# Software Requirements Specification

# PRJ666 – Fall 2023

# PRJ666 – Team No: 6

# Name of Project:   PhotoSyntheSit

**Last updated: September 12, 2023**

**Team Members:**

**1.** Ricky Chen

**2.** Muzzammil Ismathhimam

**3.**  Ian Jacobs

# TABLE OF CONTENTS

1. **Introduction/Overview - Document Information**
   1. **Document Authors**
   2. **Revision History**
   3. **Document Conventions**
   4. **Document Purpose**
   5. **Intended Audience**
   6. **Group Agreement**
2. **Project Overview**
   1. **Project Proposal**
   2. **Stakeholders and Users**
   3. **Functional Requirements**
   4. **Nonfunctional Requirements**
   5. **Project Scope**
   6. **System risks**
   7. **Operating Environment**
   8. **UI/UXD Interface Mockups**
3. **Process & Data Modeling**
   1. **UML Modeling: DFDs & Activity Diagrams**
   2. **Use Case Specification** 
      1. **Business Rules**
      2. **System Use Case Diagrams**
      3. **Use Case Description Tables**
4. **Domain Class Diagram**
5. **Database (Select either 5.1 or 5.2)**
   1. **RDBMS Artifacts**
      1. Scripts to create, populate, delete tables
      2. Data Dictionary
   2. **NoSQL Artifacts**
6. **Work breakdown Structure (WBS)**
7. **Milestones**
8. **Acceptance Criteria**
9. **Implementation Schedule**
10. **Client / Faculty Sign-off**

# 1 - Introduction/Overview - Document Information

## 1.1 Document Authors

Ricky Chen

Muzzammil Ismathhimam

Ian Jacobs

1.2 Revision History

|  |  |
| --- | --- |
| Week 03 | Section 1 (1.1 to 1.6)  Section 2 (2.1) |
| Week 04 | Section 2(2.2 to 2.5) |
| Week 05 | Section 2(2.6 to 2.8) |
| Week 06 | Section 3.1 |
| Week 07 | Section 3.2 (3.2.1, 3.2.2, 3.2.3) |
| Week 08 | System Mockups, PPT&Video |
| Week 09 | Section 4 |
| Week 10 | Section 5 |
| Week 11 | Section 6 - 8 |
| Final | Video presentation, SRS |

## 1.3 Document Conventions

Any text in red indicates an exception or error

Any text in blue is in-progress

Any text highlighted in yellow is an important point.

Any text in green was recently added.

Any text *italicized* represents definitions.

Any text with ~~strike-through~~ is deleted.

## 1.4 Document Purpose

The purpose of this document is to allow all members of this project to understand the project goals, benefits and to predict any problems that may arise. Additionally, the effectiveness of this document lies in how detailed it is.

This SRS document is a complete overview/framework that the development of this project will follow, and to ensure that all members are involved and in agreement. This includes the project parameters and goals, furthermore this document aims to assist the members of this project in completing the Software Delivery Lifecycle (SDLC), but outlining the different steps needed (Sections 1 – 10) in detail.

## 1.5 Intended Audience

The intended audience of this web application is geared towards males and females 18 and up who require assistance in maintaining their plants while they are unable to, and for those who would like to join a community where they can interact with plant enthusiasts or gain more knowledge in plant care in general. As a wide range of ages are targeted, the website will incorporate a simple, minimalistic design to make it easy to navigate around, but also display informative and captivating content. Additionally, as the younger age groups within our intended audience tend to have a greater online presence, a social hub and online store will be implemented to allow users to interact within the PhotosyntheSit community and increase our web application’s exposure.

## 1.6 Group Agreement

**TEAM AGREEMENT**

**Team #: 6**

**Project Title:** PhotoSyntheSit

**Project Time Frame: September 2022 – April 2022**

**Team Members:** Ricky Chen, Muzzammil Ismathhimam, Ian Jacobs

**Team Leadership:**

**Team Functions:**

* *We will share and upload documents through MS Teams*
* *We will use MS Teams to attend meetings with professor*
* *We will contact to team using Discord server.*

**Team Meetings:** Monday @ 10:30 AM

**Team Problems:**

* Any problems or issues that arise can be discussed through our discord server.

**Team Commitment**

**The undersigned members agree to work together on the project until the end of the PRJ666 next Semester. They recognize that as a team and individually they are responsible for the quality of all deliverables.**

**Name Date**

|  |  |
| --- | --- |
| Ricky Chen | September 12, 2023 |
| Ian Jacobs | September 12, 2023 |
| Muzzammil Ismathhimam | September 12, 2023 |

ShapeShapeShapeShapeShapeShapeShapeShape

# 2 - Project Overview

## 2.1 Project Proposal

Project Background

As niche as it may seem, a plant-sitting service is highly in demand in today's market. With plant owners leaving their expensive plants unattended to, we identified a gap in the market and put it upon ourselves to introduce a web-based plant-sitting service. Whether it is indoor plants or outdoor plants, our service aims to bring plant care to users across Toronto, and in the future globally.

**Problem Statement**

|  |  |
| --- | --- |
| The Problem of: | Plant Owners have no one to attend to their plants while they are on vacation. |
| Affects: | Plant Owners |
| The impact of which is: | Plant Owners expensive plants die. |
| A successful solution would: | The ability to connect plant owners with experienced plant sitters attend to their plants. |

**Product Vision**

|  |  |
| --- | --- |
| For | Plant Owners |
| Who | The need is the ability to schedule plant sitters to attend to a customer’s plants. |
| PhotosyntheSit | Is a service. |
| That | Allows an opportunity for plant owners to travel without worry of their plants dying. |
| Unlike | Job posting sites (Craigslist, TaskRabbit). |
| Our product | Our service connects you with experienced plant experts. |

## 2.2 Stakeholders and Users

|  |  |
| --- | --- |
| Stakeholder Name/Identifier | Category |
| Administrator/CEO | Administrators   * Executive decision maker for all content on the site. |
| Employee | Administrators, Users   * Respond to users’ complaints and inquiries. * Moderate, add, delete, edit listings and threads. * Manage plant-sitting bookings, payment, social media. * Buy plants. |
| Project Manager | Developers, Users   * Delegate tasks to project employees. * Ensure project is being conducted in a timely manner according to schedule. * Buy plants. |
| Plant-Sitter | Contracted Employee, User   * Edit their own profile listing and schedule. * Interact with customers upon being hired. * Buy plants and/or hire other plant-sitters. * Post threads on the social media aspect. |
| Customer/Client | User   * Hire plant-sitters. * Buy plants. |
| Developer | Developers, Users   * Code. * Debug. * Test website. * Buy plants. |
| UI/UX Designer | Developers, Users   * Design website. * Incorporate developer/user feedback. * Buy plants. |

## 2.3 Functional Requirements

## User Account

## Users can create new accounts

* + 1. Input email, password, name and address
    2. Validate inputs, if not send error messages
    3. User can cancel account creation
    4. User can confirm inputs and create account
    5. Verify email and finalize registration.
  1. Users can login
     1. Input email and password
     2. Validate inputs, if not send error messages
     3. Hit login and allow user to login
  2. Users can manage their account profile
     1. Users can view their personal information inputs
     2. Upload profile image
     3. Update information
     4. View purchase history

1. Service
   1. Users can book service
      1. Select Plant-sitters
      2. Choose service type
      3. Pick available time for them to pick up or visit
      4. Show summary and add to cart
   2. Users can buy plants
      1. Select plants or products
      2. Add to cart
   3. User can manage service cart
      1. Edit or delete service & plant
      2. Can change address or name
      3. Confirm purchase
   4. User can talk to plant sitters
      1. User can view chatting system in communication tab
      2. Add or delete chatting windows
   5. Users get notifications
      1. Users will get notification when plant sitters accept their request
      2. Can delete old notifications
   6. Users can leave ratings
      1. Users can rate plant sitters with a 5-star system after service is done
   7. Users can manage bookings
      1. Select booking
      2. Edit booking
      3. show available time schedule and edit
      4. Delete booking
2. Plant sitter account
   1. Plant sitter can manage their account
      1. Plant sitter can view their personal information inputs
      2. Chat with customers
      3. Upload profile image
      4. Update information
   2. Plant sitter can accept customer’s request
      1. Review customer’s request
      2. Confirm the request

## 2.4 Nonfunctional Requirements

1. Operational
   1. The system should be able to operate on any web browser software that has access to stable Internet
   2. The system should be mobile responsive
2. Performance
   1. Any interaction between the user and the system should be less than 3 seconds
   2. The system should receive updated inventory information every 30 minutes
   3. Social hub posts should not exceed a latency of 2 seconds
3. Security
   1. Signup/login and payment pages should use SSL encryption
   2. New accounts should only be successfully created if an acceptable password has been provided
   3. Only administrators can have access to all routes, plant sitter records, customer bookings, social hubs posts, and store inventory
   4. Customers can access the plant sitter chat system only during business hours
   5. Administrators are authorized to terminate customer account/social hub access if the customer has violated privacy and terms of service policies
4. Cultural & Political
   1. The system should display Canadian currencies and time zones
   2. The system should meet Web Content Accessibility Guidelines
   3. The system should adhere to AODA standards

## 2.5 Project Scope

**Project Goal and Objectives:**

The primary goal of this project is to introduce a convenient booking application for plant sitters, attracting users who struggle with hiring plant-sitters. Furthermore, the project aims to establish a community structure that can share plant management tips as well as a system that can sell and purchase plants and plant-related products.

* Users will be able to make a reservation for the plant-sitters
* Plant-sitters will be able to check their reservations
* Users will be able to upload their concerns about their plants and share the knowledge of plants
* Users will be able to purchase plants easily through application

**Project Boundaries:**

Within scope**:**

1. Create a web app that will implement the systems for plant-sitter booking where users can log in through authentication
2. Plant sitter can upload their profile and check their reservation
3. Secure payment by user’s authentication
4. Review system on plant-sitters
5. Community page to share user’s knowledge and experience
6. Notifications on reservation confirmation will be sent via email with plant sitter’s profile

Out of Scope:

1. Functionality for this situation when the application is inaccessible (no data or Wi-Fi)

Project Deliverables:

1. Front-end web application
2. Back-end application (API)
3. Database
4. Mail Service

Success Criteria:

* Users and plant sitters can successfully use the system to book plant sitting services

## 2.6 System Risks

|  |  |
| --- | --- |
| **Risk** | **Response** |
| Web application might leak users’ personal information and data | Implement webpage based on authorization feature. Add more unit test |
| Team might not be familiar with development and how to implement feature | As a team problem, help with finding solution and implement code together during meeting |
| User might exploit posting feature | Implement feature to prevent users from spam, personal advertisement etc. |
| Server Error | Constantly check our server is running and there is no malicious data in database. |

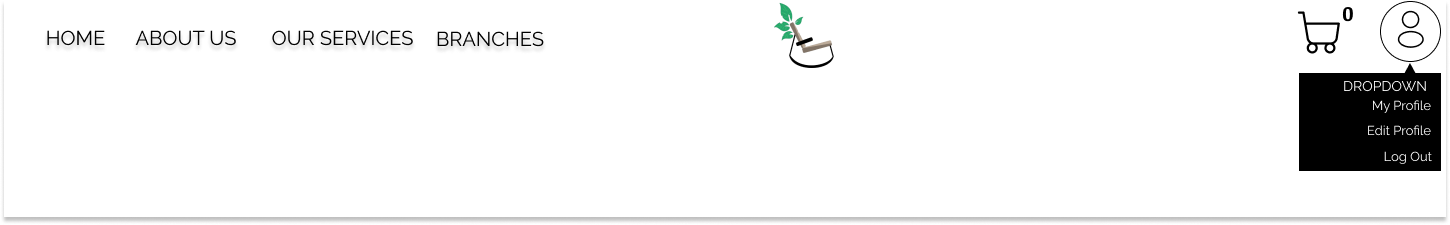
## 2.7 Operating Environment

Users can access the system with a device that can connect to the Internet and open a web browser on a compatible operating system. The system can be viewed on operating systems such as Windows, MacOS, iOS, Linux, and Android with the following versions:

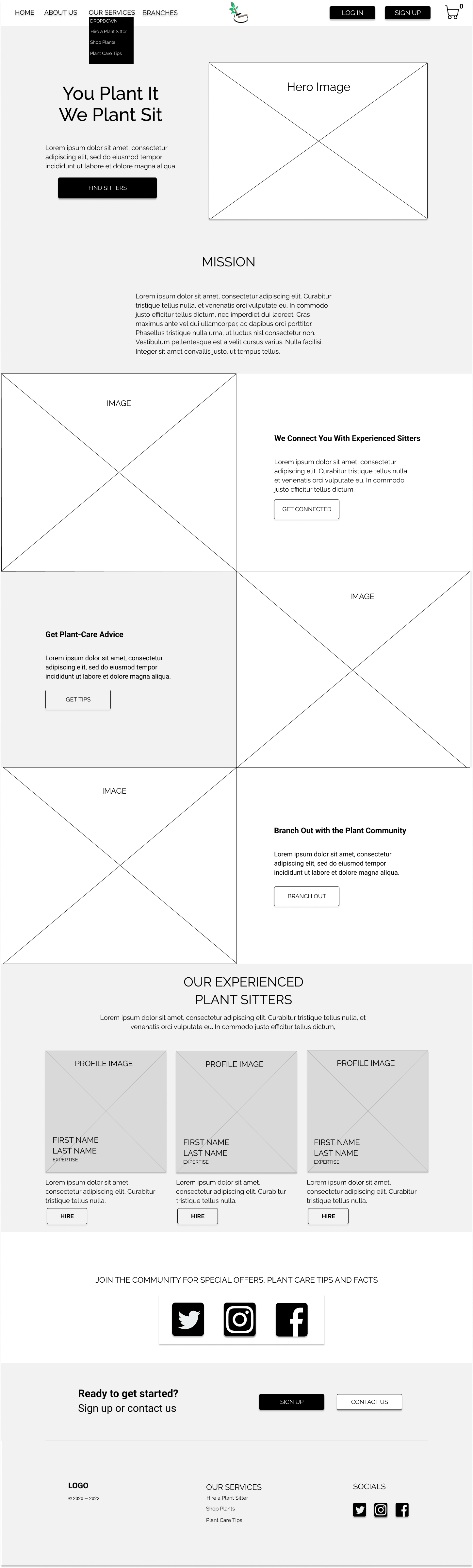
* Windows and MacOS
* Any mobile Devices

