

Software Requirements Specification

for

HealthQ

Version 1.0

**Prepared by Benny Villegas, Stuart Tresslar
Brian Deguzis, Austin Harris, Eddie Matos, Zikomo Bullock**

Team 2

**SWE 4724 Software Engineering Project
Spring Semester, 2018
Kennesaw State University
Department of Software Engineering and Game Design**

28-April-2018

Table of Contents

Introduction	4
Purpose	4
Document Conventions	4
Intended Audience and Reading Suggestions	4
Product Scope	4
References	4
Overall Description	5
Product Perspective	5
Product Functions	5
User Classes and Characteristics	5
Operating Environment	5
Design and Implementation Constraints	6
User Documentation	6
Assumptions and Dependencies	6
External Interface Requirements	6
User Interfaces	6
Hardware Interfaces	6
Software Interfaces	7
Communications Interfaces	7
Domain Model	8
System Features (Use Cases)	10
Use Case Description	10
Use Case Descriptions	10
Functional Requirements	13
Interface Requirements	13
Operational Requirements (a.k.a. Functional Capabilities)	13
Data Requirements	13
Nonfunctional Requirements	13
Performance Requirements	13
Safety Requirements	13
Software Quality Attributes	14
Other Requirements	14

Revision History

Name	Date	Reason For Changes	Version
Stuart	2/19/18	Addition of Sequence Diagram and State Chart	0.1
Stuart	3/4/18	Updated State Chart	0.5
Eddie	3/22/18	Added Non Functional Requirements	0.6
Stuart and Zikomo	4/28/18	Updated Use Case Diagram, Use Case Descriptions, and Requirements	1.0

1. Introduction

1.1 Purpose

The product that is described by this document is HealthQ. HealthQ is a software system that will be used by the general public to help analyze given data. This document is the Software Requirements Specification document and will be used to provide a detailed overview of the software product and its parameters and goals. This document also describes the project's target audience, its user interface, and hardware/software requirements.

1.2 Document Conventions

The basic template to be used is derived from IEEE Std 1058-1998, IEEE Standard for Software Project Management Plans. Sections were eliminated and added as a team.

1.3 Intended Audience and Reading Suggestions

This document is intended to give a detailed overview of our software product, its parameters and goals. It is intended to be read by all members of the project and contains details requirement specifications for our product.

1.4 Product Scope

The purpose of our product is to be able to create software that is able to analyze data and draw conclusions based on collected data. We plan on being able to integrate our software with products such as Google Home to be able to be used by the general public.

1.5 References

"Actions on Google," Google. [Online]. Available: <https://developers.google.com/actions/>. [Accessed: 09-Feb-2018].

U. S. C. Bureau, "Developers," Census.gov. [Online]. Available: <https://www.census.gov/developers/>. [Accessed: 09-Feb-2018].

U. S. C. Bureau, "Data Tools and Apps," Tools and Apps. [Online]. Available: <https://www.census.gov/data/data-tools.html>. [Accessed: 09-Feb-2018].

J. Dillard, "5 Most Important Methods For Statistical Data Analysis," Operations Consulting, Decision Analysis And Process Improvement. [Online]. Available: <https://www.bigskyassociates.com/blog/bid/356764/5-Most-Important-Methods-For-Statistical-Data-Analysis>. [Accessed: 09-Feb-2018].

“Google Genomics - Store, process, explore and share | Google Cloud Platform,” Google. [Online]. Available: <https://cloud.google.com/genomics/>. [Accessed: 09-Feb-2018].

2. Overall Description

2.1 Product Perspective

This product will be a follow-on member to the library of Google Actions and will interface with other Google Action modules, as well as external data libraries.

2.2 Product Functions

Through Google Assistant, HealthQ will provide the following functions:

Mean - given a numerical data set, HealthQ will return the sum of all the values divided by the number of values in the set.

Median - given a numerical data set, HealthQ will return the middle number of the data set.

Mode - given a numerical data set, HealthQ will return the most frequently occurring value(s) of the data set.

2.3 User Classes and Characteristics

There will only be one general user class for this system. This user will have access to all functionality including data retrieval requests, data analysis requests, and data visualization requests. Any administrative tasks such as changing the behavior of how HealthQ interfaces with other modules will be handled through the back end of development, and will not be accessible by the general user.

