**Software Implementation and Testing Document**

**For**

**Group <13>**

Version 3.0

**Authors**:

Jillian Bice

Matthew Bucataru

Brandon Specht

Brandon Cook

Libby Sauer

# Programming Languages (5 points)

We are using JSX for the components of the front end as well as CSS for the styling. For the back end, we are using Node.js. We are using SQL to query the databases.

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

For the front end, we are utilizing React as well as React Router in order to configure our routes for the website. Some elements of our CSS utilize Tailwind CSS for the styling.

We are using MySQL for the database, hosted by AWS.

# Execution-based Functional Testing (10 points)

We signed in as different users, testing the functionality of each page. We were testing for any irregularities that would be undesirable to the user.

We ensure that all the links/buttons are directing the user to the desired location.

We tested the swipe multiple times to ensure:

* the user you were swiping on was not you or repeated repeated user
* the personal data showing was correct
* names showing up were correct

We tested the sign in, sign up, and sign out

We tested that matches were displaying the correct people

We tested that the message page functioned properly

We tested that the profile page was displaying the correct name and information

We tested that the edit profile actually edited the profile data

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

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

We tested the speed of the buttons which was not fast enough so we changed them and tested them again getting better results

*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)

We did more pair programming in which one person was writing the code and another was evaluating the code as it was being written.