## 2.8 UI/UXD Interface Mock-ups

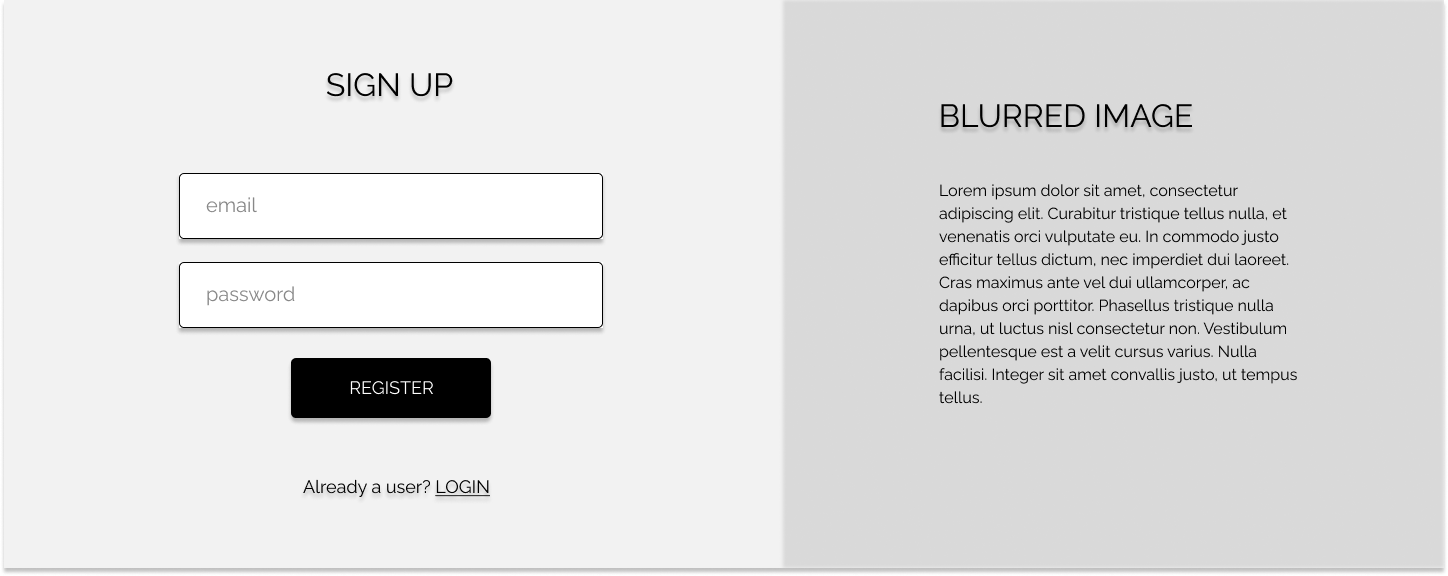
**Navbar Signed In**



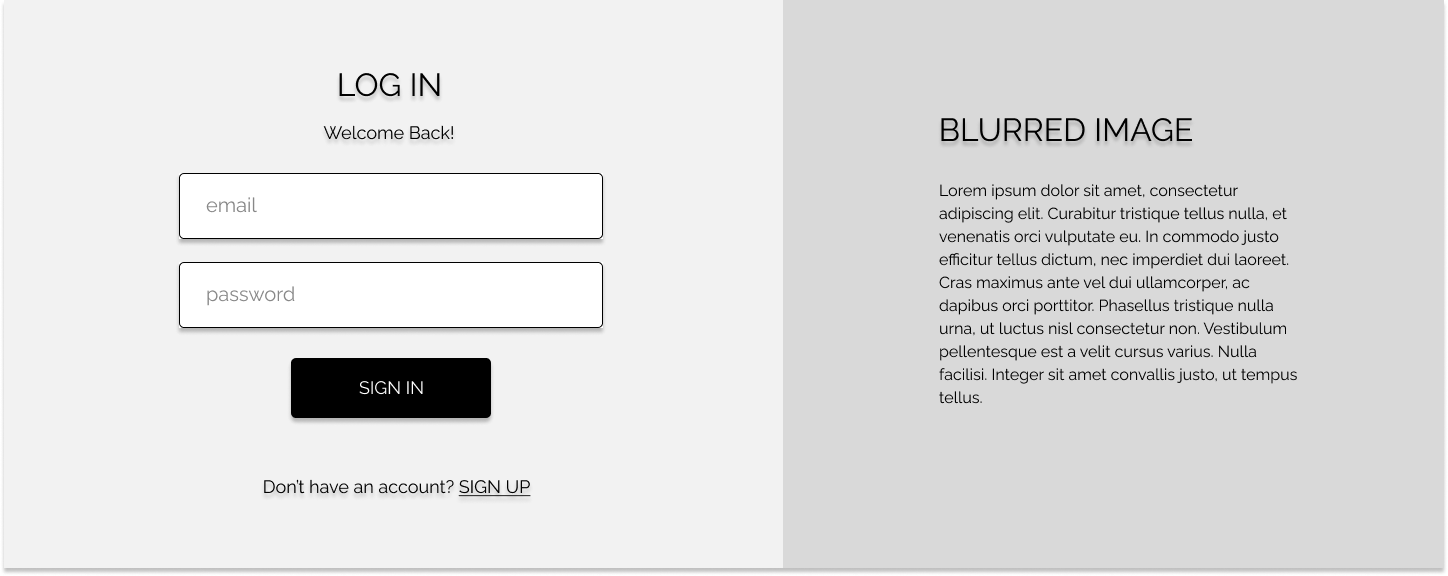
**Landing Page**



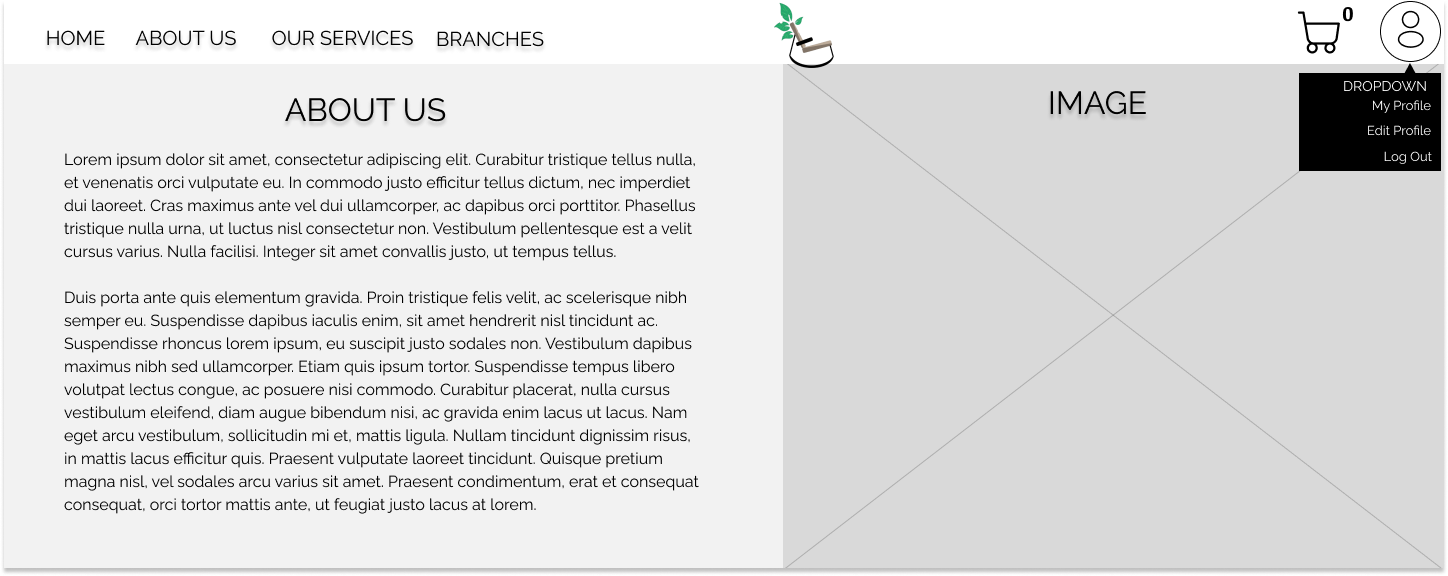
**Sign Up**



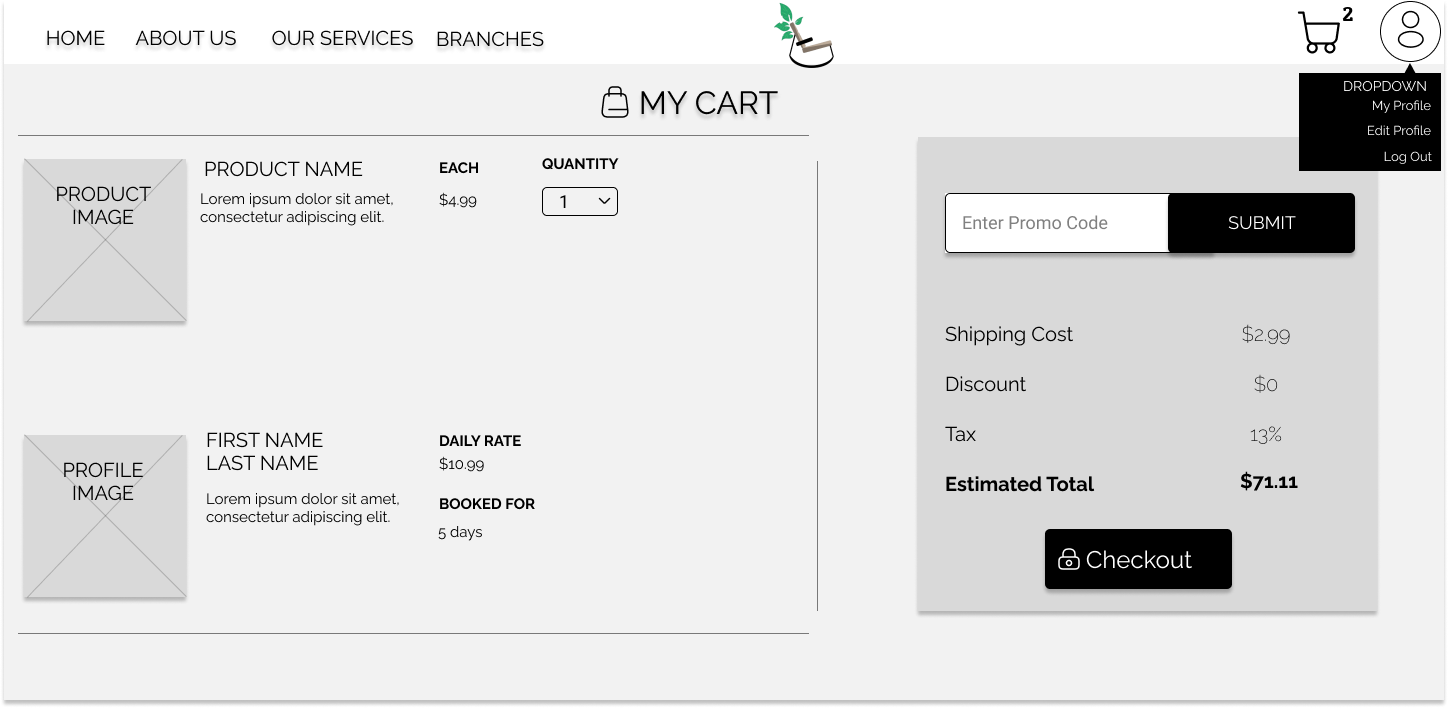
**Log In**



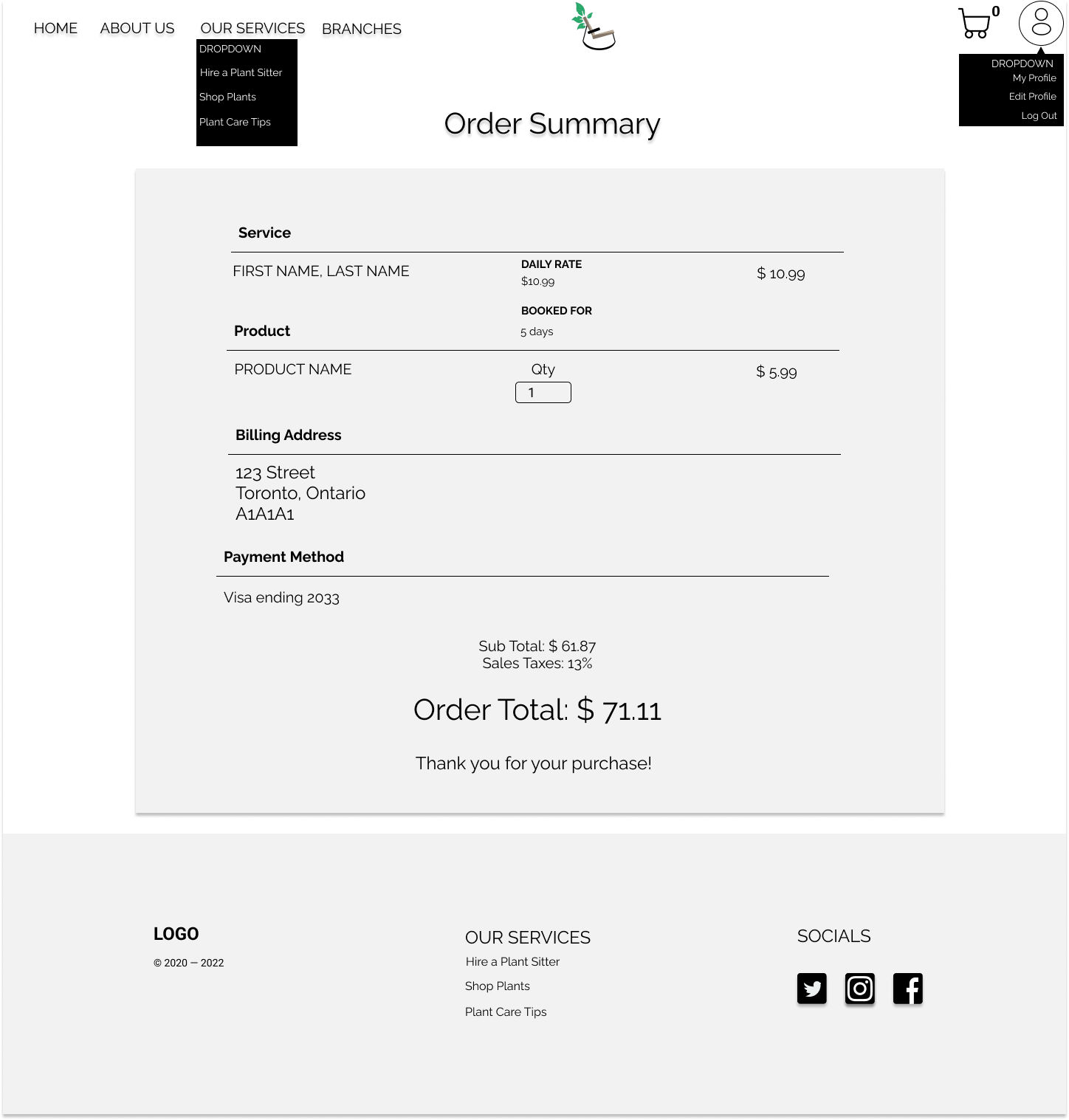
**About Us**



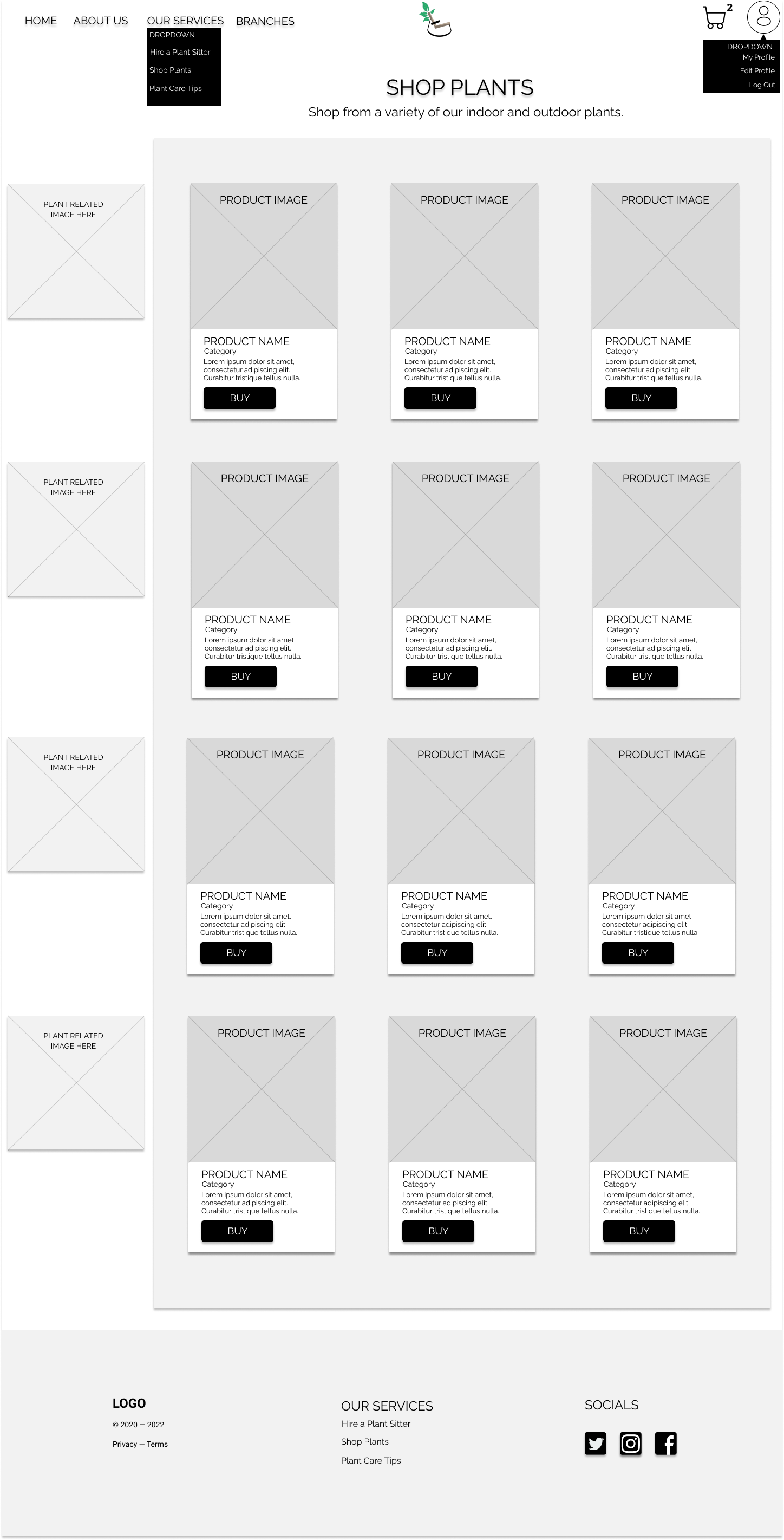
**Shopping Cart**



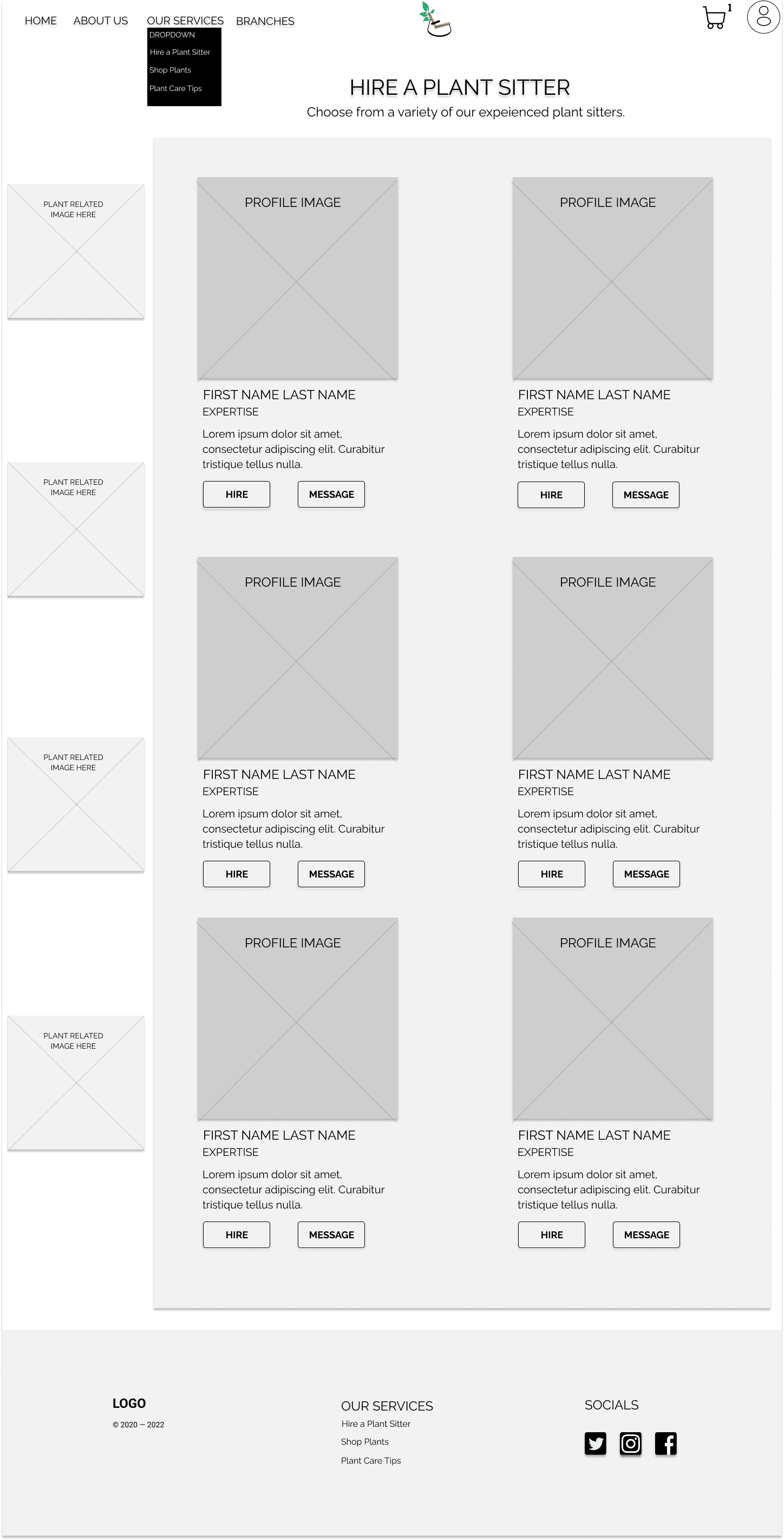
