##### MedVoice (Health Awareness Mobile Application)

Software Requirements Specification

Alva Dynamics LLC

MedVoice Bio Tech

Megan Lyn A. Del Rosario

November, 2015

1. 1. Introduction

**1.1. Purpose**

The purpose of this document is to give a detailed description of the requirements for the data analytics portion of MedVoice. It will illustrate the purpose and complete declaration for the development of the system. It will also explain system constraints, interface and applicable interactions with other external applications. This document is primarily intended to be proposed to the customer for its approval and a reference for developing MedVoice for the development team. The use of MedVoice is intended for the general public.

* 1. **Scope**

MedVoice promotes health awareness. Using voice recognition, and specific user inputs (i.e. gender, height, weight) the mobile application will display possible conditions predisposed to the user based on the analysis of the collected and input data. MedVoice also allows the user to view their health history, and form a social network with care providers and other care recipients with similar predisposed conditions and interests.

MedVoice will have the ability to retrieve and send data to and from databases upon logging into the application. Using the collected data, users will be grouped according to interests and similarities.

* 1. **Definitions, acronyms, and abbreviations**

|  |  |
| --- | --- |
| SRS | Software Requirements Specification. |
| GUI | Graphical User Interface |

* 1. **References:**

[1] IEEE Recommended Practice for Software Requirements Specifications Software Engineering Standards Committee of the IEEE Computer Society. 1998.

* 1. **Overview:**

The rest of this document will have two major sections and Appendixes and will be ended with an index for the SRS documents [1]. The second section will contain all the product outlines, perspective, function, interface, and user characteristics. The third section will include some specific requirements (e.g. hardware requirements).

1. **Overall description**

MedVoice will collect basic data such as name and email upon logging into the GUI. Data analytics will then be performed to group users based on interests and similarities.

* 1. **Product Perspective**

This section will give an overall description of the back end (data analytics) portion of the mobile application.

* + 1. **System Interfaces**

MedVoice will be able to interface with other databases and mobile applications to retrieve and send data.

* + 1. **User Interfaces**

The user interface will be the GUI to present and retrieve data.

* + 1. **Hardware interfaces**

MedVoice will support most mobile platforms and will require a mobile device capable of recording audio.

* + 1. **Software interfaces**

The interface for the software will be a web browser and programming software to design and emulate the mobile application layout and workflow.

* + 1. **Communication interfaces**

Cellular networks and/or wireless networks are required.

* + 1. **Memory constraints**

Memory is limited to the mobile device’s memory capacity (RAM).

* + 1. **Operations**

Operation will be when the program is executed until the program is exited.

* + 1. **Site adaptation requirements**

There is no site adaption requirement.

* 1. **Product functions**

MedVoice

Arbitrary

Database

Information

Username

Password

Login

Information

MedVoice

Database

Figure 1 Sequence Diagram

* 1. **User characteristics**

The projected users of the software are the general public, and male population.

* 1. **Constraints**

System should be able to handle multiple users logged into the system, and the database should be able to store multiple sets of user data.

* 1. **Assumptions and dependencies**

MedVoice will be able to retrieve data at time of login, and upon request. Data presented will be in full screen.

* 1. **Apportioning of requirements**
* MedVoice will have the ability to make correlations from the data.
* Ability to display media/data based on interests and similarities
* Loyalty programs
* Ability to detect “bread crumbs”
* Census data
* Detect geographical location

1. **Specific requirements**

This section contains all the software requirements at a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements.

* 1. **External interface**

This section contains a detailed description of possible inputs into and outputs from the software system.

* Inputs
* Login Information
* Age, Gender, Weight, etc.
* Posts
  + Pictures
  + Videos
  + “Likes”
* Audio/Video Recordings
* Medical Articles/Data
  + Results, Charts, Graphs, etc.
* Outputs (currently out of scope)
* Medical Articles/Data
* Results, Charts, Graphs, etc.
* Posts
* Pictures
* Videos
* Audio/Video Recordings
  1. **Functions**
* Group users based on similarities and interests.
* Ability to send, store, and retrieve data.
  1. **Performance requirements**
* Coding will be done to optimize the processing of multiple users using the system.
* Coding will be done to optimize data storage of multiple users using the system.
  1. **Logical database requirements**
* Ability to store login information, place of residence, and contact information upon logging into the application.
* Ability to group users based on interests and similarities.
  1. **Design constraints**
* Data storage limited by device memory capacity.
* Back end data storage limited by database memory capacity.
  + 1. **Standards compliance**

The Software will comply with software coding standards.

* 1. **Software system attributes**
     1. **Reliability**

The Software should be functional during execution.

* + 1. **Availability**

The Software will be accessible through most mobile platforms, and major mobile platforms (IOS, Android).

* + 1. **Security**

The Software should have security features for only users that have access to the Software.

* + 1. **Maintainability**

The code will be well documented and most modules will be reusable.

* + 1. **Portability**

The code used in this program will be completely portable, allowing the code to go from one application to another as long as they comply with the hardware and software requirements detailed in this document.

* 1. **Organizing the specific requirements**
     1. **System mode**
* On (normal) mode – when the application is open.
* Off mode – when the application is not open.
  + 1. **User class**
* Care Recipients
* Care Providers
  + 1. **Objects**

See the Sequence Diagram in Figure 1.

* + 1. **Feature**

Ability to group users based on interests and similarities.

* + 1. **Stimulus**
* Login information required to associate data with user.
* Posts and Newsfeed to determine similarities.
  + 1. **Response**
* Ability to group users based on interests and similarities.
  + Display users with similarities
* Ability to retrieve and store data to the database upon login, and upon request.
  + Login denied or accepted (i.e. invalid password).
    1. **Functional hierarchy**



Retrieve Data

Figure 2 MedVoice Mobile Application Workflow

* 1. **Additional Comments**

No additional comments.

1. **Support Information**

**4.4.1 Table of Contents and index**

**Table of Contents**

1. Introduction.

**1.1. Purpose.**

**1.2. Scope.**

* 1. **Definitions, acronyms, and abbreviations**
  2. **References.**
  3. **Overview.**

1. **Overall description**
   1. **Product Perspective.**
      1. **System Interfaces.**
      2. **User Interfaces.**
      3. **Hardware interfaces.**
      4. **Software interfaces.**
      5. **Communication interfaces.**
      6. **Memory constraints.**
      7. **Operations.**
   2. **Product functions.**
   3. **User characteristics.**
   4. **Constraints.**
   5. **Assumptions and dependencies**
   6. **Apportioning of requirements**
2. **Specific requirements.**
   1. **External interfaces.**
   2. **Functions.**
   3. **Performance requirements.**
   4. **Logical database requirements.**
   5. **Design compliance.**
   6. **Software system attributes.** 
      1. **Reliability.**
      2. **Availability.**
      3. **Security.**
      4. **Maintainability.**
      5. **Portability.**
   7. **Organizing the specific requirements.**
      1. **System mode.**
      2. **User Class.**
      3. **Objects.**
      4. **Feature.**
      5. **Stimulus.**
      6. **Response.**
      7. **Functional hierarchy.**