**Software Requirements and Design Document**

**For**

**Group TLH COVID TRACKER**

Version 1.0

**Authors**:

Raghav Serma

Adrian Balbuena

Evan Foglia

Prachi Kapur

Javier Roque

# Overview (5 points)

*Give a general overview of the system in 1-2 paragraphs (similar to the one in the project proposal).*

*This is a web-based application implementing CRUD functionality.*

*This application will read in data from users who enter specifics in a reactive form and read in that data and store that data. The data will then be stored in json-server, which can be pulled through API requests in the form of a table with features including filtering, sorting, and querying. The data can be deleted as well when the data becomes irrelevant or is incomplete). Login functionality to view the data.*

# Functional Requirements (10 points)

*List the* ***functional requirements*** *in sentences identified by numbers and for each requirement state if it is of high, medium, or low priority. Each functional requirement is something that the system shall do. Include all the details required such that there can be no misinterpretations of the requirements when read. Be very specific about what the system needs to do (not how, just what). You may provide a brief design rationale for any requirement which you feel requires explanation for how and/or why the requirement was derived.*

*1. Have editable table-high*

*2.. Have secure login and registration-high*

*3. Have home/info page-medium*

*4.have working reactive form-low*

*5. send data to backend and retrieve data from backend-high*

*6. store data in sql database-medium*

# Non-functional Requirements (10 points)

*List the* ***non-functional requirements*** *of the system (any requirement referring to a property of the system, such as security, safety, software quality, performance, reliability, etc.) You may provide a brief rationale for any requirement which you feel requires explanation as to how and/or why the requirement was derived.*

1. *Setting up of a secure login to view the data. This will be done through Google Firebase.*
2. *Safety of the data being entered through registered users and validity of the data*
3. *Safety of the data to not be tampered with by reporting*
4. *The reliability of the data being shown is crucial as the wrong data could produce bad or biased results with analysis.*
5. *Cross platform and inter version compatibility to run on any device. Containerize code.*

# Use Case Diagram (10 points)

*This section presents the* ***use case diagram*** *and the* ***textual descriptions*** *of the use cases for the system under development. The use case diagram should contain all the use cases and relationships between them needed to describe the functionality to be developed. If you discover new use cases between two increments, update the diagram for your future increments.*

***Textual descriptions of use cases****: For the first increment, the textual descriptions for the use cases are not required. However, the textual descriptions for all use cases discovered for your system are required for the second and third iterations.*