**Order Summary Receipt Page**



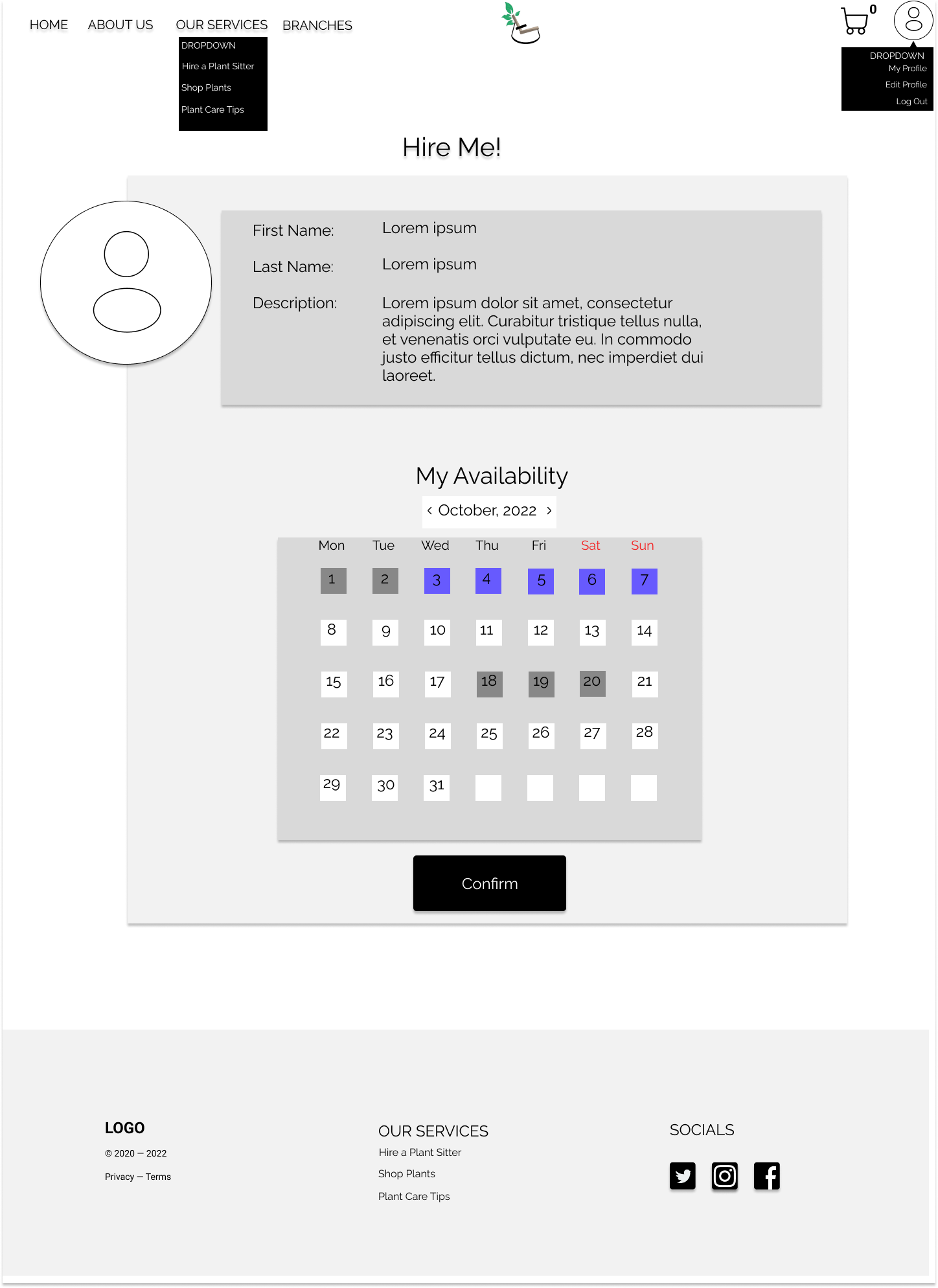
**Plant Shop Page**



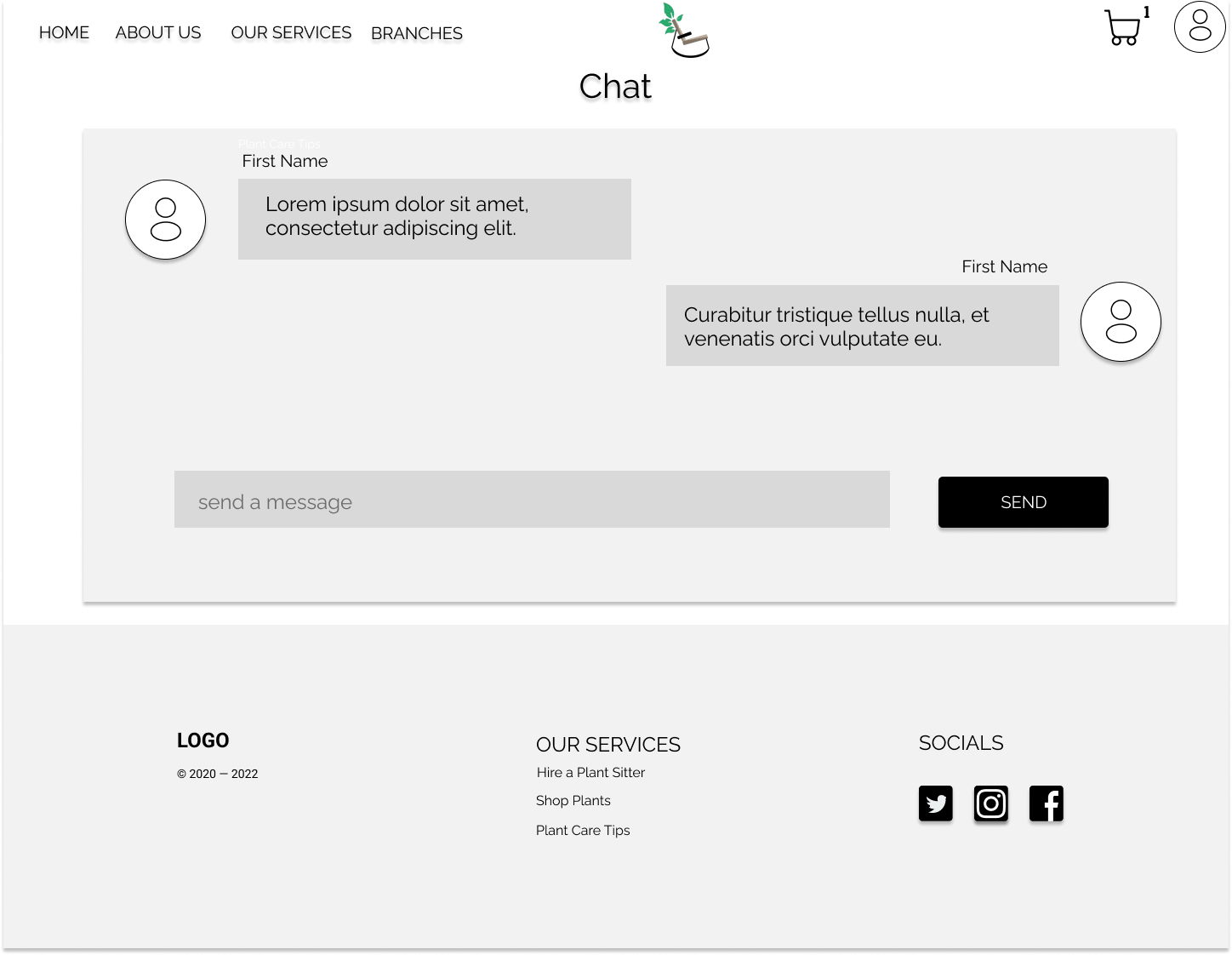
**Plant Sitters Page**



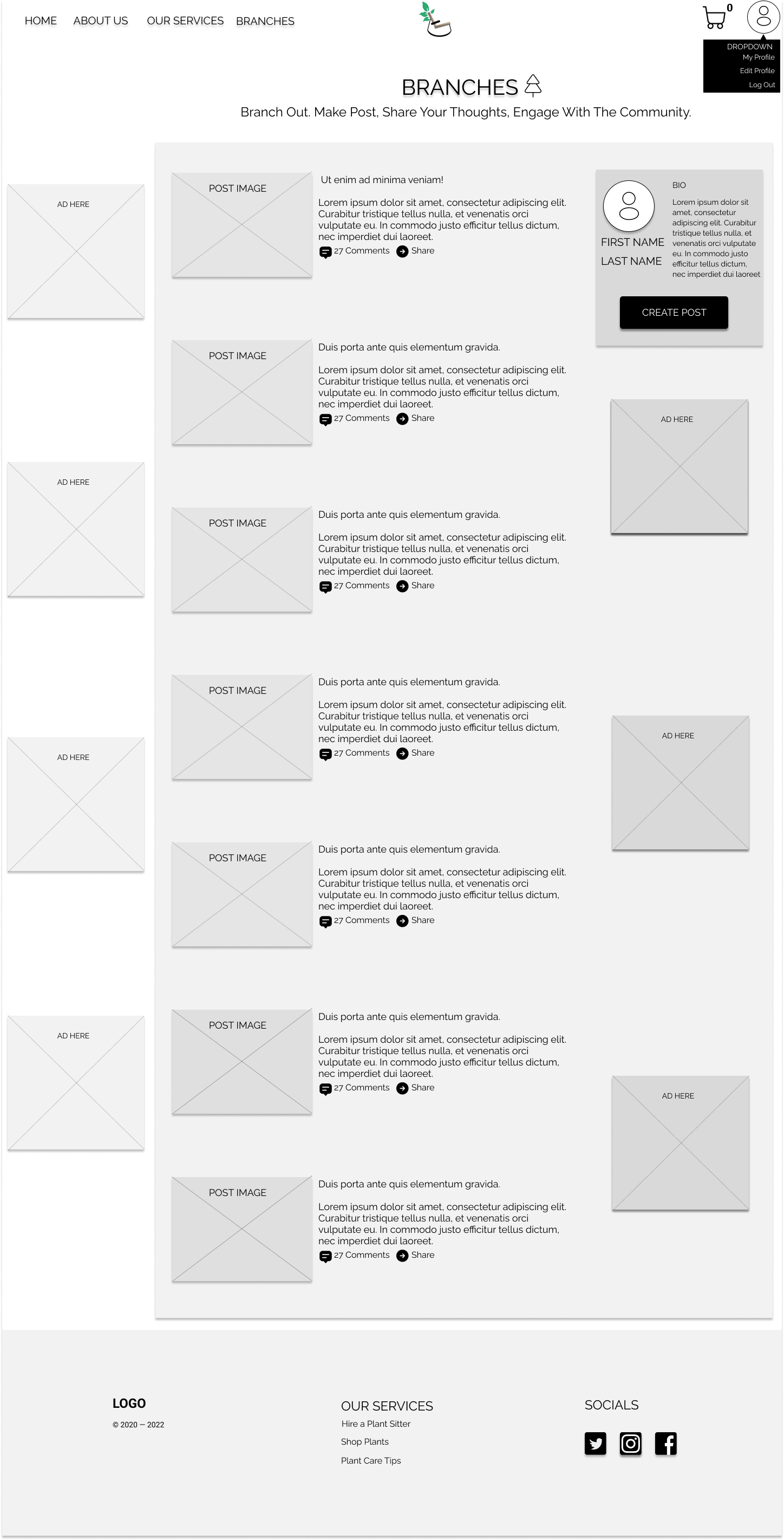
**Hire Page**



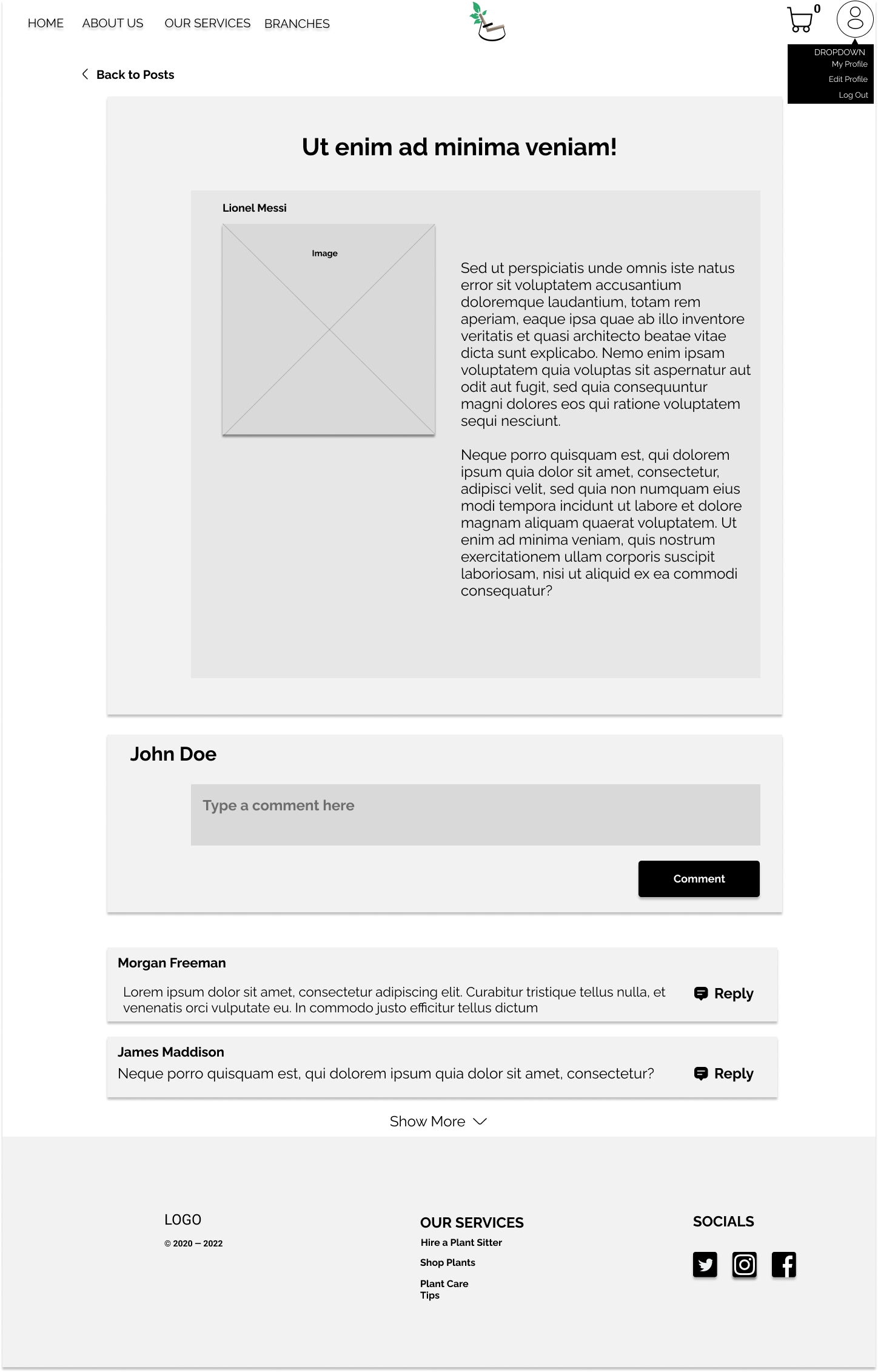
**Message Page**



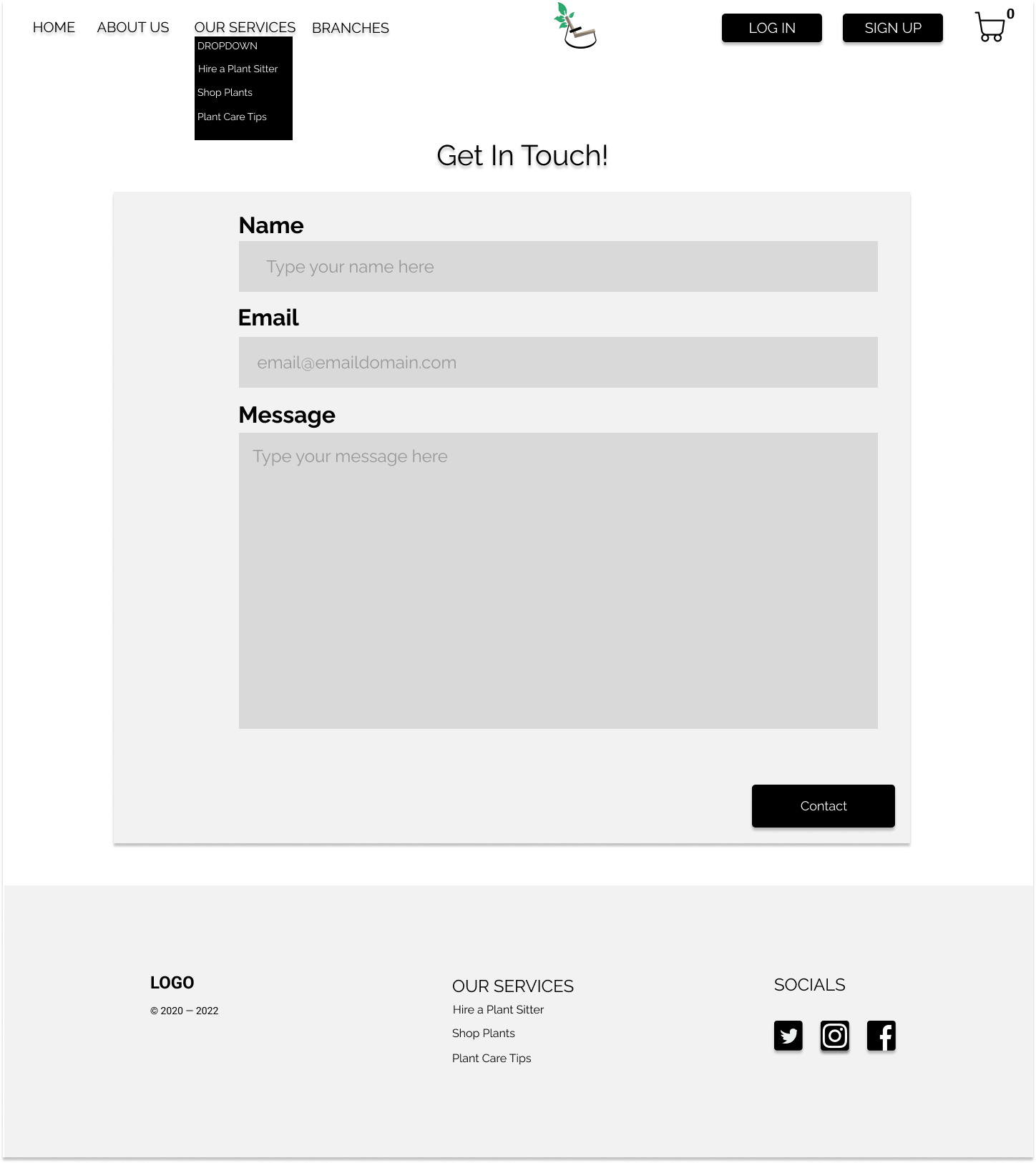
**Branches Page**



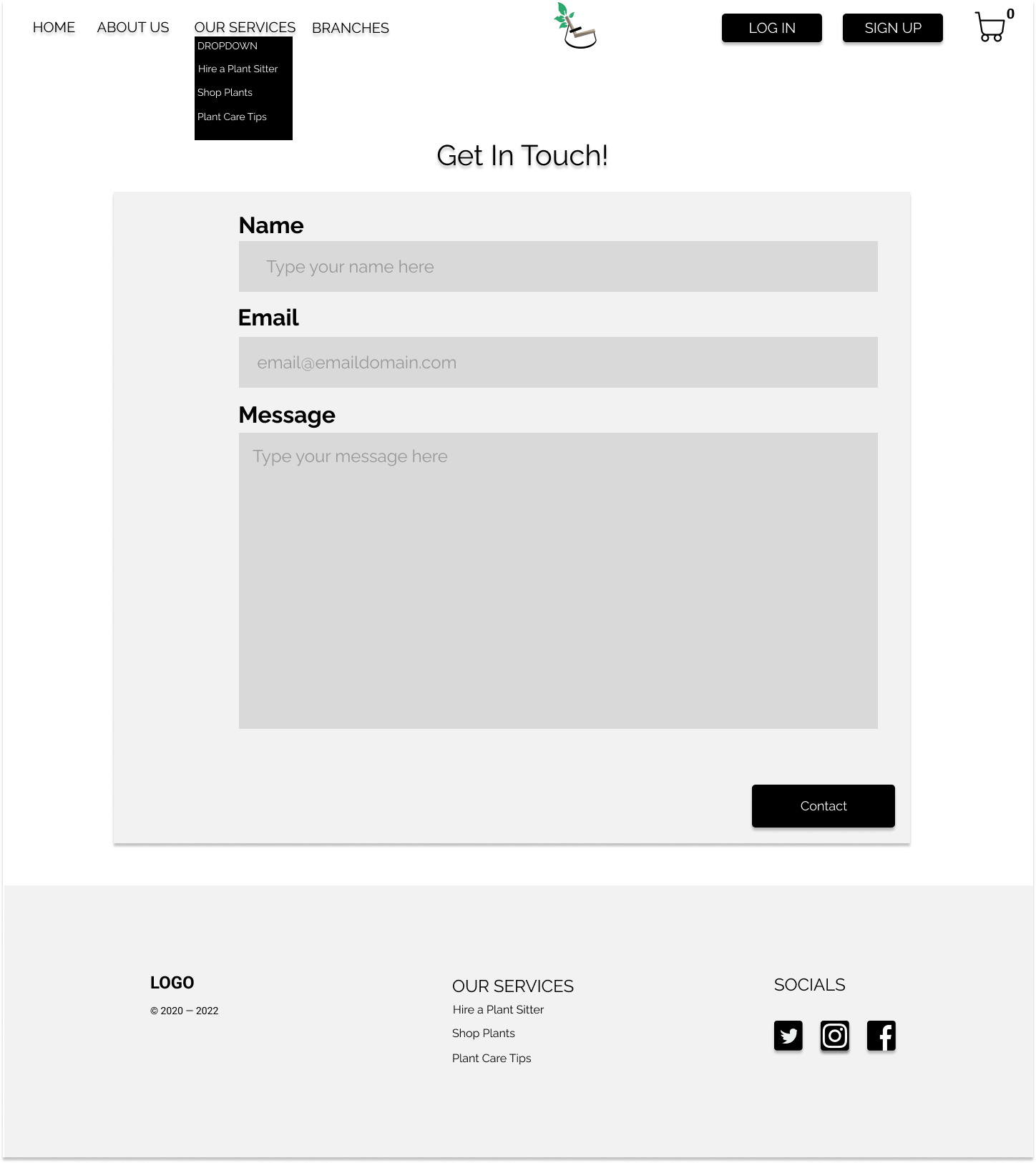
**Detailed Post Page**



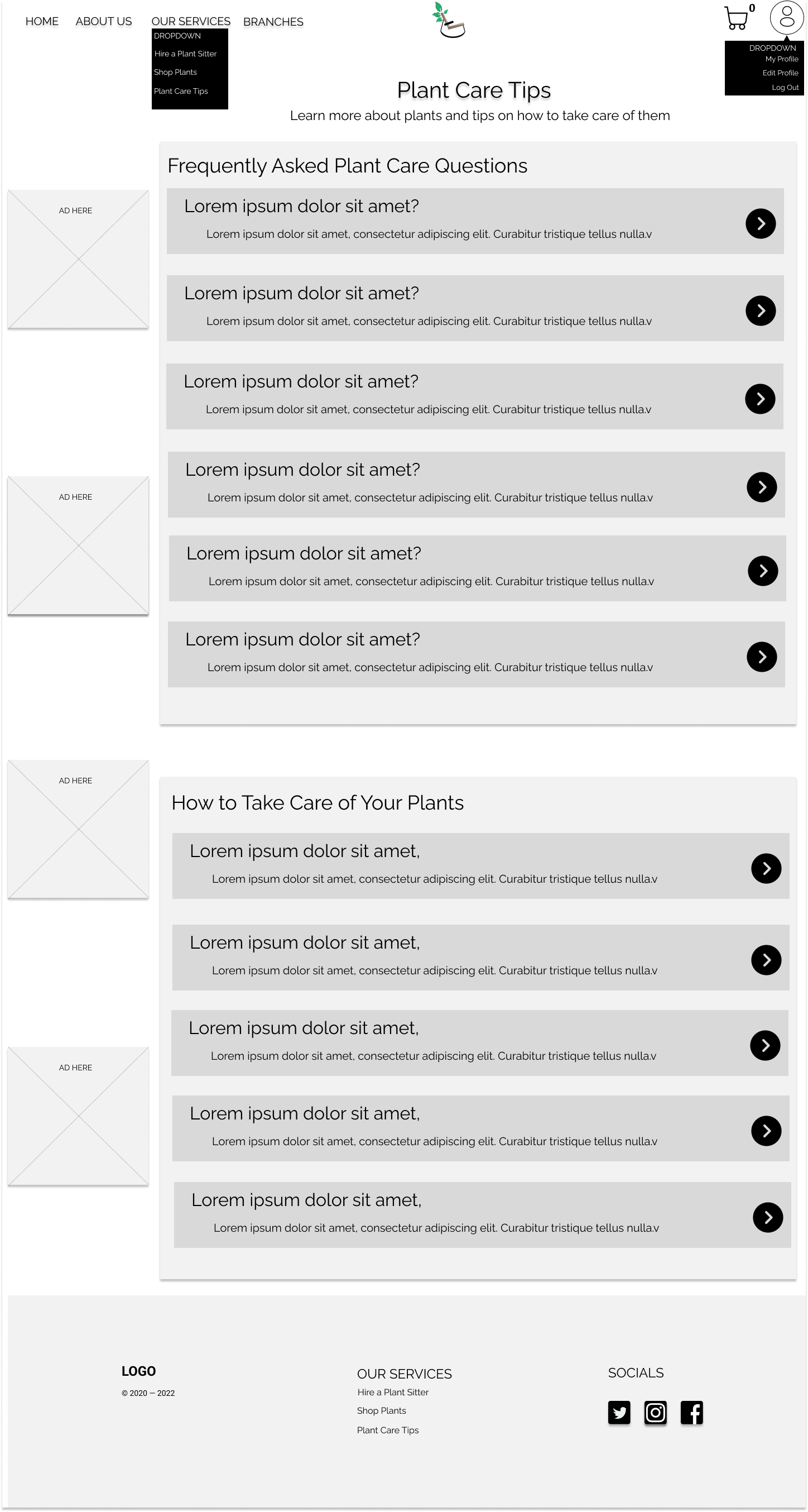
