CS 411 Stack Selection Documentation

For this project we decided to utilize Node.js, with the express framework for the backend, MySQL for the database, and React javascript for the library on the front end, which will be fully integrated in future iterations of the prototype. In constructing the technology stack, the primary decision was selecting the framework for our backend. We were initially thinking between two technology stacks:

- Stack 1: Node.js + Express + MongoDB + React.js
- Stack 2: Node.js + Angular + MySQL + React.js

We decided to use the express framework (Stack 1) for the primary reason that it was easy to learn. This was our main concern due to the limited timeframe and our own levels of experience with web frameworks. This stack toolchain also enables end-to-end continuity with javascript, which is the language most of our team members are familiar with. Additionally, there is a large selection of libraries/packages, allowing for cleaner API use and easy debugging. We also found that express was relatively fast, at least in comparison to certain alternatives, such as Python. Another factor in this decision was the educational benefit of working with the selected framework, and as a common tool in industry, Express offers valuable experience.

We decided *not* to use Angular (Stack 2) because Angular is considered to have a steep learning curve which would dominate more of the time we spent on the project than could be justified. We decided that our time would be better allocated by focusing on other aspects such as brainstorming ideas and implementation of APIs/libraries rather than familiarizing ourselves with a framework none of us had much experience in. We also figured that learning Angular wouldn't be worth the time for a classroom project of this caliber, and that ease of implementation should be one of our main priorities. Furthermore, Angular had relatively worse search optimization than our stack 1.

For these reasons, we decided on the Express framework for our technology stack.