**System Design Document**

**For**

**COVID-19 Tracking App**

Team Members:

● Sultan Abdalla Salim Saif Alteneiji

● Justin Andrews

● Bryce Cole

● Jonah Corbin

|  |  |  |
| --- | --- | --- |
| Version | Date | Team Members |
| V1 | 10/01/2020 | Justin, Bryce |
| V2 | 10/28/2020 | Justin |
|  |  |  |
|  |  |  |

**Table of Contents**

[**1 Introduction**](#_6njkmjuo0foa) **3**

[1.1 Purpose and Scope](#_pnaugng58v6) 3

[1.2 Project Executive Summary](#_pu4s62u2qab5) 3

[1.2.1 System Overview](#_4bpac5rtrlzx) 3

[1.2.2 Design Constraints](#_r12cq7pdzti9) 4

[1.2.3 Future Contingencies](#_eq0vwh7ydtyb) 4

[1.3 Document Organization](#_svqykglvdhc4) 4

[1.4 Project References](#_w6wa429vndau) 4

[1.5 Glossary](#_avi6ppa01peu) 5

[**2 System Architecture**](#_8jvxty74r2n0) **5**

[2.1 System Hardware Architecture](#_p0o96ivp1weu) 5

[2.2 System Software Architecture](#_3cgrxxpuuezv) 5

[2.3 Internal Communications Architecture](#_9xf90pv8dq50) 6

[**3 Human-Machine Interface**](#_9slxs4xp1gc9) **6**

[3.1 Inputs](#_pwc89rxposuu) 6

[3.2 Outputs](#_9j908lvixmce) 6

[**4 Detailed Design**](#_7e1j1df6ioty) **7**

[4.1 Hardware Detailed Design](#_mfduenwjr7sc) 7

[4.2 Software Detailed Design](#_xs1ddmfdfij3) 8

[4.3 Internal Communications Detailed Design](#_b701wv9fedfd) 8

[**5 External Interfaces**](#_vm255o2wkuxq) **8**

[5.1 Interface Architecture](#_fi4doz9tguzz) 8

[5.2 Interface Detailed Design](#_insbxji5q3r2) 9

[**6 System Integrity Controls**](#_yt30m3oq0ft4) **9**

# 1 Introduction

## **1.1 Purpose and Scope**

This document will describe the system overview, system design, inputs, outputs, and interface of the COVID-19 Tracking App project.

## **1.2 Project Executive Summary**

Section 1.2 gives a high level overview and introduction to the system and its constraints.

### 1.2.1 System Overview

Figure 1 is a use case diagram that gives an overview of how the user and the software interact.

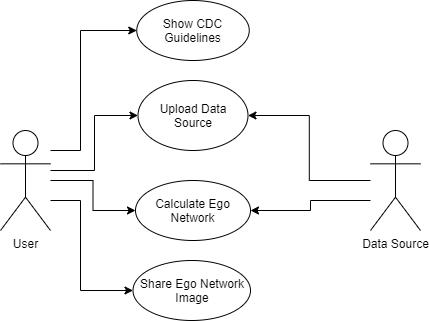


Figure 1: Use Case Diagram for COVID-19 Tracking App.

Figure 2 gives a high level overview of the system process that calculates the ego network of the user. This figure includes inputs, like the user uploading their communication data, and outputs, such as displaying the user’s ego network that has been calculated.

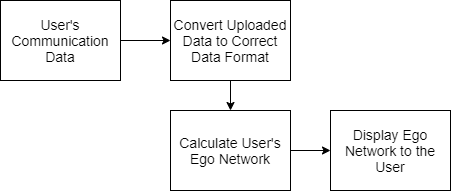


Figure 2: Overview of the process of the system

### 1.2.2 Design Constraints

There are no current constraints in the system design.

### 1.2.3 Future Contingencies

There are currently no future contingencies due to the lack of interfaces that are needed to communicate to provide issues with compatibility.

## **1.3 Document Organization**

The purpose of this Software Design Document is to give an overview of the design of the system.

## **1.4 Project References**

1.4.1 A. Sutcliffe and D. Wang, “Computational Modelling of Trust and Social Relationships,” *jasss.soc.surrey.ac.uk*, 31-Jan-2012. [Online]. Available: http://jasss.soc.surrey.ac.uk/15/1/3.html. [Accessed: 29-Oct-2020].

1.4.2 D. Wang, “Computational Model Library,” *SBH trust model*, 14-Dec-2010. [Online]. Available: https://www.comses.net/codebases/2293/releases/1.0.0/. [Accessed: 29-Oct-2020].