**Contact Us Page**



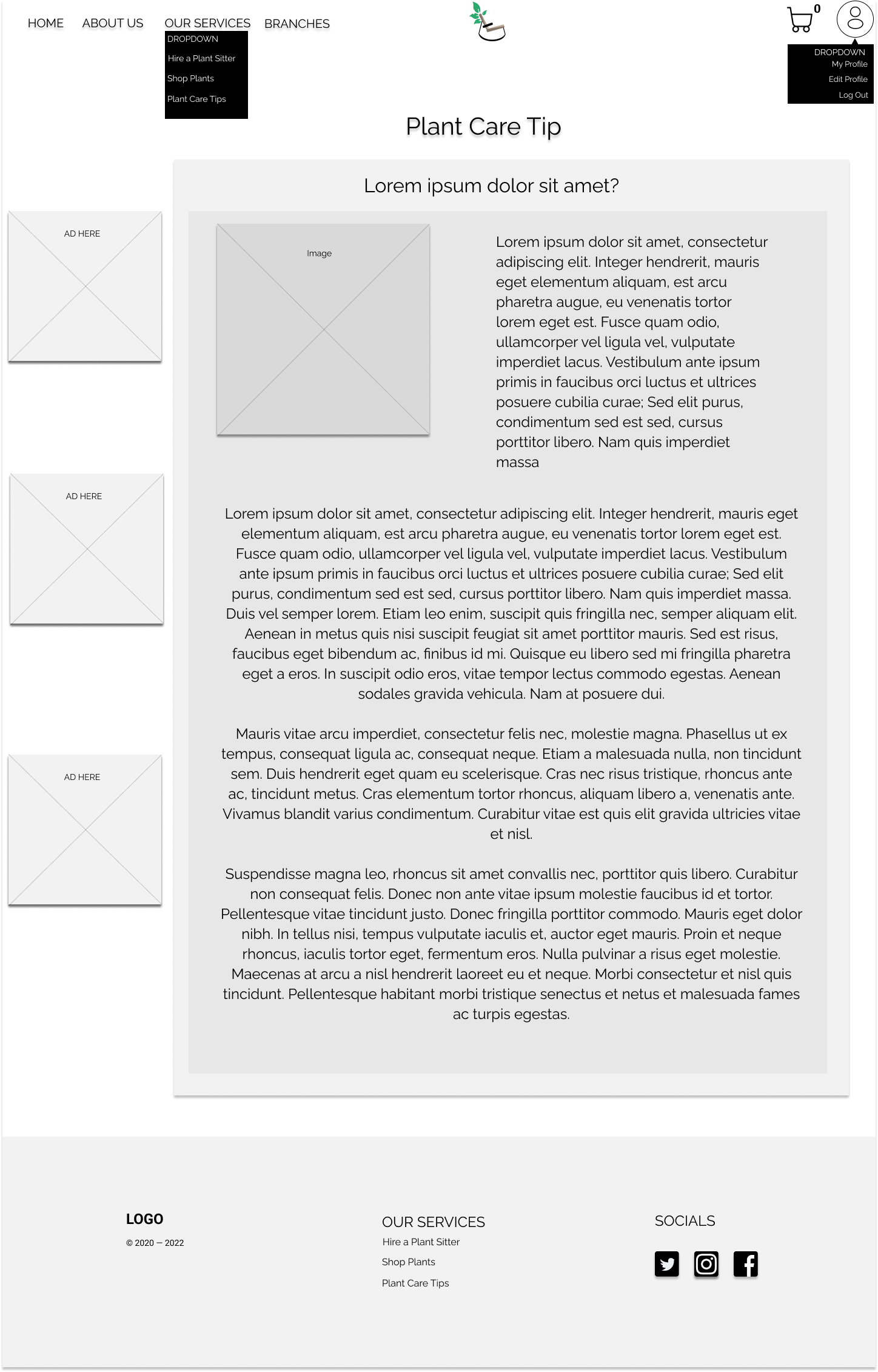
**Create A Post Page**



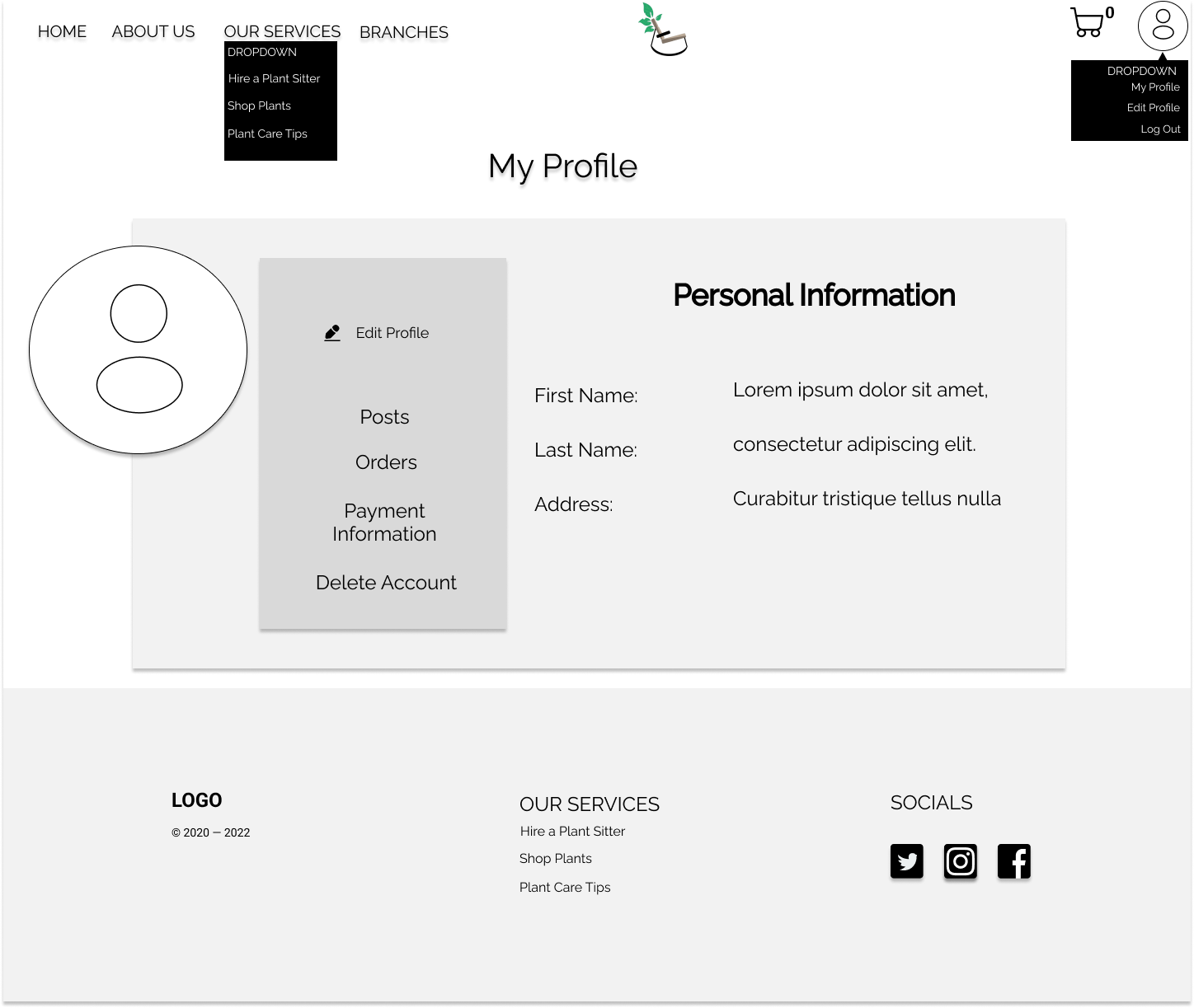
**Plant Care Tips Page**



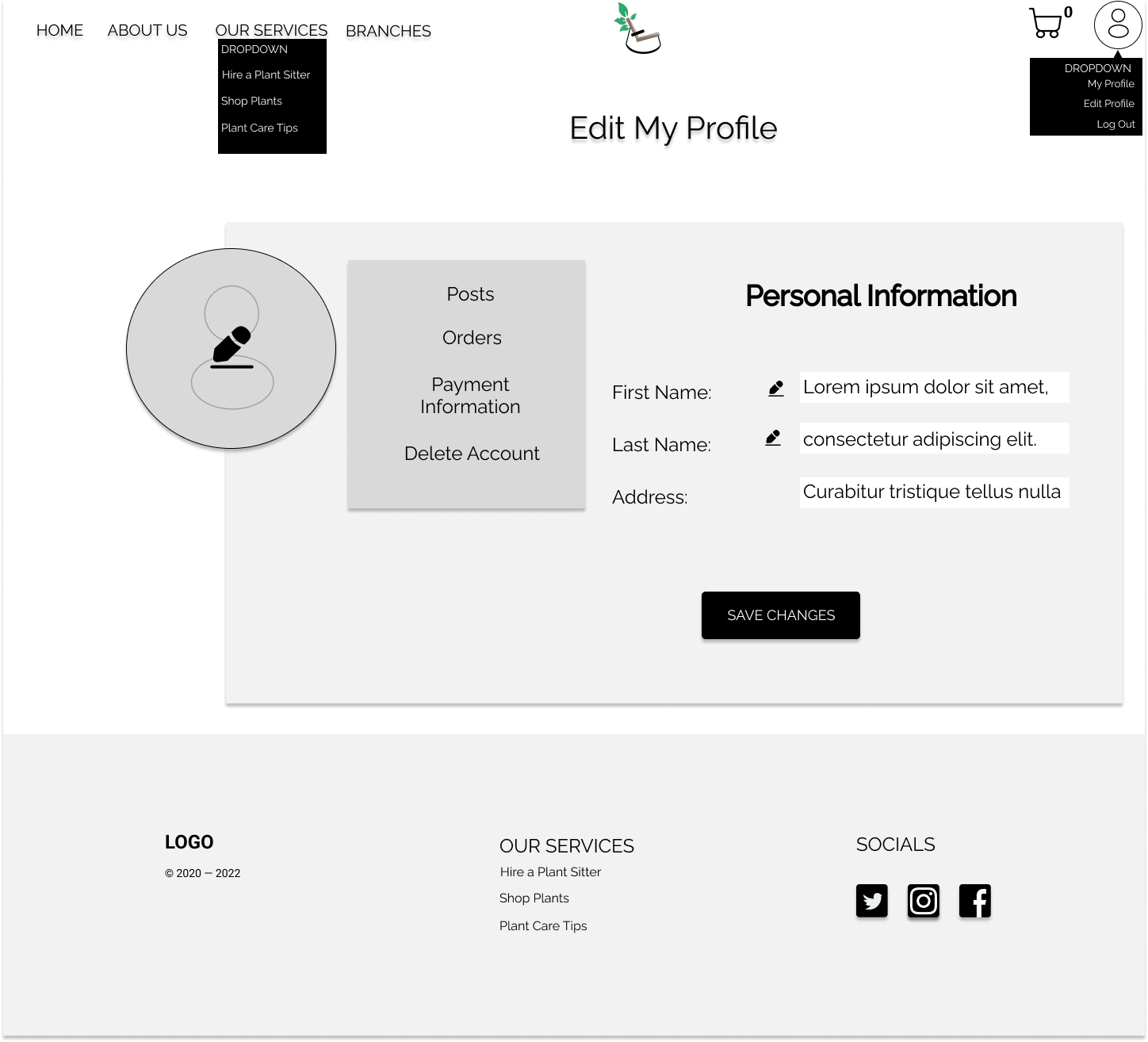
**Detailed Plant Tip Page**



**Profile Page**



**Edit Profile Page**



# Process and Data Modeling

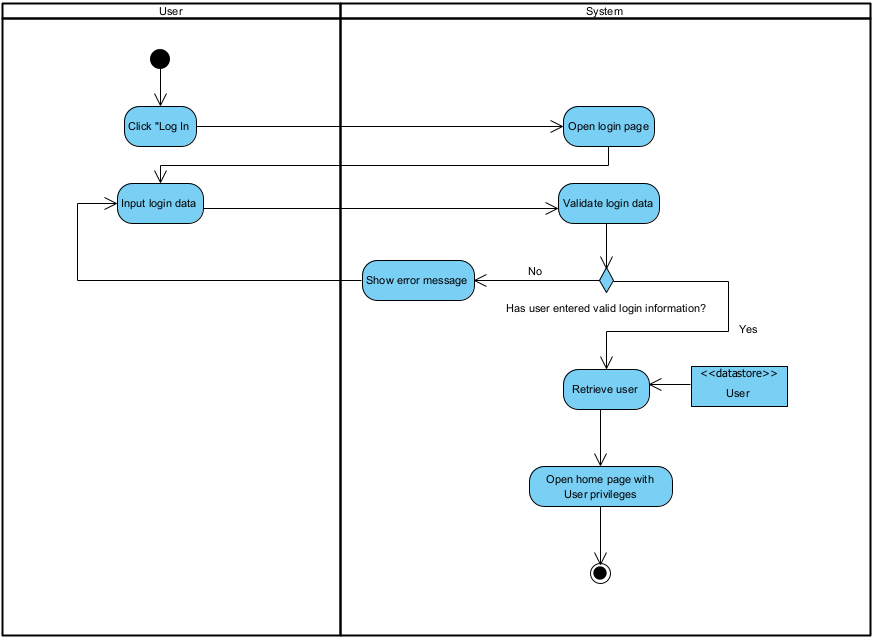
