Technology Stack Decisions

Our group will be building our application using a backed Django framework. Our front-end will be designed using HMTL and CSS. Currently, the database we are using is the internal DB SQLite. As we near completion of the project, we will migrate our database to a more scalable platform, MySQL. Our decision to build our prototype in SQLite was largely in part due to the [convenience of the database](https://stackoverflow.com/questions/711048/what-is-sqlite-used-for).

Everybody in the team has the most experience coding in Python, so we knew we would definitely be using a Python framework due to the smaller learning curve. However, we did debate between using Django vs Flask framework. One team member had built a web application in CS460 using Flask, MySQL, & HTML and was familiar with the syntax. After reading this [article,](https://dev.to/django_stars/django-vs-ruby-on-rails-web-frameworks-comparison-1fa0) we decided to build the application using Django framework. Technical incentives were useful documentation, plugins, and the REST framework. Additionally, no team members have worked with Django prior to this class, and we wanted to learn a new and useful technical skills. Team members also had very little experience with and wanted to learn HTML and CSS, and wanted to make stack decisions that would integrate well with these front-end tools.