

The Chinese University of Hong Kong
Department of Computer Science and Engineering

CSCI3100 Software Engineering
Project Final Report

Backdoor University

Group F1

Introduction	4
Project Overview	4
Objective	4
Highlights	4
Admission Chart	4
Comment	4
Language Switch - English and Chinese,	5
Expected Customers and Market	5
Cloud Integration	5
Project Statistics	5
System Architectural Design by DFD	7
System Architecture	7
DFDs	7
Login	7
Register	8
Reset Password	8
Change Password	9
View Programme Info	9
Edit Programme Info	10
Comment On Programme	10
Detailed Description of Components by UML	11
frontend	11
Structural Diagram	11
UMLs	11
Functionality	16
Procedures and Functions	17
backend	18
Structural Diagram	18
UMLs	18
ER-Diagram	18
Functionality	18
Procedures and Functions	19
Minigame - Spinning Puzzle	20
Structural diagram	20
UMLs	20
Functionality,	21
Procedures and Functions	21
User Interface Design	22
Description of the User Interface	22
Screen Images	22

Home Page	22
Tips and News	23
Programme Catalog Page	23
Statistics Page	24
Register and Login Page	25
Admin Page	26
Game Page	26
Error (404) Page	27
Test	27
Test overview and test plan	27
Bcrypt testing	27
API	27
UI Behavior testing	28
Lesson Learned	28
Conclusion	28
Appendix (Case-N)	30
Bcrypt test sets	30
Backend API test sets	30
UI Behavior Testing	37

1. Introduction

1.1. Project Overview

Nowadays, higher education is considered an entry ticket to most career paths. Therefore, students are concerned about how they should continue further education. The non-JUPAS is the primary method for community college students to continue further education in local universities. Those universities seldomly provide statistical data as references compared to JUPAS. For a better strategy to apply to suitable university programmes, this application visualizes the admission score by various fancy charts, which help efficiently understand the entry condition and plan the career path. In addition, to provide a communication platform between students and mentors.

1.2. Objective

This project aims to collect the admission data to higher education and visualize them via various charts for quick reference. It will guide students to develop an efficient strategy for the targeted programme smartly. Unlike the data officially provided by the University Grants Committee (UGC), this system is built under a bottom-up principle. Users who received an offer from a university could update the database. The database could then accordingly generate a new chart for the client-side. Users may also leave comments to notice later-comers or communicate to the mentors.

1.3. Highlights

Admission Chart

The primary function of this application is to generate admission charts for references to non-JUPAS applicants. Every local university programme would have a unique set of data for each chart. The chart shows the information sorted by the name of community colleges and sub-degree and GPA, corresponding to the data. Users who have studied in community colleges or universities could import the data. After every data input, the chart would then update.

Two charts are employed, assisted with a detailed table. One is the pie chart which shows the proportion of colleges in various colors; another is the scatter chart which shows the distribution of the GPA for each student in a specific programme or in general. A table below those charts is listing all of the information for detailed information.

Comment

This application encourages students to advise potential applicants. College or university students could update their offers to the system and leave a comment in a particular programme to convey more information that needs concern before or during application. If further advice is needed, any

users may also leave a contact in the comment. As a result, it could build strong connections between mentors and students.

Language Switch - English and Chinese.

The potential viewers are students in Hong Kong who mainly take Chinese as their first language and English as their second language. Still, most tertiary education in Hong Kong provides English information only. Therefore this website support language switching between English and Chinese by clicking the dropdown list.

Expected Customers and Market

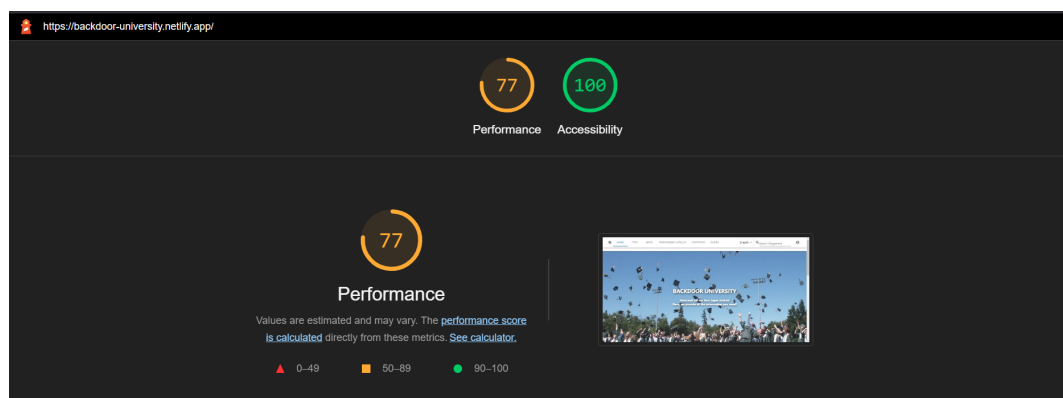
Most community colleges and international students need to apply through non-JUPAS if they are aiming at local universities. However, the system remains unclear to students, unlike the Joint University Programmes Admissions System (JUPAS), which has no transparent admission data. In addition, local universities are relatively independent in deciding on non-JUPAS entry. Hence, most of the students can only take advice based on a narrow sample size from their mentors. The traditional process causes huge uncertainty to non-JUPAS applicants. As a result, numerous students would easily overlook or underlook some programs, eventually leading to a skills mismatch in the labor market.

Thousands of community college and international students require an admission score to plan their own higher education every year. Similar systems exist but customize to other specific student groups and regions (e.g., HKDSE, Taiwanese and Japanese students).

Cloud Integration

The programme is deployed on cloud server. The front end uses Netlify (<https://backdoor-university.netlify.app>) and the backend uses Heroku (<https://backdoor-university.herokuapp.com>).

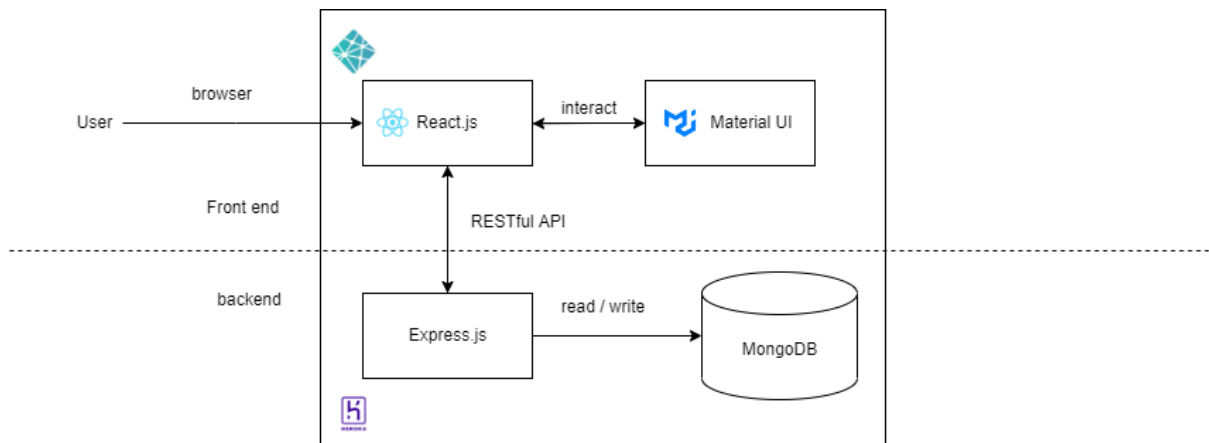
1.4. Project Statistics



Function	LOC	V(G)
sendPhoto()	5	1
userRegister()	45	4
adminLogin()	22	5
userLogin()	22	4
activateAccount()	13	3
userUpdatePassword()	33	4
userOrAdminUpdatePassword()	39	3
userForgetPassword()	25	3
userListAll()	11	2
userListOne()	27	3
userUpdate()	34	2
programmeList()	7	2
programmeCreate()	14	2
programmeUpdate()	15	2
programmeCommentUpdae()	14	2

2. System Architectural Design by DFD

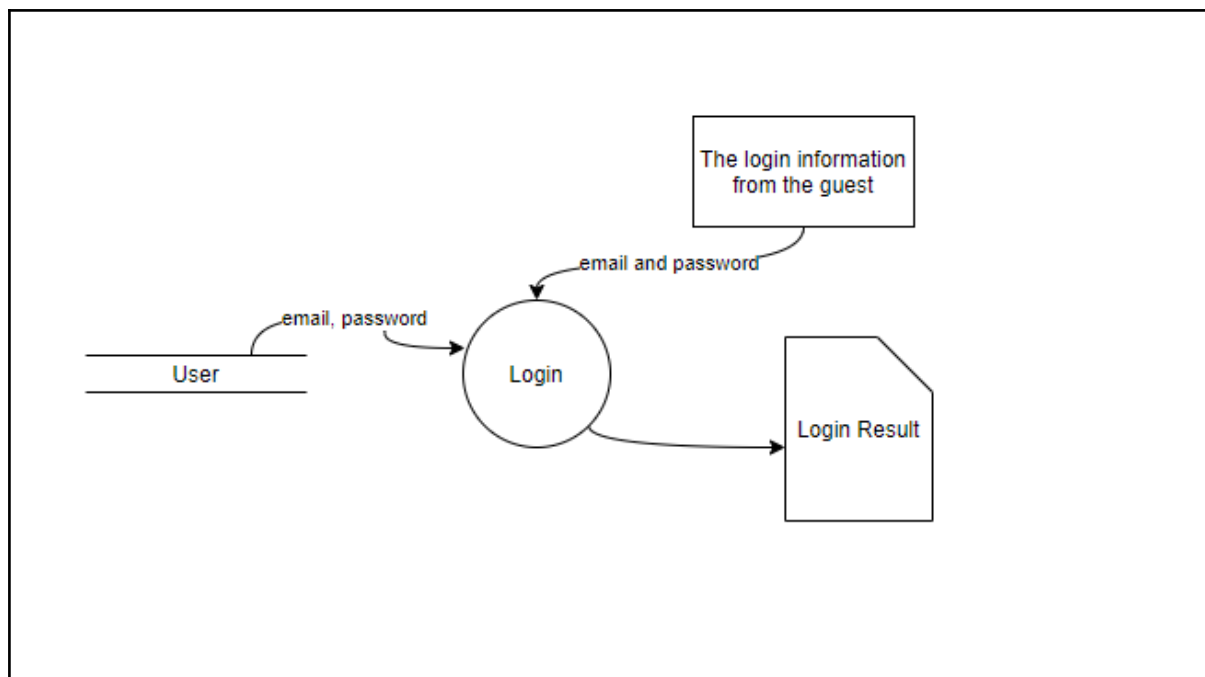
2.1. System Architecture



The front end is implemented using react.js and material UI, an extremely robust UI library that allows us to simplify the development process. The backend is implemented using Express.js, along with MongoDB as the database. The backend is responsible for authentications and database management. The user sends different requests, such as register and login, through the frontend, and the request will be handled in the backend.