## **3.1 UML/DFD Modeling and Data Modeling**

### Activity Diagrams and Data Flow diagram

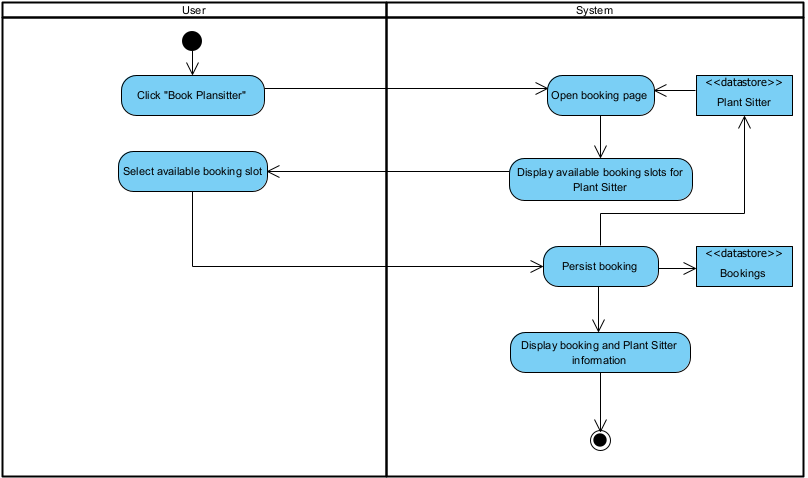
**CREATE A USER (REGISTER)**



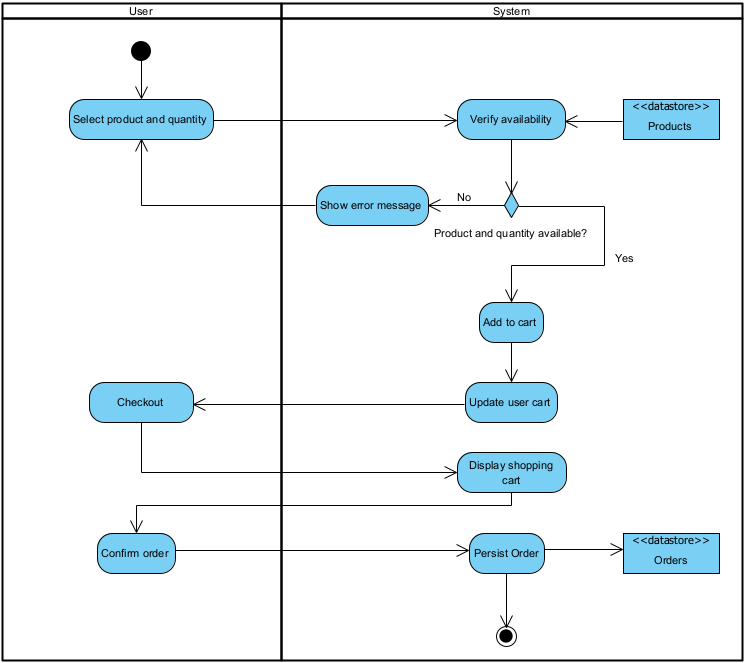
**LOG IN**

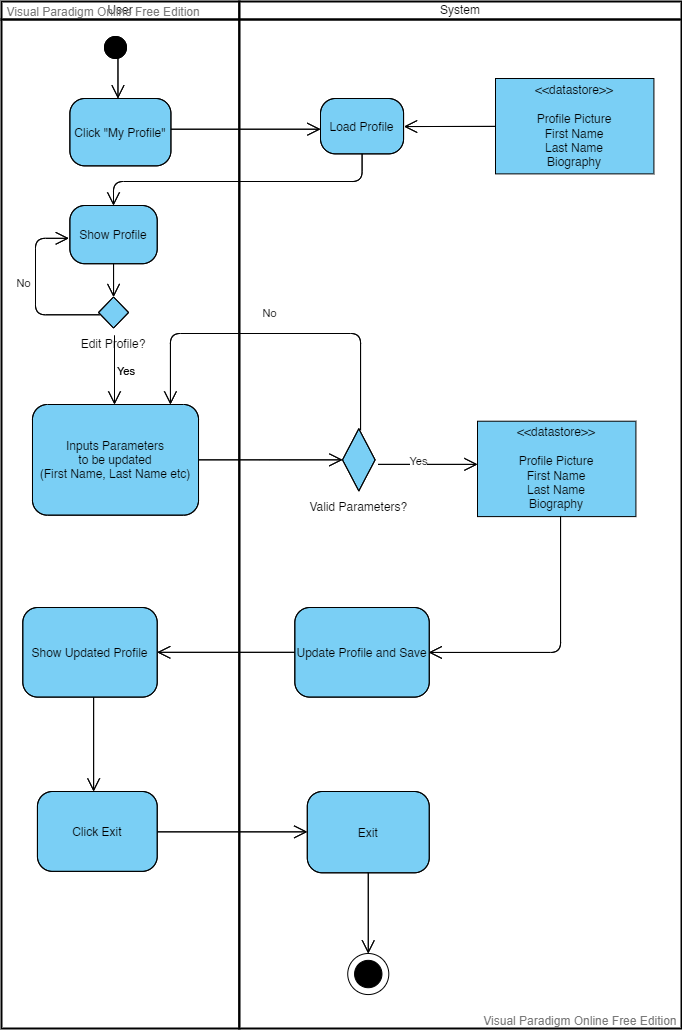


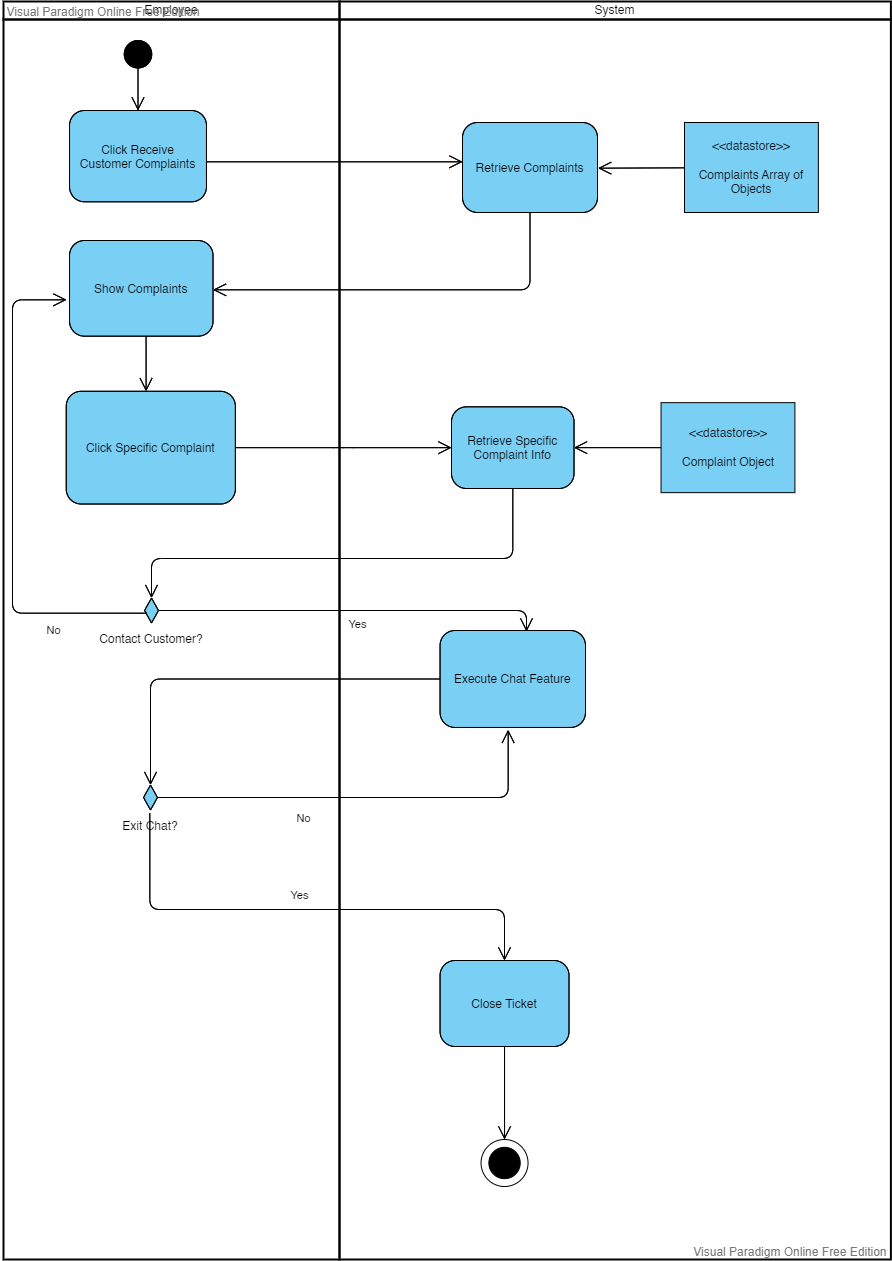
**CREATE BOOKING (BOOK PLANT SITTER)**

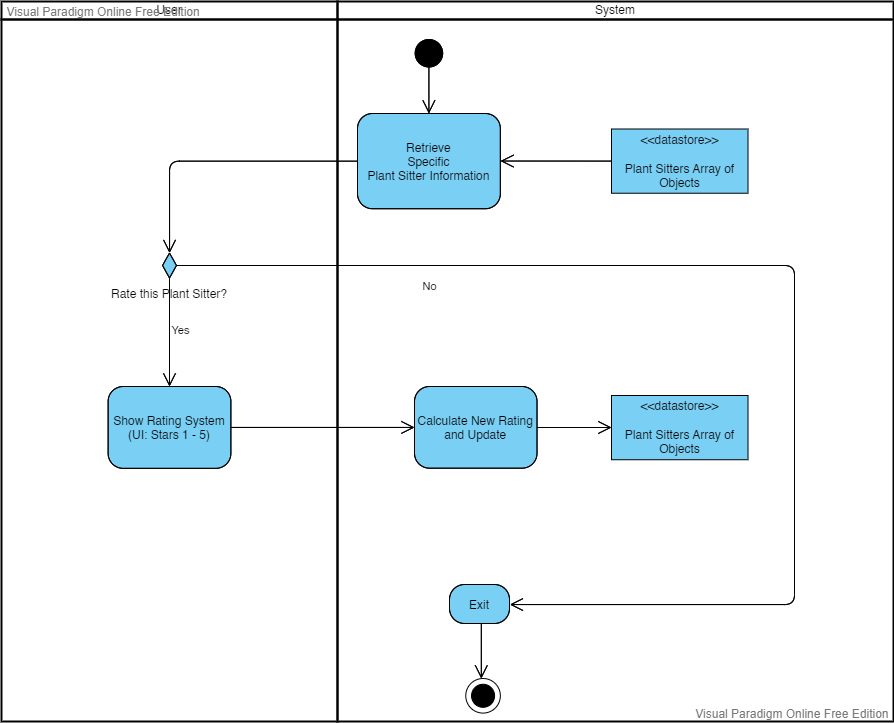


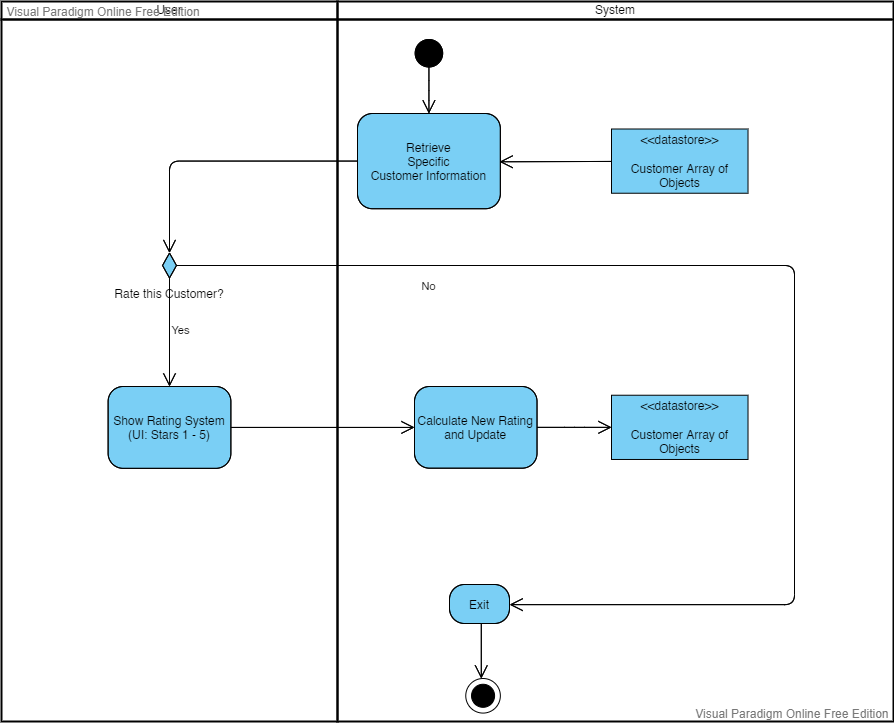
**PURCHASE PRODUCT (ADD PRODUCT TO CART AND CHECKOUT)**