2.4 Operating Environment

We will be using a couple of resources for this project. The first will be Digital Ocean which is a programming environment which will be used for the server side development of the product.

We will be using Dialogflow, which is a development suite for building voice/ conversational interfaces for Google Assistant. We will be using FTP Clients which are necessary for access into the server repository, including read, write, and execution rights. We will also be using Actions On Google which is a program run via Google Assistant for developers of “Actions”. For the hardware side we will be utilizing the physical hardware of Google Home.

2.5 Design and Implementation Constraints

Our current design and implementation constraints are with utilizing the Google Home which requires us to utilize dialog flow to successfully communicate with the device. This hardware limitation also means that a GUI may not be available; however, a chat log can be kept of the conversation for documentation and record keeping purposes. Because we are using Google, we also require firebase to be able to save much of the information being received.

2.6 User Documentation

Example interactions with our application will be provided through online instructions and guides. These will include example dialogue paths and different ways to interface with our application from external sources.

2.7 Assumptions and Dependencies

Users of this application are assumed to have some statistical knowledge and knowledge of CDC terms, or whatever dataset they are working with.

The current application depends on the CDC's API for accessing their data sets.

3. External Interface Requirements

3.1 User Interfaces

The user interface for the software shall be compatible with devices capable of running Google Assistant such as Google Home.

The user interface shall be implemented using tools such as Actions by Google, Dialogflow, etc. The user interface should accurately display data through the use of graphs and charts.

The software can also be interfaced with purely through voice on devices such as the Google Home.

3.2 Hardware Interfaces

Since the application requires the Google Assistant and internet connectivity to function correctly, all hardware required for a given device to connect to the internet will be a hardware interface for the system. This includes, but is not limited to modems and wireless routers. Due to the nature of this application, any device capable of running the Google Assistant will be supported.

The device requires a microphone and speaker for communicating directly with the application.

3.3 Software Interfaces

The application will be built using the Google Assistant virtual assistant, which is currently capable of running on the following operating systems: Android Lollipop, Android Marshmallow and iOS 9.3.

The application will be interfacing with Digital Ocean for server-side development.

The Google Cloud Console will be used for designing and creating Actions for use in the application.

Dialogue Flow will be used for designing and creating dialogues for Google Assistant.

3.4 Communications Interfaces

The communications interfaces that are being utilized are FTP clients for access to the Digital Ocean server repository.

The application will be compatible with Google Assistant's protocol for voice commands, voice searching, and voice-activated device control.