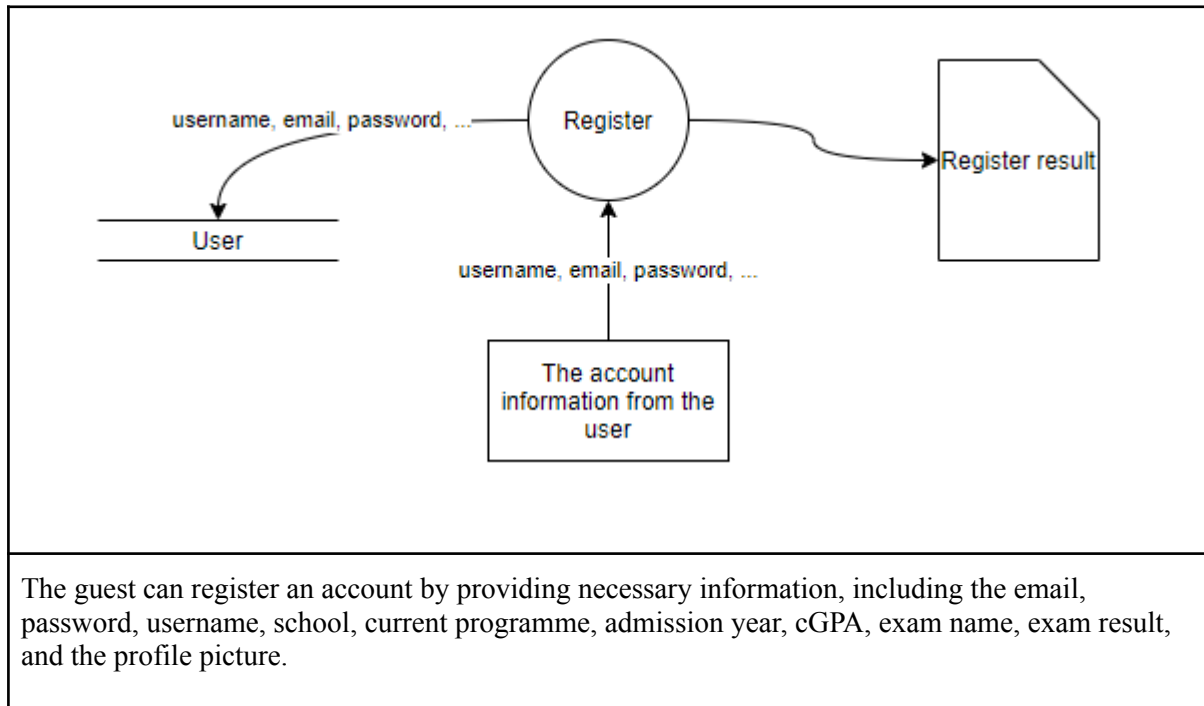
2.2. DFDs

2.2.1. Login

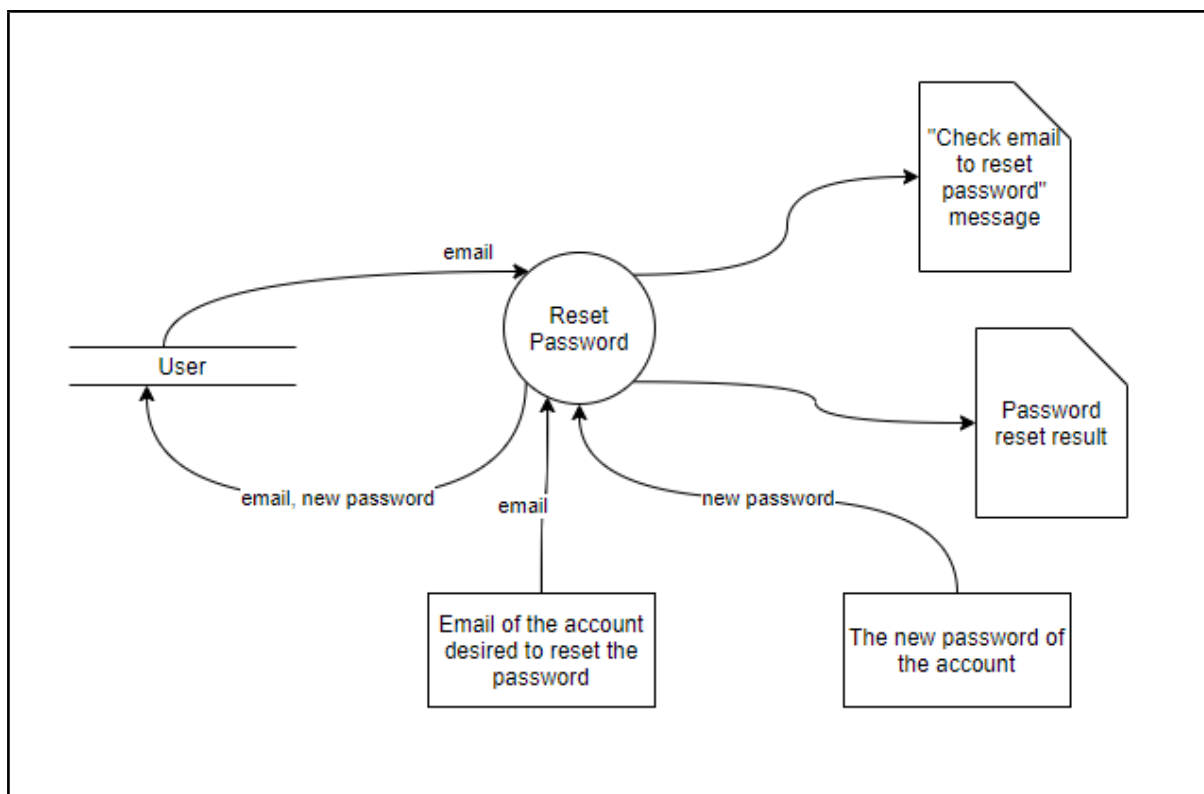


The guest can login by providing the account email and password.

2.2.2. Register

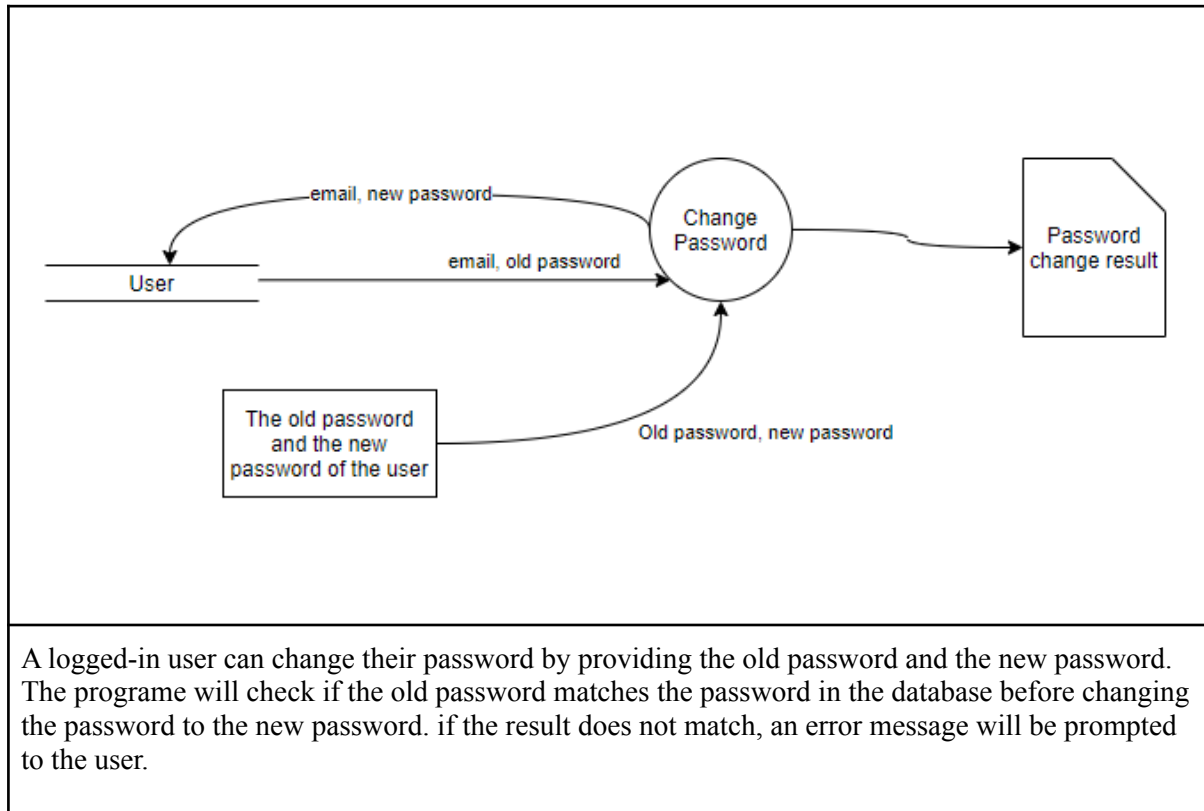


2.2.3. Reset Password

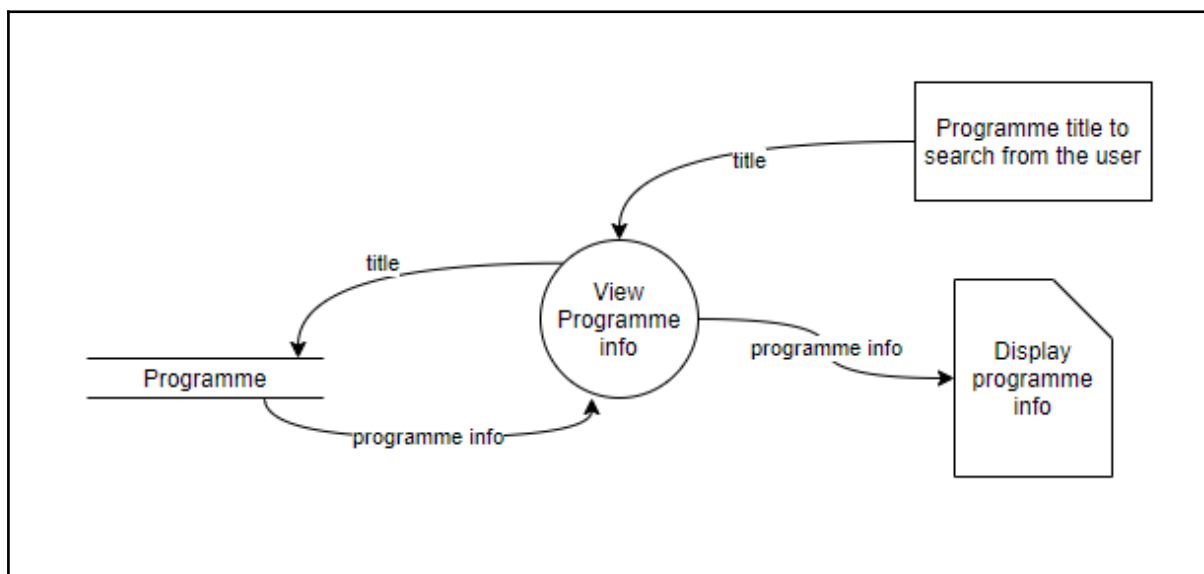


A guest user can reset their password if they have forgotten the password of their account. An email, which contains the new password, will be sent to the user's email address if the email is associated with an account.

