JnJ’s Clockwork

Software and Hardware Specification Sheet

Juan Rodriguez, Jordan Pulido, Johnson Dinh

January 24, 2019

Declaration of Sole Ownership

Proposal

Abstract

Table of Contents

Illustrations

Introduction

Technical Problem

Scope of Report and Procedures Used

The system is based on a database where the application goes hand in hand with the clockwork itself. A database will be established by us supporting the application through reading and writing temperature values and vibration features.

Purpose of Project/Objective

The purpose of the notification device is to alarm the user more effectively and to create a portable and ease of access application for consumers who are trying to get notified on time. All in all, the project is to provide an effective experience when receiving notifications from the device.

Problems Encountered

With a snooze button in the application, this can be abused by the user where the clockwork would insufficiently notify the user. Customers can also turn off their bluetooth provided in their mobile devices, which would ultimately disable the system.

Unique Approaches

Project Description

Requirement Specifications

Hardware

The hardware portion of this project will be a joint effort between each member of the group as there are many responsibilities in order for everything to function as intended. In terms of the hardware

design and enclosure, it will be handled by Juan. The functionality of each sensor will be tested and operational mainly from Jordan with help from Johnson when required. Connections between sensors and the Raspberry Pi will be accomplished by Johnson. The Integration of components may require additional help from every member due to problems that may occur during development.

The project utilizes many hardware components such as the Raspberry Pi 3B+, HTU21D-F Humidity/Temp Sensor (0x40), and a Display screen. These sensors will be inclosed using 3D printed materials with a maximum dimension size of: 12 13/16” x 6” x 2 7/8 ”. The Humidity and Haptic sensor are ready for integration with one another as they were already completed last semester. The new inclusion for the project is the Display Screen as it is crucial to display the time and temperature from the application. The android smartphone’s role will act as the device’s remote as it can communicate with the clockwork through bluetooth. A 8GB micro SD card will be used as storage as it can store the installation of the Raspbian OS, and reading and writing values from the Clockwork. The PCB (printed circuit board) acts as the structure and support of the system for sensor connections.

Software

The android application will be developed and maintained by Juan and Johnson. Add ons and additional functionality will be incorporated with the help from Jordan. The app is mostly complete in its current state. The only things that are left to work on is bluetooth functionality and debugging. The app needs to respond to the hardware in order to display desired information from the application.

The project utilizes a smartphone capable of running Android API 21 or higher. An up to date version of Android Studio was used to build the mobile application. A Raspberry Pi 3 was implemented with connection between the hardware and application. Updating the Raspbian OS to its newest version was used throughout the project. The mobile application will be used to work alongside the hardware components. Firebase real-time readings is going to be used for communication such as storing user and temperature readings.

Database

The database will be designed, created and upheld by Juan and Johnson. The database connection is established and connected to the mobile application. Reading and writing from the sensor to the database are also required. The database utilizes user-authentication to allow maximum security and protection for the users information. In order to read and write temperature, the user must be registered using a username and password through authentication processing. Offline mode allows access to the app, without the need to register and login. Offline mode skips user-authentication, and moves the user to the actual app. In offline mode, there will be no form of communications to the database. Therefore the user is unable to read/write temperature to the database.

Background

Problem

Solution

Conclusion

Recommendations

Bibliography