**Planetarium Version 1.0 Team Sprint Retrospective 2** (11/18 - 11/27)

**What went well:**

* Correcting step implementations from previous sprint
  + Behavior of tests now function as intended compared to the previous sprint
  + Learned more about Selenium’s WebDriver API such as using WebDriverWaits, ExpectedConditions, and TimeoutExceptions
* Unit testing the DAO and service layer
  + Using Mockito to mock the DAO in the service to hit other conditions in the method
    - Learned about Mockito.mock() and Mockito.when().thenReturn()
  + Learned more about when and where to use Mockito (such as using Mockito on a DAO object to help isolate the Service object to perform unit testing)
  + Learned about using JUnit’s Assert to validate expected conditions (assertThrows, assertEquals…)
* API Testing
  + Setting up a reset function to ensure no test-to-test variance
  + Automating logins
  + Learned more about using Postman such as creating pre/post requests for setting up and testing for each request and creating collections to run tests as a single unit
  + Incorporated functional programming for testing via JavaScript
* We reached the MVP in a shorter time frame (8 business days) compared to previous sprints
* Peer programming for integration tests, specifically with Postman’s Pre-Requests
* Teamwork and communication as we worked with each other whenever issues arise on our individual ends and hosted meetings for current progress

**What can be improved:**

* The Jira board could be better organized a bit more
* Better time management as we didn’t follow the Time Schedule as closely as we did from the previous sprint
  + Phase 3 took longer than what was expected on the schedule
* We were not as keen on test documentation compared to the previous sprint
* The unit tests can be expanded to run to obtain 100% code converge for the methods being covered (determining unreachable code)
* More peer programming could have been used to help each team member to be on the right track (such as learning about using Mockito earlier)
* Using a shared workspace to hold all collections of the API Testing
* To expand and fix features for Postman tests, such as learning how to analyze and test cookies generated from requests (i.e. JSESSIONID cookie)

**What work needs to be done next sprint:**

* Setting up and running Jenkins to automate build and deployment process
* Integrating the DevOps pipeline into Planetarium
* Integrating AWS features such as RDS and S3 for the database and storage of images
* Correcting defects caught in testing and continuing development for Planetarium
* Unit and integration tests for all functionality in the backend, not just use case methods
* Performance testing + exploratory testing + error guess testing
* To incorporate Java docs into our code to help outsiders/team members to understand what each test/step is performing (especially for step implementation)
* To incorporate cookie/session tests during API testing
* Perform future Acceptance testing after defects are corrected