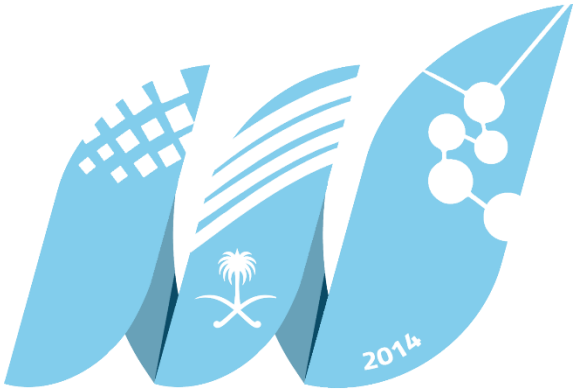



جامعة جدة

University of Jeddah

College of Computer Science and Engineering

Department of Software Engineering

Software Process Models

CCSW 315

## Sprint #0 – Bloom

| | |
|---|---|
| Team Members | 1. Lama Munir Noor \| 2211796<br>2. Lama Fathi Akbar \| 2212609<br>3. Joud Jalal Batarfi \| 2210353<br>4. Raghad Al-saiari \| 2210207<br>5. Ranad Waleed Aljhdali \| 2211806<br>6. Hajar habiburahman \| 2211350 |
| Section | E2 |

# Contents

# Project Initiation

## Project Description

**Bloom** is a desktop application designed to connect plant enthusiasts, sellers, and buyers in a user-friendly platform. The system aims to create a digital marketplace for plants while providing valuable plant care information and management tools. Built using JavaFX and deployable as an executable JAR file, **Bloom** offers a dual-interface system catering to both sellers and customers, facilitating plant sales, care guidance, and personal plant collection management.

## Problem Definition

The plant enthusiast community faces a fragmented marketplace and lacks a unified platform for plant purchases and care management. Novice owners struggle with accessing expert care instructions and maintaining proper care routines, while experienced growers have limited avenues to share their knowledge and sell plants. These disconnects results in inefficient plant care, potential plant health issues, and missed opportunities for knowledge sharing and sales. The absence of a comprehensive solution that combines a plant marketplace with personalized care management tools hinders the growth and success of both plant owners and sellers in this expanding hobby.

## The Solution

**Bloom** addresses these problems through a comprehensive, user-friendly application:

1. A unified platform for plant sellers and buyers
2. A seller interface for uploading plant details and care instructions
3. A customer interface for browsing plants and managing personal collections
4. Integrated plant care management with watering reminders and care guides
5. User-friendly design built with JavaFX
6. Offline accessibility via executable JAR file

This comprehensive solution simplifies plant care, facilitates sales, and connects plant enthusiasts. **Bloom** bridges the gap between sellers and buyers while offering valuable tools for plant care management, promoting successful plant ownership and fostering a community of plant lovers.

## The Scope

The scope of the **Bloom** includes:

1. **Seller interface:** sellers can upload plant pictures, name, type, and care instructions
2. **Customer interface:** enable customers to browse and search for plants to add to their collection, providing options to view all plants in their collection, including the ability to delete plants.
3. **Plant Purchasing:** Users can browse and purchase plants through the app.
4. **Plant delivery:** user can schedule plant delivery.
5. **Watering countdown:** users can set countdown to remind them when to water their plants depending on its instructions.
6. **Notifications:** Customizable reminders for watering, and other plant care tasks, with the ability to enable or disable reminders at any time.

**Out of Scope:**

1. **Plant database integration:** The app does not include features for identifying plants through images or descriptions.
2. **Community features:** No social sharing, forums, or community interaction within the app.
3. **Payment processing:** The app does not support payment processing for the purchase of plants or integrate with other e-commerce systems.

## The Users

1. **Sellers:**

These are users who register with the intent to sell plants. They use the system to:

   a. Upload pictures of plants they want to sell, their name and type.
   b. Set prices and quantity for their plants.
   c. Provide detailed care instructions for each plant (including soil, watering, and fertilization information).

2. **Customers:**

These are users who register with the intent to browse, learn about, and potentially purchase plants. They use the system to:

   a. Browse through plant listings.
   b. Add plants to their personal collection, purchase and get them delivered.
   c. View care instructions for plants.
   d. Set timers and countdowns for watering their plants.

# Product Backlog

## Functional Requirements

1. **User Management:**
   - **1.1.** Users shall be able to register with a username, email and password
   - **1.2.** Users shall select their account type (seller or customer) during registration
   - **1.3.** Sellers must specify their store name.
   - **1.4.** Users shall be able to log in and log out of their accounts

2. **Seller Features:**
   - **2.1.** Sellers shall be able to upload plants picture, name, characteristics, care information, fertilization options, price and quantity.
   - **2.2.** The system shall allow sellers to edit or remove plants they have added to the system
   - **2.3.** The system shall allow sellers to view the purchase history of their plants.
   - **2.4.** The system must inform the seller when a plant's quantity is running lower than 5.

