LONDON METROPOLITAN UNIVERSITY

islington college

(इस्लिङ्टन कलेज)

Module Code & Module Title

CS5002NI SOFTWARE ENGINEERING

Assessment Weightage & Type

35% Individual Coursework

Year and Semester

2022-23 Spring

Student Name: Krish Bhattarai

London Met ID: 22067570

College ID: np01cp4a220071@islingtoncollege.edu.np

Assignment Due Date: 07/05/2024

Assignment Submission Date: 07/05/2024

Word Count (Where Required): 4583

Table of Contents

Table of Figures

Table of Tables

1. Introduction

The McGregor Institute of Botanical Training is a training institute based in Ireland and located at Godawari, Lalitpur. It provides various undergraduate and postgraduate courses specializing in agriculture and horticulture and is affiliated to Dublin City University. Recently, due to the sudden rise of peoples' interest in agriculture, the institution wants to provide various short-term courses related to horticulture to the interested people. They also plan on selling different varieties of plants for minimal fee and in some instance, for no cost. They also plan to build a forum for people who require opinions from the experts or have an interest in plants where people with similar interests can discuss their ideas and organize programs.

The prototype model is chosen for this software development. A prototype is a model or a sample which is created to test the concept of a system or a process. A prototype allows designers and developers to find defects in the early stage making the vision clearer for the final product. This reduces the risk in the long run as defects can be found early. The prototype model has several advantages compared to other methodologies. In comparison to other methodologies, the prototype model is more flexible as making sudden changes to the prototypes is easy. It is less risky as creating multiple prototypes helps with identifying the risks early on. This methodology requires active participation and gathering feedback from its clients leading to a more accurate product in the end. As making changes is easier in this model, actively gathering user feedback and making changes accordingly is easier.

2. Methodology

The methodology used for this coursework is Prototype Model as this model gives the clients a hands-on experience of how the system might end up looking and functioning, leading to quicker feedback from the clients and the stakeholders. This assures that the clients and the stakeholders getting the product is what they want.

The steps for this methodology are (Rana, 2020):

- **Initial Communication:** The initial communication is about meeting the clients and stakeholders and is responsible for collection of information about the requirements and discussion of the product that the clients are looking for.
- **Planning:** The requirements gathered from the stakeholders and the clients is responsible for how the project is planned to be completed.
- **Design Modelling:** The initial design for the project, such as the user interface is created in this phase.
- **Prototype Development:** The initial design created is coded and developed so that the clients will be able to get a hands-on experience of the prototype which gives them an idea about how the system works.
- **Deployment, Delivery and Feedback:** In this phase, the clients and the stakeholders get a hands-on experience on how the system works giving them an idea about how the final product may come out to be. This stage also gives clients a chance to give feedback for the team to gather and work on. This phase keeps running until the clients are satisfied and the final product is deployed.
- **Final Product Design:** After the prototype phase, the work on the final product begins, this assures that the clients are getting exactly what they want. The final product is designed in this phase.
- **Testing, Deployment and Maintenance:** After the final product is designed and created, testing on the product begins and assures that there are no bugs in the system that the users might encounter while using it. Once the testing is completed, the product is launched into the market for the users

to use. Launching a product into the market means that it should be well maintained and consistently updated in a timely manner.

3. WBS

The proposed system is to develop a system for users to purchase plants, enroll on to different postgraduate, undergraduate and short-term courses. For this, a responsive system with a user-friendly interface providing users with an easy-to-use user experience. The system will also allow the admins of the organization to manage its users, manage the contents of the courses and manage the users inside the system.

Define Scope

Before starting the development of the system, the scope of the system must be defined. This includes what the system servers to do that is identifying the purpose of the system, the next step is to identify the right audience for the system. This helps find the right market for the system for advertising and marketing. After identifying the right audience for the system, the major features of the system should be figured out and must be prioritized first. Features such as purchase plant, enrol to courses, etc should be prioritized.

Requirement Gathering

After the scope for the system is defined, the requirement for the system needs to be gathered. This includes proper market research, interviewing the stakeholders which helps understand the vision of the system. This helps understand the system better leading to a better result. Stakeholders are kept in contact throughout the development of the system. While the requirements are being gathered, functional and non-functional requirements for the system must be gathered. This includes functional requirements like User Authentication, Order Management, etc, and non-functional requirements such as optimizing the code for less resource consumption, improving the useability of the system with easy to navigate user interface.

Concept Design

After Gathering the requirements for the system, various logos are designed, simple wireframes for the system are designed. This doesn’t necessarily show the result of the system but gives a simple concept of how the system might look like or function like on the future. This also helps to create a strategy in creating the system in the future which makes it easier to develop the system.

Prototyping

Using the knowledge from the concept design, various prototypes are created for the system. While creating prototypes, meetings with the stakeholders are done to ensure that they are satisfied with the system. Making changes in this stage is also easy to do. The prototypes are also well tested to reduce bugs and errors in the final product. Features and Accessibilities are also slowly implemented into the system during this stage.

Final Prototype

After the prototypes are created, the various prototypes are reviewed with the stakeholders. After the stakeholders have been satisfied with the prototype of their choice, developing, and optimizing of the system is done in this stage. Various external tools and services are integrated into the system to enhance the functionality of the system. The system is well tested to reduce bugs and errors in the system.

Launch Product

Before launching the final product, the system is checked for the last time. The system is then well polished. The system is then deployed for the public to use. After the system is deployed, the system is well monitored. Feedback from the users is collected and the system is then updated according to the users' feedback.

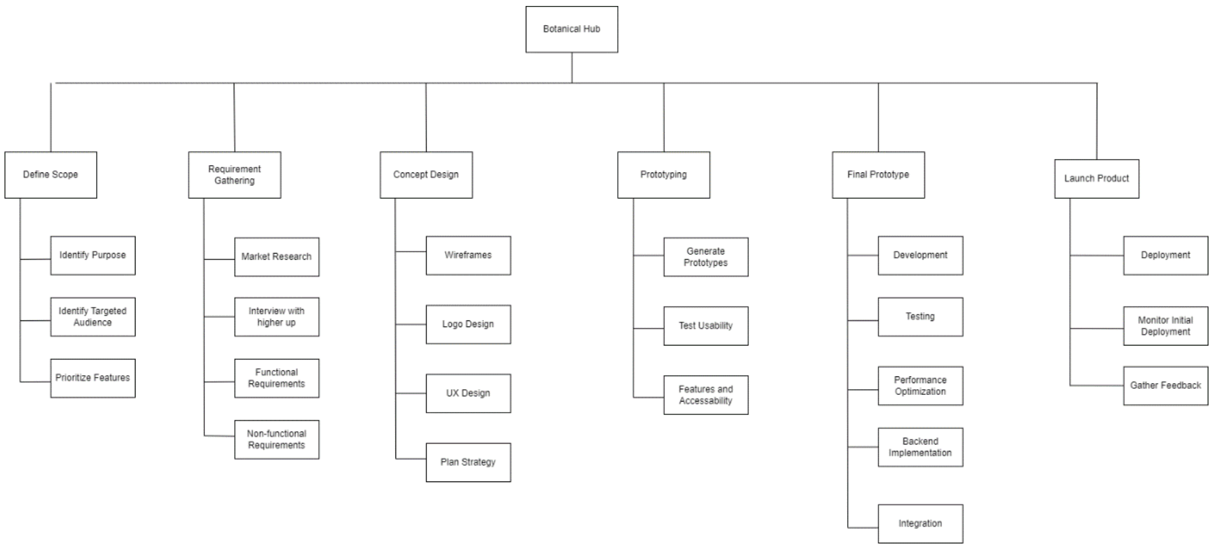


Figure 1: Work Breakdown Structure for Botanical Hub

Gantt Chart

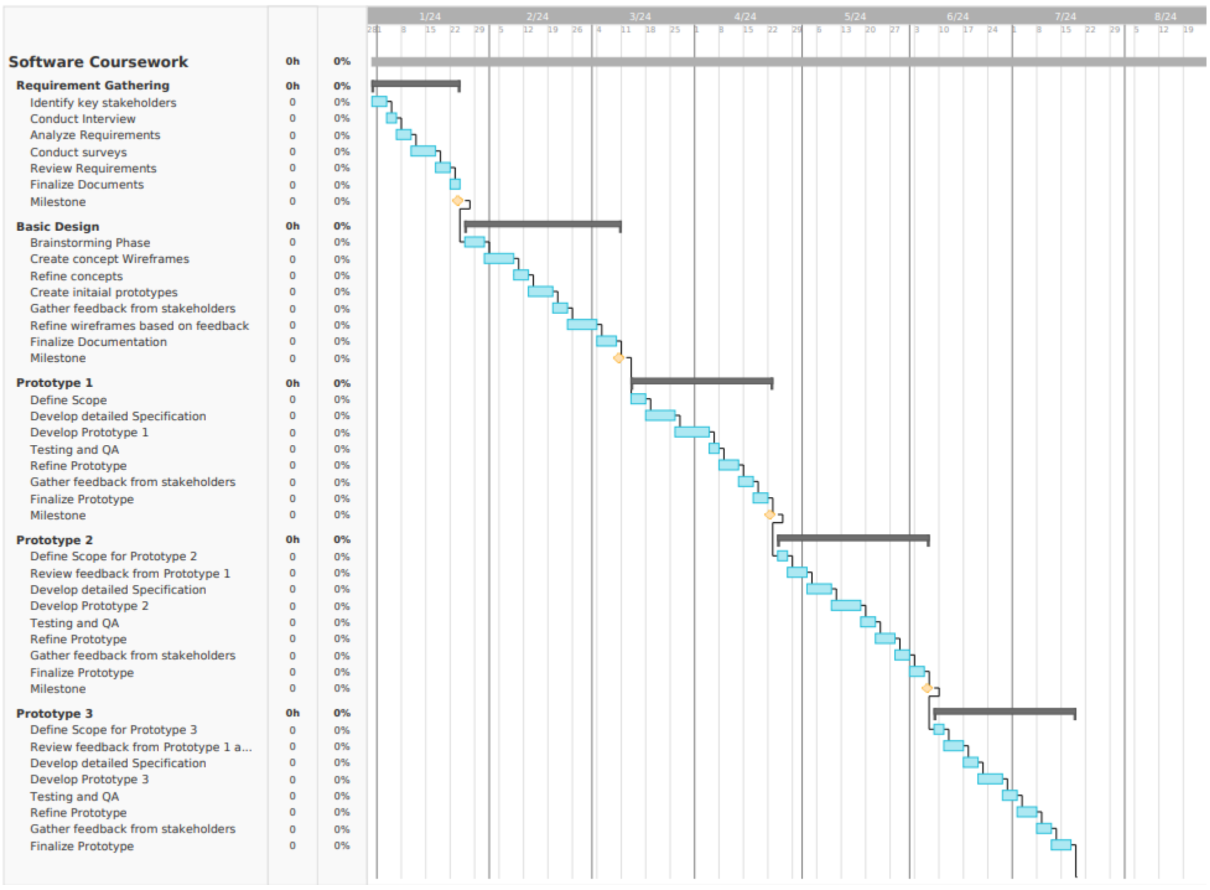
The Gantt chart is created to visualize the schedule of the activities and tasks for project management (Gantt.com, 2024).

The Requirement for the project is gathered for about a month, this includes identifying the major stakeholders for the project, conducting interviews with the stakeholders to understand their points and plans for the project. Requirements from the stakeholders are gathered and analysed. From the analysis, various surveys are conducted to review the response from the public. The requirements are then reviewed, and the initial documents are finalized.

After gathering requirements for the project, a basic design is created. This means that wireframe concepts are created to give a visual representation of the system to the stakeholders. Concepts are refined and a rough initial prototype is created. Feedback from the stakeholders is gathered and a document is created.