2.2.4. Change Password

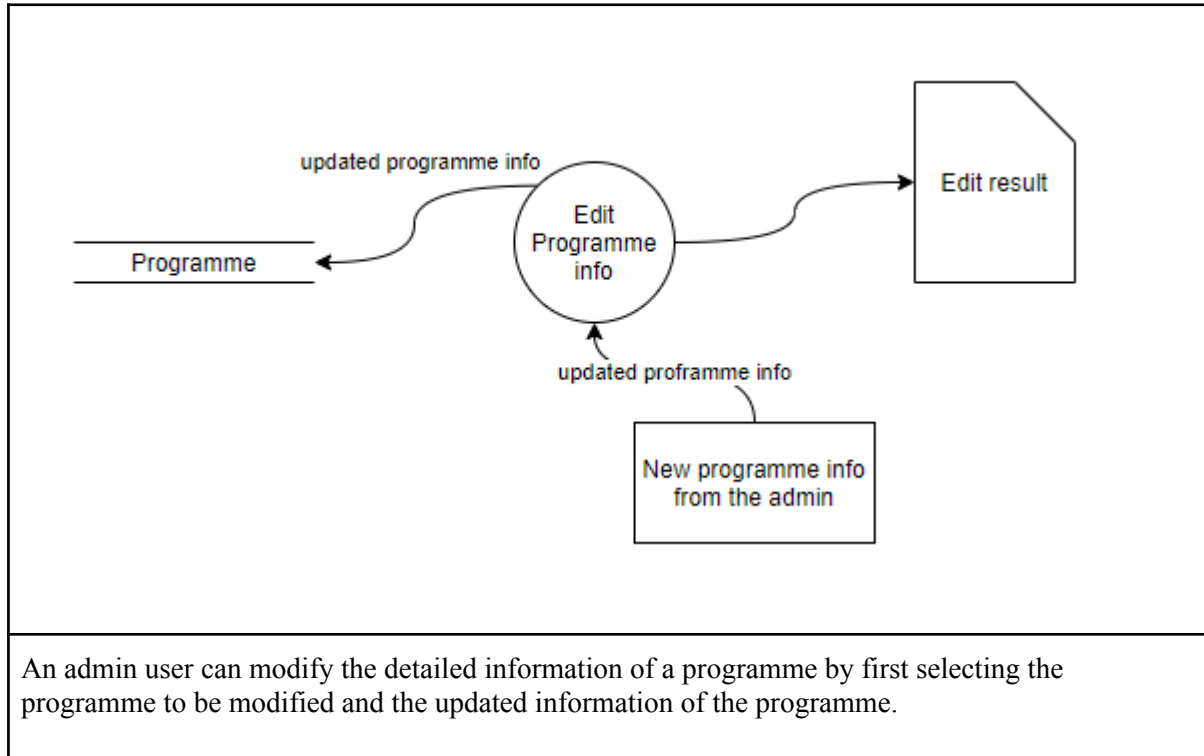


2.2.5. View Programme Info

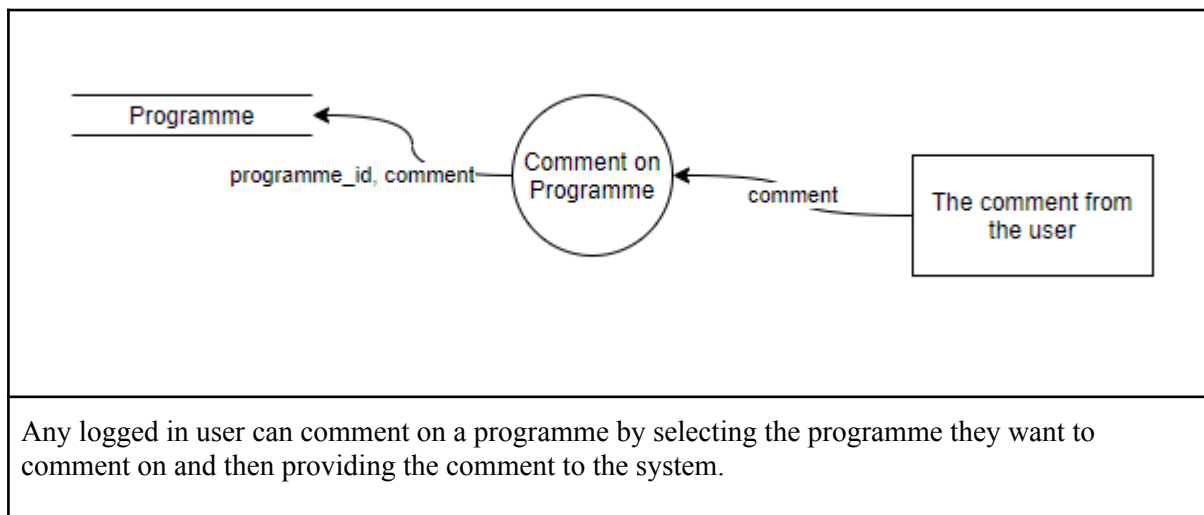


Any user can view the detailed information of a programme by selecting its title.

2.2.6. Edit Programme Info



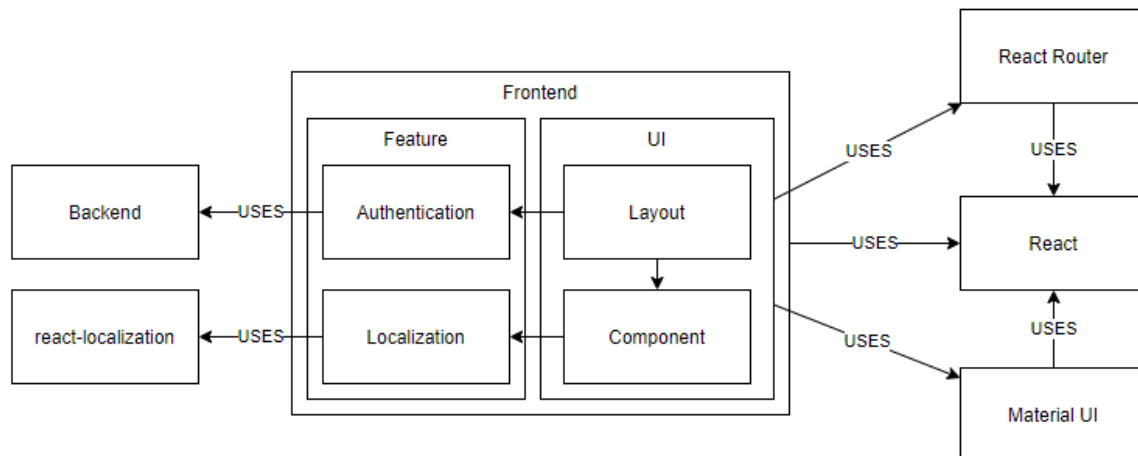
2.2.7. Comment On Programme



3. Detailed Description of Components by UML

3.1. frontend

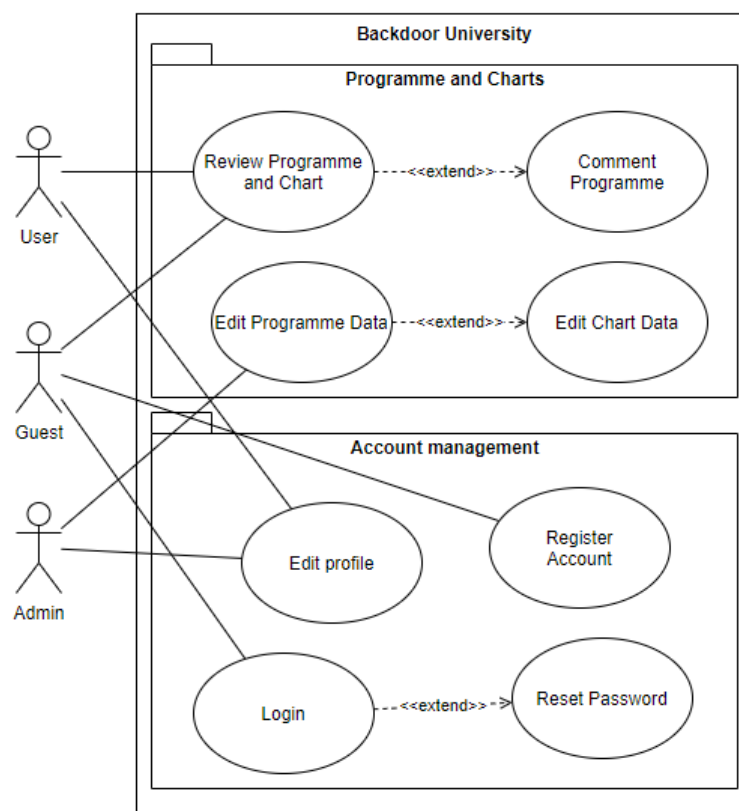
3.1.1. Structural Diagram



The frontend uses the backend and different existing modules developed by other parties.

3.1.2. UMLs

3.1.2.1. Use Case Diagram



■ Actor Description

Actor Name	User
Type	Primary Actor
Brief Description	A logged-in guest will become a standard user. A user can review programmes and their charts. A user can edit their user profile and input relevant information, such as their school and undertaking programme.

Actor Name	Guest
Type	Primary Actor
Brief Description	A guest can register an account and login to become a user. A guest can review programmes and charts.

Actor Name	Admin
Type	Primary Actor
Brief Description	A guest login with an admin account will become an admin. An admin can change the data of programmes and their charts. An admin can edit a standard user's profile and set their active state to ban them.

■ Brief Use Case Description

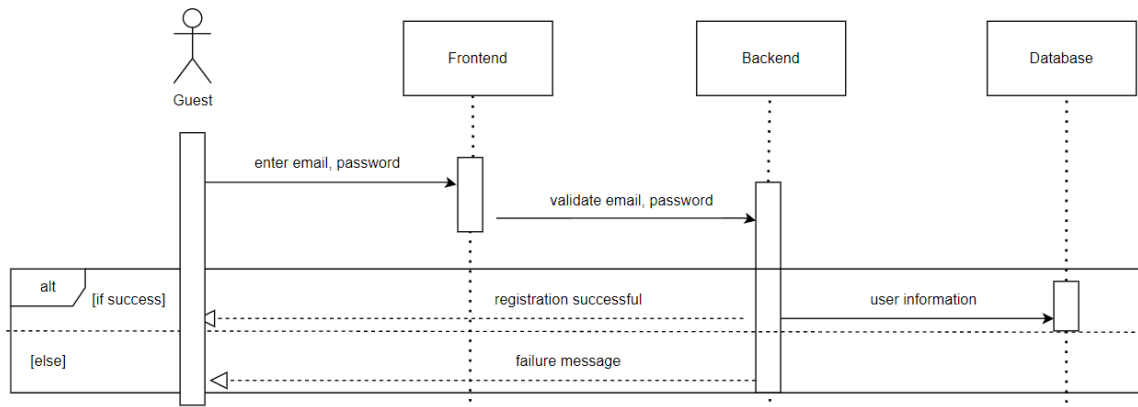
Use case name	Review Programme and Charts
Use Case ID	UC-001
Actor(s)	User, Guest
Brief Description,	An actor selects a programme and reviews its information and charts.
Pre-conditions	<ul style="list-style-type: none"> • The internet connection of the actor is available • The actor has access to the website • The cloud database and server are available
Flow of events	<ol style="list-style-type: none"> 1. The user clicks on "Programme" 2. The user selects a programme listed on screen 3. The programme information and its charts will be displayed to the actor.
Alternative flows and exceptions	<ul style="list-style-type: none"> • The actor can exit and close the website during any step above.

