# Non-Functional Requirements

**Accessibility**

Due to time constraints, the web app will not be able to assist the visually impaired. However, should enough work be finished ahead of schedule, a push can be made to make the site friendly for the visually impaired.

**Backup**

The current version control system being used is git with github operating as our remote repo host. This ensures we always have a backup of our code in case of crash on anyone else’s computer. As well, branches have been made to allocate work and streamline workflow. Each collaborator has a branch they are currently working on, the developer branch for everyone to push their latest code, and the master branch which is where we deploy our app.

**Dependancy on Other Parties**

This web app will not depend on any other parties, other than the tools to be used to build our app.

**Disaster Recovery**

The current disaster recovery plan is through our code being hosted on a remote repository. As well, our multi-branch system ensures that each collaborator has their own branch, a test branch for the group, and a master branch for app deployment. As well, we will ensure pull requests and code reviews are done before any branch merges are made.

**Documentation**

Doxygen will be extensively used to generate proper documentation on our back-end and ensure our API is simple to use. The front-end will feature comments relevant towards the ease of maintaining the front-end.

**Extensibility**

In terms of future growth, the web app could gain extra features. Features include expanding for uses outside of the US.

**Interoperability**

The web app will be designed to be working for Google Chrome usage at the least. Safari and Firefox full support will come second.

**Maintenance**

The project will be maintained by the collaborators after completion, until otherwise the life of the web app comes to an end.

**Robustness**

Currently, the web app will only be compliant for US-based results. This is due to the lack of large datasets available regarding comprehensive health records. As well, our web app will be able to specify the location of disease indicators based off of longitude and latitude, making it fairly robust when measuring disease indicators in the United States.

**Tools**

The nosophobic web app will preferably be built as a dynamic web app. As such, we will use Java as our back-end language, Tomcat server, and Spring to build our MVC. For the front-end, we will use JavaScript, HTML/CSS, bootstrap, jQuery, and Ajax.

**Testability**

We will incorporate back-end testing on our Java code through the use of junit. We will aim for 95-100% test coverage, and will employ many test suites to ensure our code works. As well after deployment, we can set up Google analytics to gather traffic data on our site. This data will be helpful in front-end unit testing and can contribute to any adjustments of our back-end.

**Usability**

The web app will be designed for ease of use with a simple user interface. This allows for any age group to use the website with little difficulty. How the data is displayed, such as through the use of charts and heat maps, can further help users understand the data.