Garvis Pool Repair App

Software Requirements Specification

CEN-3073, Spring, 2018

*Modification History:*

|  |  |  |  |
| --- | --- | --- | --- |
| *Version:* | *Date:* | *Who:* | *Comment:* |
| V1.0 | 03/05/2018 | Objective Coders | Has yet to be released. |
| V1.1 | 03/30/2018 | Objective Coders | Updated Team name, reworked use cases, updated specific requirements to meet criteria. |
| V1.2 | 04/16/2018 | Objective Coders | Rewrote all use cases, modified requirements and introduction to reflect final software product. |

**Team Name:** Tuesday One

**Team Members:**

|  |  |
| --- | --- |
| Team Members: | E-mail: |
| Jeffrey Fleurent | jeffreyfleurent@gmail.com |
| Juan Gomez | juanangelgomez@live.com |
| Edgar Meruvia Garron | emeruvia@gmail.com |
| David Murad | dsmurad1735@eagle.fgcu.edu |
| Haley Ovenhouse | hovenhouse7675@eagle.fgcu.edu |

**Contents of this document:**

1. Introduction
   * Software to be Produced
   * Applicable Standards
   * Definitions, Acronyms, and Abbreviations
2. Product Overview
   * Assumptions
   * Stakeholders
   * Use Case Diagrams
   * Use Case Descriptions
3. Specific Requirements
4. Additional Requirements

**SECTION 1: Introduction**

Software to be produced:

The Garvis Repair application is a mobile application which allows employees, managers, and owner of Garvis Pool Inc. to better communicate within their company. The application manages employee schedules by creating task called “work orders” that are assigned a date, time, client, and job to be completed. The application also keeps log of all the company’s clients, their addresses, and email addresses. All of this information is available to any employee through the use of a database, FireBase. FireBase allows the user to create, read, update, and delete work orders and client information stored in the database.

Applicable Standards:

We will keep customer personal information confidential. We will use certain information for data analysis, but that will only be available to the owners.

Definitions, Acronyms, and Abbreviations:

* noSQL – A non-relational database structure.
* FireBase – A noSQL database used to store data that is accessed and used by the application.
* Work order – A single job that is to be completed for a specific client at a designated date and time. Examples being clean filter, detect and repair leak, install pump, etc.
* API – Application Programming Interface.

**SECTION 2: Product Overview**

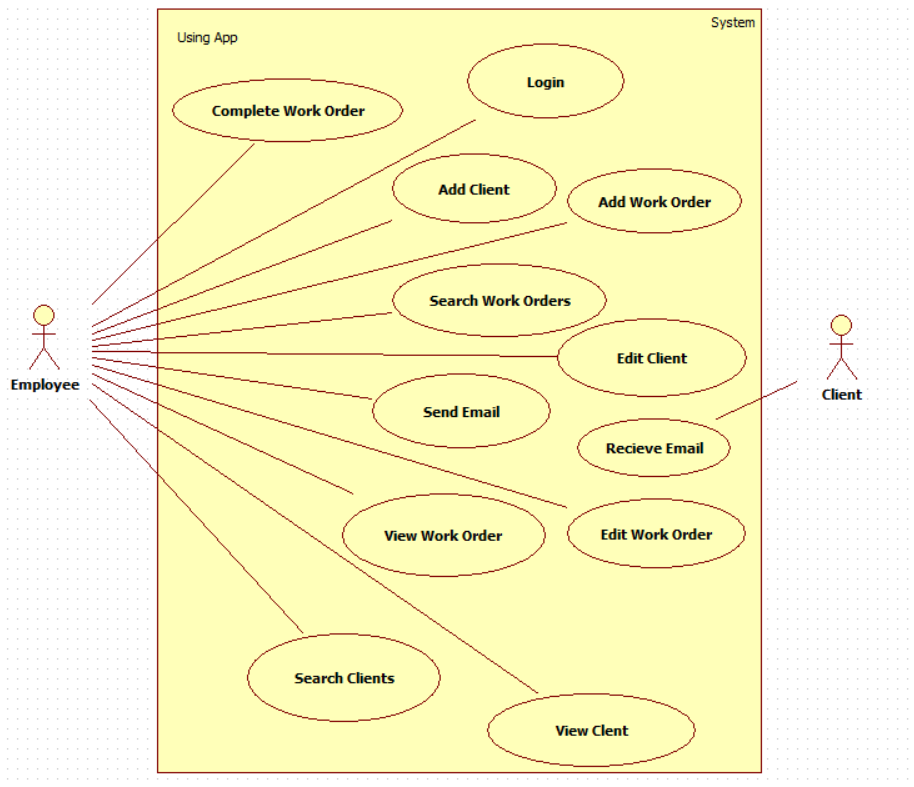
Assumptions:

1. The user will be running on a mobile device with and Android SDK minimum of 24.
2. The mobile device will have access to the internet to connect to FireBase, either through cellular service or Wi-Fi.

Stakeholders:

* Owner: Having access to all company data, schedules and client information in real time will increase productivity and allow for company to easier expand in the future.
* Employee: With an application to track their schedule, given them instant access to all client information and past work orders for said client, and directions to client addresses, employees will able to perform their jobs easier.
* Client: When a work order is scheduled, the client will receive an email notification confirming their appointment. More communication between company and client creates a better customer experience.

Use Case Diagram:



Use Case Descriptions

* Complete Work Order: Employee creates a work order when a client needs a job done. To do this the employee selects “Create A Work Order”, selects a client from the client registry, assigns the work order a specific date and time, selects the category of work to be completed, and clicks “Submit” to complete the work order.
* Login:

1. Description: When the system launches, the employee will enter an email and password into the respective fields. If the email and password match those in the database upon the employee pressing the login button, the employee will be granted access to the home screen of the application.
2. Actors:
   1. Employee.
3. Preconditions: The employee’s device has connection to the Internet and the application is able to successfully connect to the database.
4. Basic Flow of Events:
   1. Employee opens application.
   2. Employee enters their email in the email textfield and their password in the password textfield.
   3. Employee then selects the login button.
   4. The application checks the email and password against those in the database.
   5. With matching credentials, the application redirects the employee to the home screen of the application and they are logged-in.
5. Alternative Flows:
   1. When employee launches application and they do not yet have an account, they can choose to create an account by selecting the sign-up button.
   2. After employee has entered their email and password and selected the login button, the application must connect to the database to check the entered credentials against those in the database. If the connection to the database cannot be made, the credentials cannot be verified, and the employee will not be granted access to the application.
   3. An employee enters an email and password and upon selecting the login button they are denied access because the entered credentials do not match those in the database. They will remain on the login screen.
6. Exit Conditions:
   1. After employee has successfully logged-in, they are redirected to the main screen of the application where they can perform all functions.
   2. Failure to complete login process leaves employees at the login screen.

* Add Client: To add a client to the client registry, select the client tab from the main screen.` Then select create new client. Enter the clients first name, last name, email address, and home address. Once all fields are filled in the correct format, the employee selects create client. The client and their information are then added to the database.
* Add Work Order: To create a work order and add it to the schedule, the employee will select the work orders tab from the main menu, the select the blue plus button in the bottom right corner to create a new work order. The employee will select a client that the work order is for, selects what work is to be done from a drop-down menu, adds any additional notes needed, and finally assigns the work order to a date and time. Once the work order is submitted it is assigned a “work order number” and added to the database.
* Search Work Orders: To search the work orders the employee will go to the work orders tab from the main menu. In the bar at the top of the screen, the employee will click and then enter the criteria they wish to search for. For example: name of client, day, type of job, etc.
* Edit Client:

1. Description: Allows for employee to edit a client’s information, such as the name, address and email address.
2. Actos:
   1. Employee
3. Preconditions:
   1. The application is able to connect to the database.
   2. The client the changes are to be made to already exist in the client registry.
4. Basic Flow of Events:
   1. Employee enters the client registry from the main screen.
   2. Employee selects the client that the changes are to be made to.
   3. Employee selects the “Edit Client” button from the upper right-hand corner of the screen.
   4. The employee changes the field(s) that is/are to be changed.
   5. The employee selects submit button.
   6. The changes are made to the client’s information in the database.
5. Alternative Flows:
   1. If the client to be edited does not exist, there will be no way to edit anything and the process would stop at step b from above.
   2. If there is no connection to the database, the employee will be unable to search for the client and the process will fail at step a above.
6. Post Conditions:
   1. Once the employee has selected the button to submit and the changes to the client’s information and the changes have been made to the database, the employee is redirected to the client’s profile page with the new updated information.
   2. If the process is a failure, the application will be redirected to the client’s profile with the only, non-updated information.

* Sends Email: When a work order is created, the system sends the client an email with the date and time that they’re scheduled for to the email that is on file.
* Receive Email: After an employee has created a work order for a client, the client will receive an email from Garvis Pool repair with the details of their work order such as time and date that the work order is scheduled for.
* Edit Work Order: To edit a work order, an employee will select the work orders tab from the main screen, locate and select the work order that is to be updated, and then select “Edit Work Order” from the work order’s profile. The employee will make the necessary changes and then click the submit button to make the changes to the work order.
* View Work Order: An employee is able to view a work order by selecting the work order tab from the main screen and then find the specific work order. This will redirect the employee to the work order profile.
* View Client: An employee may view a client’s profile by selecting the client tab from the main screen, searching for and then selecting the specific client. This will bring up that client’s profile containing and displaying all of their information.
* Search Client: To search for a client, an employee can select the client tab from the main screen, click on the search bar, and enter the criteria that is to be searched for.

**SECTION 3: Specific Requirements**

**Functional Requirements**

|  |
| --- |
| No: 001 |
| Statement: The software shall accept usernames containing only alphanumeric characters |
| Dependency: 012, 033 |

|  |
| --- |
| No: 002 |
| Statement: The software shall require all employee first names to only contain alphabetic characters |
| Dependency: 014, 033 |

|  |
| --- |
| No: 003 |
| Statement: The software shall require all employee last names to only contain alphabetic characters |
| Dependency: 015, 033 |

|  |
| --- |
| No: 004 |
| Statement: The software shall require a password with a minimum length of 6 containing only alphanumeric characters. |
| Dependency: 013, 033 |

|  |
| --- |
| No: 005 |
| Statement: The software shall move from screen to screen as intended when the associated button is pushed. |
| Dependency: 022, 033 |

|  |
| --- |
| No: 006 |
| Statement: The software shall require the client first name field to contain only alphabetic characters. |
| Dependency: 016, 033 |

|  |
| --- |
| No: 007 |
| Statement: The software shall require the client last name field to contain only alphabetic characters. |
| Dependency: 017, 033 |

|  |
| --- |
| No: 008 |
| Statement: The software shall require all notes fields to contain only ASCII characters. |
| Dependency: 023, 033 |

|  |
| --- |
| No: 009 |
| Statement: Calendar shall have a limit of 100 work orders created for any date. |
| Dependency: None |

|  |
| --- |
| No: 010 |
| Statement: The software shall only take addresses within the United States. |
| Dependency: None |

|  |
| --- |
| No: 011 |
| Statement: The software shall close if any communication failures occur. |
| Dependency: 030 |