Post-conditions	The selected programme and its charts should be displayed to the actor
Priority	High
Non-behavioural requirements	Efficiency: The website response time should not exceed 3 seconds. This can be limited by internet quality. Availability: The programme information and charts should be available when the actor is trying to access them.
Assumptions	The user understands the display language of the website. There is no internet interruption while performing the above actions.

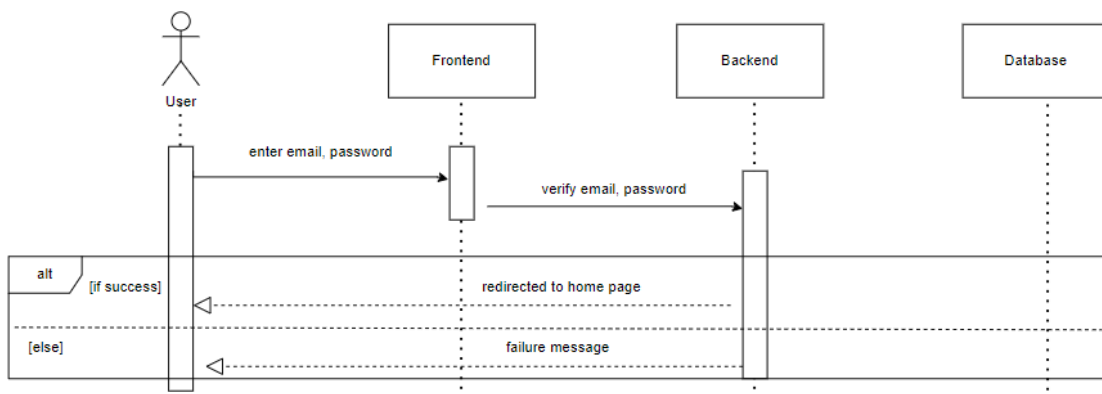
Use case name	Login
Use Case ID	UC-002
Actor(s)	Guest
Brief Description	An actor attempt to login to the system.
Pre-conditions	<ul style="list-style-type: none"> • The internet connection of the actor is available • The actor has access to the website • The actor is on the login page
Flow of events	<ol style="list-style-type: none"> 1. The actor enters the email and password 2. The actor presses the enter key of the login button 3. The user logged in to the system and returned to the homepage.
Alternative flows and exceptions	<ul style="list-style-type: none"> • The actor can exit and close the website during any step above. • In step 1, the user can click on “Forgot Password” to execute the use case “Reset Password”. After completing the above use case, the user will return to step 1. • In step 3, if either the provided email or password are incorrect, an error message will be displayed, and the actor will go back to step 1.
Post-conditions,	The result of the login attempt should be indicated
Priority	High
Behavioral requirements	<ul style="list-style-type: none"> • The password entered by the user should be hidden
Non-behavioural requirements	Efficiency: The website's response time should not be longer than 3 seconds. This can be limited by internet quality.
Assumptions	The user understands the display language of the website. There is no internet interruption while performing the above actions.