## **1.5 Glossary**

There are no abbreviations that need to be defined in this document.

# 2 System Architecture

## **2.1 System Hardware Architecture**

This project is purely software so there is no system hardware architecture needed for this project.

## **2.2 System Software Architecture**

The app has 3 major components: accepting the user’s uploaded data, calculating the user’s ego network, and displaying the ego network and its details to the user. The user uploads their communication data to the app which the app then cleans so that the data is in the format that the ego network algorithm can work with. The app can then use this data to calculate their ego network. Finally, the app displays the ego network to the user. The user can then select different levels of their ego network to get more information about that level, such as who of their friends falls within that level. This is shown below in Figure 3.

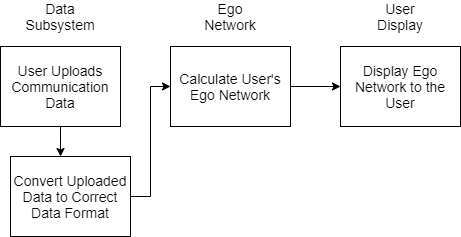


Figure 3: System Architecture Overview

## **2.3 Internal Communications Architecture**

This project is purely software so there are no internal hardware communications that must be described.

# 3 Human-Machine Interface

## **3.1 Inputs**

To use this app the user must use their device’s file explorer to upload their communication data (as a JSON file) to the app for the ego network algorithm to analyze. After uploading their data the user must provide their consent for the app to look at and analyze this data by selecting that they wish to have their ego network calculated. This process is shown below in Figure 4.

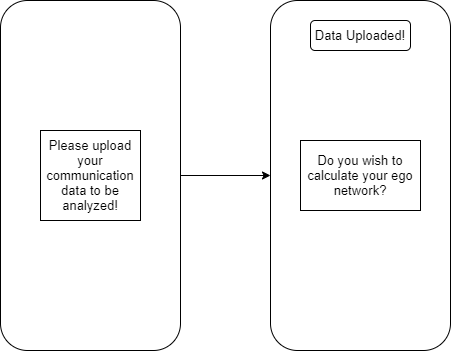


Figure 4: Input Display to the User

## **3.2 Outputs**

After the app has calculated the user’s ego network the app shall display the ego network to the user in an interactive way. The user shall be able to see how many people are in each level of their ego network but if they select a level they will be able to see a more detailed view of which of their friends are in that specific level of their ego network. This display is shown below in Figure 5. The left side of Figure 5 shows the user’s ego network as a whole. The right side of Figure 5 shows the view the user would see if they selected a level, in this case Level 2. It would show the people in that level and their friendship score.

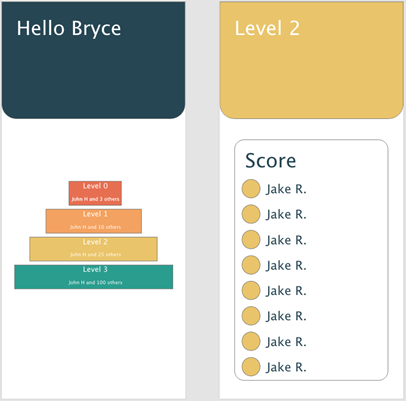


Figure 5: Output Displayed to the User

# 4 Detailed Design

## **4.1 Hardware Detailed Design**

This project is purely software so there is no hardware detailed design that needs to be described.

## **4.2 Software Detailed Design**

This app has 3 main systems; the display subsystem, the data subsystem, and the ego network subsystem. The display subsystem is in charge of displaying the app to the user. This includes displaying the input and output to the user. Next is the data subsystem, the system deals with the user’s data once it is uploaded. This system will accept the data and then clean the data so that it is in an acceptable format to the ego network algorithm. Finally, the ego network algorithm system takes the data from the data system, calculates the user’s ego network, and then gives the ego network to the user display system to be displayed. An overview of these subsystems and how they communicate can be seen below in Figure 6.

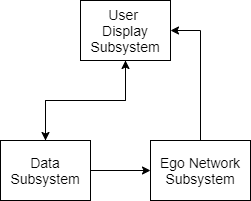


Figure 6: Software Architecture

## **4.3 Internal Communications Detailed Design**

This project is purely software so there is no internal communications to take place that would need to be described.

# 5 External Interfaces

## 5.1 Interface Architecture

The project has no interfaces that need to be described.

## 5.2 Interface Detailed Design

This project has no interfaces that need to be described.

# 6 System Integrity Controls

To protect the privacy of the user, the app will hash the file that contains the output of the ego network algorithm.