3. **Customer Features:**
   - **3.1.** Customers shall be able to browse plant listings
   - **3.2.** The system shall allow users to search for plants by name.
   - **3.3.** If a plant is not found, the system shall notify the user that the plant is unavailable.
   - **3.4.** Customers shall be able to add plants to their collection
   - **3.5.** Customers shall be able to view care instructions for plants in their collection.
   - **3.6.** Customer shall be able to explore fertilization possibility between two plants of their collection.
   - **3.7.** The system shall allow users to delete plants from their collection.

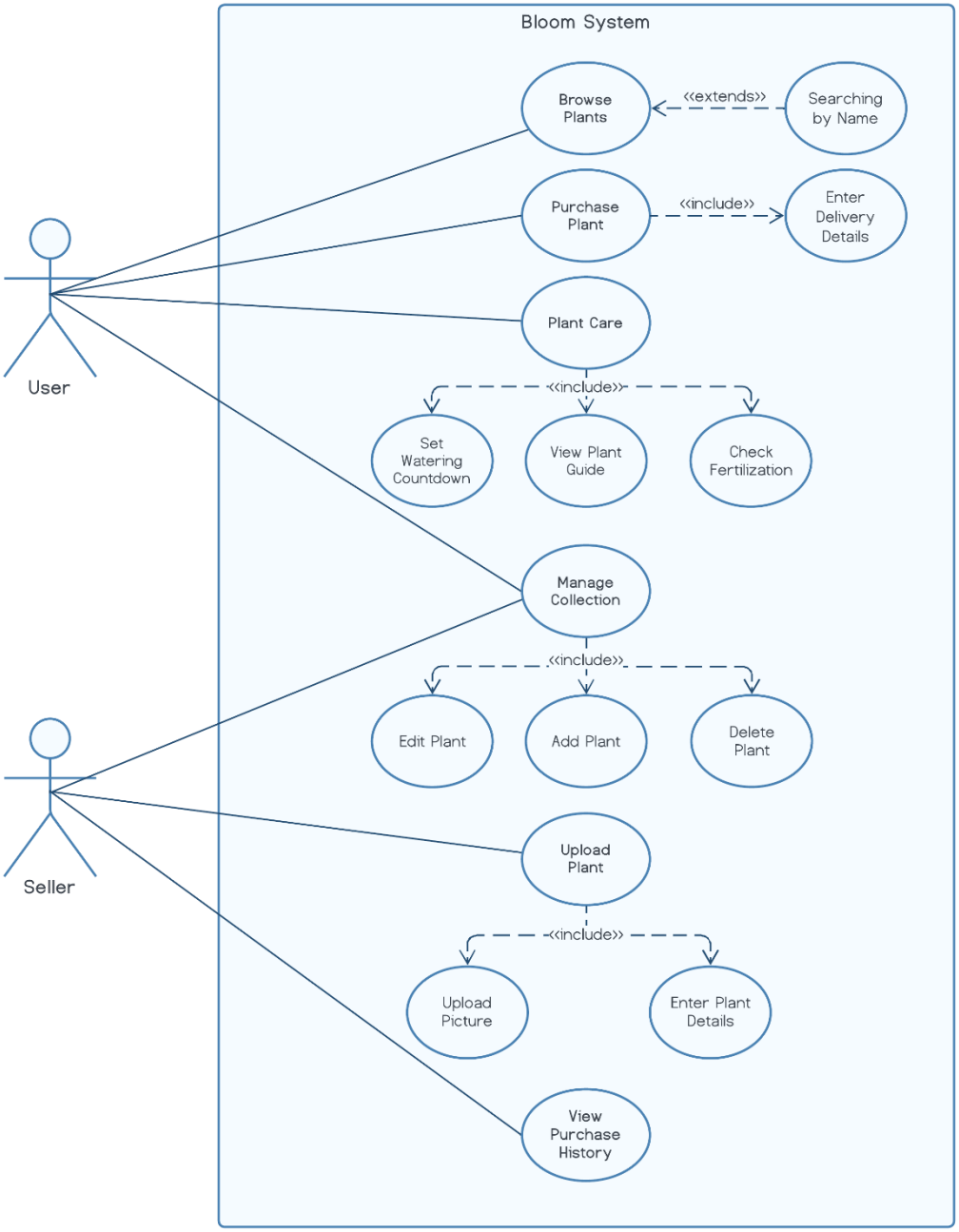
4. **Timer Functionality:**
   - **4.1.** The system shall allow setting of watering timers for each plant
   - **4.2.** The system shall provide notifications when watering is due