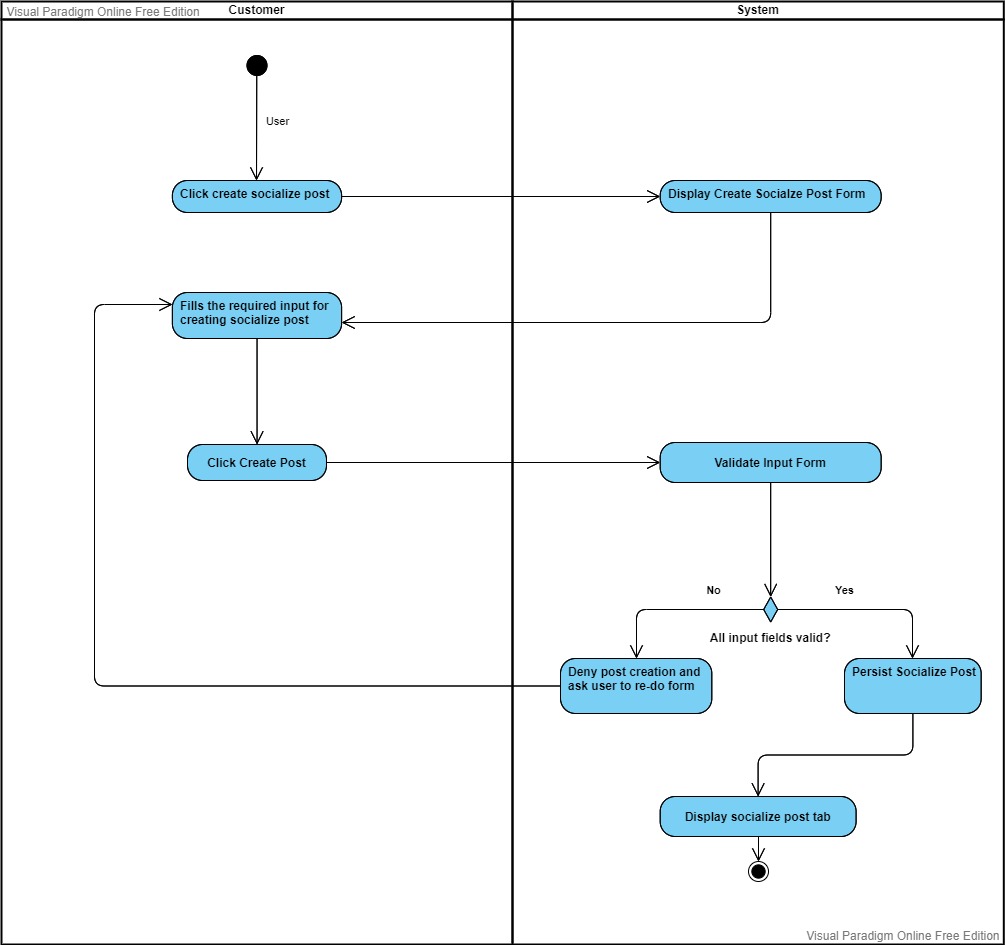


**VIEW AND EDIT PROFILE**  


**RETRIEVE CUSTOMER COMPLAINTS**  
  
  
**RATE PLANT SITTER**



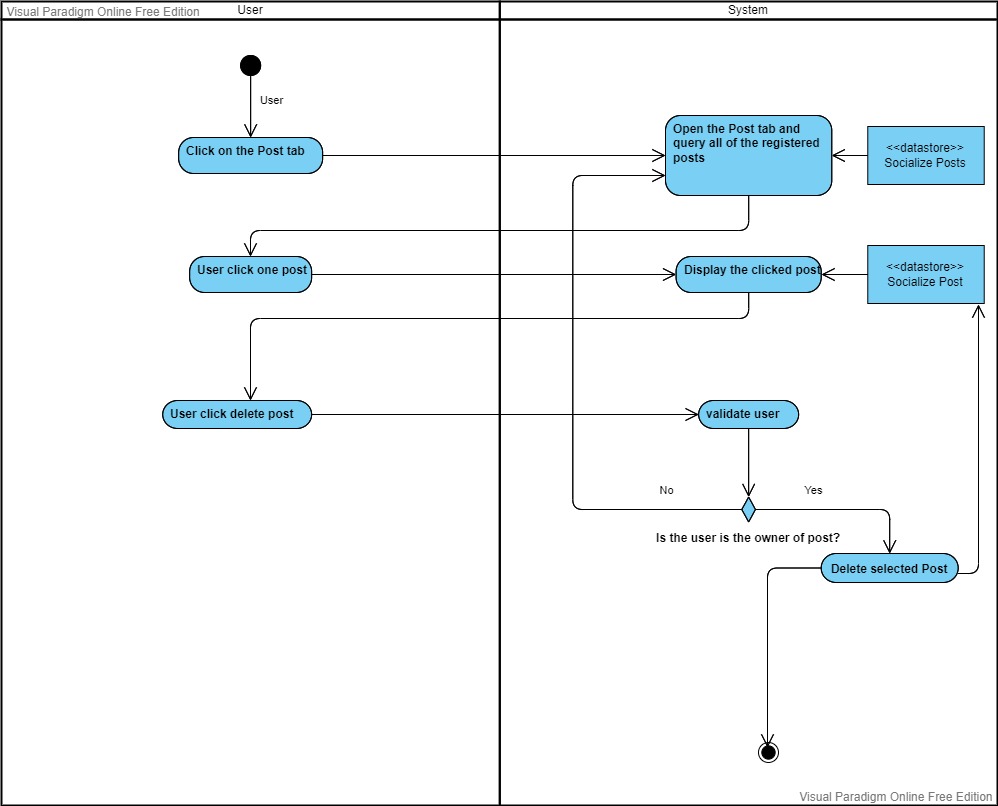
**RATE PLANT OWNER**  
  
**CREATE SOCIALIZE POST**



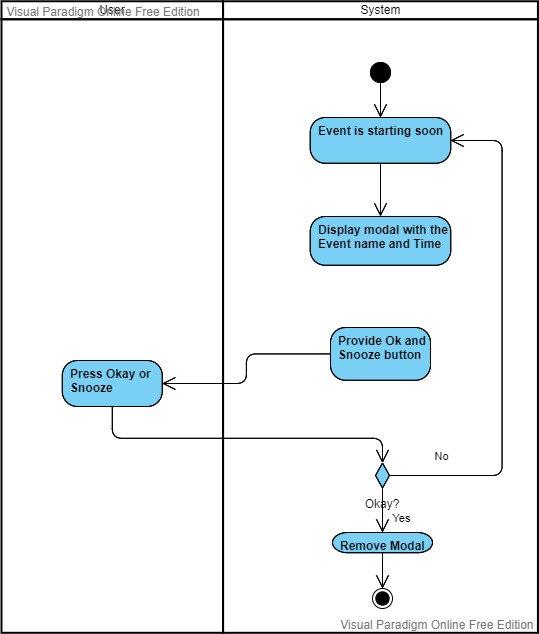
**EDIT SOCIALIZE POST**



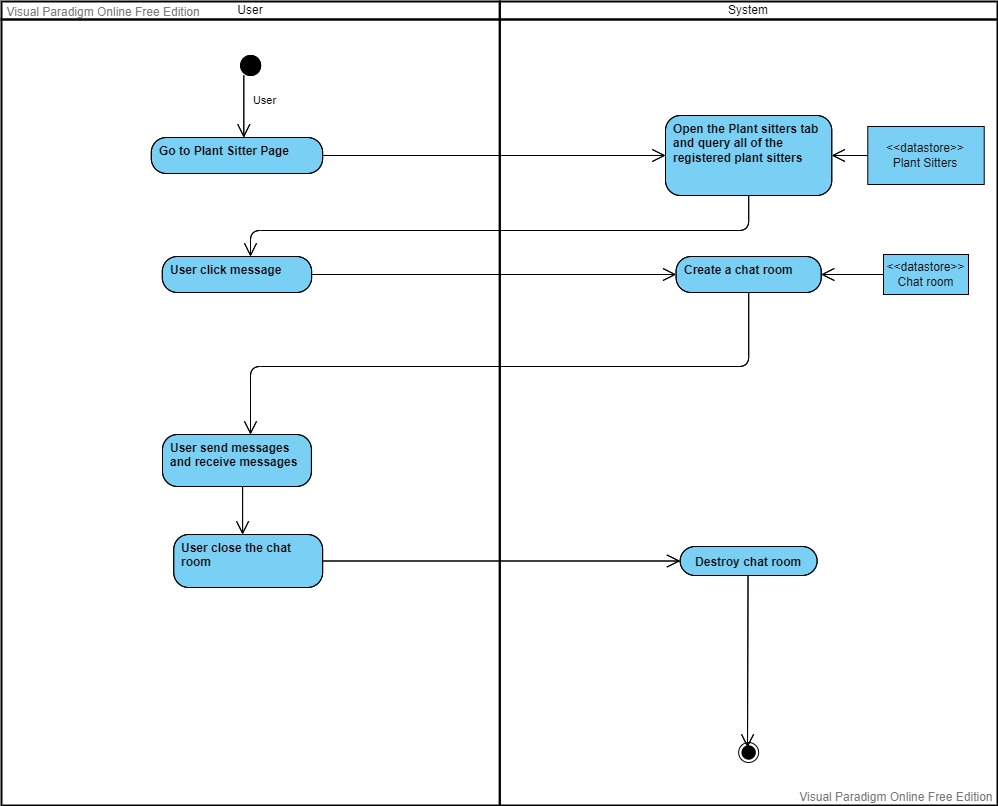
**DELETE SOCIALIZE POST**



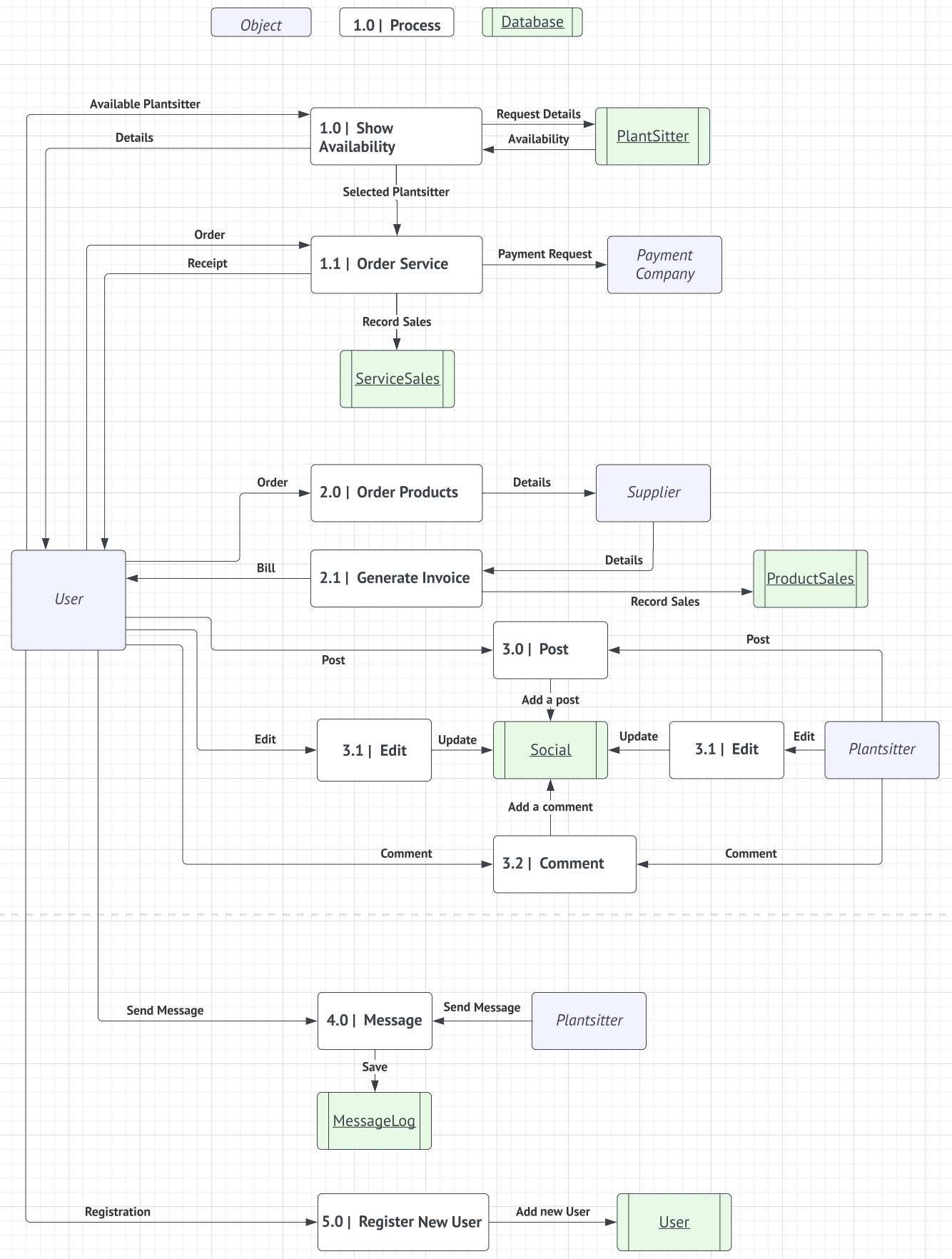
**RECEIVE NOTIFICATION**



**START CHAT**



Data Flow Diagrams

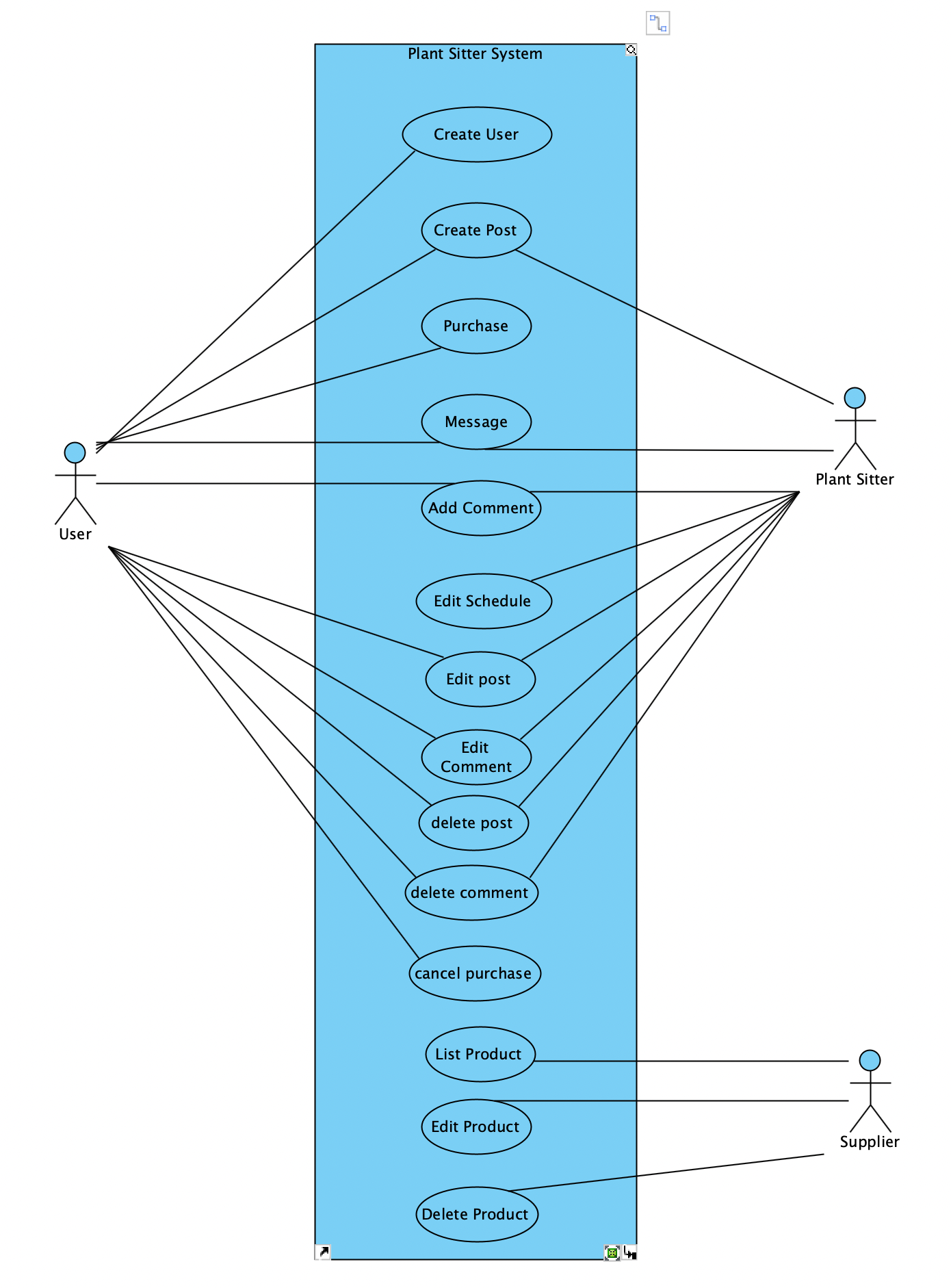


## **3.2 Business Rules**

|  |  |  |
| --- | --- | --- |
| Business Rule Number | Business Rule Description | Related UC |
| BR01 | User must provide a username, email and password to register for the website. | Create New User |
| BR02 | Email should be verified. Email & username should be unique. | Create New User |
| BR03 | Update user database after creating a new user | Create New User |
| BR04 | Maximum of words for post is 100 and Minimum is 10. | Create New Post |
| BR05 | User should be logged in to post | Create New Post |
| BR06 | Update Social database after user post | Create New Post |
| BR07 | User cannot buy 0 product | Purchase |
| BR08 | Cart cannot be empty when user proceed payment | Purchase |
| BR09 | System updates serviceSales or productSales database after payment is done | Purchase |
| BR10 | User can send text and photo. | Chatting |
| BR11 | System should not accept a malicious text or code to hack system | Chatting |
| BR12 | MessageLog database will save every chatting history | Chatting |
| BR13 | User must provide correct info to login | Login |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## **3.3 Use Case Specifications with corresponding interface mockups:**

