applied for the course.	Student should be able to check the status of the applied course.
Faculty should be able to add new course	1. Faculty should be logged into the system.	Faculty should be able to add new course without any error.
Faculty can modify the course	1. Faculty should be logged into the system. 2. Course should exist in the system	Faculty should be able to modify the course successfully.
Faculty can delete the course.	1. Faculty should be logged into the system. 2. Course should exist in the system	Faculty should be able to modify the course successfully.
Faculty can access the list of the application applied by the student.	1. Faculty should be logged into the system. 2. Student should have applied for the course.	Faculty should be able to view the list of the application applied by the student without any error.
Faculty can Disapprove the application for the applied course by the student.	1. Faculty should be logged into the system. 2. Student should have applied for the course.	Faculty should be able to able to Disapprove the application applied by the student and email should be sent to student and status should change to rejected.
Faculty can Approve the application for the applied course by the student.	1. Faculty should be logged into the system. 2. Student should have applied for the course.	Faculty should be able to approve the course and email should be sent to student and status should change to interview scheduled.
Faculty can select the application after the interview.	1. Faculty should be logged into the system. 2. Student should have applied	Faculty should be able to select the student after interview and email should be sent to the student and

	for the course.	status should change to selected.
Faculty can reject the application after the interview.	<ol style="list-style-type: none">1. Faculty should be logged into the system.2. Student should have applied for the course.	Faculty should be able to reject the student after interview and email should be sent to the student and status should change to rejected.