4. Domain Model

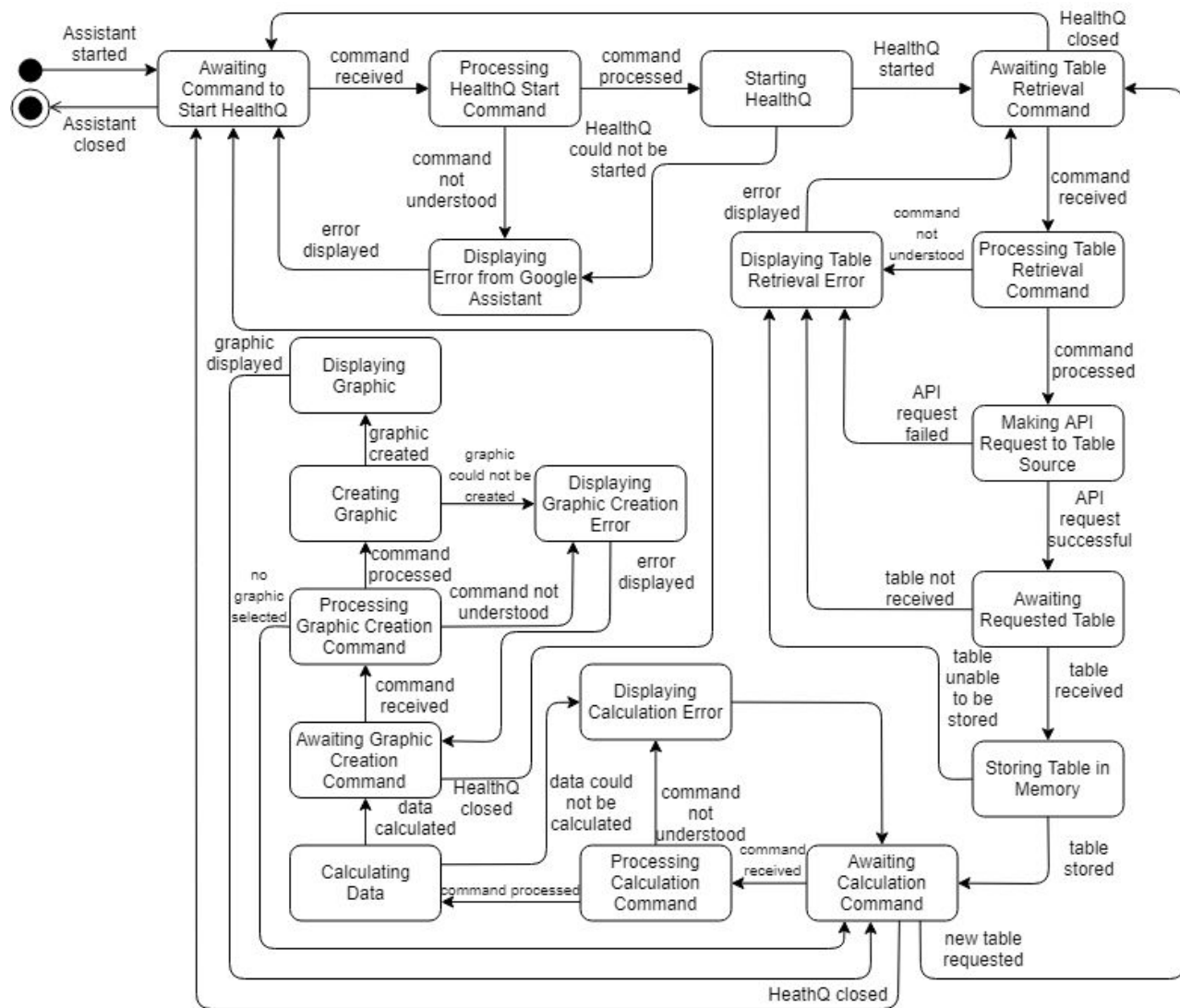


Figure 4-1 HealthQ State Chart Diagram

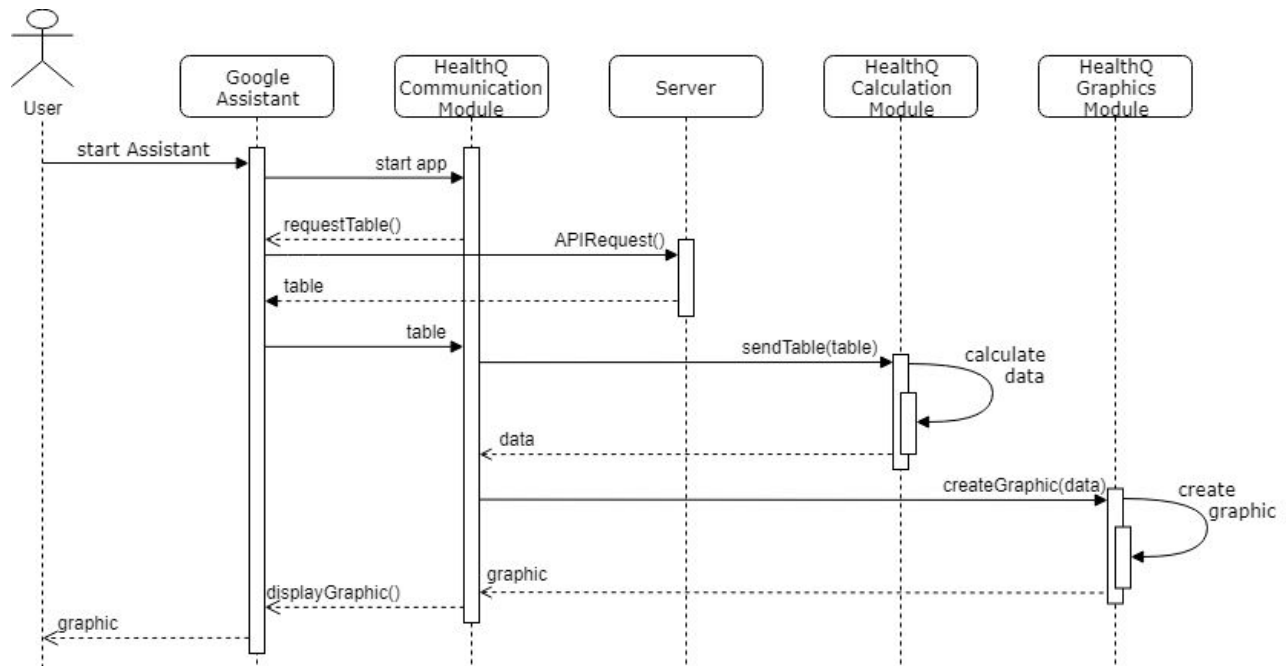


Figure 4-2 HealthQ Sequence Diagram

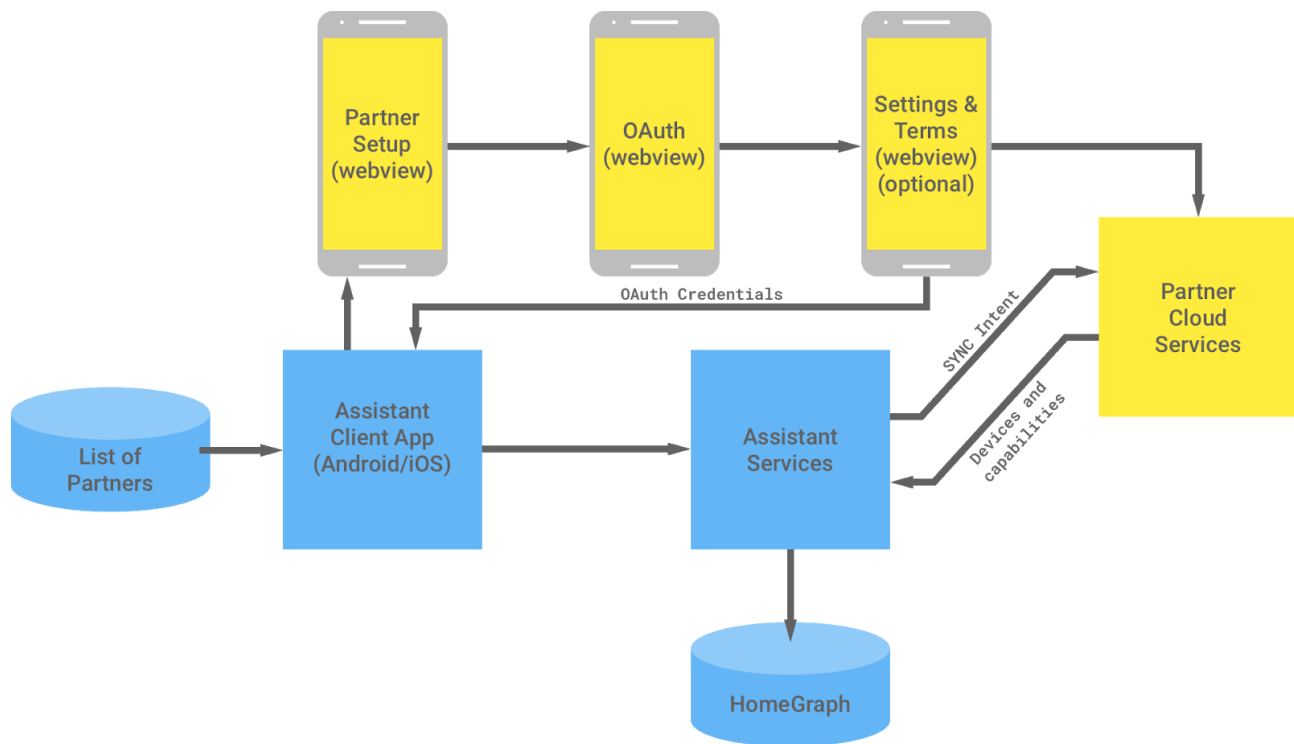


Figure 4-3 Google Assistant Component Diagram

5. System Features (Use Cases)

5.1 Use Case Description

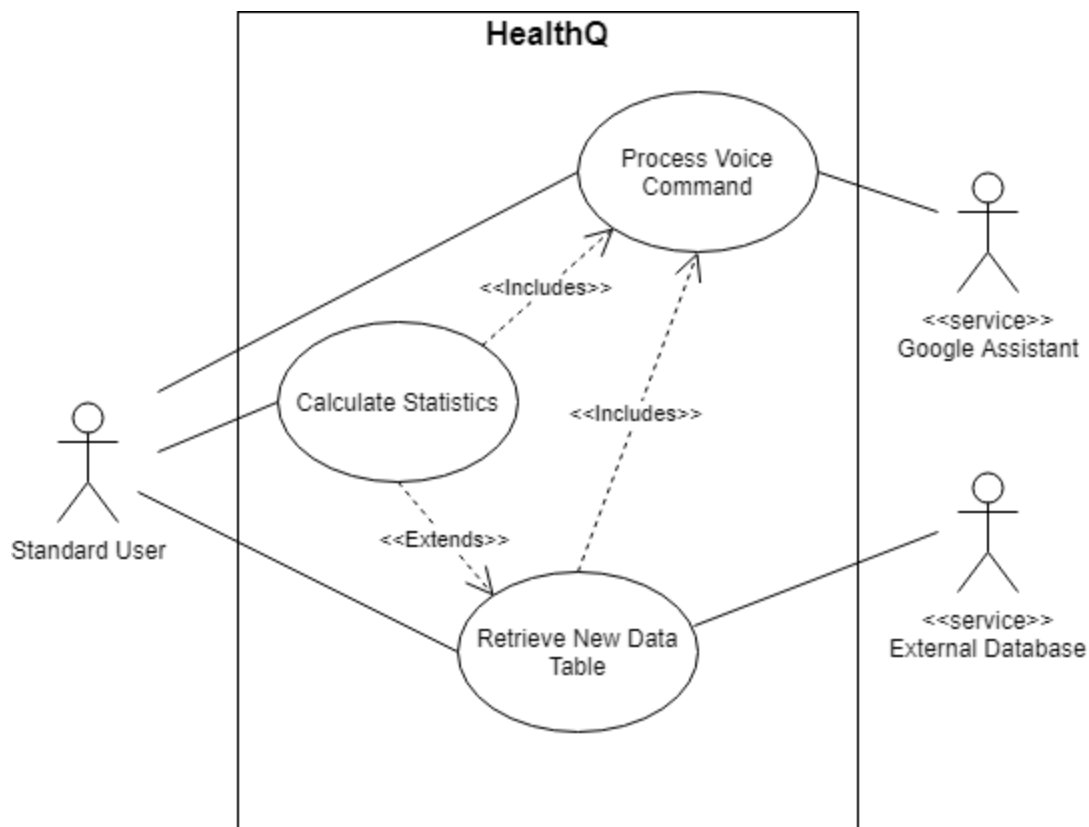


Figure 5-1 HealthQ Use Case Diagram

5.2 Use Case Descriptions

ID	UC_HQ_01
Name	Process Voice Command
Author	Zikomo Bullock and Stuart Tresslar
Date	February 27, 2018
Description	Allows the user to retrieve a data set to use with HealthQ

Actors	Standard User, Google Assistant
Preconditions	Health Q must currently be active
Assumptions	The standard user is utilizing Google Assistant via voice commands
Triggers	The Standard User prompts the device with “Hey Google”.
Normal Flow	<ol style="list-style-type: none"> 1. Google Assistant awaits a command. 2. Standard User vocalizes a command. 3. Google Assistant saves the command. 4. Google Assistant turns command into a query. 5. Google Assistant references the query within HealthQ’s library of voice commands. 6. Google Assistant starts the appropriate HealthQ command.
Exceptional Flow	<ol style="list-style-type: none"> 1. Google Assistant awaits a command. 2. Standard User vocalizes a command. 3. Google Assistant saves the command. 4. Google Assistant turns command into a query. 5. Google Assistant references the query within HealthQ’s library of voice commands. 6. Google Assistant is unable to find appropriate HealthQ command. 7. Google Assistant outputs error message.
Post Conditions	Appropriate HealthQ response has been given.