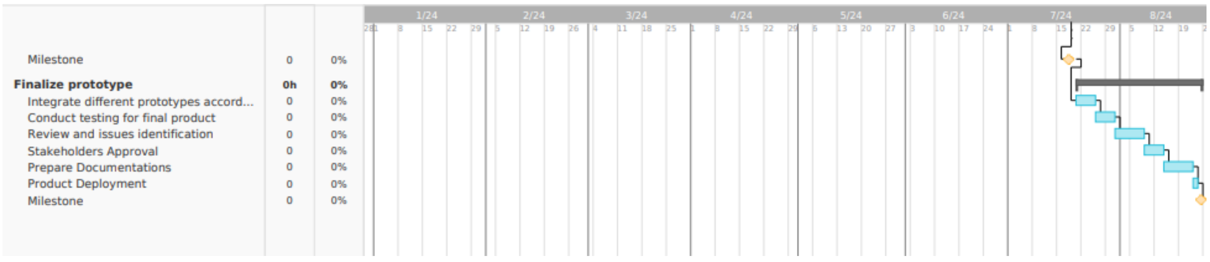
After the stakeholders have agreed with the initial prototype. Various prototypes are created and refined. The prototypes are tested to reduce bugs and errors for the final product. Feedback from the stakeholders is gathered and various prototypes are created until the stakeholders are satisfied.

A final prototype is created. Testing for the final prototype is done to have a well-polished prototype. The prototype is then reviewed, and issues are identified and fixed. The system is then reviewed with the stakeholders and after getting an approval from the stakeholders, the prototype is finalized. A final document is then prepared for the system which includes various plans for the system including deployment and maintenance plan. The product is then deployed to the market and user feedback is gathered.



Software Coursework	0h	0%
Requirement Gathering	0h	0%
Identify key stakeholders	0	0%
Conduct Interview	0	0%
Analyze Requirements	0	0%
Conduct surveys	0	0%
Review Requirements	0	0%
Finalize Documents	0	0%
Milestone	0	0%
Basic Design	0h	0%
Brainstorming Phase	0	0%
Create concept Wireframes	0	0%
Refine concepts	0	0%
Create initaial prototypes	0	0%
Gather feedback from stakeholders	0	0%
Refine wireframes based on feedback	0	0%
Finalize Documentation	0	0%
Milestone	0	0%
Prototype 1	0h	0%
Define Scope	0	0%
Develop detailed Specification	0	0%
Develop Prototype 1	0	0%
Testing and QA	0	0%
Refine Prototype	0	0%
Gather feedback from stakeholders	0	0%
Finalize Prototype	0	0%
Milestone	0	0%
Prototype 2	0h	0%
Define Scope for Prototype 2	0	0%
Review feedback from Prototype 1	0	0%
Develop detailed Specification	0	0%
Develop Prototype 2	0	0%
Testing and QA	0	0%
Refine Prototype	0	0%
Gather feedback from stakeholders	0	0%
Finalize Prototype	0	0%
Milestone	0	0%
Prototype 3	0h	0%
Define Scope for Prototype 3	0	0%
Review feedback from Prototype 1 a...	0	0%
Develop detailed Specification	0	0%
Develop Prototype 3	0	0%
Testing and QA	0	0%
Refine Prototype	0	0%
Gather feedback from stakeholders	0	0%
Finalize Prototype	0	0%

Figure 2: Fig Gantt Chart



Milestone	0	0%
Finalize prototype	0h	0%
Integrate different prototypes accord...	0	0%
Conduct testing for final product	0	0%
Review and issues identification	0	0%
Stakeholders Approval	0	0%
Prepare Documentations	0	0%
Product Deployment	0	0%
Milestone	0	0%

Figure 3: Fig Gantt Chart

4. Requirement Gathering

Goals and Objectives

The goal is to create a system for McGregor Institute of Botanical Training as there has been a sudden increase in people's interest in agriculture. Due to this sudden increase in interest towards agriculture, McGregor Institute plans in introducing different short-term courses, certification courses and selling various plants charging minimal fee and in some cases for free. The system also features a platform for its users to ask for recommendations which the experts in the system can answer. Apart from that, users get to post forums and interact with other plant enthusiasts.

Meetings with Stakeholders

After identifying the goal of the system, interview and meeting with the stakeholders should be held. This helps understand the needs of the system requirements and helps clear ambiguities. Interviews with the stakeholders helps build trust as it shows commitment over for the work. Several assumptions are made before meeting the stakeholders, communicating with them helps validate the assumptions made.

User Identification

The most important step before creating a system is identifying the targeted audience. This is very important as the system is to be developed for the users. This is important as the interface needs to be designed accordingly, the features and functionality of the system needs to accommodate the users of the system. Identifying the users helps with targeted market communication as it is important to test the software in the right market which helps with usable user feedback to fix issues before launching the system.

Functional Requirements

Functional Requirements describes how the system functions. This includes what the system does, how it behaves, etc. This is to ensure that the system has proper functionality to run and doesn't lack necessary functionalities. The system offers functionalities such as buying of plants, creating forums for users, enrolling into various courses, etc.

Non-Functional Requirements

Non-Functional Requirements relates to optimization of the system. This includes things like scalability, reliability, code optimization, etc. This is to improve performance of the software as this decreases resource consumption decreasing the system requirement to run and make the system run smoothly.

5. Use Case

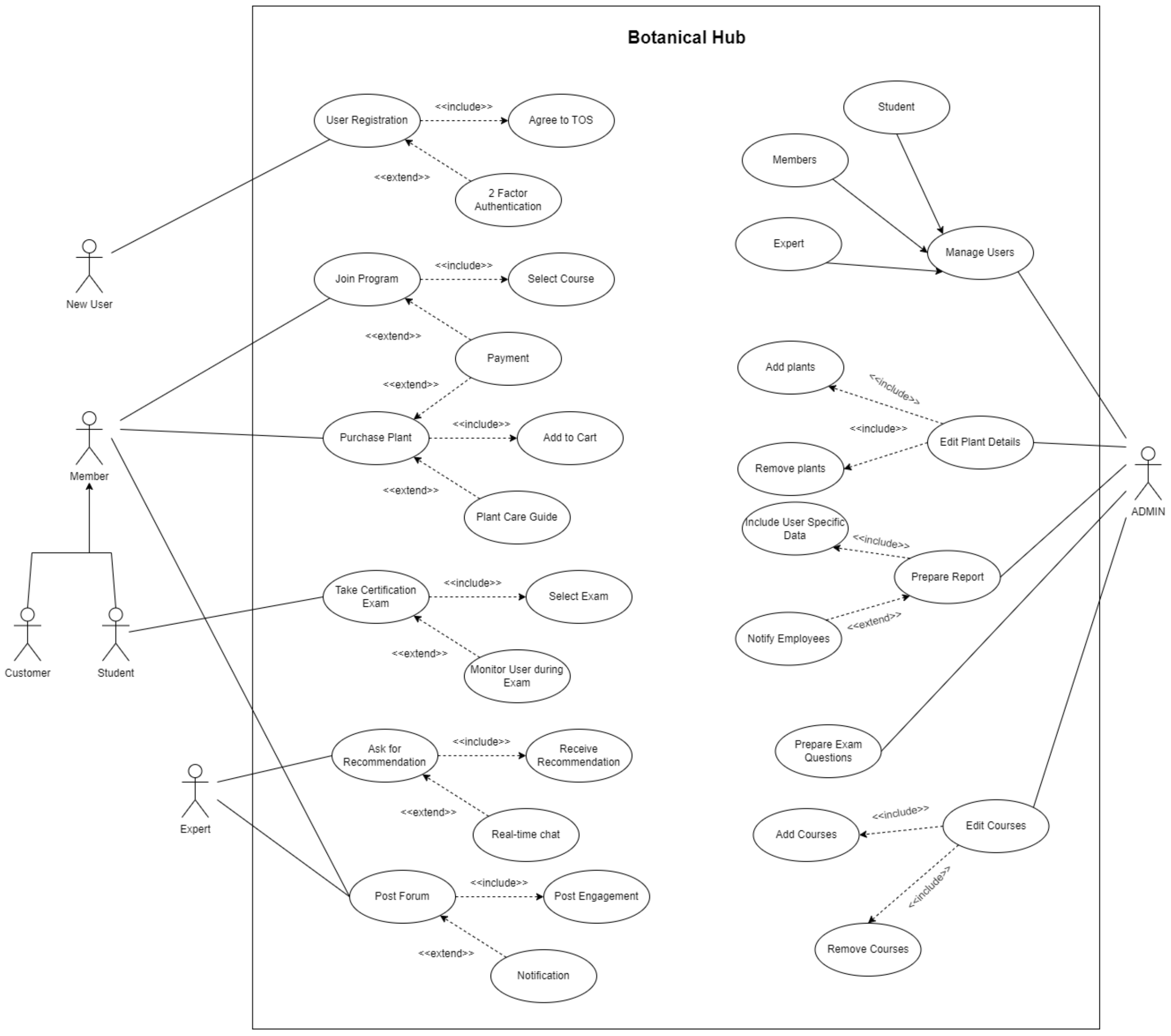


Figure 4: Use case for Botanical Hub

High level Use case for User Registration

Use Case: User Registration

Actor: New User (initiator)

Description: A new user provides their required personal details into the system. Before completing the registration process, the user must agree to the Terms and Conditions of the website. After providing the required information and agreeing to the terms and conditions, the user is registered into the system.

High level Use case for Join Program

Use Case: Join Program

Actor: Member (Initiator)

Description: Members have the option to join an available program. The student selects an available program of their choice. The student pays for the course if required. The payment is validated, and the role of the member is changed into student. Some courses are available for free.

High level Use Case for Purchase Plant

Use Case: Purchase Plant

Actor: Member (Initiator)

Description: Members can select the plants from the given options. The selected plants are added to the cart. When the payment is verified, the plant is delivered to the user. A separate care guide for plants can be purchased which the plant if needed.

High level Use Case for Take Certification Exam

Use Case: Take Certification Exam

Actor: Student (Initiator)

Description: Students who have been enrolled to a program can take certification exam. After a chapter of the course has been completed, the user is tasked to take a certification program. The results are monitored and if the student gets less than a certain points, the user isn't allowed to proceed further and must take the exam again. If the student gets equal to or exceeds certain points, the student is allowed to proceed further.

High level Use Case for Post Recommendations

Use Case: Post Recommendations

Actor: Member (Initiator)

Description: Members can post recommendations asking for recommendations. The experts reply to the members providing them proper recommendations. The members also have the option to have a real time text conversation with an expert if the experts are available.

High level Use Case for Post Forums

Use Case: Post Forums

Actor: Member (Initiator)

Description: Users can post forums and engage with the posted forum. Post Engagements such as upvotes, downvotes, comments, etc. can be done. The members have the option to turn on notifications for when someone creates a new post or engages with their own post.

High level Use Case for Manage Users

Use Case: Manage Users

Actor: Admin (Initiator)

Description: The admin has the ability to manage users. Roles such as expert is set by the admin. The admin can also manage users such as student, members and expert.

High level Use Case for Edit Plant Details

Use Case: Edit Plant Details

Actor: Admin (initiator)

Description: The admin can add or remove plants. Set prices of the plants and add descriptions.

High level Use Case for Prepare Exam Questions

Use Case: Prepare Exam Questions

Actor: Admin (initiator)

Description: The admin prepares questions for the certification exams.

High level Use Case for Edit Course

Use Case: Edit Courses

Actor: Admin (initiator)

Description: The admin can remove or add certain courses if required. The admin can also edit the contents inside the courses.

Expanded Use case Description for Join Program.

