Software Engineering
Software Requirements Specification (SRS) Document

Plant-Hydrate

09/20/2022

0.1

By: Kayla Abreu, Laura Love, Aimade Yacouba

**I HAVE ABIDED BY THE UNCG *Academic Integrity Policy*
ON THIS ASSIGNMENT**

Ay, LL, KA

Table of Contents

1. Introduction

1.1 Purpose:

> The goal of our project is to help users keep their plants alive and healthy by sending them reminders on the specific days that their plants need to be watered.

1.2 Document conventions:

> The purpose of this Software Requirements Document (SRD) is to illustrate the requirements for the Plant-Hydrate (P-H20) system. In it, we will clarify the requirements from both the client and developers sides. The requirements from the client side will describe the types of user that will use the system and how they will interact with the user interface. From the software developer's side, the SRD will elaborate on the necessary software and hardware requirements needed to construct, use, and maintain the system at optimal levels.

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition. Acronym, Abbreviation
CSS	Cascading Style Sheets (CSS) works alongside HTML to format content on web browsers.
HTML	The HyperText Markup Language (HTML) is used to display content on web browsers.
Java	A popular object-oriented programming language that we will use to construct our application.
MySQL	An open-source database management system.
P-H20	An abbreviation for the Plant-Hydrate system.
Spring	A framework that makes the creation of Java applications more efficient.

1.4 Intended audience:

The entire SRD is intended to assist the software development team and the instructor of the CSC 340 class. Users of the application may benefit from reading the general description and the operating system and compatibility sections to see if they are interested in using the application and to confirm the application will run on their personal systems.

1.5 Project Scope:

The completion of the P-H20 system coincides with the goals of the CSC 340 class since it is a requirement of the class.

1.6 Technology Challenges:

1.7 References:

2. General Description

2.1 Product perspective:
The context and origin of the product is a group project that consists of three members. Members are to invent an application that saves data in a persistent format and must incorporate a third party API for data retrieval.

2.2 Product features:
When a user logs in, they will see a dashboard where they can add new plants. Then they will be able to schedule a reminder for that specific plant to be watered.

2.3 User class and characteristics:
Our web application will expect users to be able to access a web browser and to be self-educated on how often their plants need to be watered to set an accurate reminder time.

2.4 Operating environment:
The software is to be operated as a web app, running on traditional computers and phones.

2.5 Constraints:
- Use of specific 3rd party APIs such as the 'google calendar api'

2.6 Assumptions and dependencies:
- Possible use of reactJS
- Possible use of Bootstrap
- Possible use of a Google calendar api

3. Functional Requirements

3.1 Primary
- FR0: The system will allow the user or administrator to login into their own account to set up their own personalized water planting system. (LL)
- FR1: The system will allow the user to enter their own specific plant into a plant-storage database. (KA)
- FR2: The system will allow the user to determine how frequently they want to be reminded and how, e.g. email. (Ay)
- FR3: The system will allow the user to change remindement times.(Ay)
- FR4: The system will allow the user to add more plants or remove them. (KA)
- FR5: The system will hold the users plant information.(LL)

3.2 Secondary:

Each member will be responsible for two functional requirements.

4. Technical Requirements

4.1 Operating System & Compatibility
4.2 Interface requirements
4.2.1 User Interfaces

First screen would be the login page with a create account link to sign up, if not a user. Then the next screen would take the user to the dashboard to add a new plant. Then the user can set up a reminder time to water plants.

4.2.2 Hardware Interfaces

Any device that supports JVM and can connect to a web browser will be able to run the P-H20 system. The system will use HTTP protocols.

4.2.3 Communications Interfaces

HTTPS will be used for the communication standards by the software as part of the project.

4.2.4 Software Interfaces

HTML/CSS will be used for the frontend to create a stylized user interface. Java will be used for the backend development with a Spring framework and Maven dependencies to make the application function. MySQL will be used for the database management system to maintain records of user logins, plant lists, and reminder schedules.

5. Non-Functional Requirements

5.1 Performance requirements

- NFR1(R): The beginner user will be able to set up their own account in less than a minute.
- NFR2(R): The beginner user will be able to set up their own reminder and add their plant in less than 5 minutes.
- NFR3(R): The novice user will be able to set up their own reminder and add their plant in less than 2 minutes.

5.2 Safety requirements

- Safe storage of passwords

5.3 Security requirements

- NFR4(R): Login credentials from user.