# Interface Requirements

|  |
| --- |
| No: 012 |
| Statement: The software shall only accept usernames of string data type. |
| Dependency: None |

|  |
| --- |
| No: 013 |
| Statement: The software shall only accept passwords of string data type. |
| Dependency: None |

|  |
| --- |
| No: 014 |
| Statement: The software shall only accept employee first names of string data type. |
| Dependency: None |

|  |
| --- |
| No: 015 |
| Statement: The software shall only accept employee last names of string data type. |
| Dependency: None |

|  |
| --- |
| No: 016 |
| Statement: The software shall only accept client first names of string data type. |
| Dependency: None |

|  |
| --- |
| No: 017 |
| Statement: The software shall only accept client last names of string data type. |
| Dependency: None |

|  |
| --- |
| No: 018 |
| Statement: The software shall only accept email addresses of string data type. |
| Dependency: None |

|  |
| --- |
| No: 019 |
| Statement: The software shall only accept phone numbers of string data type. |
| Dependency: None |

|  |
| --- |
| No: 020 |
| Statement: The software shall only accept work order numbers of int data type. |
| Dependency: None |

|  |
| --- |
| No: 021 |
| Statement: The software shall only accept phone numbers of string data type. |
| Dependency: None |

|  |
| --- |
| No: 022 |
| Statement: The software shall only accept job fields of string data type. |
| Dependency: None |

|  |
| --- |
| No: 023 |
| Statement: The software shall only accept notes of string data type. |
| Dependency: None |

|  |
| --- |
| No: 024 |
| Statement: New work orders shall only be created if address, client and job fields have been correctly filled out and the scheduled date is not before the current date. |
| Dependency: 005, 006, 007, 016, 017, 022 |

|  |
| --- |
| No: 025 |
| Statement: The software shall only accept client information client if the first name, last name, address, and email fields have been correctly filled out. |
| Dependency: 006, 007, 016, 017, 026 |

|  |
| --- |
| No: 026 |
| Statement: The software shall require all email addresses to follow the format x@y.com |
| Dependency: 018 |

|  |
| --- |
| No: 027 |
| Statement: The software shall require all phone numbers to follow the format +########## where # represents any numeric digit. |
| Dependency: 021 |

|  |
| --- |
| No: 028 |
| Statement: The software shall refresh all database data items every time the user enters a new page. |
| Dependency: 031, 033 |

# Physical Environment Requirements

|  |
| --- |
| No: 029 |
| Statement: The software shall run on Android SDK minimum 24 |
| Dependency: None |

|  |
| --- |
| No: 030 |
| Statement: The software shall only run with access to the Internet. |
| Dependency: None |

**User and Human Factors Requirements**

|  |
| --- |
| No: 031 |
| Statement: The software shall support Employees to access the application and all up to date information. |
| Dependency: 033 |

|  |
| --- |
| No: 032 |
| Statement: The software shall require a maximum of one-hour training time to be able to carry out all functions the application provides; such as creating/editing a work order, creating/editing a client, logging in, searching for a work order or client, and using the Google API for directions. |
| Dependency: None |

**Data Requirements**

|  |
| --- |
| No: 033 |
| Statement: The software shall use FireBase for all database needs. |
| Dependency: 030 |

|  |
| --- |
| No: 034 |
| Statement: The software shall prevent adding a new client if the email for the new client is already being used by another client. |
| Dependency: 033 |

|  |
| --- |
| No: 035 |
| Statement: The software shall not retain any data on application shut down. All data will be populated on user log-in and page refresh. |
| Dependency: 033 |

|  |
| --- |
| No: 036 |
| Statement: The database shall be current up to one hour of creation, edit or delete of data items. |
| Dependency: 033 |

**Quality Requirements**

|  |
| --- |
| No: 037 |
| Statement: The software shall be available to use as long as FireBase is accessible and available. |
| Dependency: 030, 033 |

|  |
| --- |
| No: 038 |
| Statement: The software shall not crash on user input error. If user input error occurs, messages informing users of their error and the software shall allow users to input correct data. |
| Dependency: 012, 013, 014, 015, 016, 017, 018, 019, 020, 021, 022, 023, 024, 025, 026, 027 |

|  |
| --- |
| No: 039 |
| Statement: The software shall allow users to sign in and access data on any android device in which the software is installed. |
| Dependency: 033 |

|  |
| --- |
| No: 040 |
| Statement: The software shall require updates dependent on updates to FireBase. |
| Dependency: 033 |

**SECTION 4: Additional Material**