Use Case: Join Program

Actors: Student (initiator)

Description: A student provides their required personal details into the system. The student then chooses their specified course. The user then pays for the specified course if required. If the payment if required is done, the student is enrolled.

Typical Course of Events:

Actor Action	System Response
1. A student provides their personal details for course enrolment. 3. A student chooses a course. 5. The student pays the course fees.	2. The system presents the available courses to the student. 4. The student is displayed the price of the chosen course. 6.The student is enrolled into the course.

Table 1: Expanded Use Case for Join Program

Alternative Courses:

Line 7: The new student does not find a specific course. Use Case ends.

Line 8: The new student has insufficient funds in their bank account. Use Case ends.

Expanded Use case Description for Purchase Plant.

Use Case: Purchase Plant

Actors: User (initiator)

Description: A user logs into the system. The logged in user views the purchase plant section. The user chooses a plant or plants to buy. The user than pays for the selected plants. If the payment is verified, the plants are shipped to the user.

Typical Course of Events:

Actor Action	System Response
1. The user proceeds to the page with various plant.	2. The user clicks on the desired plant.
3. The user proceeds to the plant detail page.	4. The system views the plant detail page.
5. The user chooses a plant to buy.	6. The chosen plant is placed in the cart.
7. The user pays for the chosen the plant.	8. The payment is verified, and an invoice is generated.

Table 2: Expanded Use case for Purchase Plant.

Alternative Courses:

Line 9: The user doesn't see a specified plant. Use Case ends.

Line 10: The payment is not verified. Use Case ends.

6. Collaboration Diagram with Steps

A collaboration diagram is a visual representation of the interaction and representation of objects in a system. It is also referred to as communication diagram and sows how objects interact with each other to perform a specific task in in a particular flow of event (GeeksforGeeks, 2019). The following collaboration diagram is created for PurchasePlant. It represents a detailed flow of how the purchasing of a plant is done.

Possible classes for the PurchasePlant UseCase:

- CartPage
- PaymentGateway
- OrderSuccess
- GenerateInvoice

Description for the objects of the PurchasePlant usecase:

- A CartPage object is created to store the details of the plants that the users want to purchase.
- A PaymentGateway object is created to process the payment of the plants.
- An OrderSuccess object is created to verify and confirm the payment.
- A GenerateInvoice object is created to store the invoice generated.

The objects for PurchasePlants include the following:

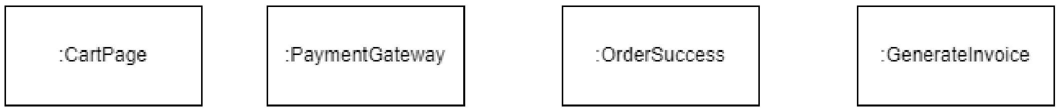


Figure 5: Objects for PurchasePlants.

A control object is added:

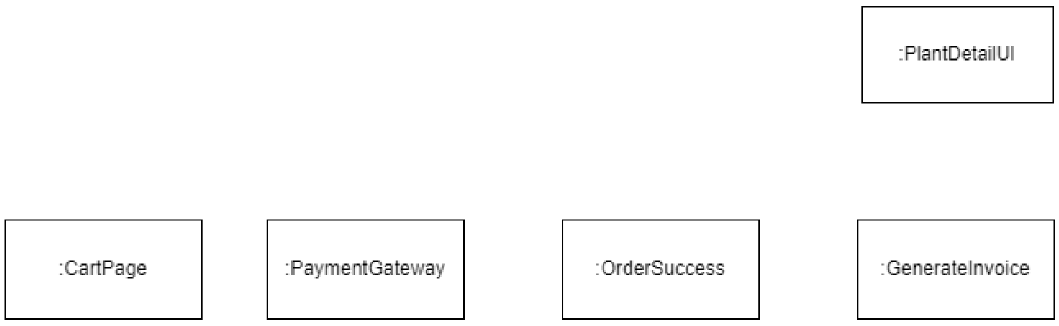


Figure 6: Control object.

A boundary object is then added:

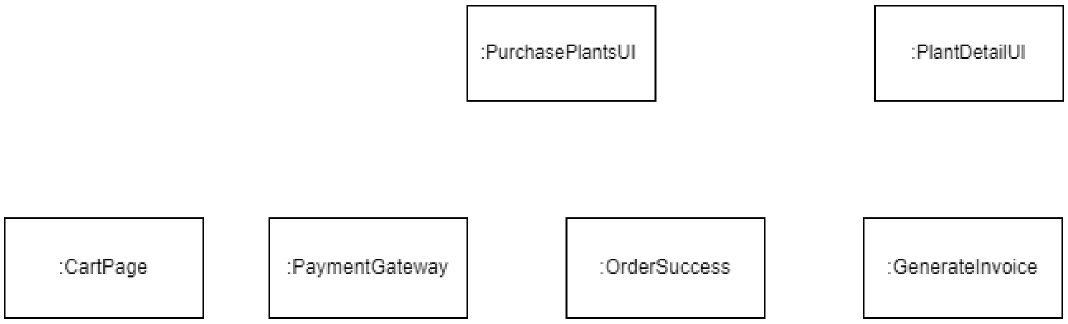


Figure 7: Boundary object.

The objects are properly connected to each other to send and receive messages. The following is the figure after the connections are made.

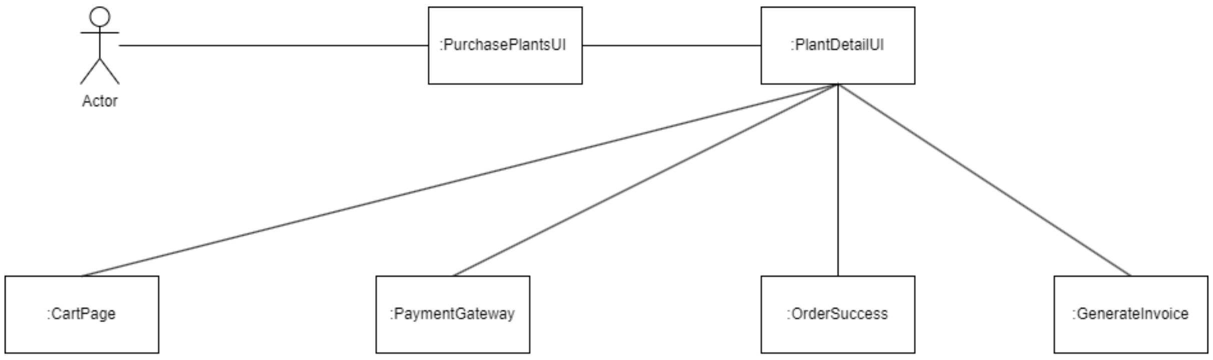


Figure 8: Connected Objects.

In the final collaboration diagram, messages are passed from one another to further explain how the system works:

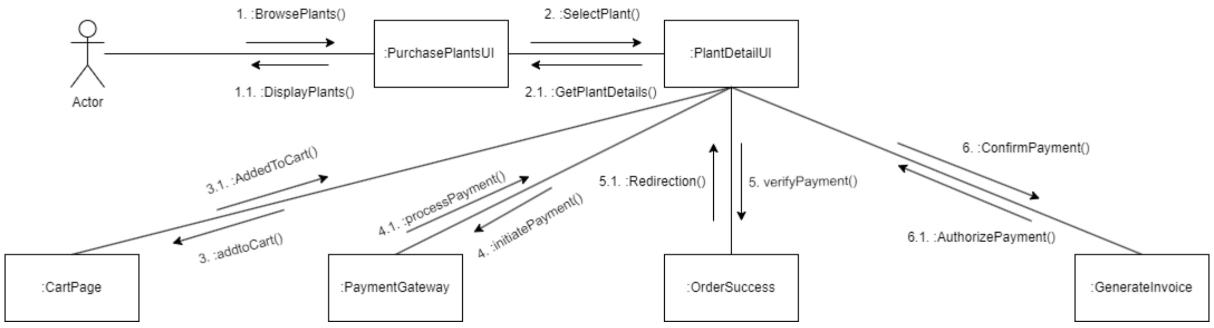


Figure 9: Final Collaboration Diagram.

After the user views the PurchasePlantsUI, the user selects a desired plant to purchase. The users then views the PlantDetailUI and clicks on the Buy button or the Add to cart button which adds the plant to the cart page. The user then views the cart page and pays for the plant. The payment is then processed in the system. The processed payment is then verified which then shows a success message and redirects the user to the PlantDetailUI. An invoice is then generated and provided to the user.

7. Sequence Diagram Explain Diagram

A sequence diagram visualizes the sequence of messages and the interactions between the objects. It contains a group of objects which are represented by lifelines. They exchange messages overtime while interacting with each other. (IBM, 2021)

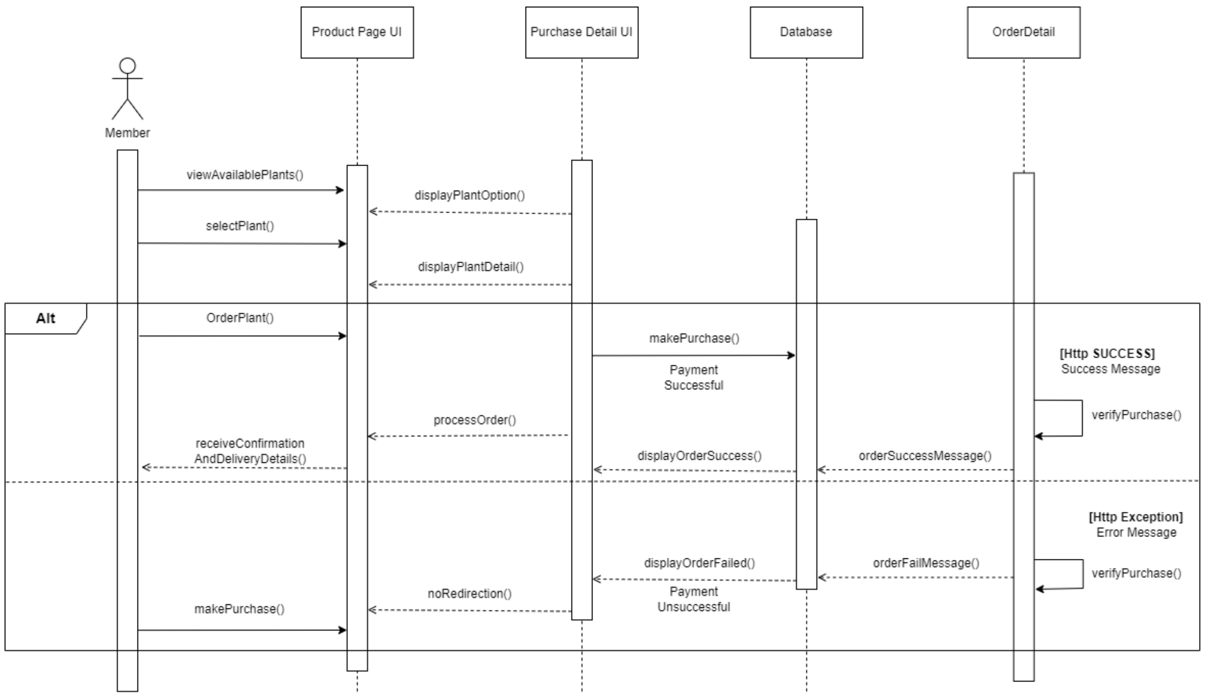


Figure 10: Sequence Diagram.

Notation:

- Object
- Actor
- Activation Bar
- Synchronous Arrow
- Asynchronous Arrow
- Return Message
- Reflexive Message

8. Class Diagram Explain

A class diagram is a type of UML diagram which visualizes the different objects and relationships of the objects within the system (creately, 2023).

Class Diagram Notation:

- Class
- Aggregation
- Composition
- Inheritance

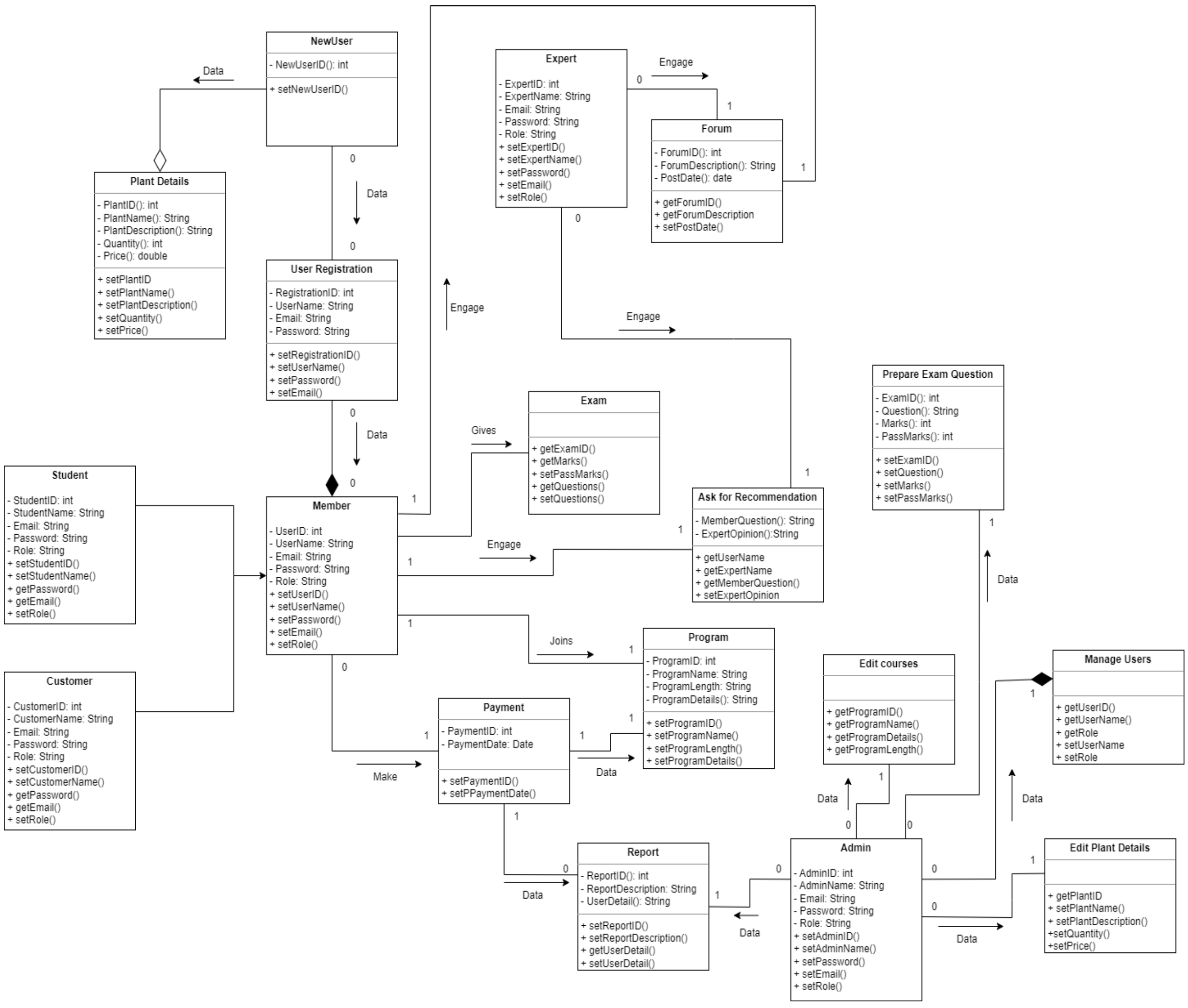


Figure 11: Class Diagram.

9. Further Development

Architectural Choice

The choice of Architecture for this system would be Microservice Architecture. In Microservice architecture a program isn't built all at once but is built in smaller programs. The components are deployed as separate units. This allows the components to be individually deployed which makes this highly scalable. This is well suited for prototype methodology as various prototypes are created before the actual product is developed. The flexibility in the choice of technology allows for experiments to be conducted in a wide scale.

Development Plan

Defining a scope and purpose is crucial before planning a project and it helps with identifying the requirements of the project. This ensures that the expectation of the stakeholders have been met. In this stage, the time, budget, and the resource requirement are allocated and frequent interviews and meetings with the stakeholders is held.

Creating a wireframe is crucial as this helps visualise the software. The tool planned to create wireframes is Figma. Figma can be used to create a detailed and polished wireframe, which makes the development process easier and smoother.

The frontend to be used for the system is planned to be React.js as it uses component-based architecture, declarative syntax making it easier to understand and manage the code.

The backend planned to be used for the system is Node.js as it is universal meaning that the code can run on both client and server side. It also supports server-side rendering which helps improve performance as it loads the initial HTML on the server and sends it to the user.

The database planned to be used for the system is MySQL. It is an open-source database which is well known for its reliability and scalability. It is well optimized meaning that it handles large volumes of data with fast server response time. It is also cost effective as it is open source.

Design Pattern

The design pattern used in this system is MVC design pattern. This design pattern consists of a data model, presentation information and control information. The ‘M’ stands for ‘Model’ which contains the application data. The ‘V’ stands for ‘View’ which incorporates the UI design of the system. The ‘C’ stands for ‘Controller’, it takes the event triggers and processes the triggers to provide a meaningful output. This design pattern is followed by many developers as it simplifies the structure of the process. The reason MVC pattern is used for the development of this software is that it makes the development process efficient by managing and organizing the code of the software. This makes the simplifies the development and maintainability of the software. (HCL Commerce, 2024)

Testing Plan

The process of exercising a program to find errors and flaws before the full deployment of the system is known as testing. Its main objective is to find errors, check the performance and quality of the system. Testing is done to ensure that the software developed functions as expected.

There are two kinds of testing, they are:

Whitebox Testing

Whitebox Testing also known as glass box testing, is a testing methodology in which the internal code structures are examined. This verifies the data flow from Input to Output. It helps improve the usability and security of the system (Ashtari, 2022). The white box testing of the system involves testing the code and ensuring the code works as expected. The code of the software should be studied at the runtime level to examine the utilization of the resources. The test cases should be designed according to the priority of the functionality.

The test cases include:

- Unit Testing: The unit testing is done to ensure that the component of the system is working properly. For the unit testing of this system, individual components of the system are to be tested. For example, the individual UI components such as buttons, product cards, etc.

- Mutation Testing: The mutation testing checks the consistency of the code by making small random changes to the code and checking if the tests still pass. For the mutation testing of this system, a framework is to be setup. The existing unit tests are then executed against the mutated code.
- Integration Testing: Integration testing checks the integration points of the internal components of the system with external system. The integration testing for this system includes testing whether the frontend and the backend of the system work together as expected.
- White Box penetration Testing: The White Box Penetration testing is done to ensure that the system cannot be easily penetrated from the outside, typically the hacker. For the White Box Testing of this system, an ethical hacker attempts to hack the internal system of the website.
- Static Code Analysis: The static code testing for this system is planned to be done without executing the code and checking the actual code for potential bugs and other reliability issues.

Blackbox Testing

Blackbox testing is a testing methodology in which the tester does not require the knowledge of the code structure. The term “black box” refers to an exterior covering of the application, which keeps testers from viewing the internal operations of the system. This ensures that the testing is done from the user’s perspective where the testers have no knowledge of the code (Ashtari, 2022). The black box testing of the system involves testing the functionality and the usability of the website such as:

- Testing the Registration feature with valid and invalid information.
- Testing if the navigation bar works.
- Testing if the users can browse, purchase plants.
- Testing if the forums can be posted, interact with the posts.

Maintenance Plan

After the system is created, it must be well maintained to have good customer service and feedback. The plan to keep the system up to date and maintain the system the following can be done:

- User feedback should be frequently gathered.
- Regular backups of the system should be maintained.
- The backups are to be stored off site to prevent data loss.
- Frequent software updates with bug fixes and security updates will be rolled out.
- Identification and fixing of vulnerabilities should done before hackers exploit the vulnerability.
- The contents of the website like products and prices are to be updated.
- Using the user feedback, optimize the UI-UX of the system.
- Provide customer support.

Deployment Plan

Before deployment of the software, a proper deployment plan must be created for a smooth deployment of the software. The following is the plan for the deployment of the system:

- **Monitor and Maintenance:** After deployment of the software, the system must be monitored for unexpected bugs and errors and should be maintained.
- **Rollback:** If a major vulnerability is found in the system, a rollback plan must be prepared. This lessens the chance of the software failing.
- **Staff Training:** Staffs must be properly trained for further development to ensure the system functions well.
- **Further Optimization:** The system must be further optimized to ensure less resource utilization which results in better performance of the software.
- **Availability:** The system is to be available for all of the system, for this, the traffic to the system is to be distributed to various servers to manage the traffic and provide a smoother experience for the users.

10. Prototype

A prototype is a sample which is created to test a product or the concept of a product to ensure that the concept works and is valid for further development (Ramirez, 2018).

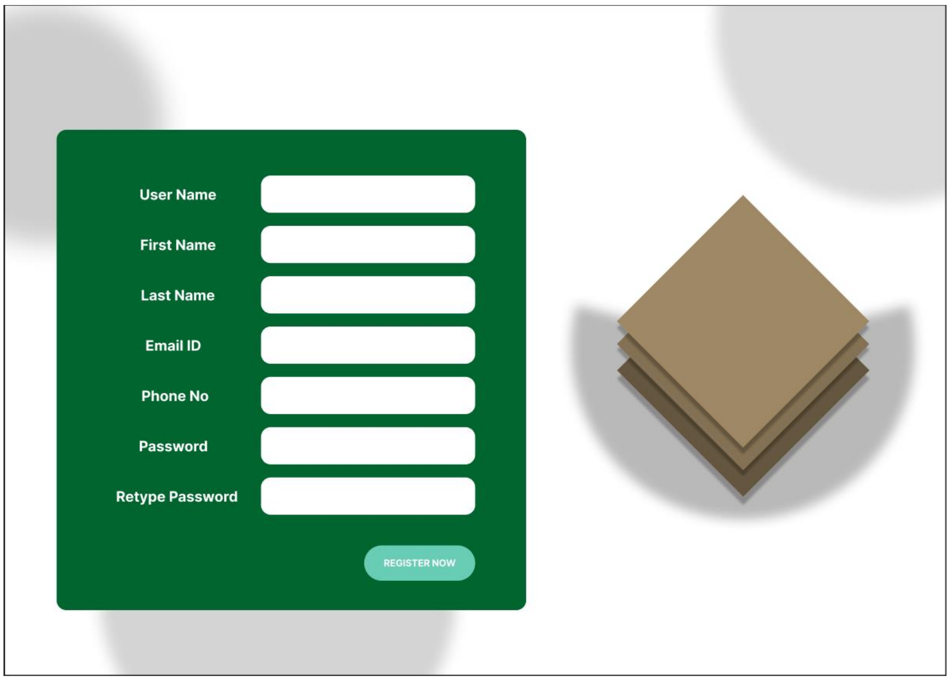


Figure 12: Prototype for Register Page.

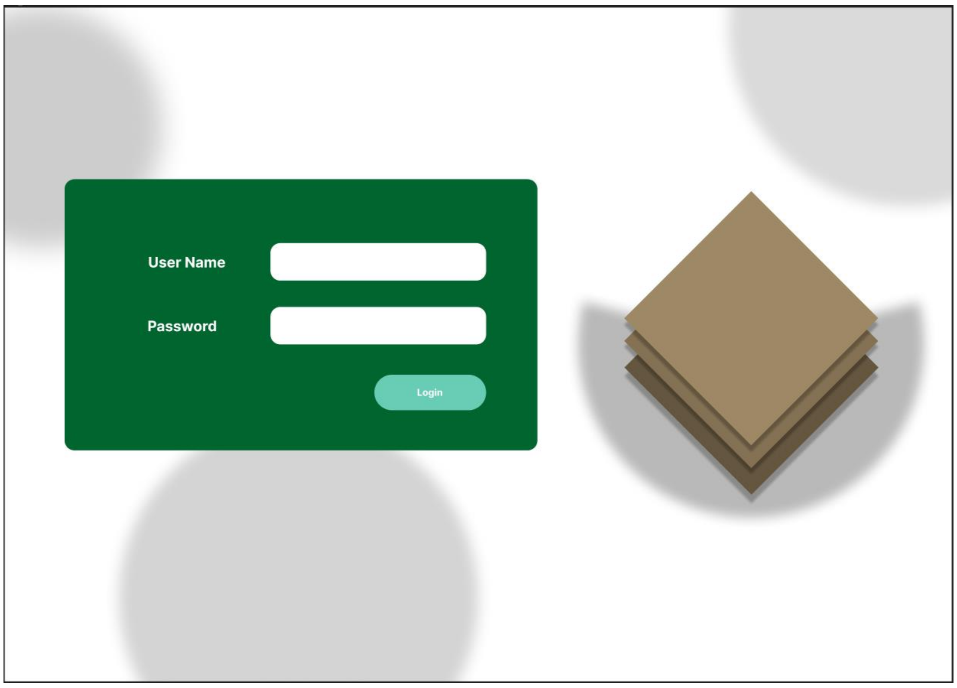


Figure 13: Prototype for Login Page.

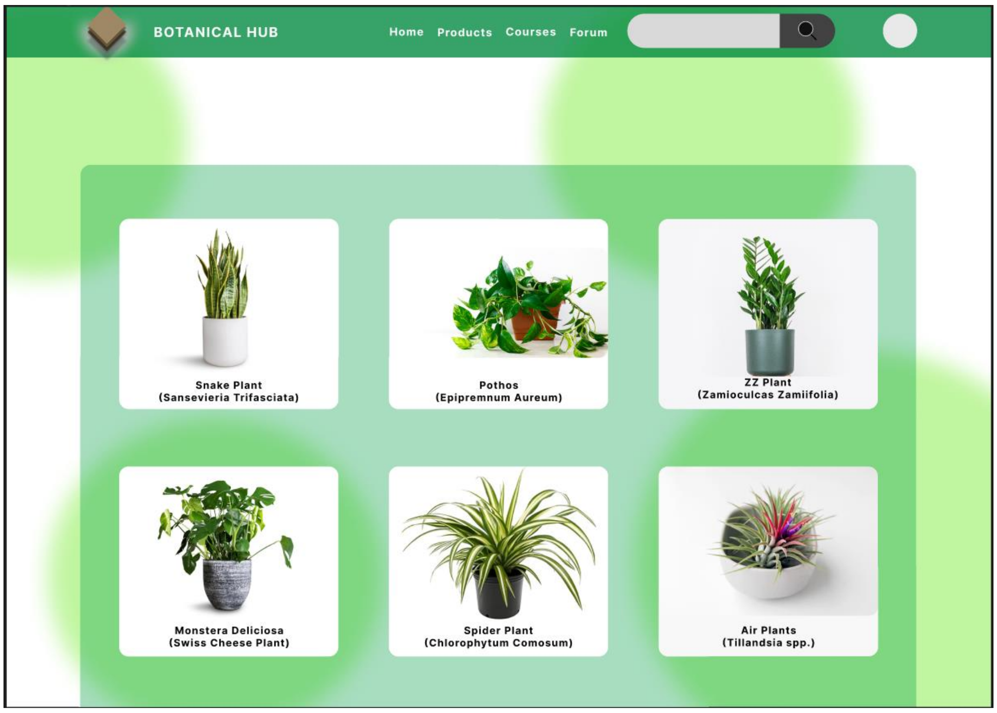
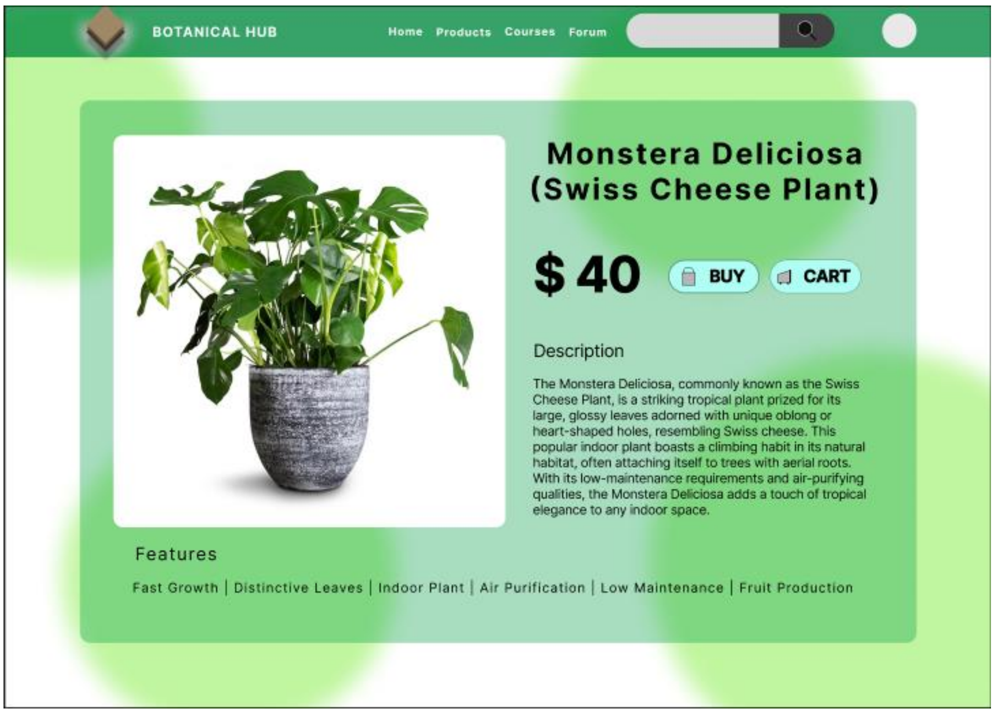


Figure 14: Prototype for Plants page.



Monstera Deliciosa
(Swiss Cheese Plant)

$ 40

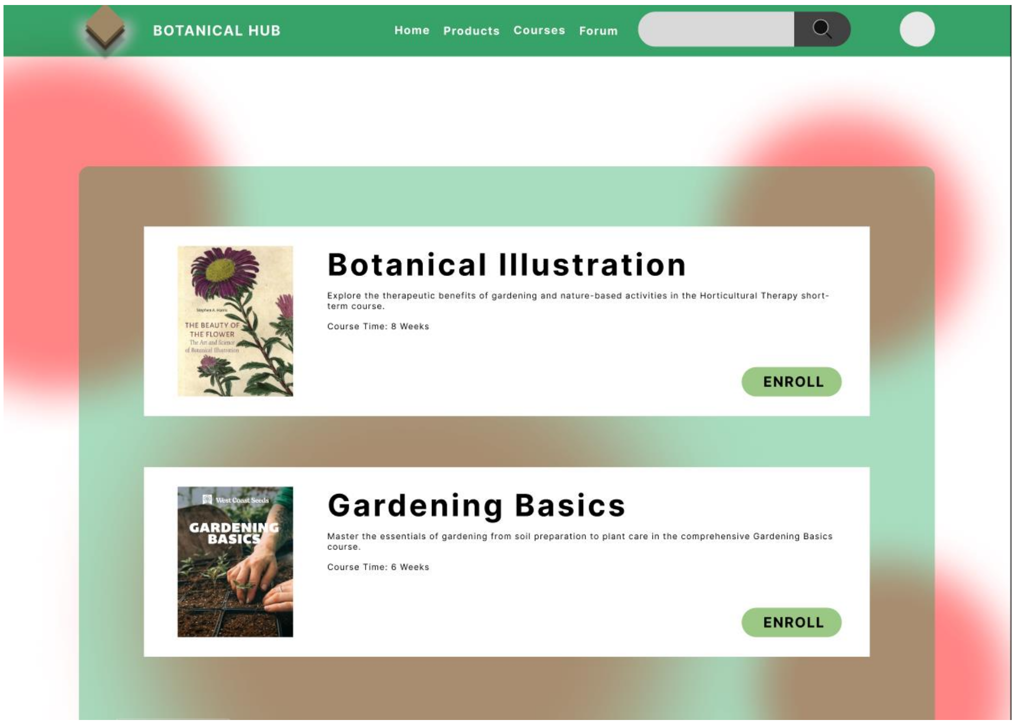
Description

The Monstera Deliciosa, commonly known as the Swiss Cheese Plant, is a striking tropical plant prized for its large, glossy leaves adorned with unique oblong or heart-shaped holes, resembling Swiss cheese. This popular indoor plant boasts a climbing habit in its natural habitat, often attaching itself to trees with aerial roots. With its low-maintenance requirements and air-purifying qualities, the Monstera Deliciosa adds a touch of tropical elegance to any indoor space.

Features

Fast Growth | Distinctive Leaves | Indoor Plant | Air Purification | Low Maintenance | Fruit Production

Figure 15: Prototype for Plant Detail Page.



Botanical Illustration

Explore the therapeutic benefits of gardening and nature-based activities in the Horticultural Therapy short-term course.

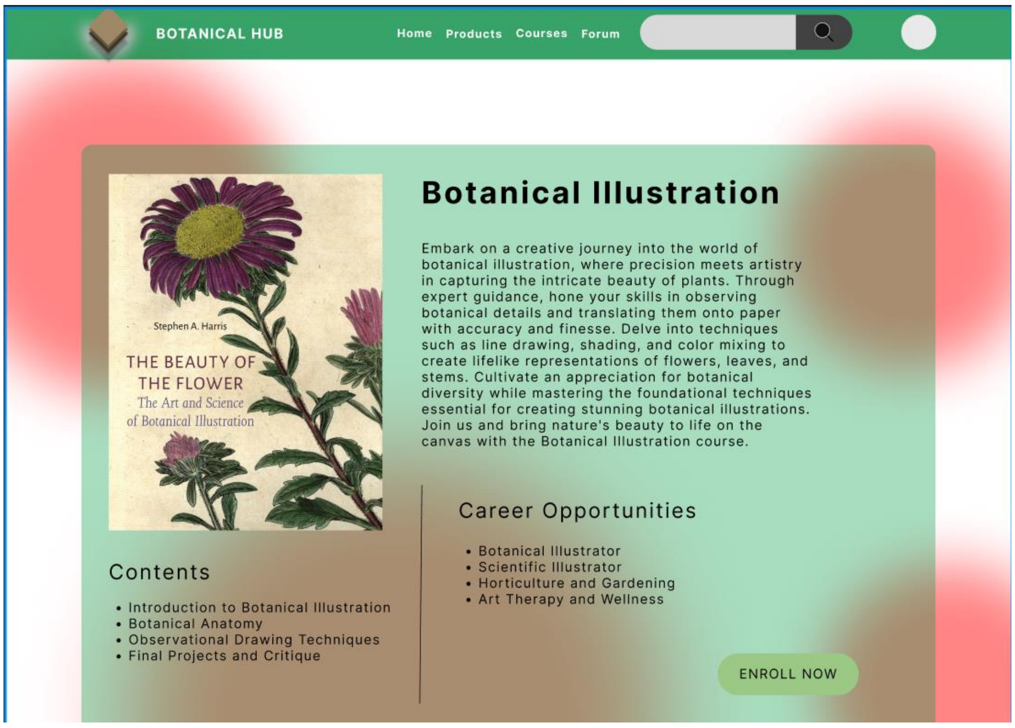
Course Time: 8 Weeks

Gardening Basics

Master the essentials of gardening from soil preparation to plant care in the comprehensive Gardening Basics course.

