Annie Tran

September 29, 2015

Engineering World Health

Computer Science Subteam

Technology Proposal

**Framework**

* Ruby on Rails:
  + Pros:
    - Popular and widely used; therefore, lots of resources for information/assistance with bugs and errors
    - Gives user lots of power and freedom in design
* Django:
  + Pros:
    - Uses Python
    - Code readability is of upmost importance
    - Structured, consistent and simple

**Database**

* Microsoft SQL Server 2012:
  + Pros:
    - Provides strong data integrity and security built right into the database without having to rely on application rules and logic
    - SQL is easy-to-use language to retrieve and query data
    - Excellent way of representing/working with sets of data
    - Widely used to store data for web applications
  + Cons:
    - Only 10GB of free storage for Microsoft SQL Server Express
    - Bad horizontal scalability
* MongoDB:
  + Pros:
    - Free
    - Fast data retrieval
  + Cons:
    - Does not offer an easy way to join tables, e.g. multi-dimensional data types
    - Not as secure as MySQL

**Visuals**

* D3.js:
  + Pros:
    - Uses HTML and CSS which most of us are familiar with
    - Capable of advanced visualization with complex data sets
    - Ease of use with the “fantastic user interaction support”
* Google Charts
  + Pros:
    - Highly flexible with a great set of developer tools
    - Built-in animation and user interaction controls

**Alerts**

* Twilio:
  + Pros:
    - Popular and easy to integrate to web applications
    - Better pricing than it’s competitors
    - Nice interface, easy to use
  + Cons:
    - Cost
* Plivo
  + Pros:
    - Added features to Twilio’s API so increased degree of functionality
  + Cons:
    - Cost

Personal Opinions

**Framework**:

**Ruby on Rails**: Main reason I would choose Ruby on Rails is because I want to learn the framework. It is widely used in industry and would be a great skill to have.

**Database**:

**MySQL**: For the data we want to represent, MySQL is a better choice to structure the data. Also, the added reliability/security is tempting even though I do not feel as though security is a huge issue in this project.

**Visuals**:

**D3.js**: I do not have much knowledge on visualization, but if this is based on HTML and CSS then it should be easy to pick up and use. My stance on this decision is not very strong.

**Alerts**:

**Twilio**: Twilio gets the job done and is easy to use. The only thing that would set another alert system before Twilio is if it was completely free. Unfortunately, I failed to find something completely free of charge.