**Each use case needs to have the following:**

**2- System Use Case Diagrams.**  


**3- Use Case Descriptions.**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Create a User | | |
| Triggering Event | User wants to create a personalized account | | |
| Brief Description | This use case enables the user to create a login and become a registered user | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules | BR001, BR2, BR03 | | |
| Preconditions | User has opened the Create Account page | | |
| Post Conditions | User account is created and saved in the system | | |
| Flow of activities | Actor | | System |
|  | 1. | Requests to create new account | Displays “Create Account” form. Prompts for first name, last name, email address, and password |
|  | 2. | Enters first name, last name, email address, and password | Verifies that the first name, last name, email address, and password have been entered and satisfy requirements. |
|  | 3. | Requests to save | Saves the new user to the system |
| Exception Conditions | User chooses to cancel creating an account | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | User Log in | | |
| Triggering Event | User wants to log into their account | | |
| Brief Description | This use case enables the user to log into their account and access user privileges | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules | BR13 | | |
| Preconditions | User has opened the Login page | | |
| Post Conditions | User is logged into their account | | |
| Flow of activities | Actor | | System |
|  | 1. | Requests to log in | Displays “Log In” form. Prompts for email address and password |
|  | 2. | Enters email address and password | Verifies if the email address and password match an existing user within the system. If successful, redirects to Home page. |
| Exception Conditions | User chooses to cancel logging into account | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Create a Booking | | |
| Triggering Event | User wants to book a Plant Sitter for an allotted time slot | | |
| Brief Description | This use case enables the user to create a booking for a Plant Sitter | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules |  | | |
| Preconditions | User has clicked on the Booking button beside Plant Sitter profile | | |
| Post Conditions | Booking for desired Plant Sitter has been created and saved to the system | | |
| Flow of activities | Actor | | System |
|  | 1. | Requests to book a Plant Sitter | Displays a list of available time slots, prompts for selection |
|  | 2. | Selects available time slot | Verifies a time slot was selected. Displays Plant Sitter details and time slot |
|  | 3. | Requests to confirm | Saves booking to the system |
| Exception Conditions | User chooses to cancel creating a booking | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Add to Cart/Purchase product | | |
| Triggering Event | User wants to add product(s) to the shopping cart and purchase | | |
| Brief Description | This use case enables the user to products to a shopping cart and persist an order | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules | BR07, BR08, BR09 | | |
| Preconditions | User has confirmed a booking or clicks the “Add to Shopping Cart” button under a product | | |
| Post Conditions | Product is placed in the shopping cart and an order will be placed | | |
| Flow of activities | Actor | | System |
|  | 1. | Selects product and quantity | Verifies if the product and quantity is available. Product added to Shopping Cart. Updates User Shopping Cart |
|  | 2. | Clicks on the Shopping Cart icon to checkout | Displays a list of products placed within the Shopping Cart. Prompts to confirm products and place the order |
|  | 3. | Requests to confirm order | Order is created and saved to the system. User Shopping Cart is cleared. |
| Exception Conditions | User chooses to cancel purchasing a product or removes all products from shopping cart | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Purchase | | |
| Triggering Event | User clicks to purchase products and services | | |
| Brief Description | User purchases products and service | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules | B07, B08, B09 | | |
| Preconditions | A user is signed in. | | |
| Post Conditions | Purchase history is stored in database | | |
| Flow of activities | User | | System |
|  |  | Requests to purchase | Check if quantity of items should not be 0. Shopping cart should not be empty. |
|  |  | Proceed payment | Verify payment. |
|  |  | Confirm | Update sales database. |
| Exception Conditions | User chooses to cancel | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Create New Socialize Post | | |
| Triggering Event | User clicks to create a post | | |
| Brief Description | Create a new thread | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules | B04, B05, B06 | | |
| Preconditions | A user is signed in. | | |
| Post Conditions | A new post is created and stored in a database | | |
| Flow of activities | User | | System |
|  |  | Requests to add a new thread | Display required fields.  Prompts for title and content. |
|  |  | Fill in a title and content. | Verify inputs. Check the minimum and maximum words. |
|  |  | Upload | Update database and create thread. |
| Exception Conditions | User chooses to cancel | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Edit Socialize Post | | |
| Triggering Event | User clicks to edit a post | | |
| Brief Description | Edit a thread | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules |  | | |
| Preconditions | A user is signed in. | | |
| Post Conditions | A modified post is stored in a database | | |
| Flow of activities | User | | System |
|  | 1 . | Requests to edit a thread | Displays a list of posts and prompts for selection |
|  | 2. | Select a post | Prompts for modification |
|  | 3. | Modify a title or content. | Verify inputs. Check the minimum and maximum words. Prompts for confirmation |
|  | 4. | Confirm | Update database and create thread. |
| Exception Conditions | User chooses to cancel | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Delete Socialize Post | | |
| Triggering Event | User clicks to delete a post | | |
| Brief Description | delete a thread | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules |  | | |
| Preconditions | A user is signed in. | | |
| Post Conditions | A selected post is removed from a database | | |
| Flow of activities | User | | System |
|  | 1 . | Requests to delete a thread | Displays a list of posts and prompts for selection |
|  | 2. | Select a post to delete | Check if the author of the post is the same as the user. Prompts for confirmation |
|  | 3. | Confirm | Delete the post and update database |
| Exception Conditions | User chooses to cancel | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Receive Notification | | |
| Triggering Event | User can receive notification | | |
| Brief Description | User receive notification | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules |  | | |
| Preconditions | A user is signed in. | | |
| Post Conditions |  | | |
| Flow of activities | User | | System |
|  |  |  | System send a notification |
|  |  | Click the notification | Display the detailed information that the notification shows |
| Exception Conditions | User can ignore the notification | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Chatting | | |
| Triggering Event | User can message with plant sitters | | |
| Brief Description | Send message | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Business Rules | B10, B11, B12 | | |
| Preconditions | A user is signed in. | | |
| Post Conditions | Chatting is stored in a database | | |
| Flow of activities | User | | System |
|  | 1. | Request to chat with plant sitters | Display a list of plant sitters and prompts for selection |
|  | 2. | Select a plant sitter | Display a plant sitter’s information and prompt for starting chatting |
|  | 3. | Click “Start Chatting” | Display a chatting box |
| Exception Conditions | User is out of chatting room | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | View and Edit Profile. | | |
| Triggering Event | User wants to modify view or edit profile. | | |
| Brief Description | Allows a logged in user to view and edit profile. | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Preconditions | The user is logged in. | | |
| Post Conditions | Profile is shown and/or updated. | | |
| Flow of activities | Actor | | System |
|  |  | Clicks “My Profile” | Loads Profile information from database and displays it.  Prompts to edit profile. |
|  |  | Chooses to edit profile and inputs sections to be updated. | Validates parameters.  Data is valid and updates profile and saves.  Displays updated profile. |
|  |  | Clicks Exit. | Exits. |
| Exception Conditions | Edit Profile is not selected. | | |

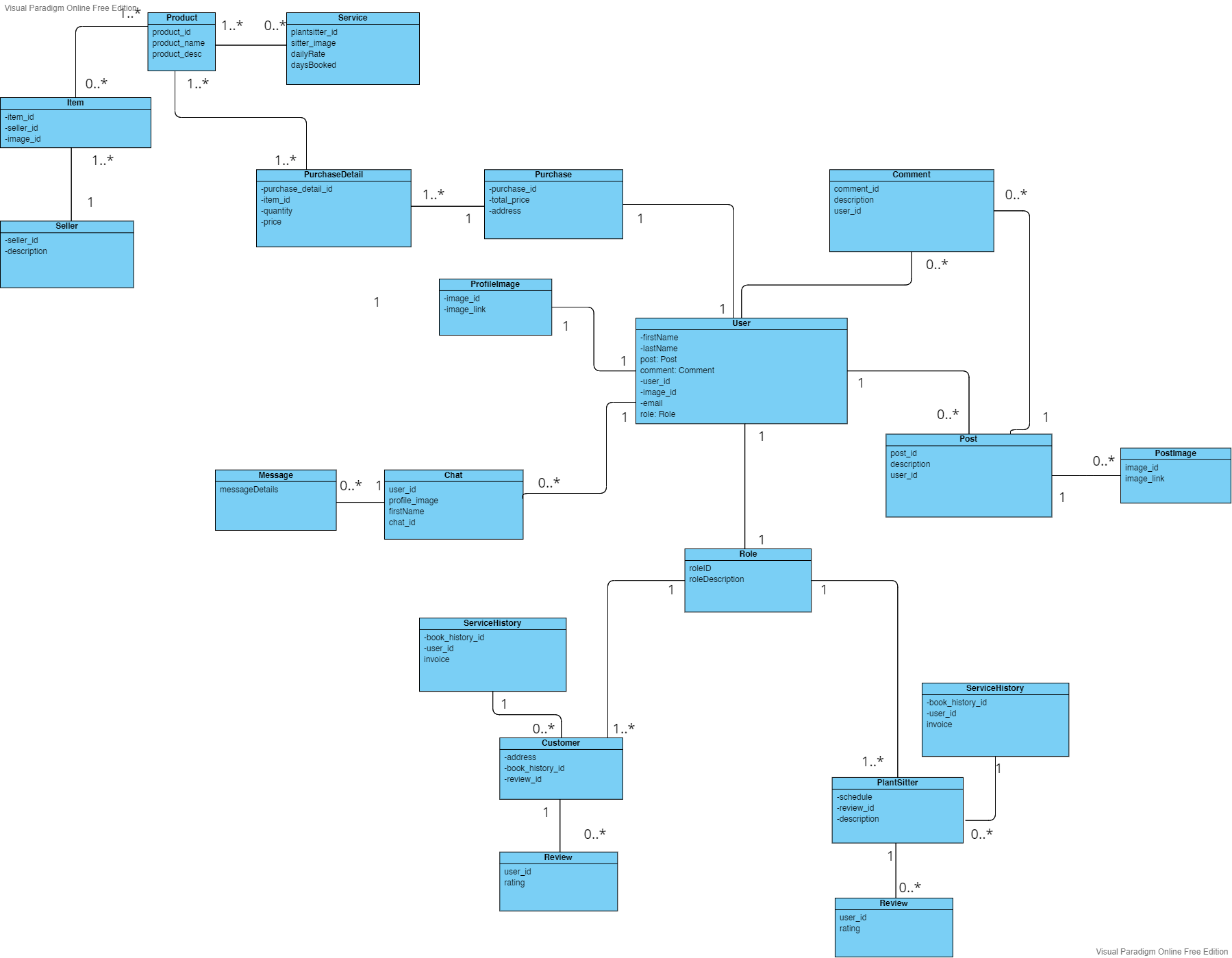
|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Retrieve Customer Complaints. | | |
| Triggering Event | Employee wants to retrieve customer complaints and/or resolve them. | | |
| Brief Description | Allows a logged in employee to view and solved complaints. | | |
| Actors | Employee | | |
| Related Use Cases | Start Chat | | |
| Preconditions | The Employee is logged in. | | |
| Post Conditions | Customer complaints are shown and/or solved. | | |
| Flow of activities | Actor | | System |
|  |  | Clicks “Receive Customer Complaints” | Retrieves complaints from database.  Displays customer complaints. |
|  |  | Clicks specific complaint to solve. | Retrieves specific complaint information from database.  Displays specific complaint detailed information.  Prompts to contact customer. |
|  |  | Clicks “Start Chatting” | Displays Chat feature.  Button prompt to exit chat whenever user is finished. |
|  |  | Clicks “Exit Chat” | Complaint is marked as “Closed” in database.  Re-displays customer complaints from database. |
| Exception Conditions | Does not click to receive customer complaints. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Rate Plant Sitter | | |
| Triggering Event | User wants to rate a plant sitter after their service. | | |
| Brief Description | Allows a logged in user give a plant sitter a rating of 1-5 stars. | | |
| Actors | User | | |
| Related Use Cases |  | | |
| Preconditions | The user is logged in and has had a plant sitting service completed. | | |
| Post Conditions | Rating is saved to database and plant sitter’s profile. | | |
| Flow of activities | Actor | | System |
|  |  |  | Retrieves specific plant sitter information from database.  Prompts to rate plant sitter by displaying Rating System (1-5 stars) |
|  |  | Clicks their desired rating. | Stores new rating, calculates, and updates new rating into database.  Exits UI automatically. |
| Exception Conditions | User chooses not to rate plant sitter. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | Rate Plant Owner | | |
| Triggering Event | Plant Sitter wants to rate a plant owner after servicing them. | | |
| Brief Description | Allows a logged in plant sitter to give a plant owner a rating of 1-5 stars. | | |
| Actors | Plant Sitter | | |
| Related Use Cases |  | | |
| Preconditions | The plant sitter is logged in and has completed servicing a customer. | | |
| Post Conditions | Rating is saved to database and customer’s profile. | | |
| Flow of activities | Actor | | System |
|  |  |  | Retrieves specific customer information from database.  Prompts to rate customer by displaying Rating System (1-5 stars) |
|  |  | Clicks their desired rating. | Stores new rating, calculates, and updates new rating into database.  Exits UI automatically. |
| Exception Conditions | Plant Sitter chooses not to rate a customer. | | |