Course Time: 6 Weeks

Figure 16: Prototype for Courses Page.



Botanical Illustration

Embark on a creative journey into the world of botanical illustration, where precision meets artistry in capturing the intricate beauty of plants. Through expert guidance, hone your skills in observing botanical details and translating them onto paper with accuracy and finesse. Delve into techniques such as line drawing, shading, and color mixing to create lifelike representations of flowers, leaves, and stems. Cultivate an appreciation for botanical diversity while mastering the foundational techniques essential for creating stunning botanical illustrations. Join us and bring nature's beauty to life on the canvas with the Botanical Illustration course.

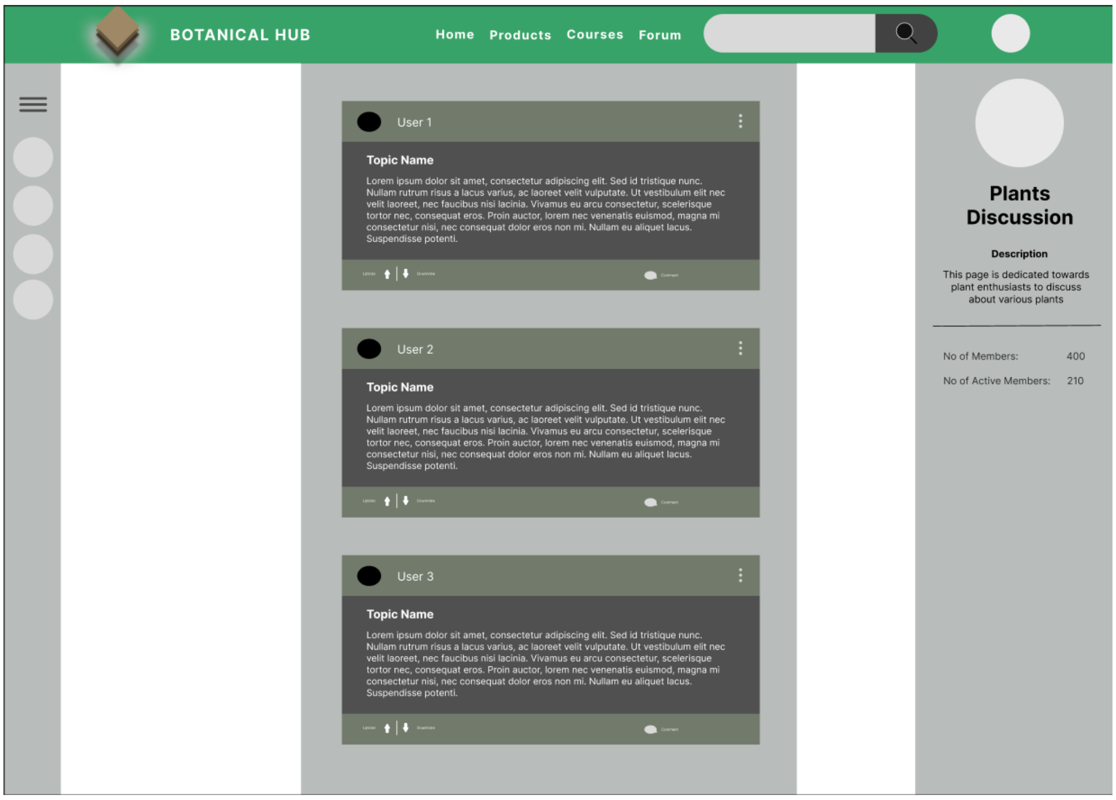
Contents

- Introduction to Botanical Illustration
- Botanical Anatomy
- Observational Drawing Techniques
- Final Projects and Critique

Career Opportunities

- Botanical Illustrator
- Scientific Illustrator
- Horticulture and Gardening
- Art Therapy and Wellness

Figure 17: Prototype for Course Derail Page.



User 1

Topic Name

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id tristique nunc. Nullam rutrum risus a lacus varius, ac laoreet velit vulputate. Ut vestibulum elit nec velit laoreet, nec faucibus nisi lacinia. Vivamus eu arcu consectetur, scelerisque tortor nec, consequat eros. Proin auctor, lorem nec venenatis euismod, magna mi consectetur nisi, nec consequat dolor eros non mi. Nullam eu aliquet lacus. Suspendisse potenti.

User 2

Topic Name

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id tristique nunc. Nullam rutrum risus a lacus varius, ac laoreet velit vulputate. Ut vestibulum elit nec velit laoreet, nec faucibus nisi lacinia. Vivamus eu arcu consectetur, scelerisque tortor nec, consequat eros. Proin auctor, lorem nec venenatis euismod, magna mi consectetur nisi, nec consequat dolor eros non mi. Nullam eu aliquet lacus. Suspendisse potenti.

User 3

Topic Name

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id tristique nunc. Nullam rutrum risus a lacus varius, ac laoreet velit vulputate. Ut vestibulum elit nec velit laoreet, nec faucibus nisi lacinia. Vivamus eu arcu consectetur, scelerisque tortor nec, consequat eros. Proin auctor, lorem nec venenatis euismod, magna mi consectetur nisi, nec consequat dolor eros non mi. Nullam eu aliquet lacus. Suspendisse potenti.

Plants Discussion

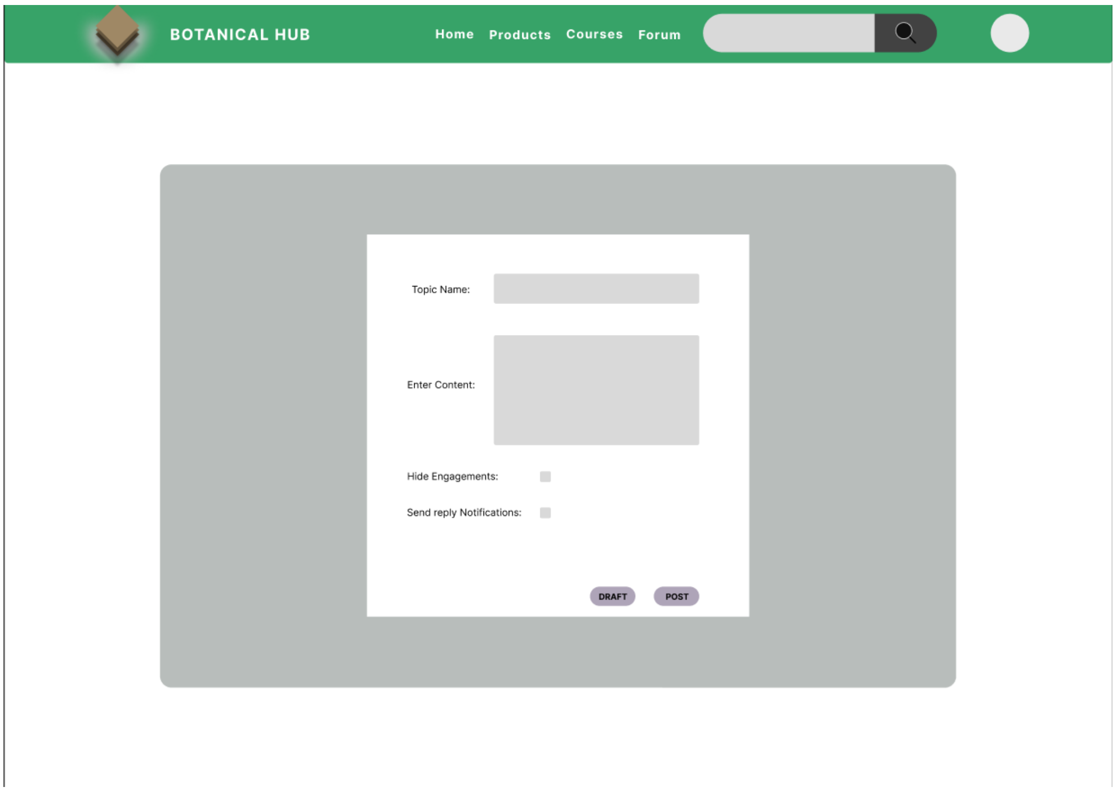
Description

This page is dedicated towards plant enthusiasts to discuss about various plants

No of Members: 400

No of Active Members: 210

Figure 18: Prototype for Forum Page.



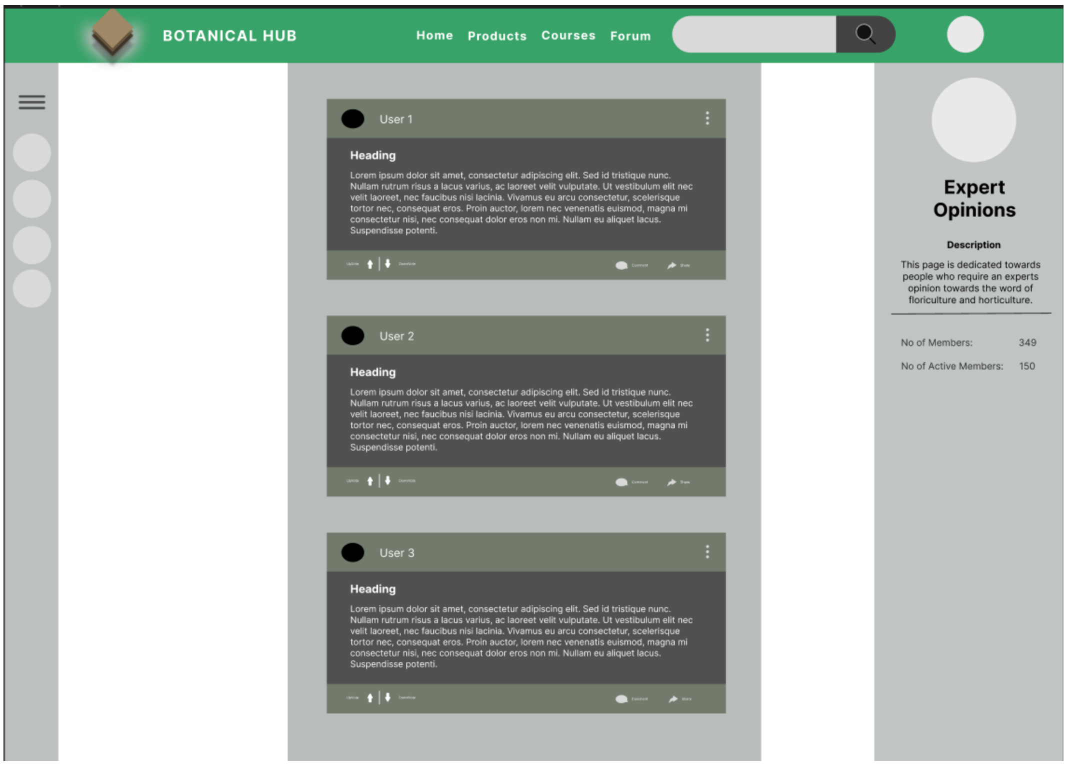
Topic Name:

Enter Content:

Hide Engagements:

Send reply Notifications:

Figure 19: Prototype for Post Forum Page.