ID	UC_HQ_02
Name	Retrieve New Data Table
Author	Zikomo Bullock and Stuart Tresslar
Date	February 27, 2018
Description	Allows the user to retrieve a data set to use with HealthQ
Actors	Standard User, External Database
Preconditions	HealthQ must currently be active and UC_HQ_01 finished its normal flow.
Assumptions	An API for the table is present.
Triggers	Standard User requests table retrieval

Normal Flow	<ol style="list-style-type: none"> 1. HealthQ makes an API request. 2. External Database receives the request. 3. External Database responds to the request. 4. HealthQ receives access to the requested table.
Exceptional Flow	<ol style="list-style-type: none"> 1. HealthQ makes an API request. 2. External Database could not be connected to.
Post Conditions	HealthQ has access to the requested table.

ID	UC_HQ_03
Name	Calculate Statistics
Author	Zikomo Bullock and Stuart Tresslar
Date	February 27, 2018
Description	HealthQ calculates desired statistics for the Standard User
Actors	Standard User
Preconditions	HealthQ has access to a data table.
Assumptions	Standard User has knowledge of statistics they require.
Triggers	Standard User makes a request to calculate a statistic.
Normal Flow	<ol style="list-style-type: none"> 1. HealthQ reads table. 2. HealthQ parses table for required data. 3. HealthQ performs calculations with said data. 4. HealthQ outputs results of calculation.
Exceptional Flow	<ol style="list-style-type: none"> 1. HealthQ reads table. 2. HealthQ parses table for required data. 3. HealthQ is unable to find required data. 4. HealthQ outputs error regarding absence of requested data.
Post Conditions	HealthQ has communicated result.

5.3 Functional Requirements

5.3.1 Interface Requirements

INT.1.1 – The system shall be able to process sound based commands.

INT.1.2 – The system shall return results back to the user in an easy to understand format.

INT.1.3 – The system shall be able to return an error if a task is unsuccessful.

5.3.2 Operational Requirements (a.k.a. Functional Capabilities)

OPR.1.1 – The system shall be able to turn voice commands into a query.

OPR.1.2 – The system shall be able to reference a query within a library of voice commands.

OPR.1.3 – The system shall be able to make API requests.

OPR.1.4 – The system shall be able to receive data from external database.

OPR.1.5 – The system shall be able to read a data table.

OPR.1.6 – The system shall be able to parse a table for data.

OPR.1.7 – The system shall be able to perform mathematical calculations.

5.3.3 Data Requirements

DAT.1.1 – The system shall be able to take a location as user input.

DAT.1.2 – The system shall be able to take a date as user input.

DAT.1.3 – The system shall be able to take a calculation as user input.

6. Nonfunctional Requirements

6.1 Performance Requirements

PR.1.1 – The system shall be able to respond to a users request within 10 seconds.

PR.1.2 - The system should start up within 10 seconds of initiation.

6.2 Safety Requirements

SR.1.1 - The system should not pull data from unsafe external resources.

SR.1.2 - The system should only respond to prespecified voice commands.

SR.1.3 - The system should adhere to privacy laws.

6.3 Software Quality Attributes

- SQA.1.1 - The system should be available every hour of every day.
- SQA.1.2 - The system should be able to be used by visually impaired users.
- SQA.1.3 - The system should log any error encountered in use for maintenance purposes.
- SQA.1.4 - The software should be able to be compatible with any devices capable of running Google Assistant.
- SQA.1.5 - The software should provide correct results one hundred percent of the time.
- SQA.1.6 - The software should allow for updates to be installed remotely.

7. Other Requirements

- OR.1.1- The system should be able to be used by people who speak different languages.

Appendix A: Glossary

Term	Definition
SRS	Software design document is a written description of a software product, that a software designer writes in order to give a software development team overall guidance to the architecture of the software project.
INT	Interface requirements
OPR	Operational requirements
DAT	Data Requirements
PR	Performance Requirements
CDC	Center for Disease Control
SQA	Software Quality Attribute
SR	Safety Requirements