5.4 Software quality attributes

5.4.1. Availability
5.4.2. Correctness
5.4.3. Maintainability
5.4.4. Reusability
5.4.5. Portability

5.5 Process Requirements

5.5.1. Development Process Used
We will use the incremental development process for the P-H20 System.
5.5.2. Time Constraints

A prototype will need to be prepared by 10/20/2022 and a functional version of the application will need to be presentable by 11/22/2022.
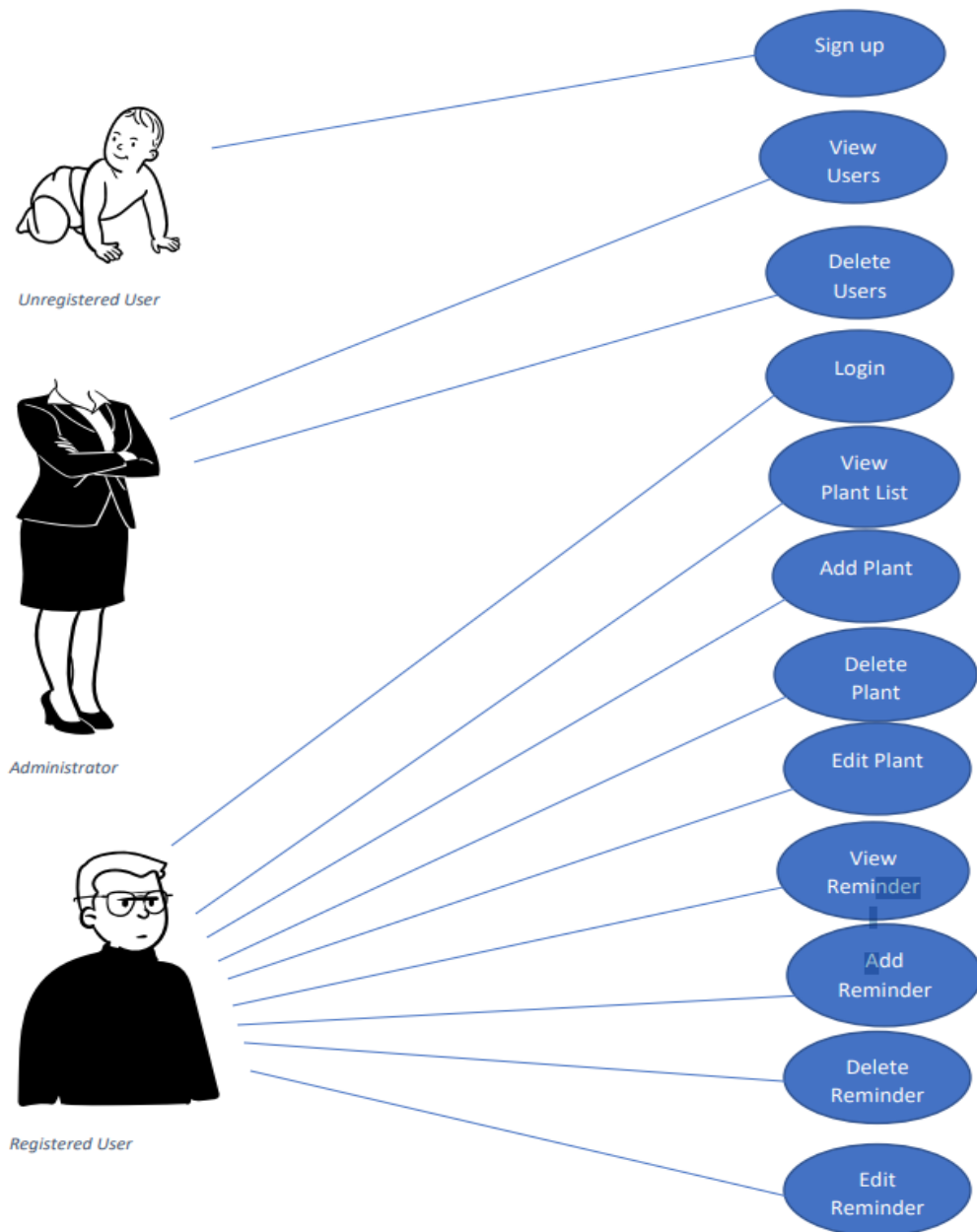
5.5.3.   Cost and Delivery Date
There should be no additional costs in the development of the P-H20 system and it will be ready by 11/22/2022.

5.6 Other requirements

6. Appendix

6.1 Use-Case Model

6.1.1 Use-Case Diagram

6.1.2 Use-Case Description

- **Signup (Laura)**
  - **An unregistered user can sign up with a username and password.**
- **View Users (Laura)**
  - **Administrators can view all registered users.**
- **Delete Users (Laura)**
  - **Administrators can delete all registered users.**
- **Login (Laura)**

- **Registered users and administrators can login. Administrators will have the username "ADMIN" with a unique password of their choice. Registered users can login with the username and password they signed up with.**
  - **View Plant List (Kayla)**
    - **Registered users can view their plant list.**
  - **Add Plant (Kayla)**
    - **Registered users can add plants to their plant list.**
  - **Delete Plant (Kayla)**
    - **Registered users can delete plants from their plant list.**
  - **Edit Plant (Kayla)**
    - **Registered users can name or rename their plants.**
  - **View Reminders(Ay)**
    - **Registered users can view which plants have a reminder to get watered at a certain time.**
  - **Add Reminder(Ay)**
    - **Registered users can add a reminder to their plants to get watered at a certain time.**
  - **Delete Reminder(Ay)**
    - **Registered users can delete reminders that are set to their plants.**
  - **Edit Reminder(Ay)**
    - **Registered users can edit their reminder times.**

    6.1.3   Use-Case Scenarios

Signup

    i. Initial Assumption: A new user will not have a registered account with the Plant-Hydrate system.

    ii. Normal: The new user will create a unique username and password.

    iii. What can go wrong: The new user will try to sign up with a username that is already in use or will enter a username or password that is invalid.

    iv. Other activities: If the new user is an administrator, their information will be saved to a different section.

    v. System state on completion: The user will be registered with the system. Their username and password will be entered into the system's database. They will be able to successfully login to the system.