User 1

Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id tristique nunc. Nullam rutrum risus a lacus varius, ac laoreet velit vulputate. Ut vestibulum elit nec velit laoreet, nec faucibus nisi lacinia. Vivamus eu arcu consectetur, scelerisque tortor nec, consequat eros. Proin auctor, lorem nec venenatis euismod, magna mi consectetur nisi, nec consequat dolor eros non mi. Nullam eu aliquet lacus. Suspendisse potenti.

User 2

Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id tristique nunc. Nullam rutrum risus a lacus varius, ac laoreet velit vulputate. Ut vestibulum elit nec velit laoreet, nec faucibus nisi lacinia. Vivamus eu arcu consectetur, scelerisque tortor nec, consequat eros. Proin auctor, lorem nec venenatis euismod, magna mi consectetur nisi, nec consequat dolor eros non mi. Nullam eu aliquet lacus. Suspendisse potenti.

User 3

Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id tristique nunc. Nullam rutrum risus a lacus varius, ac laoreet velit vulputate. Ut vestibulum elit nec velit laoreet, nec faucibus nisi lacinia. Vivamus eu arcu consectetur, scelerisque tortor nec, consequat eros. Proin auctor, lorem nec venenatis euismod, magna mi consectetur nisi, nec consequat dolor eros non mi. Nullam eu aliquet lacus. Suspendisse potenti.

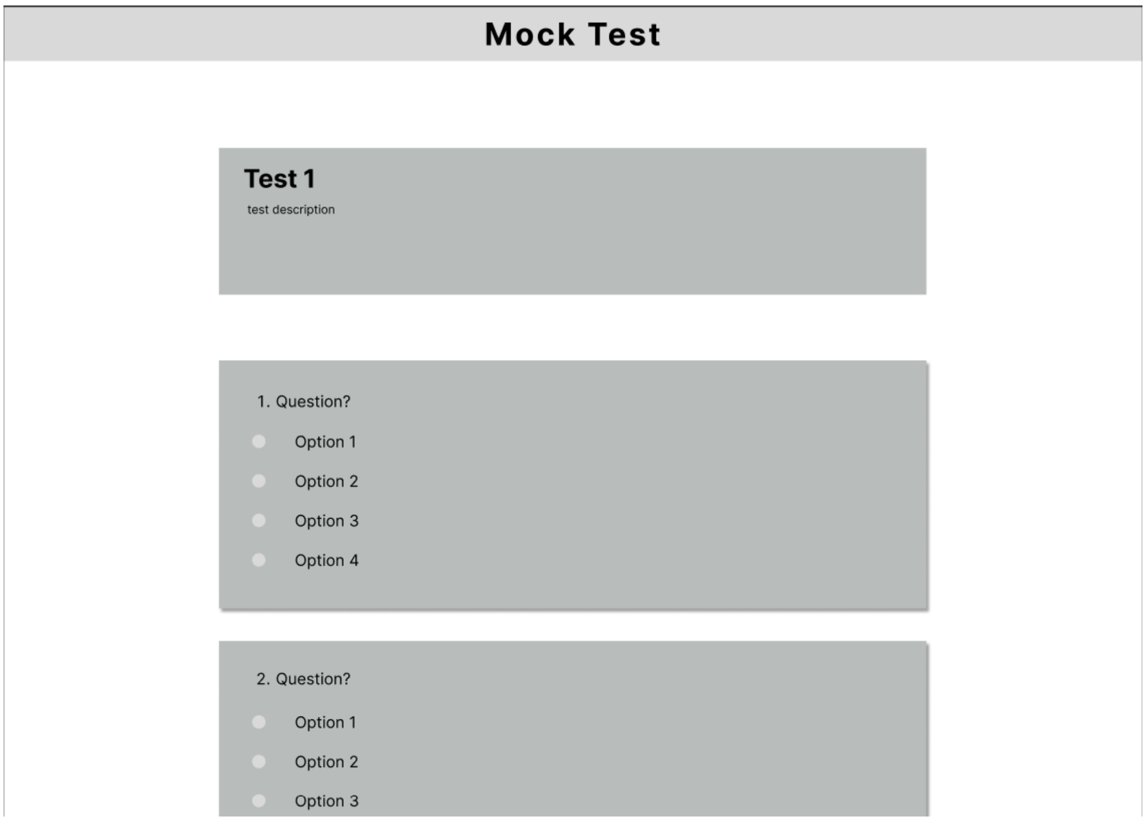
Expert Opinions

Description

This page is dedicated towards people who require an experts opinion towards the word of floriculture and horticulture.

No of Members:	349
No of Active Members:	150

Figure 20: Prototype for Expert Opinion Page.



Mock Test

Test 1

test description

1. Question?
   - Option 1
   - Option 2
   - Option 3
   - Option 4

2. Question?
   - Option 1
   - Option 2
   - Option 3

Figure 21: Prototype for Mock Test Page.

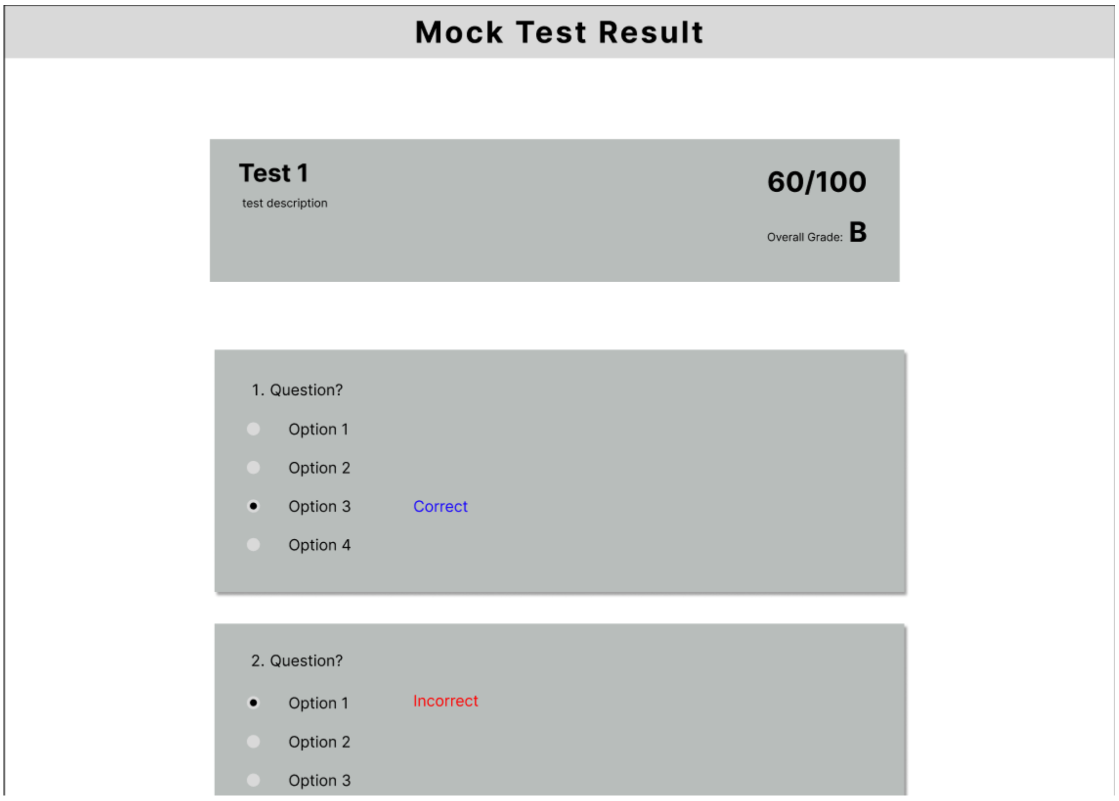


Figure 22: Prototype for Mock Test Result Page.

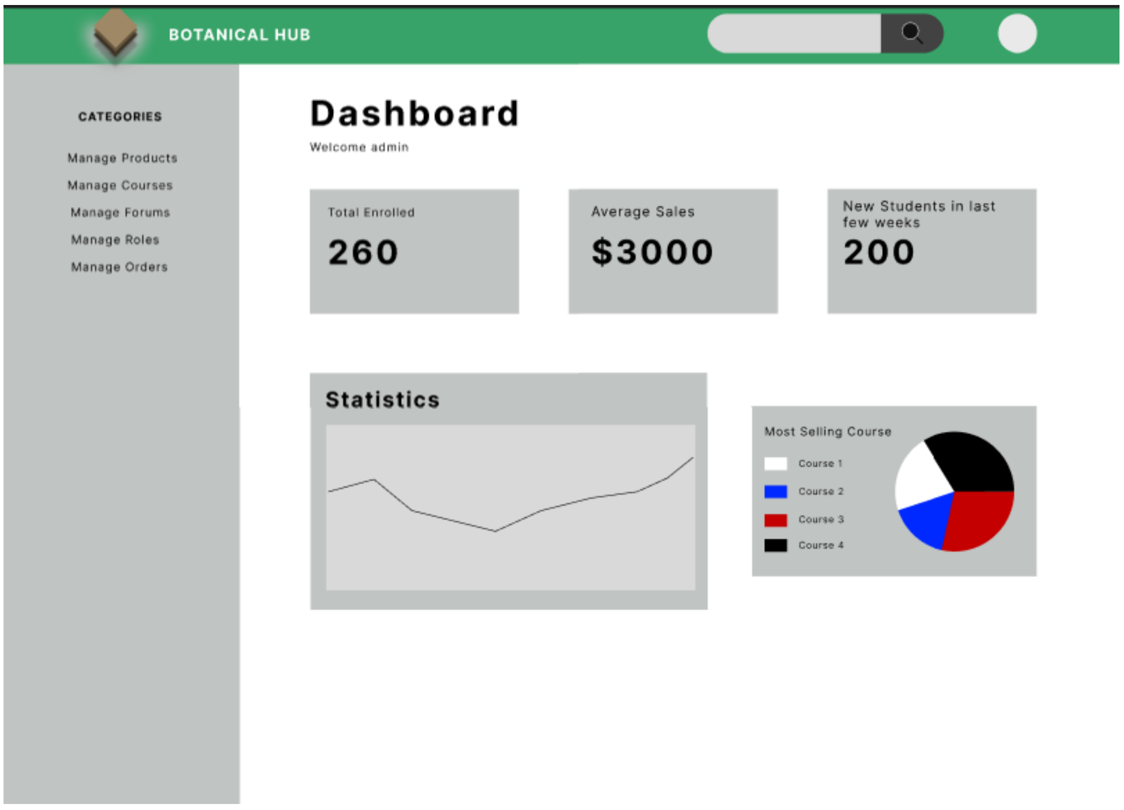


Figure 23: Prototype for Admin Dashboard.