# Class Diagram and/or Sequence Diagrams (15 points)

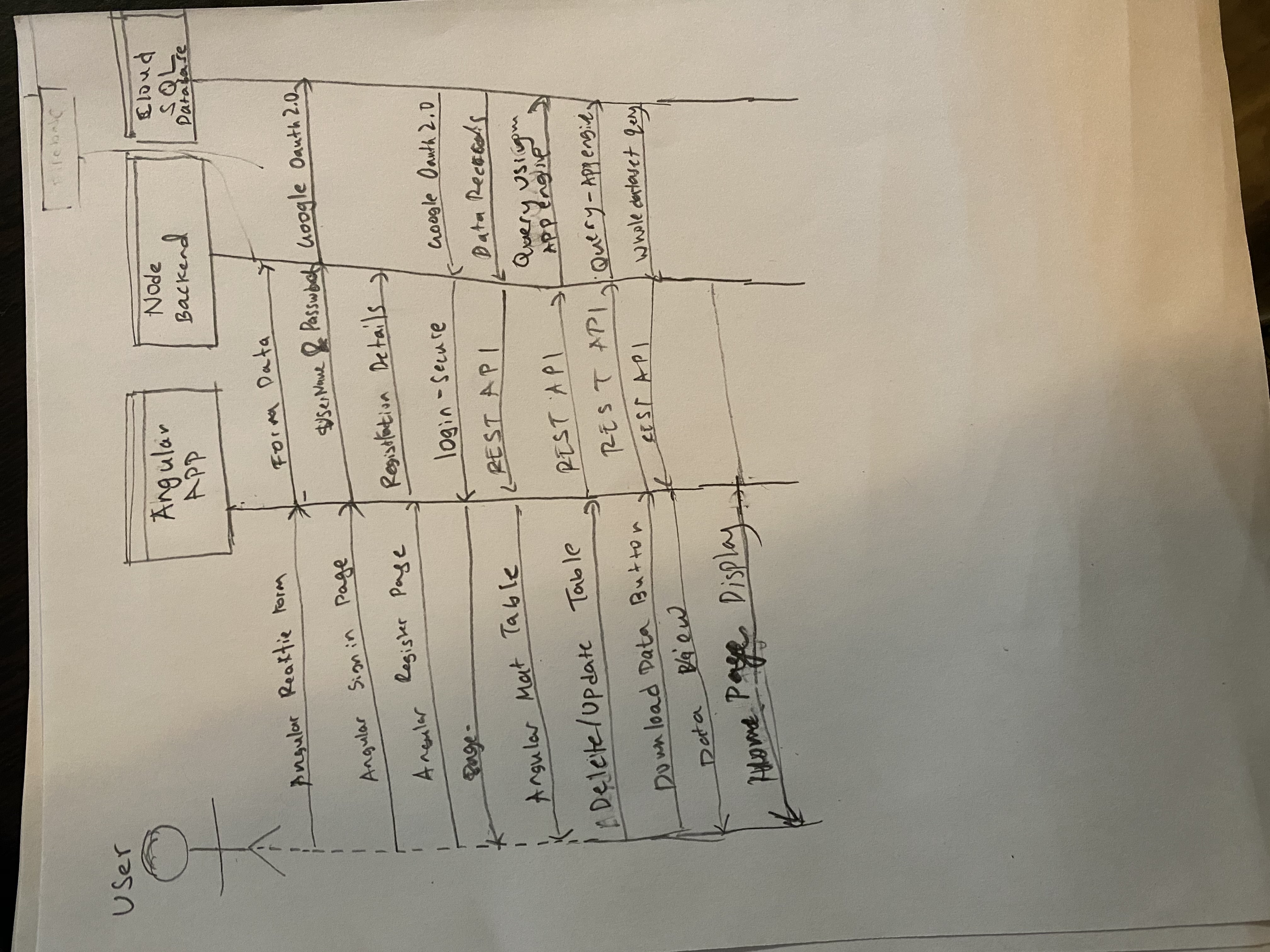
*This section presents a high-level overview of the anticipated system architecture using a* ***class******diagram*** *and/or* ***sequence diagrams****.*

*If the main* ***paradigm*** *used in your project is* ***Object Oriented*** *(i.e., you have classes or something that acts similar to classes in your system), then draw the* ***Class Diagram******of the entire system and Sequence Diagrams for the three (3) most important use cases in your system.***

*If the main* ***paradigm*** *in your system is* ***not Object Oriented*** *(i.e., you* ***do not*** *have classes**or anything similar to classes in your system) then only draw* ***Sequence Diagrams****,* ***but for all the use cases of your system.*** *In this case, we will use a modified version of Sequence Diagrams, where instead of objects, the lifelines will represent the functions in the system involved in the action sequence.*

***Class Diagrams*** *show the* ***fundamental objects/classes*** *that must be modeled with the system to satisfy its requirements and* ***the relationships*** *between them. Each class rectangle on the diagram* ***must also include the attributes and the methods of the class*** *(they can be refined between increments). All the* ***relationships between classes and their multiplicity*** *must be shown on the class diagram.*

*A* ***Sequence Diagram*** *simply depicts* ***interaction******between objects*** *(or* ***functions -*** *in our case - for non-OOP systems) in a sequential order, i.e. the order in which these interactions take place. Sequence diagrams describe how and in what order the objects in a system function.*



# Operating Environment (5 points)

*Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.*

*The app can run on any web-enabled device as it is a web application. It can run on any modern web browser but not older versions as the features of Angular 9 are new. Tis will be overcome with the help of the polyfills file. It will run on npm, or node package manager. The backend of this is on node, so Angular has easy access to using node as a backend.*

# Assumptions and Dependencies (5 points)

*List any assumed factors (as opposed to known facts) that could affect the requirements stated in this document. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project.*

Assuming client access to firease.

Assuming gcp will be on free trial

Assuming google OAuth2.0 is supported on Angular

Assuming forms are reactive and not template driven

Assuming backend will have increased functionality with angular application.

Assuming SQL databse is normalized.