SOFTWARE REQUIREMENTS SPECIFICATIONS

PLANT CARE SYSTEM

Group Members:
BT19CSE009 Ruchika Pandharikar
BT19CSE047 Neha Kalbande
BT19CSE056 Hemanshu Chaudhari
BT19CSE077 Diwyani Kemekar
BT19CSE095 Ekta Singh

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the PlantCare based on the images . It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for the developers of the system.

1.2 Intended Audience:

This application is useful for all the people who are passionate about keeping houseplants. This document is intended any age group people

1.3 Intended Use

Users can search for the detailed info of the plant by using an image from storage or click a live image. The details users will be able to view are Common Name, Scientific Name, Family Name, category, maxGrowth, origin, poisonous, images , description, temperature, light, watering, soil,etc.

1.4 Scope

The main purpose of our application is to help plant parents take care of their plants by understanding the unique qualities and requirements (e.g nutrition) of each plant. The app allows you to look up numerous types of houseplants, add plants to your garden, set up reminders with notifications such as water, fertilizer, and repotting, personalize your plant by renaming it and adding a picture of it, as well as browse through various interesting articles about your house plants.

1.5 Overview

The overall description part of this document gives the description of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements. The Requirements Specification section of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.
Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2. Overall Description

Product Perspective

The perspective of our application is to help plant parents take care of their plants by understanding the unique qualities and requirements (e.g nutrition) of each plant.

Product Functions

2.1 User Needs:

Having house plants is increasingly becoming popular especially among the millennials.

However a lot of these users have little or no knowledge about plants and their requirements like recognising a particular plant or knowing the right nutrition for their houseplants. PlantCare solves this by providing relevant information about these plants to the users.

2.2 Assumptions and Dependencies

The app can only identify and suggest info for the plants that are a part of its database.

2.3 Operating Environment

- Operating System: Android (Cell phone)
- Database: Firebase Firestore
- Hardware Platform: Cell Phone (Android)

2.4 Design and Implementation Constraints

Constraints includes:

- Class diagram and its implementation.
- Automatic SQL commands calling with the help of the data provided.
- Implementing the database using the centralized database management system.

3. System Features and Requirements

3.1 Functional Requirements

- The app provides a provision to users to add the plants they have planted in the garden.
- The user can click the picture of their plant and the app will identify the plant and provide the description.
- The user can also view a detailed description of the plant including their family, kingdom, if the plant is poisonous or not and other details. The favorable temperature for their growth, the nutrients and the amount of water required for the proper growth of the plant.
- The user can delete the plant added.
- Also user can search for the plant

3.2 External Interface Requirements

User Interfaces: Mobile App (Android): React Native, Redux, Expo, Firebase.

Hardware Interfaces: Android Smartphone with camera

Software Interfaces:

- In mobile users we have picked Android as the primary OS.
- Firebase will be used to interact with the centralized server.
- In order to save the plants info Firebase Firestore database will be used.
- Internet connectivity is required.

Communications Interfaces

This will support id verification using Gmail, all kinds of browser and most likely all updates of Android devices.

3.3 Non Functional Requirements

(i) Secure access to consumer's confidential data.
(ii) 24X7 availability.
(iii) Accessible to any number of users at any time.
(iv) Various other Non-Functional Requirements are: Security , Reliability , Maintainability, Portability, Extensibility , Reusability, Compatibility Resource Utilization

Performance Requirements

Response time: less than 1.00 second.
Workload: 1000 users per second.

Safety Requirements

Since this platform is authenticated with the Firebase, we won't have to deal with safety issues much often. But to avoid unnecessary risks, and potential crises, we will generate Token Verification using a private key.

Security Requirements

Users will be able to sign up with their Google accounts. This will verify them as legit users. Since user privacy is concerned, users' confidential information will not be visible to other users. Error messages will be popped up, when invalid inputs are given in order to inform the user that the input provided is not accepted.

Software Quality Attributes

- Accurate and precise processes must be performed by the system to avoid problems.
- The system must be flexible. Should be easily modified.
- Performance: Execution process must be fast and responsive.
- The system should possess maintainability, should be able to cope up with any encountered problems. UML will be used in the development process.
- The UI of the system will be user friendly.
- Ensuring portability. The software is an app-based application and is built in React Native and Firebase. So it is platform independent.
- Security: The system contains authentication thus providing users the needed security for their respective accounts.
- Availability: The system should be always available with very negligible downtime.

Preliminary Schedule

1st increment: 1 month.

Business Rules

We can create partnerships with different users and encourage people to take care of the nearby plants. If our app becomes popular our earth will not face global warming issues.

CODE LINK: https://github.com/nehakalbande/PlantCare-Management