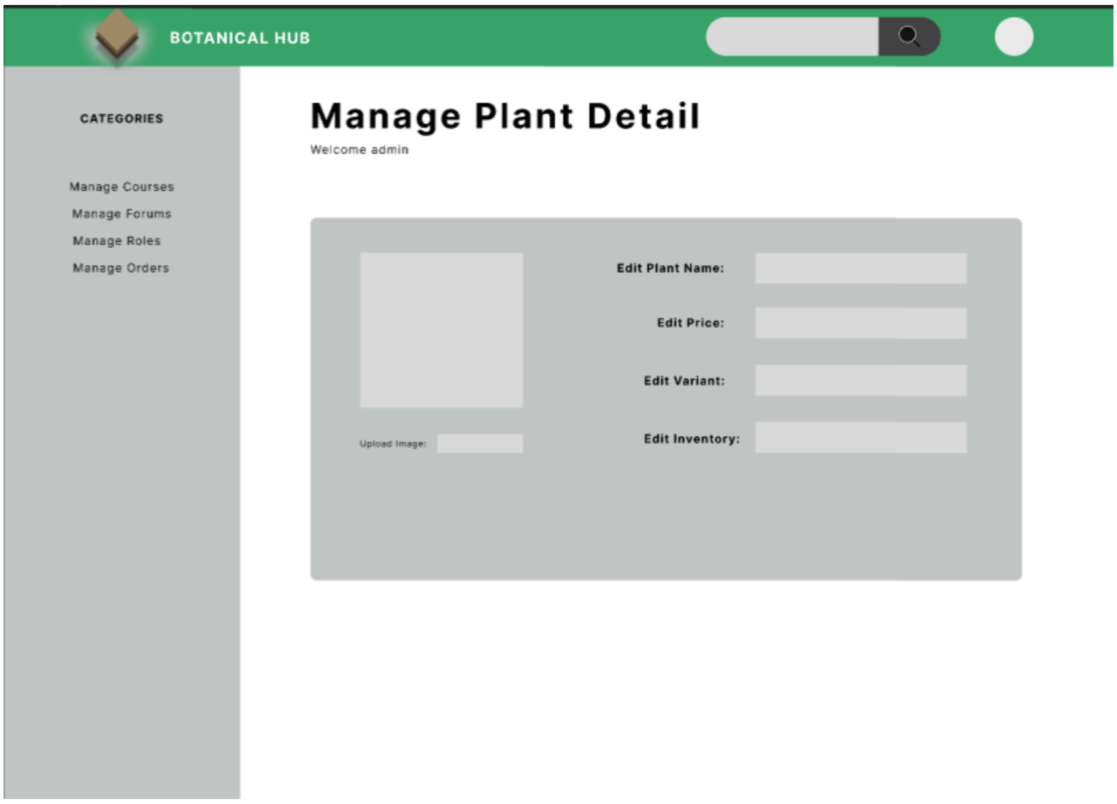


Figure 24: Prototype for Manage Plant Detail Page.

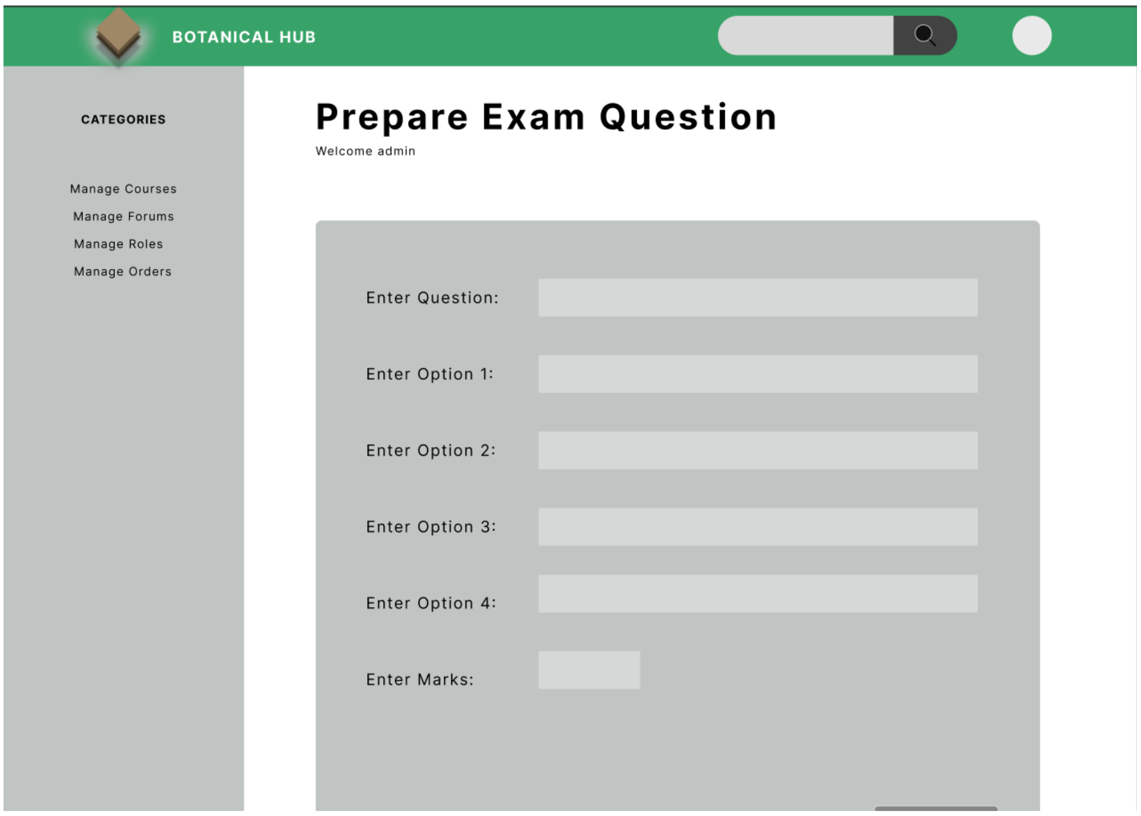
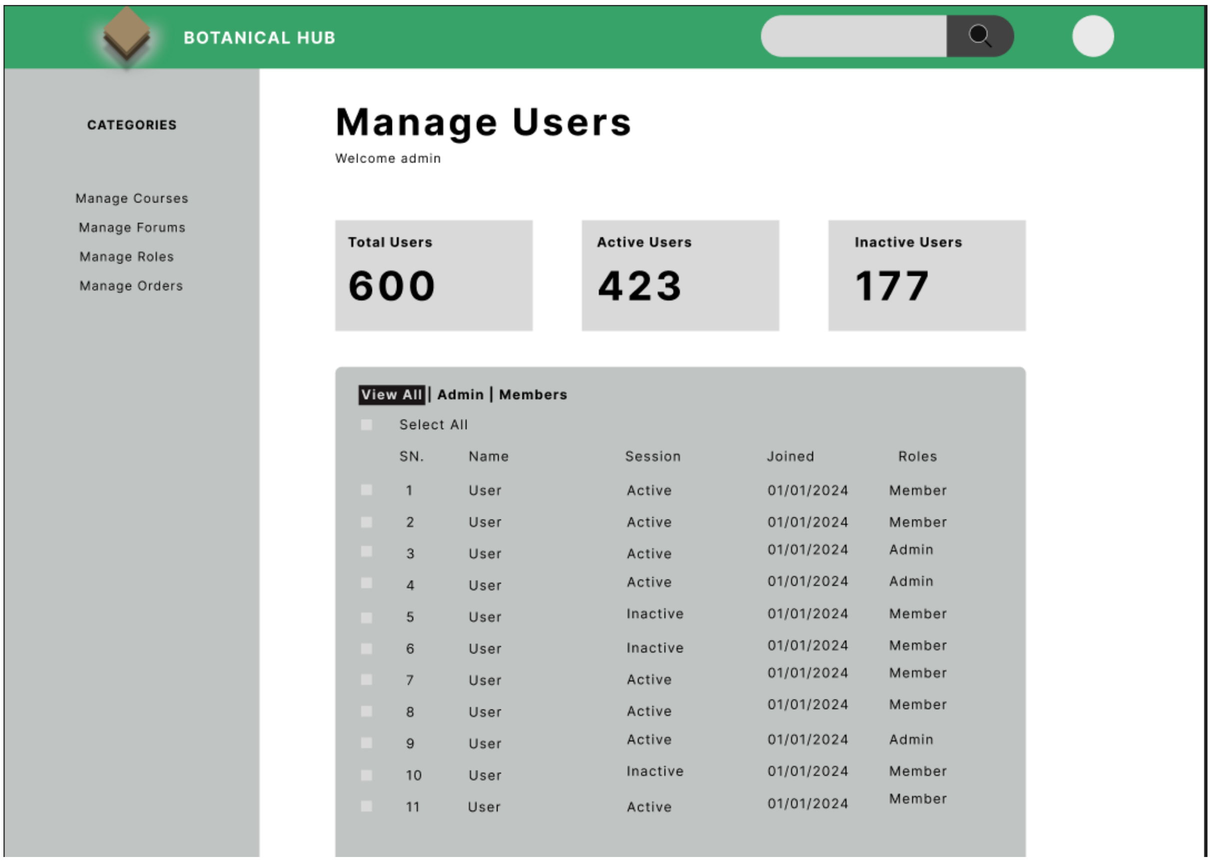


Figure 25: Prototype for Prepare Exam Page.



BOTANICAL HUB

CATEGORIES

Manage Courses
Manage Forums
Manage Roles
Manage Orders

Manage Users

Welcome admin

Total Users 600

Active Users 423

Inactive Users 177

View All | Admin | Members

Select All

	SN.	Name	Session	Joined	Roles
	1	User	Active	01/01/2024	Member
	2	User	Active	01/01/2024	Member
	3	User	Active	01/01/2024	Admin
	4	User	Active	01/01/2024	Admin
	5	User	Inactive	01/01/2024	Member
	6	User	Inactive	01/01/2024	Member
	7	User	Active	01/01/2024	Member
	8	User	Active	01/01/2024	Member
	9	User	Active	01/01/2024	Admin
	10	User	Inactive	01/01/2024	Member
	11	User	Active	01/01/2024	Member

Figure 26: Prototype for Manage user Page.

11. Conclusion

In conclusion, this coursework requires creation of a system. It requires to carry out Object Oriented Analysis and Design of the system. The McGregor Institute of Botanical Training aims to create a system that provides various short term related to people with increasing demand for agriculture and horticulture. Apart from providing courses, it also plans to sell various plants charging minimal fee and even for free in some cases. They also plan on creating a forum where plant enthusiasts get to engage with other plant enthusiasts. The forum also provides a platform for the users to ask for recommendations to the experts.

The system aims to provide functionalities such as payment, courses, post forums, etc. providing valuable resources to the users. For this a well-documented plan for the system is created which provides a proper work breakdown structure, schedule for specific tasks. A use case, collaboration diagram and class diagram is created for better understand of the system and how it works. Furthermore, a development plan is created to give an idea of how the system is to be developed. Testing and maintenance plan for the system is also created and a deployment plan is created for better deployment of the system. Various prototypes for the system is created for a visual representation of how the system is to be developed.

12. References

Ashtari, H., 2022. *Black Box vs. White Box Testing: Understanding 3 Key Differences.* [Online]
Available at: https://www.spiceworks.com/tech/devops/articles/black-box-vs-white-box-testing/
[Accessed 4 May 2024].

creately, 2023. *The Easy Guide to UML Class Diagrams | Class Diagram Tutorial.* [Online]
Available at: https://creately.com/blog/software-teams/class-diagram-tutorial/
[Accessed 6 May 2024].

Gantt.com, 2024. *What is a Gantt Chart?.* [Online]
Available at: https://www.gantt.com/
[Accessed 6 May 2024].

GeeksforGeeks, 2019. *Difference between Sequence diagram and Collaboration diagram.* [Online]
Available at: https://www.geeksforgeeks.org/difference-between-sequence-diagram-and-collaboration-diagram/
[Accessed 27 April 2024].

HCL Commerce, 2024. *Model-View-Controller design pattern.* [Online]
Available at: https://help.hcltechsw.com/commerce/9.1.0/developer/concepts/csdmvcdespat.html
[Accessed 2 May 2024].

IBM, 2021. *Sequence diagrams.* [Online]
Available at: https://www.ibm.com/docs/en/radfws/9.6.1?topic=diagrams-sequence
[Accessed 6 May 2024].

Ramirez, V., 2018. *What is a Prototype?.* [Online]
Available at: https://medium.com/nyc-design/what-is-a-prototype-924ff9400cfd
[Accessed 6 May 2024].

Rana, K., 2020. *Prototype Model.* [Online]
Available at: https://artoftesting.com/prototype-model
[Accessed 25 April 2024].