3.1.2.2. Sequence Diagram

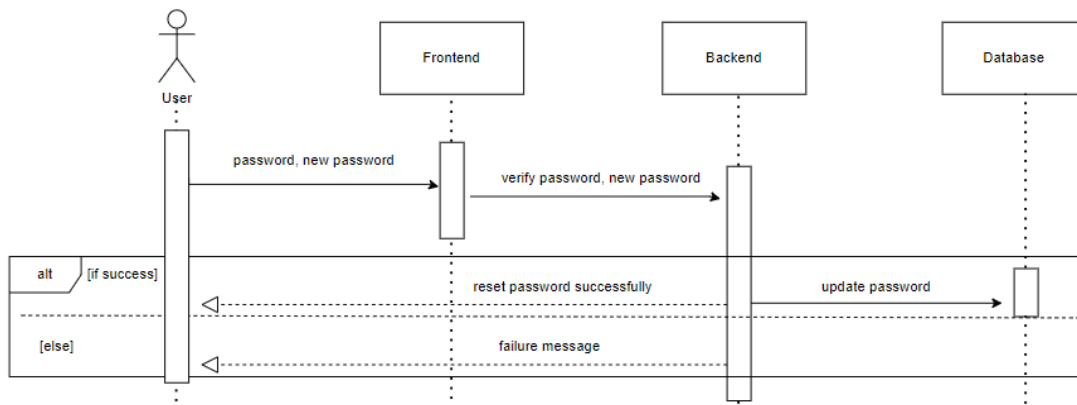
■ Registration



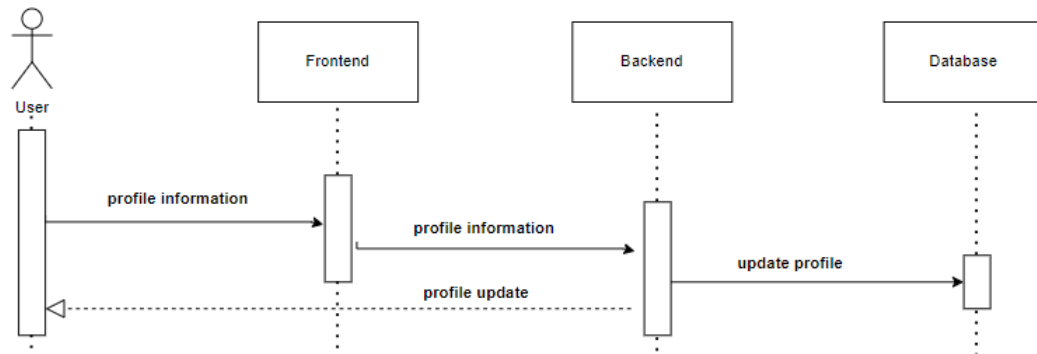
■ Login



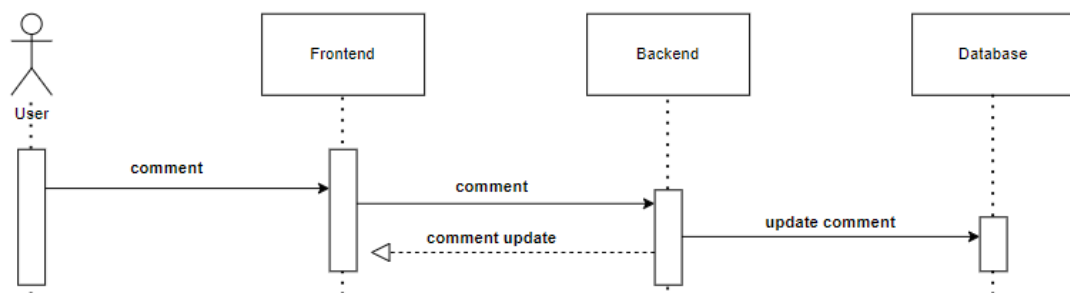
■ Reset Password



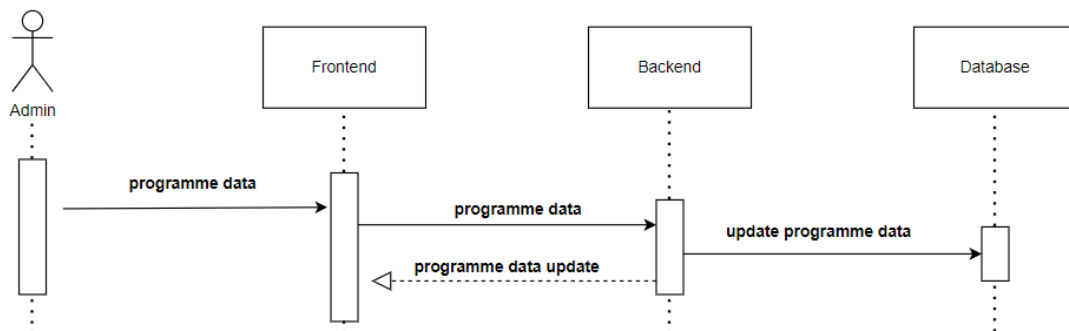
■ Edit Profile



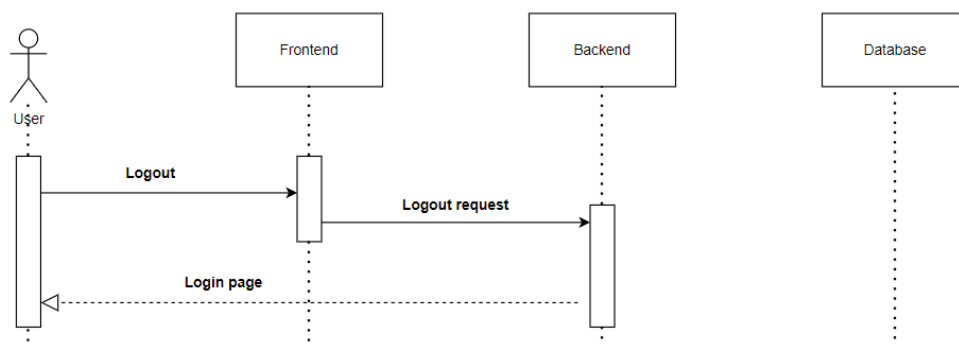
■ Comment



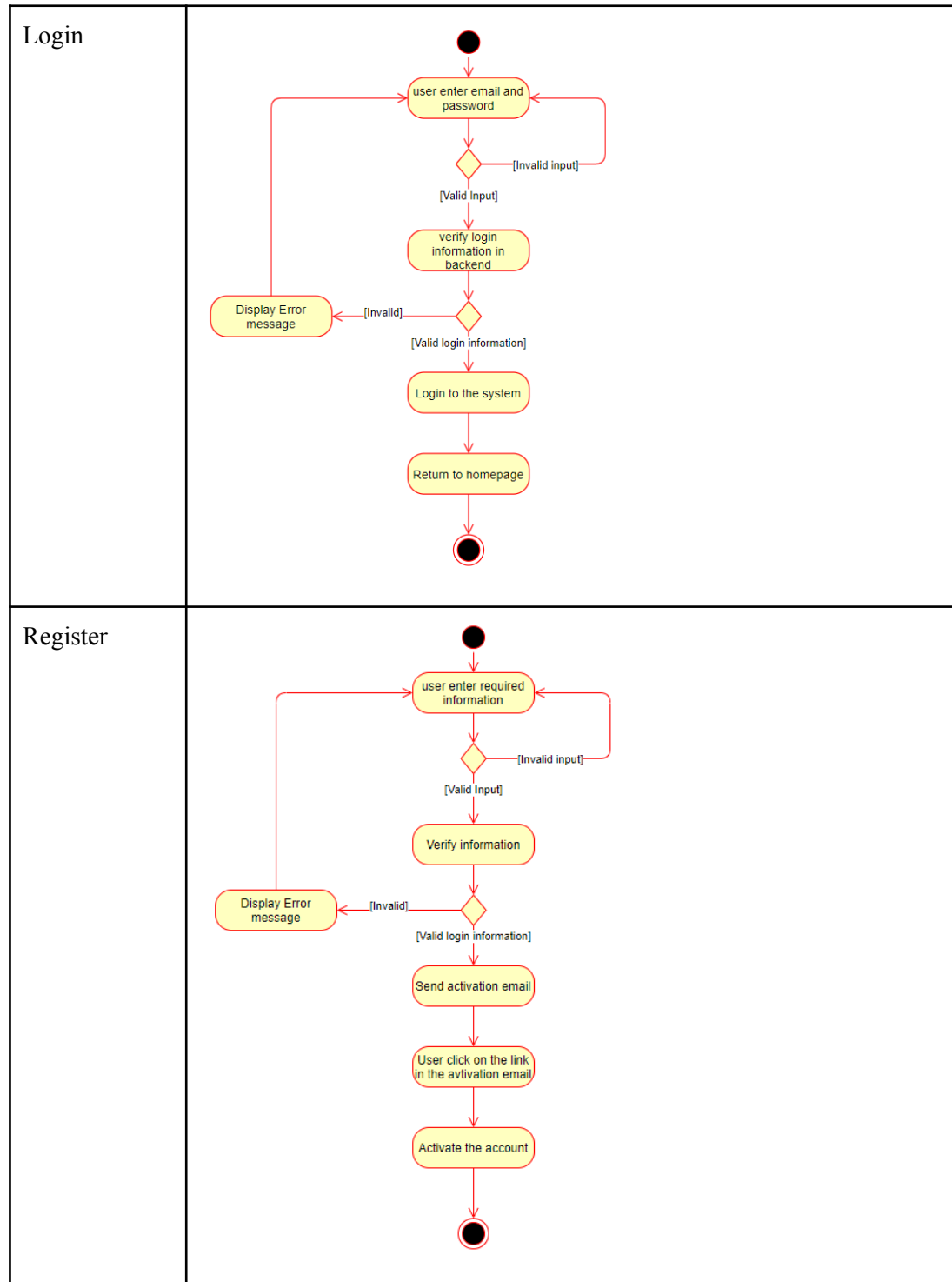
■ Edit Programme Data



■ Logout



3.1.2.3. Activity Diagram



3.1.3. Functionality

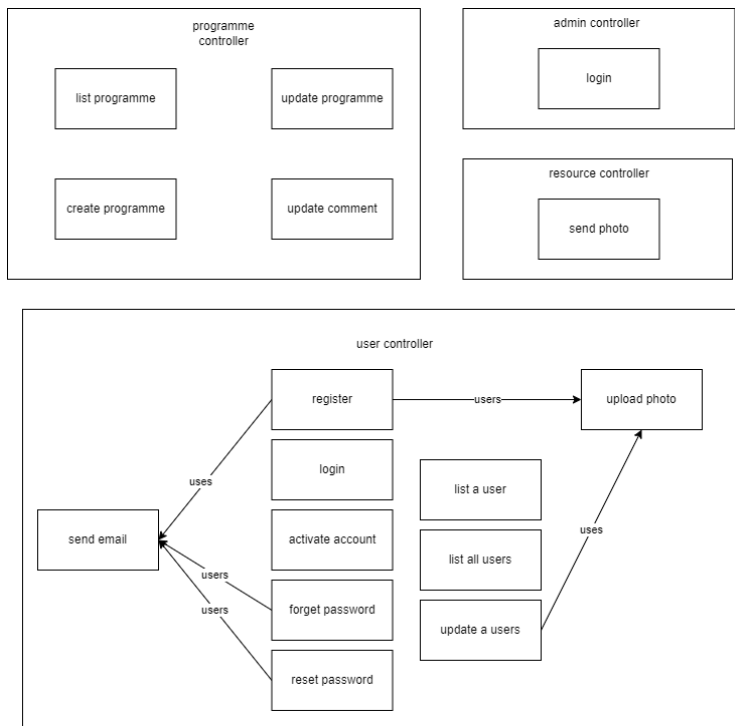
This component provides a graphical user interface for the users to interact. Any actor should be able to find all the features they have access to in the front end. The display of the front end should be changed dynamically according to the type of actor the current user is, and actors should not be able to access the features that they do not have access to by any means. For example, a general user should not access the admin-specific features.

3.1.4. Procedures and Functions

Function	Description
Register()	The user can create a new account by inputting the college or university email, nickname, password, school, programme, admission year, cGPA, public exam result, and uploading a profile picture. The function will first check whether the username or email is registered, which users cannot use the registered information. If all the input is valid, then the system will send an activating email. Once the user clicks the activated link, the account will be registered.
Login()	The user inputs the school email and password for a registered account. If the email or password does not match the one in the database, it will output an alert message.
ResetPassword()	The user input the registered email, and the system will send an email to the corresponding email. The user could then follow the instruction to reset the password.
DisplayCollegeChart()	Once redirected to a specific programme page or statistics page; the page would generate a pie chart that shows the background college from the database.
DisplayGPADistributionChart() ()	Once redirected to a specific programme page or statistics page; the page would generate a scatter chart which shows the GPA distribution from the database.
DisplayInfoTable()	Once redirected to a specific programme page or statistics page; the page would generate a table which shows all the comments with corresponding information from the database.
Comment()	If the user has signed in, the user could input a comment to a specific programme and update the database.
ChangeLanguage()	Change the interface language of the website.

3.2. backend

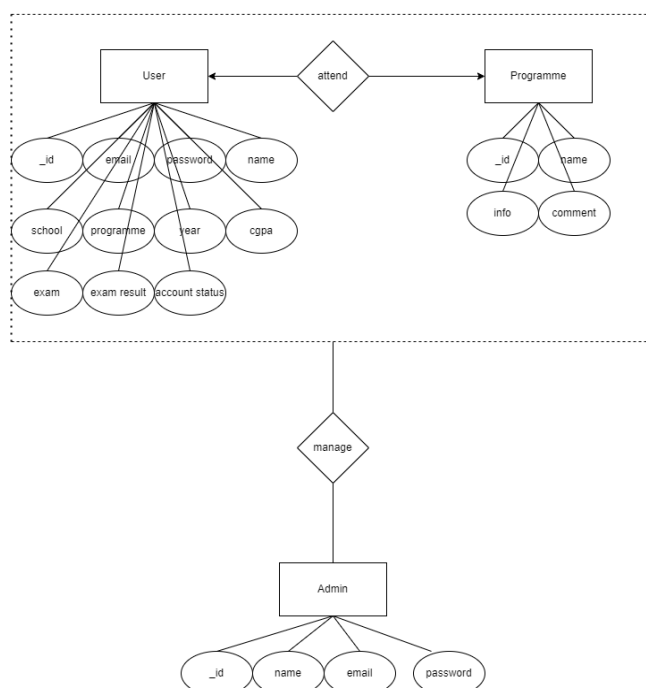
3.2.1. Structural Diagram



3.2.2. UMLs

The UML use case diagram, sequence diagram, and activity diagram are mentioned in the front end UMLs section.

3.2.3. ER-Diagram



Since MongoDB is a NoSQL database. The ER diagram cannot change to schema directly like SQL database. Therefore, the schema is decided according to the display of the front end, so all documents in a database can be fetched in one IO. Thus, the IO won't have a "join" operation, so the efficiency is enhanced.

3.2.4. Functionality

The backend component has several functions supporting the front end features. The user can register and then activate his account. Furthermore, he can login, forget, and reset his password. For admin, he can log in, and update the user's information. Regarding user

and programme data, it can be created, listed and updated.

3.2.5. Procedures and Functions

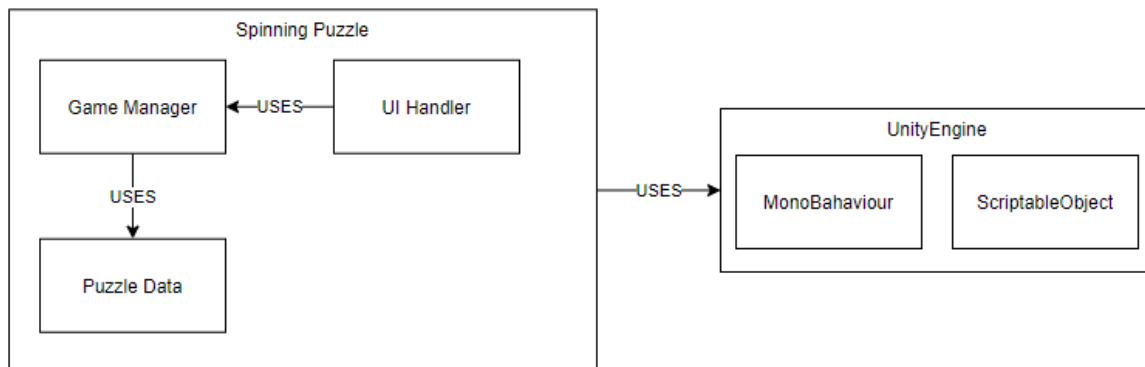
The procedures and functions are described from the perspective of the backend. The order is arranged according to the position of the routes.js function.

Function	Description
sendPhoto()	Send a photo to the client.
userRegister()	Register an account for the user.
adminLogin()	Admin logins his account.
userLogin()	User logins his account.
activateAccount()	Activate a user account after the user clicks the URL in the activated email.
userUpdatePassword()	Update a user's password.
userOrAdminUpdatePassword()	Update an admin or a user's password.
userForgetPassword()	The user attempted to reset the password with the email. The server will first check if the email is registered, and then send out an email to the mailbox if the email is registered.
userListAll()	Return a list of data of all users.
userListOne()	Return the data of a specific user by email.
userUpdate()	Update the info of a specific user. Return the status of whether the operation succeeded or not.
programmeList()	Return a list of all programmes.
programmeCreate()	Add a new programme to the database. Return the status of whether the operation succeeded or not.
programmeUpdate()	Update the info of a specific programme. Return the status of whether the operation succeeded or not.

programmeCommentUpdae()	Add a new comment and add it to the database when a user submitted a new comment. Return the status of whether the operation succeeded or not.
-------------------------	--

3.3. Minigame - Spinning Puzzle

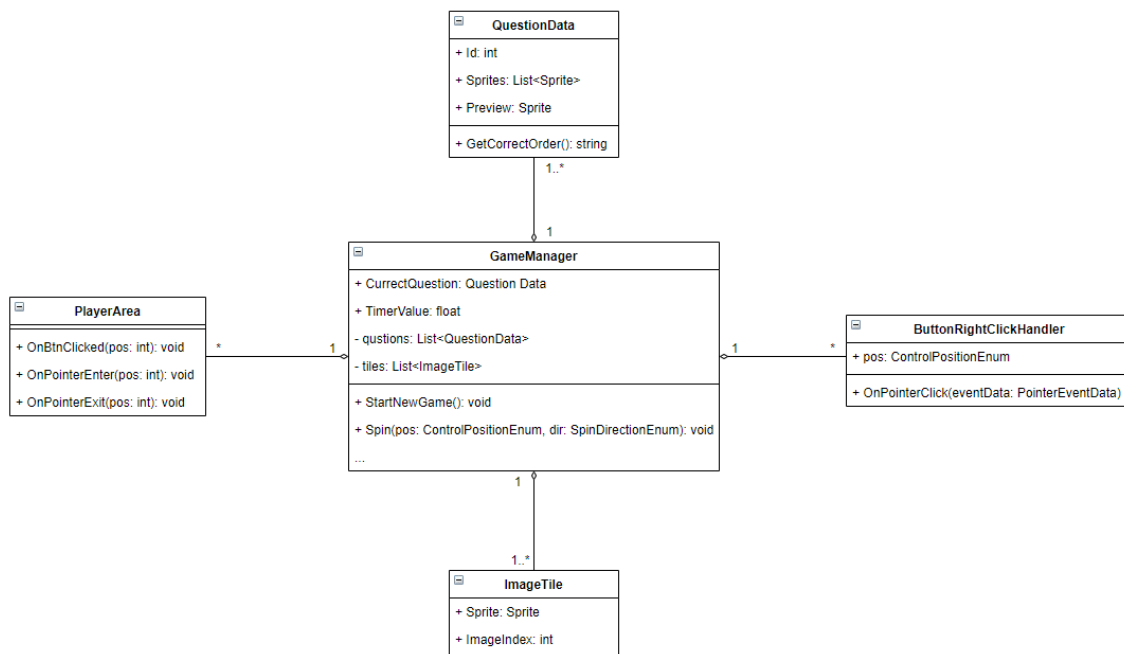
3.3.1. Structural diagram



The game uses many different modules from the Unity Engine, including but not limited to MonoBehaviour and ScriptableObject.

3.3.2. UMLs

3.3.2.1. Class Diagram



The class GameManager is the main class that governs the game flow, and contains the essential functions, that many other classes will use.

3.3.3. Functionality,

The spinning puzzle minigame is a small game project developed with Unity, a popular game engine, and is embedded in the website by WebGL. The users will only have access to the game after logging in to the system. The game starts with a picture divided into 3 by 3 images shuffled randomly; the users need to restore the puzzle by clicking on the specific region to rotate the position of a 2 by 2 region. The game is intended to train the user's puzzle-solving skills. In the future, a leaderboard will be implemented so that users can compete with each other on the minimum time spent to solve the puzzle.

3.3.4. Procedures and Functions

Procedure	Description
IsCorrect(): boolean	Determine whether the current puzzle is solved correctly.

Function	Description
Spin(pos: ControlPositionEnum, dir: SpinDirectionEnum)	Rotate the position of images in a 2 by 2 region. The argument <i>pos</i> determined which 2 by 2 region to rotate. Possible values included <i>TopLeft</i> , <i>TopRight</i> , <i>BottomLeft</i> , and <i>BottomRight</i> . The argument <i>dir</i> determined which direction to rotate; possible values included <i>Clockwise</i> and <i>CounterClockwise</i> .
ChangeDirection()	Toggle the rotating direction of the images between clockwise and counterclockwise.
StartNewGame()	Reset the timer and start a new game with a random image. The current game will be abandoned.
GameOver()	The current game will be over upon calling this function. The timer will be stopped and a game over message will be displayed to the player.

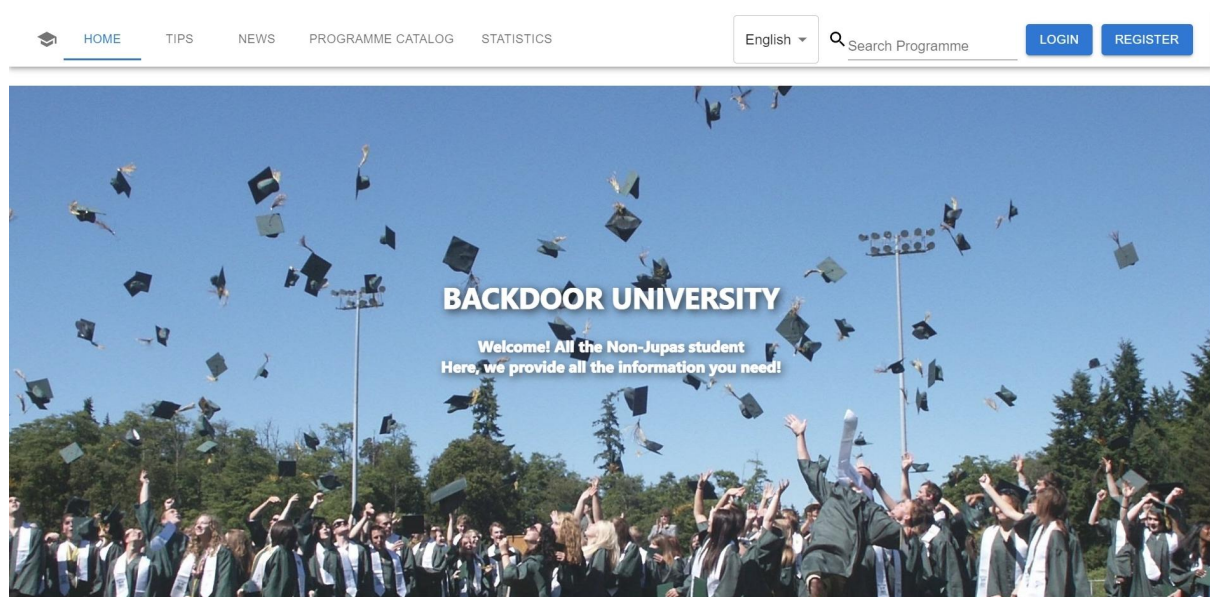
4. User Interface Design

4.1. Description of the User Interface

To enhance user-friendliness, we use Material-UI for the component design. Material-UI is a library based on Google's material design that the users are more familiar with. Thus, we need further explanation for the design. For example, we have provided a navigation bar on the top of the website, providing a clear destination.

4.2. Screen Images

Home Page



This is the home page UI shown to all users visiting the Backdoor University website. All other pages provide the navigation bar on the top. The primary purpose of the home page is to provide a general overview and introduce the mission and function of our website. It could intuitively indicate the current page to users and where they can go by buttons that could redirect to different pages by clicking.

Tips and News

University Interview





-How to prepare for a university interview

Being asked to attend an interview is the mark of an excellent application but remember that you'll be competing against other strong candidates. So, it's important to prepare well. These university interview tips listed below will help increase your chances of success.

-Practice answers to common questions

Most interview will ask 'Why do you want to study the subject?' and 'Why do you want to go to this university?'. Prepare the response to these question well, better memorize and polish it before the interview, but need to speak naturally and not like scripted.

Application Deadline




 <p>The Chinese University of Hong Kong (CUHK)</p> <p>Early Round Deadline: 17-11-2022 Regular Round Deadline: 05-01-2023 Extended Application Deadline: 31-05-2023 Apply Now</p>	 <p>The Hong Kong University of Science and Technology (UST)</p> <p>Early Round Deadline: 19-11-2021 Regular Round Deadline: 14-01-2022 Extended Application Deadline: N/A Apply Now</p>	 <p>The University of Hong Kong (HKU)</p> <p>Early Round Deadline: 17-11-2022 Regular Round Deadline: 24-08-2022 Extended Application Deadline: N/A Apply Now</p>	 <p>The City University of Hong Kong (CityU)</p> <p>Early Round Deadline: 15-11-2021 Regular Round Deadline: 13-01-2023 Extended Application Deadline: N/A Apply Now</p>
---	---	---	--

These two pages are to provide general advice and news on non-JUPAS to users.

Programme Catalog Page

Non-JUPAS Programmes

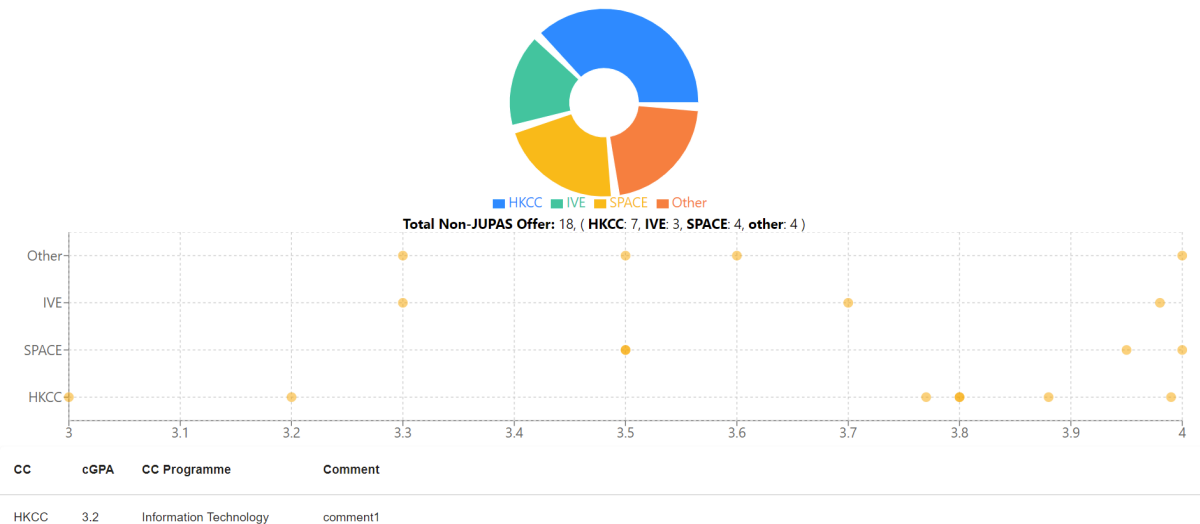
Choose a programme to check the specific statistics

	The Chinese University of Hong Kong (CUHK)	^
>	Artificial Intelligence	
>	Computer Engineering	
>	Computer Science	
	The University of Hong Kong (HKU)	▼
	The Hong Kong University of Science and Technology (UST)	▼

The programme catalogue is to list all of the non-JUPAS programmes and redirects the users to a specific programme. For easier searching, list buttons are employed to help categorise each by the university. Since there are hundreds of them in Hong Kong, the UI could be improved to nested list buttons or search by typing text.

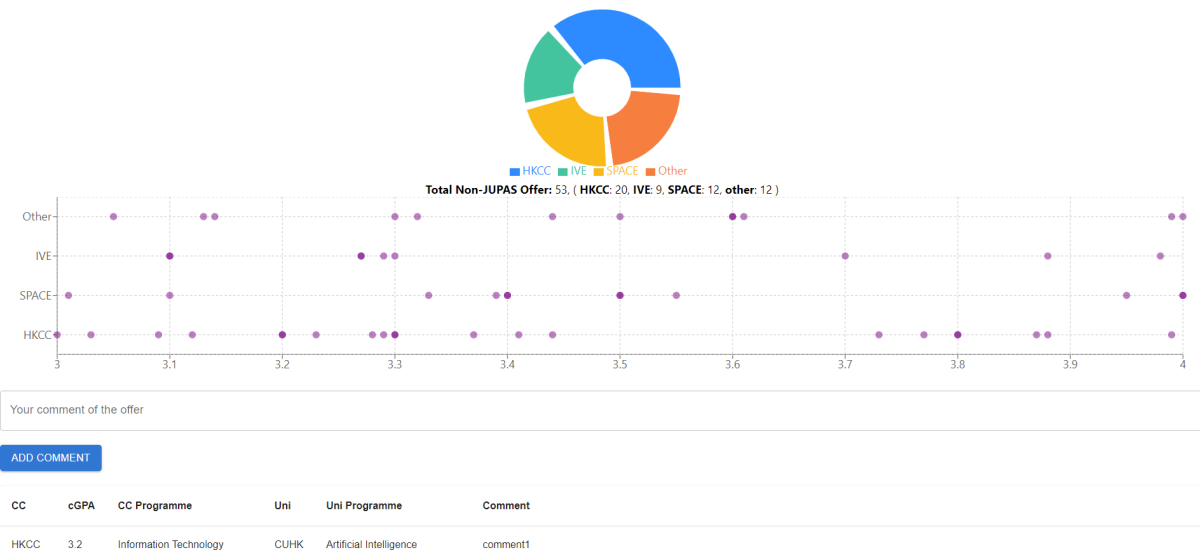
After redirecting to a specific programme, the user could view the shared admission information, including the college background by a pie chart, GPA distribution by scatter graph, and a detailed table for reference.