## Non-Functional Requirements:

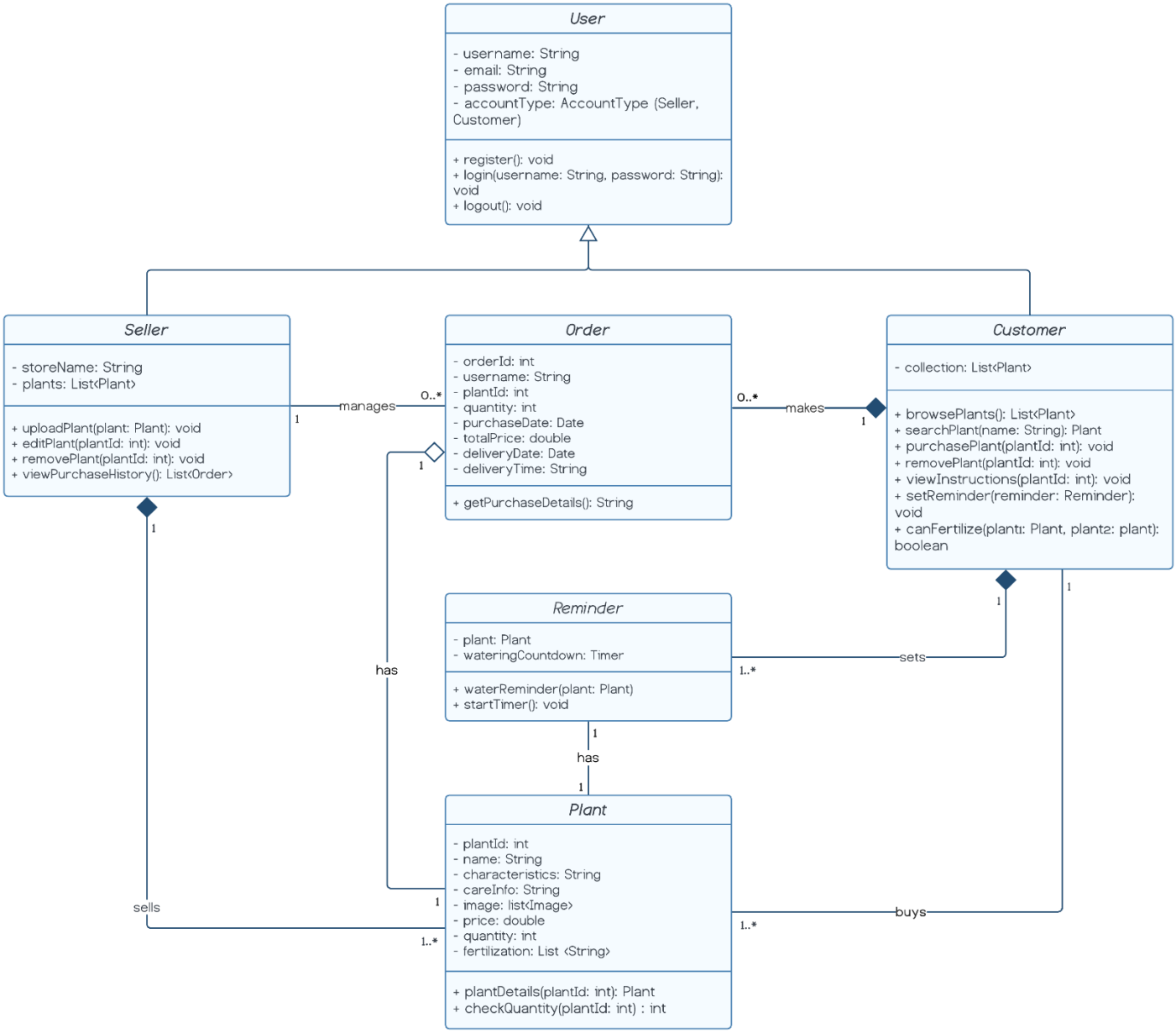
1. **Performance:**
   - **1.1.** The application shall load plant listings within 3 seconds on a standard broadband connection
2. **Usability:**
   - **2.1.** The user interface shall be intuitive and require no more than 5 minutes of learning for basic operations
3. **Reliability:**
   - **3.1.** The system shall have an uptime of 99.9% excluding scheduled maintenance
4. **Security:**
   - **4.1.** The system shall enforce strong password policies to enhance user account security.
5. **Compatibility:**
   - **5.1.** The application shall run on Windows, macOS, and Linux operating systems with Java Runtime Environment 8 or higher
6. **Maintainability:**
   - **6.1.** The codebase shall follow object-oriented design principles and be well-documented
7. **Storage:**
   - **7.1.** The application shall not exceed 100MB in size for the executable JAR file
8. **Response Time:**
   - **8.1.** User interactions (e.g., adding a plant to collection) shall have a response time of less than 1 second

# System modelling

## Use Case Diagram



## Class Diagram



# Tools

## Integrated Development Enviroment (IDE)

**Visual Studio Code, Netbeans, and Eclipse**



## Agile Planning Tool

**Jira**

## Modeling App

**Lucidchart**

## Git Tools

**GitBash and Github**


## Messaging App

**Slack**