View Users

    i. Initial Assumption: The user is logged in as an administrator.

    ii. Normal: The administrator will be able to click on the 'View Users' button and retrieve a list of current users.

    iii. What can go wrong: The administrator will not be able to view the list of users or the list of users will be incomplete.

    iv. Other activities:

    v. System state on completion: The administrator will be able to view the list of users and be able to select a specific user.

Delete Users

    i. Initial Assumption: The user is logged in as an administrator and is able to view the list of users and select a specific one.

    ii. Normal: The administrator will be able to delete the account of the selected user.

    iii. What can go wrong: The administrator will not be able to delete the account of the selected user or may delete the account of the wrong user.

iv. Other activities: A popup will be displayed for the administrator to confirm they are deleting the correct account.
v. System state on completion: The selected user's account will be deleted.

Login

    i. Initial Assumption: The user or administrator is already registered with the system. The correct login credentials are saved in the system.
    ii. Normal: The user or administrator will enter their username and password to login to their account.
    iii. What can go wrong: The user or administrator's login credentials are not accepted because either the username or password does not match the credentials in the database. The user or administrator should be able to request a new password.
    iv. Other activities: The user or administrator can reset their password or username using the forgotten password/username link.
    v. System state on completion: The user or administrator will be logged in. They can view their separate dashboards. The users will be able to view their plants and reminders and administrators can view the list of users.

View Plant List

    i. Initial Assumption: The user is logged into their account and can see their dashboard.
    ii. Normal: The user will click on the 'View Plant List' button to view the plants they want to add reminders for.
    iii. What can go wrong: The user will not be able to view their plant list or the list may not show all of the correct plants.
    iv. Other activities: The user will be able to add new plants or edit incorrect plants.
    v. System state on completion: The user will be able to see every plant that they have added.

Add Plant

    i. Initial Assumption: The user is logged into their account and clicked on the 'View Plant List" button.
    ii. Normal: The user will click on the 'Add Plant' button to add a new plant to their plant list.
    iii. What can go wrong: The new plant will not be saved to the plant list. Plant must have a name before being added to the plant list.
    iv. Other activities:
    v. System state on completion: The user will be able to see the new plant added to their plant list.

Delete Plant

    i. Initial Assumption: The user is logged into their account and clicked on the 'View Plant List' button.
    ii. Normal: The user will click on the 'x' next to the plant to delete the plant from their plant list.
    iii. What can go wrong: There are no plants to delete.
    iv. Other activities:
    v. System state on completion: The plant is deleted from their plant list.

Edit Plant

    i. Initial Assumption: The user is logged into their account and can edit the plant they added, e.g. change name.
    ii. Normal: The user will click on 'Edit Plant' and change the plant they previously added.
    iii. What can go wrong: The change doesn't go through.
    iv. Other activities:
    v. System state on completion: The user will be able to edit and make changes to their added plant.

View Reminders
    i. Initial Assumption: The user is logged into their account and can see their dashboard.
    ii. Normal: The user will click on the 'View Reminders' button to view the reminders they have scheduled.
    iii. What can go wrong: The user will not be able to view their reminder list or the list may not show all of the correct reminders.
    iv. Other activities: The user will be able to add new reminders or edit incorrect reminders.
    v. System state on completion: The user will be able to see every reminder that they have scheduled.

Add Reminder
    i. Initial Assumption: The user is logged into their account and can see their dashboard.
    ii. Normal: The user will click on the 'Add Reminder' button. They will be asked what plant they want to set a reminder for. Then they will be asked how often they want to be reminded.
    iii. What can go wrong: The reminder will not be set or the reminder will be set for the wrong plant or for the wrong time.
    iv. Other activities: The user will be able to edit an incorrect reminder.
    v. System state on completion: A reminder will be set for a specific plant.

Delete Reminder
    i. Initial Assumption: The user is logged into their account and can see their dashboard.
    ii. Normal: The user will be able to select a reminder and cancel it.
    iii. What can go wrong: The reminder will still be set.
    iv. Other activities:
    v. System state on completion: The reminder will be removed.

Edit Reminder
    i. Initial Assumption: The user is logged into their account and can see their dashboard.
    ii. Normal: The user will be able to select a reminder and change the frequency of the reminder.
    iii. What can go wrong: The reminder will remain unchanged.
    iv. Other activities:
    v. System state on completion: The reminder will be updated.