CUHK, Artificial Intelligence Non-JUPAS Admission Statistics



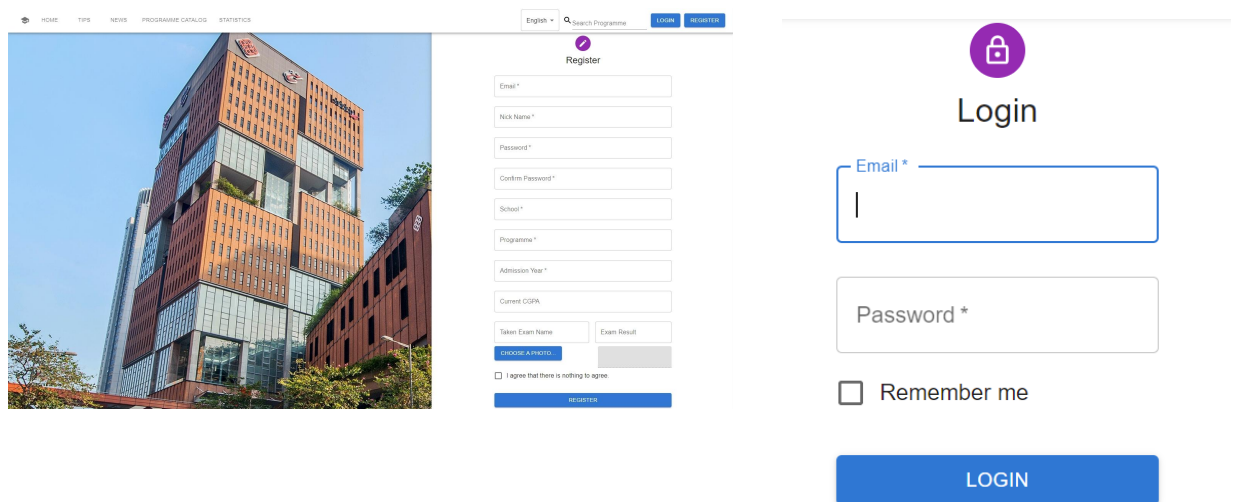
Statistics Page

General non-JUPAS Admission Statistics



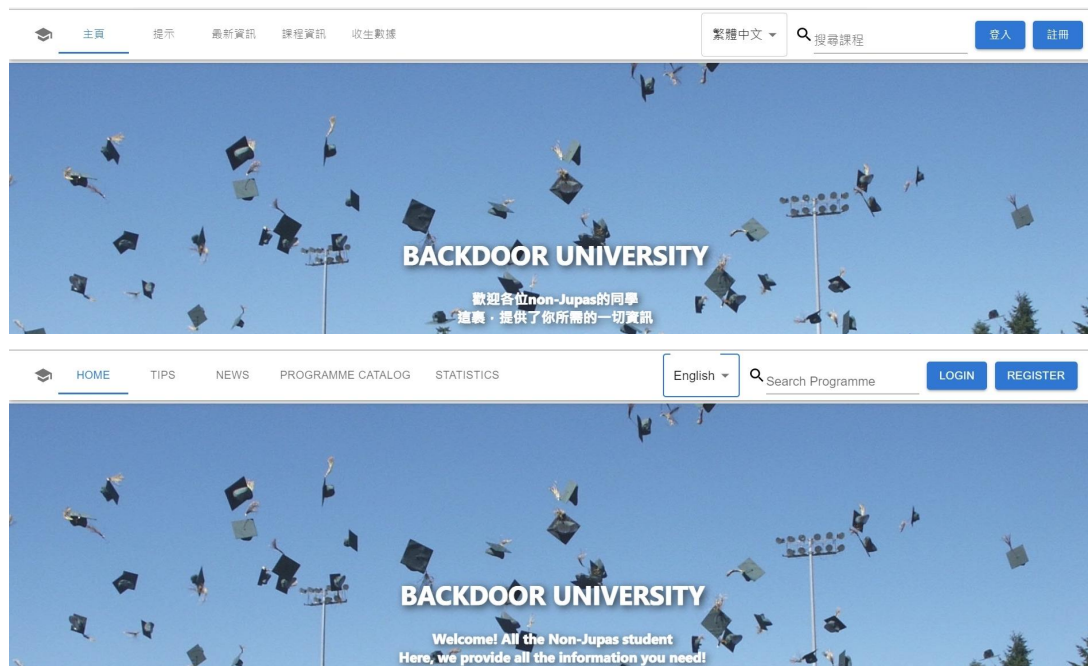
Similar to the page of a specific programme, this page gathers all programmes' data. All of the data are collected on this page. If the user had logged in, the user could comment and update the information, including the pie chart, scatter chart, and table on this general page and the specific programme page.

Register and Login Page



After clicking the register/login button in the upper right corner, it will redirect the user to these pages. Before signing in, users must sign up for an account by using a university email. The system will send an activating email; the user could activate the account by clicking the provided link. It enables the user to log in and then comment.

Switching between Chinese and English



Every page can change to the English or Chinese version by clicking the dropdown list on the navigation bar.

Admin Page

Email	Name	School	Programme	Status
user@user.com	user	CUHK	Computer Engineering	active
adc@lol.com	eral	HKCC	Computer Washing	active
tbc@slkdjfsdklfsdlk.com	flksdjfsdlkf	there	Computer Smashing	active
1155149068@link.cuhk.edu.hk	andrew	HKCC	Computer Science	active
1155142308@link.cuhk.edu.hk	eric4	HKCC	B.Sc. Computer Science	active
1155129248@link.cuhk.edu.hk	ricky	cuhk	eleg	unverified
1155142308cuhk@gmail.com	na1sdf	HKCC	Information Technology	active

1 row selected
Rows per page: 100
1-7 of 7

CHANGE PASSWORD
EDIT PROFILE
ACTIVATE ACCOUNT

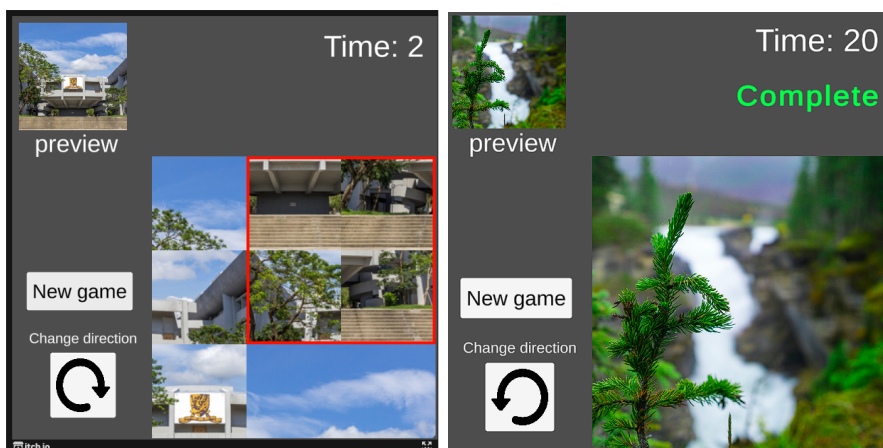
ID	Programme	Info	School	Type
625671a1d39706b8569cc523	B.Eng. in Computer Engineering	The Computer Engineering (CENGN) programme ...	CUHK	asso
6273c11ddb64f55d8f97bb79	B.Eng. in ds	Very good	CUHK	undergrad

1 row selected
Rows per page: 100
1-2 of 2

ADD PROGRAMME
EDIT PROGRAMME

After the admin login, they can see and edit the information of the user and programme data in the dashboard.

Game Page

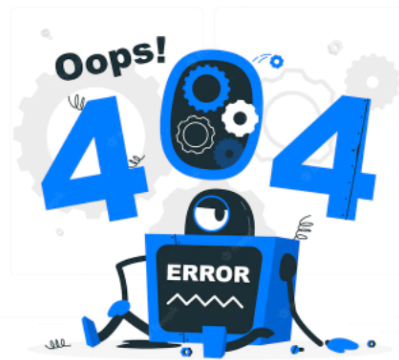


To provide some entertainment, a small puzzle game is provided to the user after they have logged in. The user must complete the puzzle by clicking the photo such that the photo shown inside the red frame will be rotated clockwise.

Error (404) Page

404 Not Found

The page you are looking for does not exist.



Server error vector created by stonyset - www.freepik.com

[← GO BACK TO HOME](#)

Whenever users try to reach a website that could not be found on the server, they will be redirected to this page. They can go back to the Home Page by clicking the button.

5. Test

5.1. Test overview and test plan

For the backend, we will conduct Bcrypt testing and API testing using black-box testing for the backend.

5.1.1. Bcrypt testing

Regarding passwords, the program encrypts them.

For initializing the password, the password is encrypted by Bcrypt with the salt equal to 10 and stored in the database, so the admin cannot get the password.

For checking the password, the procedure is the following. When the user login, he will need to enter the password, which will be encrypted, then compared to the encrypted password. As a result, the database password can never expose anyone except the user.

The detailed test cases are in the appendix.

5.1.2. API

Backend API black-box testing, we use Postman as a driver. The detailed test sets are attached in the appendix.

5.1.3. UI Behavior testing

Black box testing will be conducted to test if the UI behaves correctly, such as by displaying the correct output and content. The detailed test sets are attached in the appendix.

6. Lesson Learned

We have learned two main aspects: one is software technology, and another one is project management.

Since the lecture did not teach solid skills to students in terms of software technology, we had to seek suitable frameworks for the project. To challenge ourselves toward a professional standard, we have chosen the most popular framework: React.js, Express.js, MongoDB, and Cloud employment. Half of our team members knew nothing about the framework, but another half of our team had some experience, which led us to achieve a standard close to the commercial level. Therefore, some of us learned the front-end framework and its corresponding programming skills; on the other hand, the rest became more sophisticated to the full-stack level and developed better front-end and back-end skills.

The third parties services made a significant impact on the project as well. Firstly, we had used many times to debug the email sending bug: the sending function suddenly did not work, then we eventually found out that the Google Cloud Platform changed its API policy. Hence, the Gmail service API is not available after some time. The documentation of API is confusing, and the answers from StackOverflow and the blogs are outdated. As a result, we decide to change the email client to Yahoo email to solve this problem. Secondly, the change of deployment method of Heroku affected our project. Originally, we used Github and Heroku integration to make a simple pipeline to deploy our backend. However, Heroku decided to drop this feature and disabled it recently. Therefore, with the rapid change made, we remade the pipeline. As a result, our group learned that we need to stay alerted about service modification when using “as a service (*aaS)”.

Since the given time is limited for project management, we have learned the importance of communicating with each team member and distributing workload according to different skill levels. Poor communication caused many difficulties; one is about not knowing the hurdle of each task, then some of our tasks just worked behind schedule by one or two team members. Therefore, if we could redo it, we should cooperate more between the front end and backend. Due to a lack of communication, we have wasted some effort on some features, such as some useless APIs. The performance of some features, like commenting, is prolonged. It takes about 1 second to reflect the change of input text, and third-party email service takes about a few minutes, which is unacceptable for user-friendliness. However, our group could not find an excellent solution to this problem.

