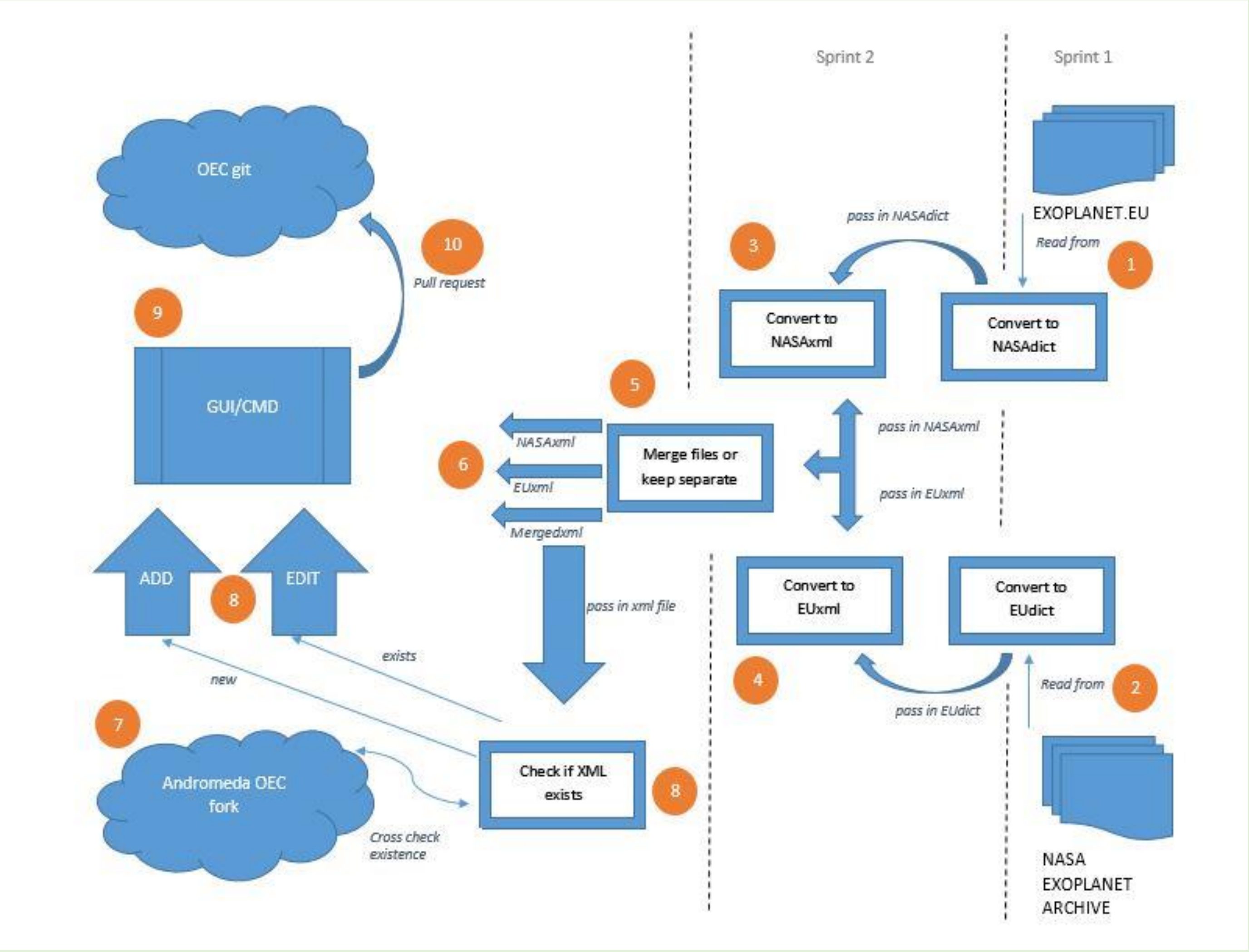
Team 15 – Andromeda

Deliverable 3: Report

[Sprint 1]

Table of Contents:

System Design



Subsystems:

1. Read from Exoplanet.eu and convert exoplanet data into a base dictionary
2. Read from NASA EXOPLANET ARCHIVE and convert exoplanet data into a base dictionary
3. Convert the base Eu dictionary to valid OEC xml file
4. Convert the base NASA dictionary to valid OEC xml file
5. Merges XML files if they refer to the same system, or passes two different if they are not of the same system
6. Return valid OEC xml file to compare with subsystem 7 through subsystem 8
7. Fork the OEC database
8. Compare and denote if files from subsystem 6 exist in subsystem 7, thereby denoting them as an edit to an existing file or a new addition
9. The subsystem representing the backend and frontend of the CMD or GUI user interface, giving user access to designated features
10. If the user accepts an add or an edit create the corresponding pull request to the OEC

Completed Components of the System

Sprint 1

- Read from Exoplanet.eu and convert exoplanet data into a base dictionary
- Read from NASA EXOPLANET ARCHIVE and convert exoplanet data into a base dictionary
- Subsystem 1 and 2 demo in driver.py

Product Backlog

Cost: DH (Developer Hours)
Priority: 1 – lowest, 5 – highest

- **User Story # 1:** As Professor Smith, the client, I want the application to run on Unix environment by using software and packages that are UNIX compatible
  - Cost: 0 DH & Priority: 5

- **User Story #2:** As Professor Smith, the client, I want the application to fetch data from the NASA and EU catalogues and I want it to be organized by each star system.
  - Cost: 5 DH & Priority: 5

- **User Story #3:** As Professor Smith, the client, I would like the application to organize the retrieved data by each star system.
  - Cost: 25 DH & Priority: 5

- **User Story #4:** As Professor Smith, the client, I would like the data that can representable in the OEC to be retrieved from the two catalogues.
  - Cost: 10 DH & Priority: 5

- **User Story #5**: As Professor Smith, the client, I want the application to produce data files in XML format that are usable as OEC XML files.
  - Cost: 12 DH & Priority: 4

- **User Story #6:** As Professor Smith, the client, I want the application to use the generated OEC compatible XML files and check if the data already exists in the OEC repository and report an edit or new addition respectively.
  - Cost: 10 DH & Priority: 3

- **User Story #7:** As Professor Smith, the client, I want the application to work on a fork version of the OEC master repository on GitHub.
  - Cost: 9 DH & Priority: 4

- **User Story #8:** As Professor Smith, the client, I want the application to receive daily updates, if any, from the NASA and EU catalogues.
  - Cost: 7 DH & Priority: 3

- **User Story #9:** As Professor Smith, the client, I would like to be notified of all updates available when I log into the application
  - Cost: 13 DH & Priority: 3

- **User Story #10:** As Professor Smith, the client, I would to view the log of changes where I can analyze and approve of the generated updates.
  - