

# CS1530 SPRINT-3 DELIVERABLE

Project: Webfreesurfer  
Group: Open Science Grid - 2

Due Date: **27 OCTOBER 2015**

## Group Member

William McKibbin

Phillip Washy

Robert Colleran

Nathan Dorman

Zane Hernandez

## Role

Project Manager

QA/Dev

Developer

Developer

QA/Dev

## User Stories (<https://github.com/wmckibbin/webfreesurfer>)

**One user story was completed during this sprint:**

1c. As a **user**, I want to be able create an account on the website, and modify my information after my account is created so I can keep my account up to date and secure.

**A large amount of work was put into the following user stories, but they were unable to be completed and merged before the sprint ended:**

1b. As an **administrator**, I want to be able to access and modify account information of users within the site so I can support and audit the users of the site.

1d. As an **administrator**, I want users to have to confirm their email address via a dynamically generated link upon creating an account.

2a. As a **user**, I want to be able to submit a freesurfer job in a few clicks so I don't have to worry about the technical aspects of running the job on the OSG.

2b. As an **administrator**, I want job data to be stored in a Postgre SQL database with a schema of my choosing so a record of past and present jobs can be monitored and audited.

**The following user stories remain in the backlog with little/no work done on them:**

2c. As an **administrator**, I want brain scan data to be anonymized and I want the user to confirm this so no confidential medical records are exposed.

2d. As a **developer** I want want the freesurfer jobs to be submitted, monitored, and deleted via a RESTful Api call to a middleware named OSGConnect

2e. As a **power user** I want to be able to use a CLI to submit, monitor and delete jobs so I can test the software and have increased functionality.

2f. As a **user**, I want to be notified via Email when my job has completed and be provided a download link to view the results so I can have better awareness and access to the results.

2g. As a **user**, I want to be able to monitor the progress of a job and be provided statistics about its runtime so I can be aware about the status of my job.

## **Accomplishments**

### **Asynchronous development**

Our team was able to more effectively utilize git to do asynchronous development. Devs were able to branch off with their tasks and develop features independently of one another. This also encouraged our team to develop from a modular fashion. Each developer was able to create separate classes to satisfy the particular task they were assigned. In previous sprints, we had trouble with keeping our code separated and encapsulated which made it difficult to contribute to the same codebase. We acknowledged that functionality like database operations, cryptography, and email messaging needed to be encapsulated and made efforts to push this functionality out of our main methods and into classes.

### **Communication**

Our team was able to make moderate improvements in our organization by making better use of our slack channel. We were able to utilize our entire team on separate aspects of the project simultaneously, something we've struggled with in the past.

We need to improve on our face to face time. We thought that we could get away with just online communication in addition to our scrum meetings after class, but we need more human interaction to coordinate our long term efforts. The customer has expressed that the software be extensible and readable, aspects all good software should have, but especially important in our case. We are going to meet for a group development session once per week on Thursday nights and expect to have good attendance.

### **Agility**

Our team needs to focus on making our software bug free and operational at the end of each sprint. We have been focusing primarily on implementation and getting features coded. We need to switch gears and focus on creating the highest quality software that can be merged and integrated.

### **TDD**

Our team was able to make improvements to our testing and QA approach. We defined tests before and after writing code and were able to detect issues before they were merged into the main codebase. We plan to utilize TDD even more next sprint as we begin integrating code that interfaces with the customer's middleware.

### **Documentation**

This sprint one of our efforts was to create javadocs per Prof. Laboon's suggestion. We plan to continue ensure we have well documented code for the

remainder of project. We need to ensure that the software we write can be managed by another group of developers.

## **Customer Interactions**

### **Communications with the Customer**

Our project manager reached out to our two point of contacts to establish a timeline of when their middleware server would be available for testing and integration. We were able to get a rough timeline that a testing version of the middleware would be available to us soon.

We also were able to clarify certain aspects of how the middleware by speaking with the technical contact who was added to our slack channel. He gave us a detailed specification on how the users would be authenticated via a cryptographic token generated from unix time and a shared secret.

### **Wasn't ready to demo yet**

Our project was not able to be demoed this sprint. Our software was not yet ready to be shown. We did not finish our front end or user creation aspects until the later days of the sprint. The customer is primarily concerned with the job creation functionality of site which is not yet operational on our end or the technical contact's middleware. We expect fully to show the software this sprint and give the customer a worthy demo. We recognized that this was the best decision to make in order to maintain customer confidence and interest.

### **Demo coming soon**

We plan to demo our new bootstrapped front end with a semi-operational MVP that implements user creation, modification, and administration as well as job creation, submission and monitoring. Depending on the customer's middleware we will begin integration tests with their software ensuring that we are properly using their RESTful API otherwise we will put in dummy values for the demo.

### **Reconnecting**

Next Sprint we plan to reach out to the customer more frequently. As of now interactions with the customer were not as necessary since they knew exactly what they wanted and how they wanted it implemented. As we move forward out of the base functionality we will need their feedback to more to correctly satisfy their needs.

## **Defects**

### **Password Change**

A defect was discovered with the user account modification which prevented a user from logging in after changing their password. The defect was discovered through simple user testing of the feature. Our passwords are created by using a series of hashes that take produce strings as output. One thought was there was an encoding problem from switching from byte arrays to strings and back again, but the answer is still unclear. The bug is ongoing into the next sprint.

### **Account Creation**

Account Creation is experiencing an intermittent bug that causes the account to fail. It was discovered by our devs during add user tests. The bug is ongoing and the cause is unknown. We plan to create many users at once until the problem is reproduced and we can analyze the issue.