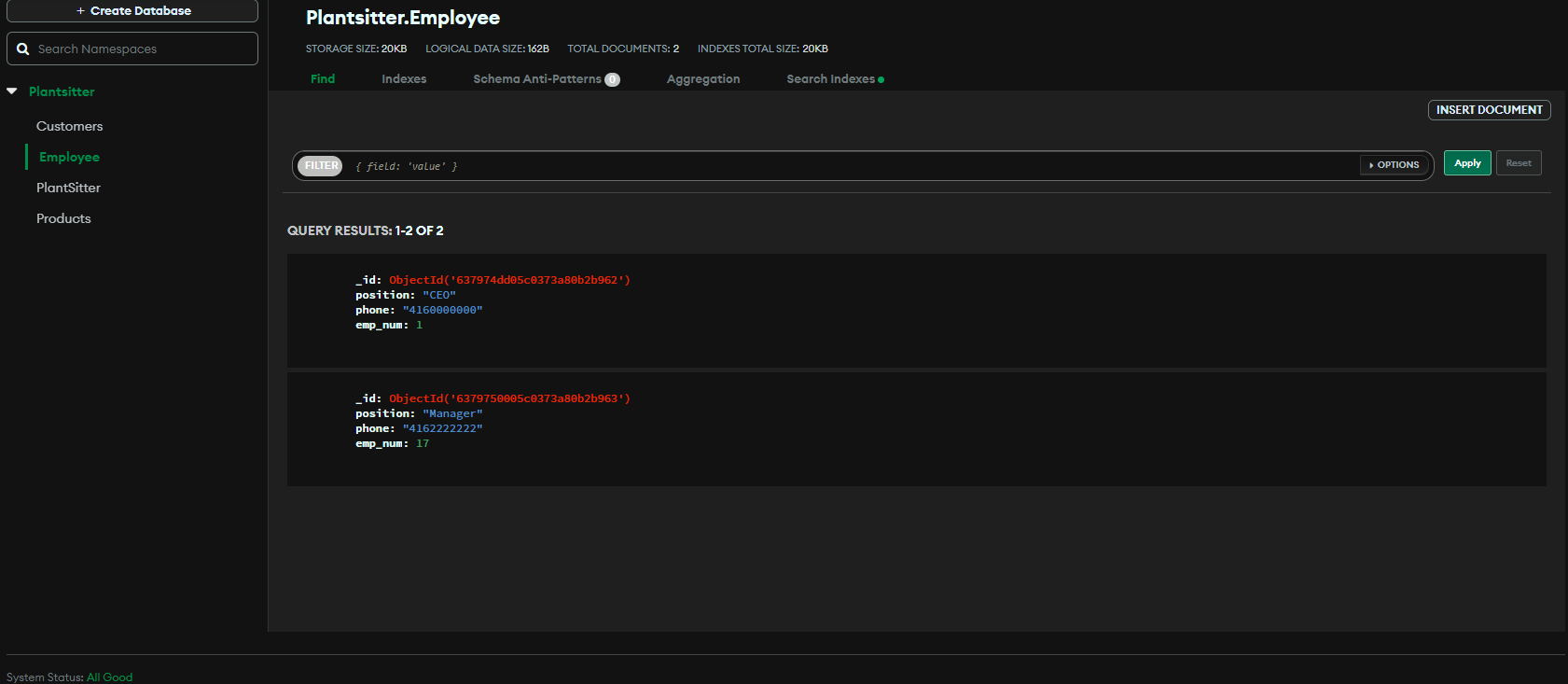
**4- Corresponding Mockups**

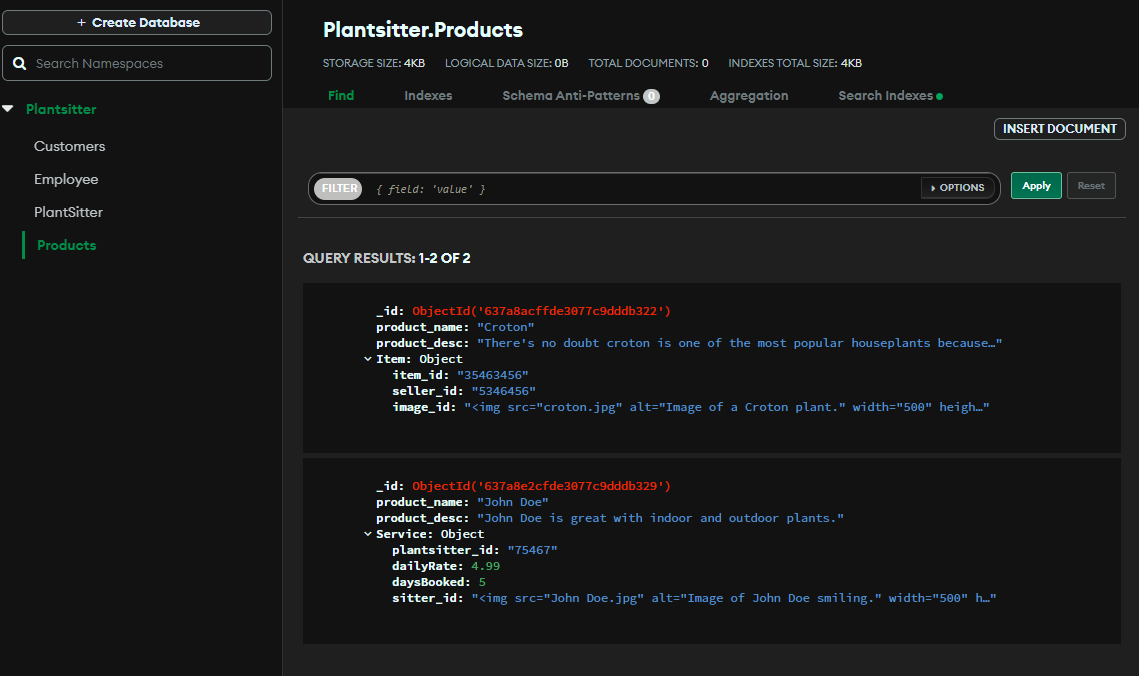
# Domain Class Diagram

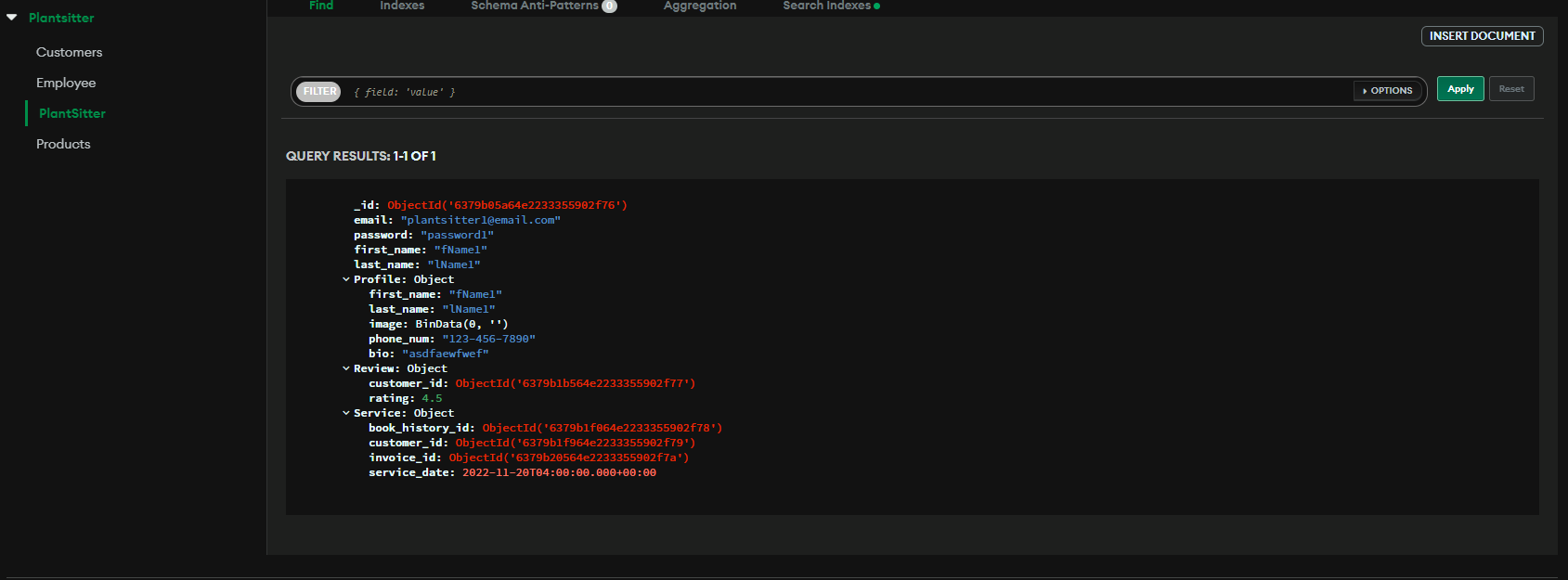


# Database

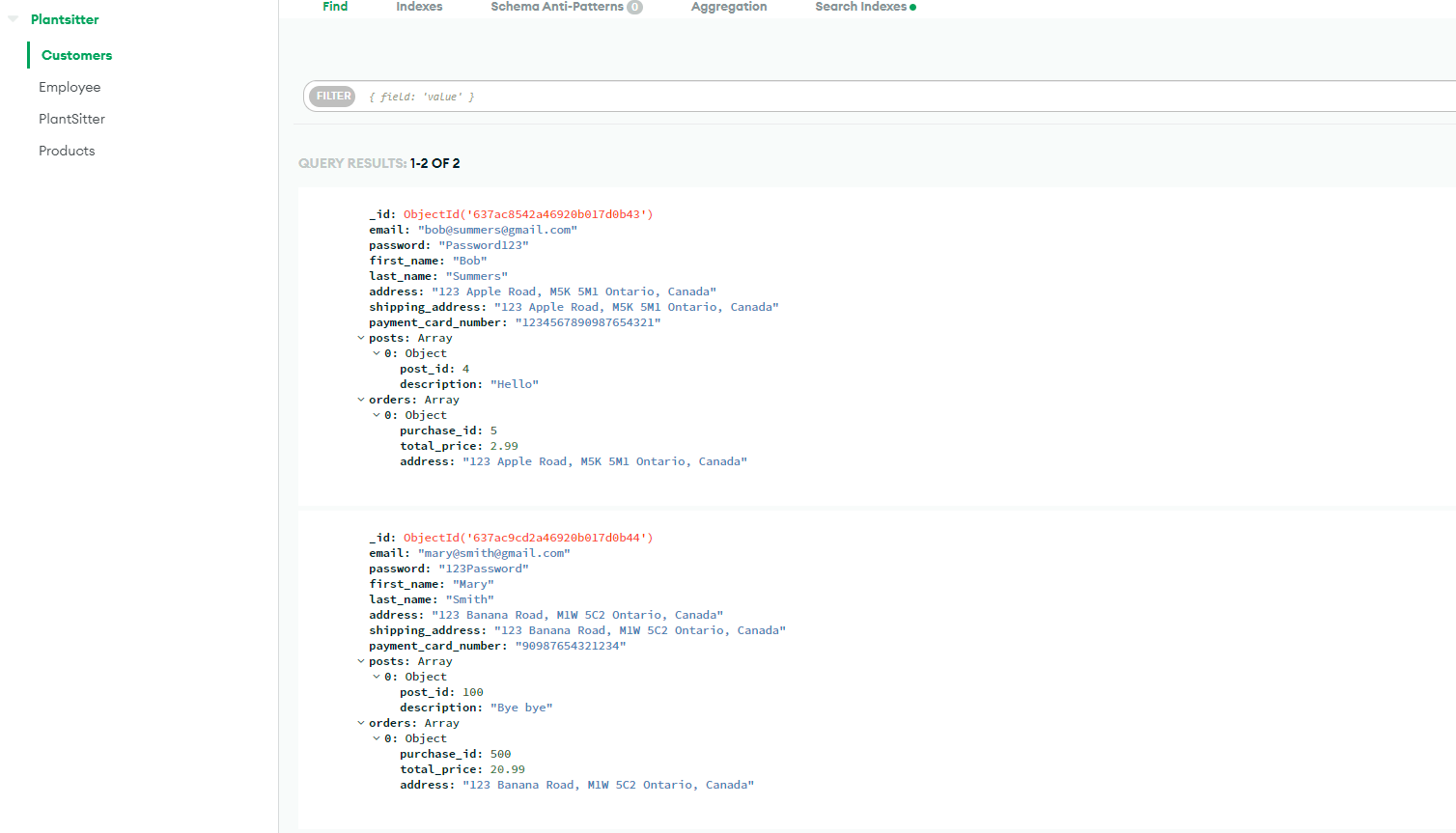
**MongoDB Database**

**Employee**  


**Products**  


**PlantSitter**  


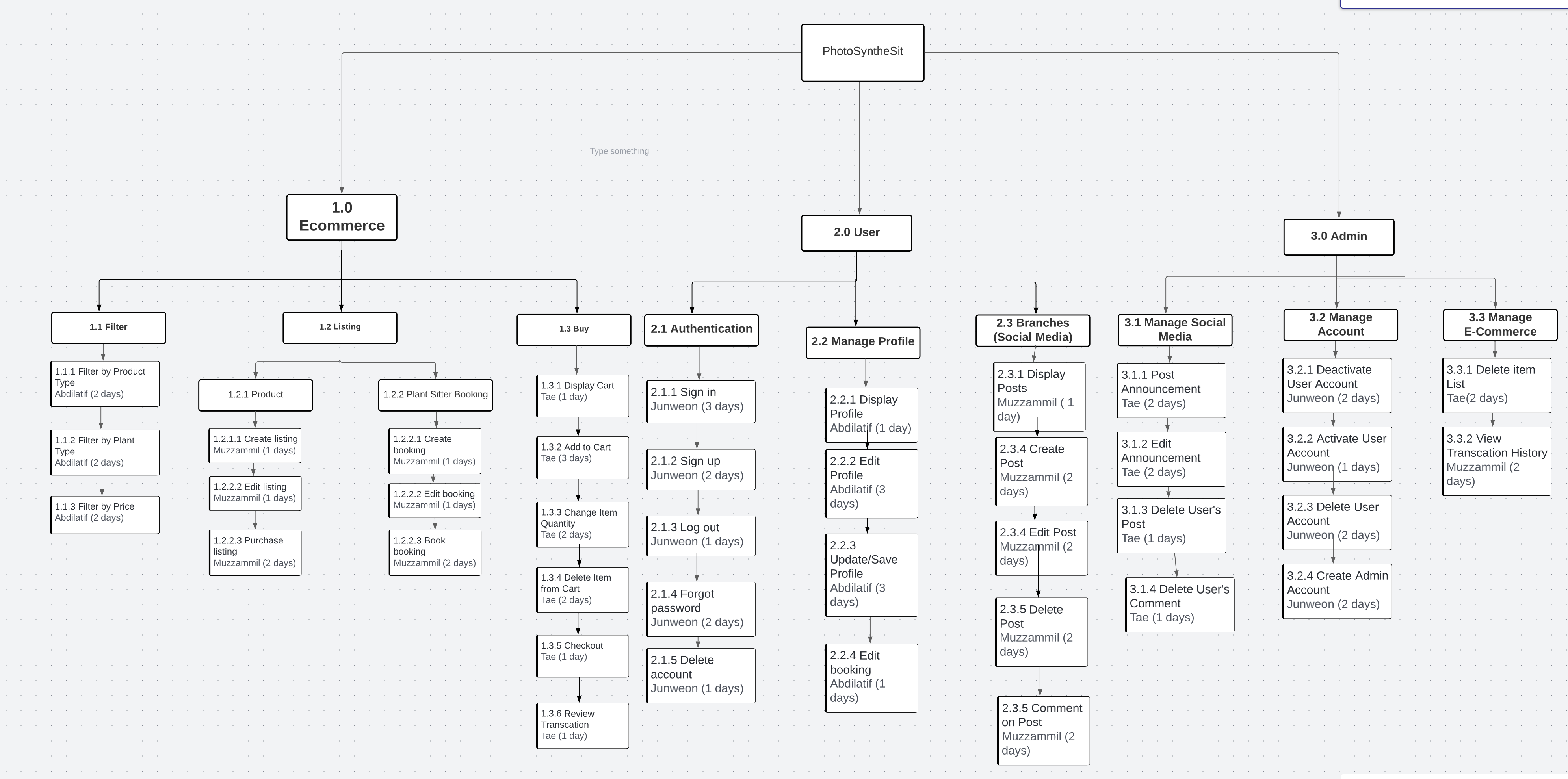
**Customers**



# Work Breakdown Structure (WBS)

## 

## Work Breakdown Structure



# Milestones

1. Basic Ecommerce function implemented (Product, Booking, Shopping cart)  
   - Allow users to filter product by product type, plant type, and price  
   - Allow users to create their own lists to store products and edit then and purchase them  
   - Allow users to create, edit, and confirm booking  
   - Allow users to manage their shopping cart
2. User Setup  
   - Authentication  
   - Profile  
   - Branch
3. Admin Setup  
   - Manage Branch  
   - Manage Users  
   - Manage E-Commerce

# Acceptance Criteria

Milestone1

* Users can filter products with product type, plant type and price
* Users can create listing
* Users can edit listing
* Users can purchase listing
* Users can create booking
* Users can edit booking
* Users can confirm booking
* Users can view shopping cart
* Users can add items to their shopping cart
* Users can change item quantity
* Users can delete item
* Users can make a purchase
* Users can review transaction

Milestone 2

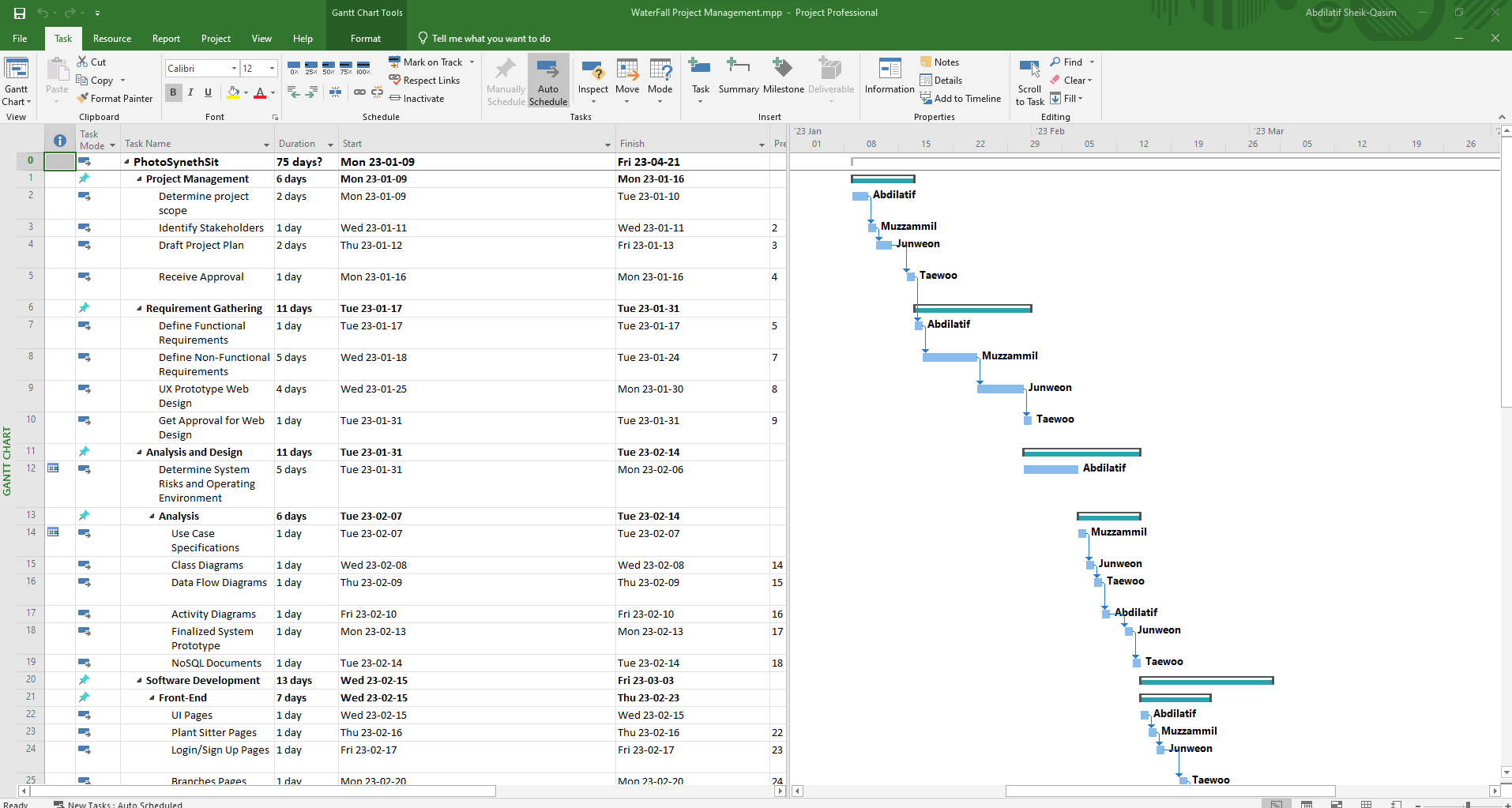
* Users can create accounts
* Users can login to their accounts
* Users can logout of their accounts
* Users can reset their account’s password
* Users can delete their accounts
* Users can see other users’ profile
* Users can edit their profile
* Users can edit booking
* Users can see other users’ post
* Users can create post
* Users can edit post
* Users can delete post
* Users can comment on other users’ post

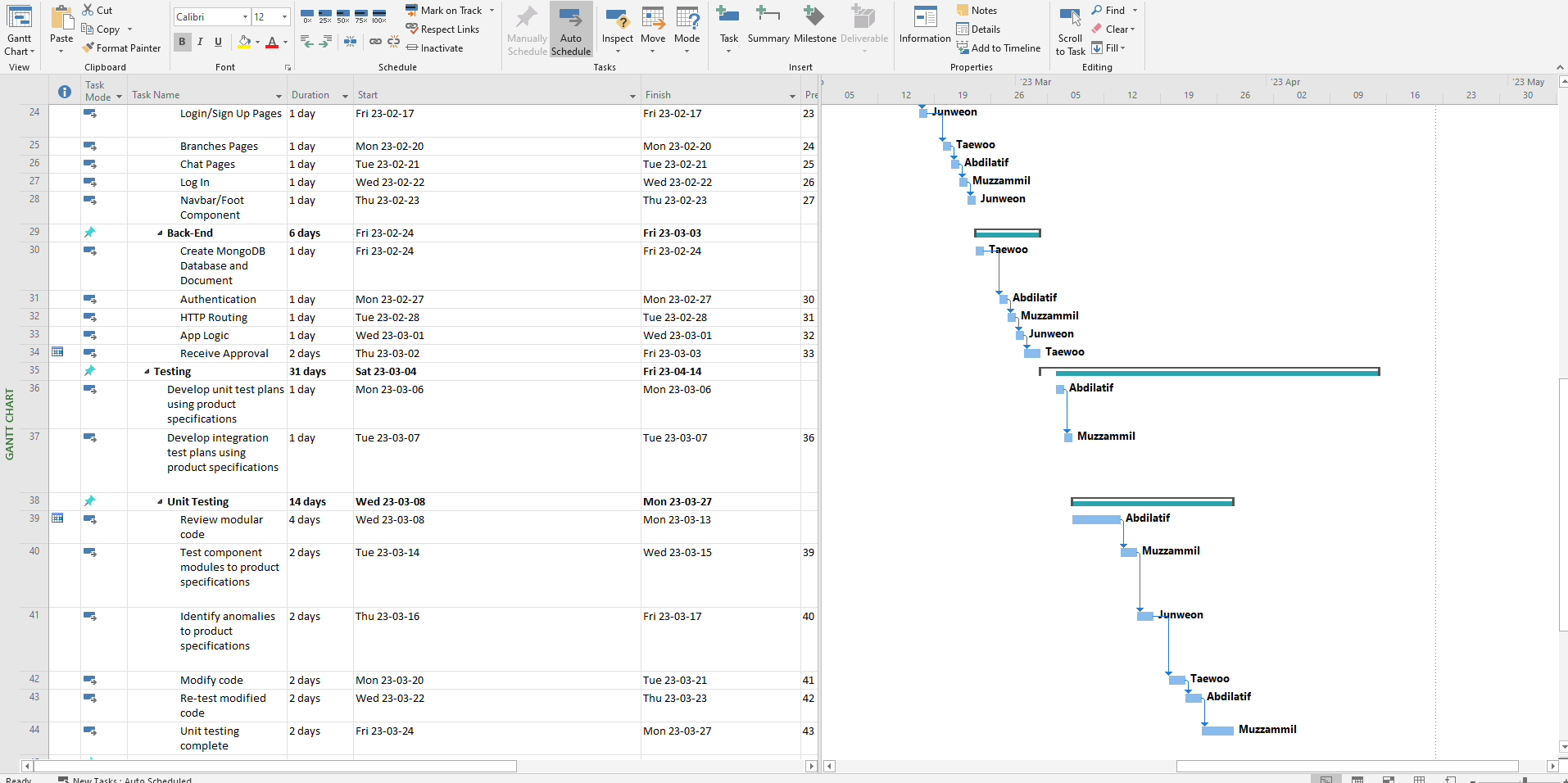
Milestone 3

* Admin can manage branch
  + Admin can post announcements on branch
  + Admin can edit the announcements
  + Admin can delete users’ post
  + Admin can delete users’ comments
* Admin can manage accounts
  + Admin can deactivate users’ accounts
  + Admin can activate users’ accounts
  + Admin can delete users’ accounts
  + Admin can create an admin account
* Admin can manage E-commerce
  + Admin can delete item list
  + Admin can view transaction history

# Implementation Schedule

## Implementation Schedule using MS Project







# Client / Faculty Sign-off

**Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

X .

Name of Client/Rep/Professor

Company Name