

For our front end, we were debating using React.js or using Angular. In the end we decided to use React.js instead of Angular because one of our members has experience with React.js. In addition, we believed that React.js would be a better technology stack for our project upon further research. Firstly, people online said that React.js was easier to learn and since we wanted as many people as involved as possible, we figured React.js would be the better option. Furthermore, some members also have experience with HTML and CSS which are more compatible with React.js. Lastly, since our user would be constantly sending information to the website, we wanted to make sure that whenever the user updated a part of the website, it wouldn't affect other parts of the interface unless necessary.

For our backend, we considered using Flask or Node.js and Express. We decided to go with Flask because one of our team members had direct experience with the framework, and most team members were familiar with Python, but not particularly familiar with JavaScript. We also found that Flask is more user-friendly and intuitive than Express through our research, contributing to our overall decision to use Flask and Python.

For our database, we thought about using PostgreSQL, Firebase, and SQLite. We decided to go with Firebase's Realtime Database because we could also utilize Firebase's login authentication feature, so it made the most sense to minimize the technologies that we were going to use.

For deployment, we considered Heroku, AWS, Firebase. We decided to go with Firebase, for the sake of simplicity and comfort over AWS. Heroku is also no longer free, so it made the most sense to go with Firebase.