7. Conclusion

The purpose of Backdoor University is to provide Non-JUPAS information to the student to help them make better decisions. We create Backdoor University to collect the admission statistic from the mentors so that the users can understand their ability and apply to the university program wisely.

As some of us enter university through Non-JUPAS, we understand their confusion, especially since there is not much information. We are glad to see Backdoor University well implemented. We know that Backdoor University has room to improve. We will keep enhancing the UI and add more functions to provide a better user experience. We hope that we can publish a more completed version of it and be able to help the students.

Appendix (Case-N)

Bcrypt test sets

Test set 1

Purpose	To encrypt a password
Inputs	password: 12345678
Expected Outputs	password in the database is "\$2a\$10\$kqqYZPCuG4jrMKcKxfNEheNMPD5C7z2kv0KT5EPJXaq0esz0UMuei"
Pass/ Fail Criteria	The test fails if the password is not encrypted

Test set 2

Purpose	To compare a password
Inputs	password: 12345678, password in the database: "\$2a\$10\$kqqYZPCuG4jrMKcKxfNEheNMPD5C7z2kv0KT5EPJXaq0esz0UMuei"
Expected Outputs	true
Pass/ Fail Criteria	The test fails if the output false

Backend API test sets

Test set 1

Purpose	To register an account for the user
Inputs	email:1155142308@link.cuhk.edu.hk password:123 name:eric8 school:HKCC programme:Information Technology admissionYear:2 cgpa:3 examname:DSE result:20
Expected Outputs	200 { "msg": "Please check the verification email, including spam folder" }

Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong or return code is incorrect.
---------------------	--

Test set 2

Purpose	To log in as a user or admin
Inputs	email:1155142308cuhk@gmail.com password:12345678
Expected Outputs	200 { msg: "Login successful" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.

Test set 3

Purpose	To activate account 1. Go to the registered email 2. Click the activated link
Inputs	activated link
Expected Outputs	200 { msg: "Account is activated" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.

Test set 4

Purpose	To update password
Inputs	email:1155142308cuhk@gmail.com old password:12345678 new password: 123456789
Expected Outputs	200 { msg: "Modify password successful" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or return code is wrong.

Test set 5

Purpose	To forget and reset a password
Inputs	email:1155142308cuhk@gmail.com

Expected Outputs	200 { msg: "The system has reset a new password. Please check your email for it" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or return code is wrong.

Test set 6

Purpose	To list a user by email or user name
Inputs	email:1155142308cuhk@gmail.com
Expected Outputs	200 { "currProgramme": { "school": "HKCC", "programme": "Information Technology", "type": "undergrad", "admissionYear": 2, "cgpa": 3 }, "exam": { "name": "DSE", "result": "20" }, "_id": "6273a62aced9d954d775da08", "email": "1155142308@link.cuhk.edu.hk", "name": "eric8", "photo": "river.jpg", "status": "active" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or return code is wrong.

Test set 7

Purpose	To list all users
Inputs	
Expected Outputs	200 [{ "currProgramme": { "school": "CUHK", "programme": "Computer Engineering", "type": "undergrad", "admissionYear": 2014, "cgpa": 0.1 }, },]


```

    "exam": {
      "name": "DSE",
      "result": "11"
    },
    "_id": "6255bc3aac33b1a3f6e673e7",
    "email": "user@user.com",
    "name": "user",
    "photo": "4c9b06a5-0793-4f15-8eac-17be57e699b9",
    "status": "active"
  },
  {
    "currProgramme": {
      "school": "HKCC",
      "programme": "Computer Washing",
      "type": "undergrad",
      "admissionYear": 2002,
      "cgpa": 0.1
    },
    "exam": {
      "name": "A Level",
      "result": "3"
    },
    "_id": "6255bc90ac33b1a3f6e673e9",
    "email": "adc@lol.com",
    "name": "eral",
    "photo": "0d57952e-0a6b-4325-b913-f879c3c94a33",
    "status": "active"
  },
  {
    "currProgramme": {
      "school": "there",
      "programme": "Computer Smashing",
      "type": "undergrad",
      "admissionYear": 2000,
      "cgpa": 0
    },
    "exam": {
      "name": "A Level",
      "result": "3"
    },
    "_id": "6255bce2ac33b1a3f6e673eb",
    "email": "tbc@sldkjfsdklfjsdlk.com",
    "name": "flksdjfsdlkf",
    "photo": "979d98e3-20f5-4b29-a860-ef5b1a966e1f",
    "status": "active"
  },
  {
    "currProgramme": {
      "school": "HKCC",
      "programme": "Computer Science",
      "type": "undergrad",
      "admissionYear": 2018,
      "cgpa": 3.5
    },

```

	<pre> "exam": { "name": "DSE", "result": "19" }, "_id": "62569cd5825237c032313057", "email": "1155149068@link.cuhk.edu.hk", "name": "andrew", "photo": "6255bdd8-a623-4209-a3de-aa8df3a5127d", "status": "active" }, { "currProgramme": { "school": "HKCC", "programme": "Information Technology", "type": "undergrad", "admissionYear": 2, "cgpa": 3 }, "exam": { "name": "DSE", "result": "20" }, "_id": "6273a62aced9d954d775da08", "email": "1155142308@link.cuhk.edu.hk", "name": "eric8", "photo": "river.jpg", "status": "active" }, { "currProgramme": { "school": "cuhk", "programme": "eleg", "type": "undergrad", "admissionYear": 2019, "cgpa": 4 }, "exam": { "name": "", "result": "" }, "offer": { "school": "cuhk", "programme": "eleg" }, "_id": "6273ab40bf3aa76bc12d0e08", "email": "1155129248@link.cuhk.edu.hk", "name": "ricky", "photo": "eb10fcbc-7209-48e8-b0f4-0ea2bd8b0062.jpg", "status": "unverified" }] </pre>
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or return code is wrong.

Test set 8

Purpose	To update a user's information
Inputs	email:1155142308cuhk@gmail.com name:eric4 school:HKCC programme:B.Sc. Computer Science admissionYear:2 cgpa:3 examname:DSE result:20 offerSchool:CUHK offerProgramme:B.Eng. Computer Engineering type:asso
Expected Outputs	200 { "msg": "User information is modified" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.

Test set 9

Purpose	To update a user or admin's password If the caller is an admin, he can modify the password without validating the user's information
Inputs	email:1155142308cuhk@gmail.com
Expected Outputs	200 { "msg": "User information is modified" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.

Test set 10

Purpose	To list all programmes
Inputs	
Expected Outputs	200 [{

	<pre> "_id": "625671a1d39706b8569cc523", "school": "CUHK", "programme": "B.Eng. in Computer Engineering", "type": "asso", "info": "The Computer Engineering (CENGN) programme was formally established when the Faculty of Engineering was inaugurated in 1991. It is a balanced programme with an emphasis on both computer hardware and software. It distinguishes itself from others by offering specialized training for students in computer design, mobile embedded systems, microprocessors and very large-scale integrated circuit (VLSI) design.", "comments": [] }] </pre>
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.

Test set 11

Purpose	To create a programme
Inputs	school:CUHK programme:B.Eng. in AI type:undergrad info:AI is good
Expected Outputs	200 { "msg": "Programme created successful" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.

Test set 12

Purpose	To update a programme
Inputs	newSchool:CUHK newProgramme:B.Eng. in ds type:undergrad info:Very good comments:[] oldSchool:CUHK oldProgramme:B.Eng. in AI
Expected Outputs	200 { "msg": "Programme is modified successfully" }

Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.
---------------------	---

Test set 13

Purpose	To update a programme
Inputs	school:CUHK programme:B.Eng. in ds content:In data science we trusted email:1155142308cuhk@gmail.com
Expected Outputs	200 { "msg": "Comment submitted" }
Pass/ Fail Criteria	The test fails if any input is missing, his type is wrong, or the return code is wrong.

UI Behavior Testing

Test set 1

Purpose	To go to the home page
Inputs	url: https://backdoor-university.netlify.app/
Expected Outputs	The home page of the website is displayed
Pass/ Fail Criteria	The test fails if by any means the homepage is not displayed. For example, a blank page is displayed, or any other page is displayed instead of the home page.

Test set 2

Purpose	To switch between languages
Inputs	1. Click on the selection box and select “Chinese” 2. Click on the selection box and select “English”
Expected Outputs	The interface language is changed according to the selected language, and the selection box will display the current selected language.
Pass/ Fail Criteria	The test fails if the interface language is not the language we have selected at the moment, or the selection box does not display the current displayed language correctly.

Test set 3

Purpose	To test if the website will be displayed in the same language as the system
---------	---

	language by default, or display in English if the website does not provide the system's language. This has to be tested in incognito mode since after the website first run, the default language will be cached locally as the user preference.
Inputs	<ol style="list-style-type: none"> 1. Change the system language to Chinese 2. Open the website in a incognito window 3. Close the window 4. Change the system language to Spanish 5. Open the website
Expected Outputs	<ul style="list-style-type: none"> - In step 2, the interface language of the website should be Chinese. - In step 5, the interface language of the sweb site should be English.
Pass/ Fail Criteria	<p>The test fails if:</p> <ul style="list-style-type: none"> - The interface language is not the language we have selected at the moment - The selection box does not display the current displayed language correctly. - The texts in the website do not displayed or do not displayed correctly if the system language is neither Chinese or English.

Test set 4

Purpose	To test if the website will direct the user to the 404 page if the route of the url is incorrect
Inputs	url: https://backdoor-university.netlify.app/9qja90fh90w3rj
Expected Outputs	The 404 page is displayed
Pass/ Fail Criteria	The test fails if the 404 page is not displayed by any means.

Test set 5

Purpose	To test if the error message will be displayed if the user is trying to access the user profile of a user that does not exist.
Inputs	url: https://backdoor-university.netlify.app/user/{Random_String_That_Does_Not_Exist_In_The_User_Database}
Expected Outputs	The user not found message is displayed instead of the user profile
Pass/ Fail Criteria	The test fails if the message is no displayed

Test set 6

Purpose	To test if the UI is adaptive to the mobile view
Inputs	Shrink the window size of the browser.
Expected Outputs	For the navigation bar, the buttons should be collapsed into a single button, which will open a drawer that displays all the buttons.

Pass/ Fail Criteria	The test fails if the buttons in the navigation bar remain unchanged.
---------------------	---

Test set 7

Purpose	To test if the UI will be dynamically changed according to the type of the current user.
Inputs	<ol style="list-style-type: none"> 1. Open the website unlogged-in 2. Login as a general user 3. Logout 4. Login as an admin user
Expected Outputs	<ul style="list-style-type: none"> - In step 1, the login button should be displayed in the top right - After step 2, the login should be replaced by the profile icon of the current logged-in user. Clicking on the icon will displays the “profile” button, which lead the user to their profile page, and a “logout” button. - After step 3, the display should be the same as in step 1. - After step 4, the display should be the same as in step 2, except that instead of a “profile” button, the “dashboard” button should be displayed. The user should be directed to the admin page after clicking on the button.
Pass/ Fail Criteria	The test fails if the buttons in the navigation bar remain unchanged.