**Software Implementation and Testing Document**

**For**

**Group 31**

Version 2.0

**Authors**:

Aiden Livingston

Dustin Haire

Maria Penalver

McKenna Warrick

Maria Penalver

# Programming Languages (5 points)

* JavaScript
* CSS
* HTML

# Platforms, APIs, Databases, and other technologies used (5 points)

***Technologies***

* *JavaScript: A versatile programming language used for client-side and server-side web development* 
  + *React: JavaScript library for building user interfaces, particularly single-page applications*
* *Node.js: A JavaScript runtime built on Chrome's V8 JavaScript engine, allowing you to run JavaScript on the server* 
  + *react-router: A standard routing library for React, used to handle navigation in single-page applications*
  + *firebase admin: A server SDK that lets you interact with Firebase from privileged environments to perform various administrative tasks*
  + *axios: A promise-based HTTP client for making requests to APIs from both the browser and Node.js*
  + *nodemon: A utility that monitors for changes in your source and automatically restarts your server*
  + *cors: A Node.js package for providing a Connect/Express middleware that can be used to enable CORS (Cross-Origin Resource Sharing)*
* *Express.js: A minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications*
* *CSS: Cascading Style Sheets, a style sheet language used for describing the presentation of a document written in HTML or XML* 
  + *SCSS: Sassy CSS, a preprocessor scripting language that is interpreted or compiled into CSS*
* *Firebase: A platform developed by Google for creating mobile and web applications* 
  + *Firestore NoSQL Database: A flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud Platform*

# Execution-based Functional Testing (10 points)

At this stage of the project, we have not yet conducted functional testing. However, we plan to implement execution-based functional testing by the third increment of development. During this phase, we will validate the functional requirements specified in our RD Document by executing test cases for user and API interactions.

# Execution-based Non-Functional Testing (10 points)

*Describe how/if you performed non-functional testing for your project (i.e., tested for the* ***non-functional requirements*** *listed in your RD).*

# Non-Execution-based Testing (10 points)

*Describe how/if you performed non-execution-based testing (such as code reviews/inspections/walkthroughs).*