Team 17 Sprint 2 Planning Document

Northrop Grumman Xetron Seismic Activity Map

Aaron Peters, Charles Cho, Zhihao Hu, Paul Lohmuller, Ben Whorley

**Sprint Overview**

**Overview:**

Now that our team is in our second sprint, we will be able to accomplish much more now that we are more familiar with the basic parts of our project. First, we will learn more about how to use Vaadin. The three main goals of this sprint are:

1. Finish the unfinished work from first sprint,

2. Implement more functions on our web page, and

3. Implement the database on our Amazon Web Services (AWS) instance so that it can be accessed from anywhere, not just on machines which have MySQL installed.

Below, we have described the next round of user stories that we will try to tackle. The estimated time for each task is an arbitrary number with a base of 1 and a max of 8, with the numbers in-between being Fibonacci numbers (1, 2, 3, 5, 8, 13, 21, 34, 55, 89).

**Scrum Master:** Aaron Peters

**Scrum Meeting Time:** 7PM T and R, 7PM Su

**Risks/Challenges:** Learning new function in Vaadin. Use the API from public sensor to access the information from them. Making sure git commands are understood and executed successfully.

**Current Sprint Detail**

**User Story:** As a user, I want to be able to access this system from any web browser on any device.

|  |  |  |
| --- | --- | --- |
| Task Description | Estimated Time | Owner |
| Research the required methods for launching our Vaadin project onto a public IP address rather than launching it onto our localhost server. | 8 | Aaron |
| Once the necessary methods have been researched, implement our Vaadin project to launch onto our Amazon IP address | 2 | Aaron |
| Try to connect to this public server from different devices (i.e. phone, laptop, different WiFi connections) | 5 | Aaron |

**Acceptance Criteria:** If this user story is implemented successfully, a tester should be able to type in our public IP address and view our program from any machine.

**User Story:** As a user, I want to have an easy-to-use interface for viewing database records of seismic events.

|  |  |  |
| --- | --- | --- |
| Task Description | Estimated Time | Owner |
| Create a button on the seismic events view page. | 2 | Aaron |
| Make the button functional to show some information of seismic events. | 8 | Charles |
| Implement the functionality of grabbing information from the database and then displaying it on the screen. | 5 | Charles |
| The information window can be closed if user wants it to. | 2 | Aaron |
| Make sure the information window is always showing the latest information from the database. I.e., if the database gets updated and the user clicks on the button again, they should see the updated information. | 2 | Charles |

**Acceptance Criteria:** If this user story is implemented successfully, a tester should see a button showing in the seismic events view page and after clicking it, they should see an information window describing all events that have taken place in the US over the past 20 years.

**User Story:** As a developer, I want to read seismic data from public sensors so that I can begin to gather real seismic data instead of using mock data.

|  |  |  |
| --- | --- | --- |
| Task Description | Estimated Time | Owner |
| We need to first look up any APIs that are available from public seismic sensor sources, like the USGS. | 3 | Charles |
| Write methods that can connect to these APIs which will pull the data associated with this seismic sensor. | 8 | Zhihao |
| These sensors will likely return a large amount of information, so we will need to describe what attributes we are interested in storing. | 5 | Zhihao |
| Add the information that we deem important to each sensor listed on the sensor view map page. | 8 | Paul |
| When the user clicks on a sensor, the information window that pops up should show this important information related to this sensor. | 5 | Paul |

**Acceptance Criteria:** If this user story is implemented successfully, a tester should be able to open the sensor view map page, click on a certain sensor, and see the relevant information related to that sensor in an information window that will pop up above the sensor marker on the map.

**User Story:** As a developer, I want to add a slider bar to the seismic event map so that the users will eventually be able to view only events of certain intensity.

|  |  |  |
| --- | --- | --- |
| Task Description | Estimated Time | Owner |
| Make a slider bar in the seismic events view page. | 5 | Ben |
| Add descriptions to the slider bar so that the user understands exactly what they are changing when moving the slider. | 3 | Paul |
| Add a listener to it so we can respond after move the slider bar. | 5 | Ben |

**Acceptance Criteria:** If this user story is implemented successfully, a test should see a slider bar which will be used to manage which events are viewable, and at least some hard coding to demonstrate how its functionality should work when it’s completed will be present in the source code.

**User Story:** As a developer, I want to connect our database to our AWS instance so that our program can access it from any device.

|  |  |  |
| --- | --- | --- |
| Task Description | Estimated Time | Owner |
| Research the required XML code needed to connect our Java Database Connectivity tool (our database tool) to an instance of MySQL on AWS instead of connecting to a local MySQL, which is what we currently do. | 8 | Ben |
| After the research is complete, modify the XML code accordingly. | 2 | Zhihao |
| Test the solution by trying to connect to the database from a machine that doesn’t have MySQL installed on it. | 2 | Zhihao |

**Acceptance Criteria:** If this user story is implemented successfully, a tester will be able to view our program in their browser without MySQL installed on their system, and view all parts of our program that require the stored data in our database in order to run.

**Remaining Backlog**

**Functional**

1. As a user, I want to be able to see the epicenter of an event.
2. As a user, I want to see the radius of an event on the map.
3. As a developer, I want to implement a system that categorizes seismic events based on data it produces so that users can see what kind of event is probably occurring.
4. As a user, I want to be able to see what areas on the map are least active seismically and which areas are most active seismically so that I can determine the best areas to test new sensors.
5. As a user, I want to be able to select and deselect certain seismic sensors so that I can view certain regions of seismic activity.
6. As a user, I want to be able to filter the seismic activity threshold to visualize the activity in different colors.

**Non-Functional**

1. As a developer, I want to calculate the epicenter of the seismic data.
2. As a developer, I want to calculate the predicted radius that the seismic event will affect.
3. *As a developer, I want to implement security features for our seismic database so that we can prevent unauthorized access (If Time Permitting)*
4. *As a developer, I want to standardize the information for the user about each seismic event in a uniform manner (If Time Permitting)*