The reason why we selected MySQL as our database is because it is a relational database, meaning that it is suited for storage and retrieval. The relational part of the database allows easier and wider range of access to data that will allow us to manipulate the data in our system to its full potential.

We chose MySQL because it is easier to use. Given our limited experiences in database language, we want to use a system that will allow us to do the most without spending too much time on studying how a different database works. Moreover, MySQL is also open source software that is owned by Oracle, which means that our group won’t have to spend any money on it. Needless to say, the MySQL is also very popular amongst programmers. There is a great online community that we can take an advantage of to learn, and optimize our database for the best result.

We did take several limitations of MySQL into account. Some of them are poor performance scaling, limited functionalities, and lesser in quality than others. (1) We decided that poor performance scaling wouldn’t be a big issue due to the fact that we are not operating with a ton of data. Next, we are also not anticipating a great number of functionalities to implement inside our database, as we will have backend codes that will do the job. Our intended use of the database should be straightforward push and pull requests. Lastly, quality and stability is not something we are currently targeting for in our prototype phase. Our goal to set up the environment first with everything work before we start to optimize our environment.

|  |  |
| --- | --- |
| Pro | Cons |
| Easy to use | Poor performance scaling |
| Open source | Limited functionalities |
| Great online community | Inferior quality to top-tier databases |
| Suitable performance | Stability issues |

<https://www.datarealm.com/blog/five-advantages-disadvantages-of-mysql/>

Frontend

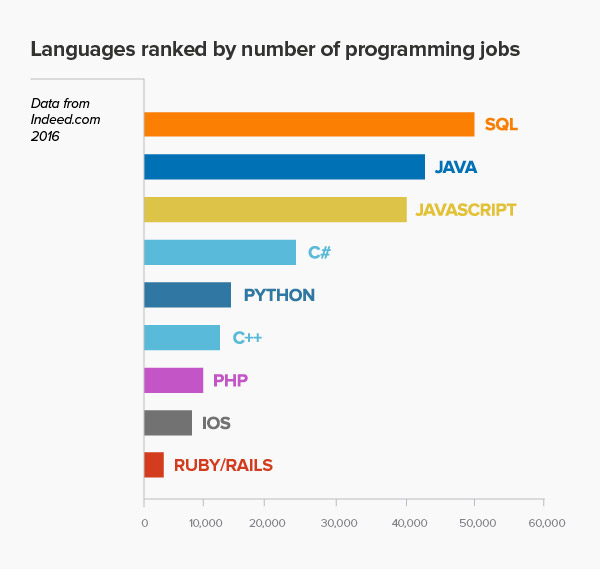
Since our team has the most experiences in JavaScript, we decided to design our front end in JavaScript, HTML 5, and CSS. We chose JavaScript because it remains one of today’s most used languages in websites. It is easier to code than server-side languages, and it offers a great range of varieties when it comes to dynamic views and displays for our users. Moreover, JavaScript is entirely a client side programming, meaning the computation is done on our users’ computer rather than having data sent back to the server and back out to the user. We rely on this mechanism to make it easier for our servers, so we can reach up to more users.

HTML 5 and CSS will be used to make our website more visibly attractive to our users. We hope that our front-end code will be able to take a full advantage of our design to create the best user experiences possible. Nevertheless, these two languages are easy to learn and implement since they are ubiquitous in today’s websites.



Backend

We want to link our frontend code to our database through a backend written in Java. Mainly because Java is still one of the most popular languages out there in the industry and many of us have experiences with them. Java offer a wide range of functionalities and online communities to help us achieve what we want.



Other Technologies

Draw.io

We used Draw.io to do most of our Entity-Relationship diagrams. It is an easy to use online program that allows us to organize our diagrams neatly.

Asana

We use Asana to plan most of our daily activities and group goals. It offers a clear view of what our goals are and how to get there, if we put in the work to create it.

GitHub

GitHub is where our repository is located. We have access see our peers’ work as well as the progression of our big project. Moreover, it is very convenient for each of us to edit and make commentary about other’s work.

GroupMe

We use GroupMe mostly to communicate with one another.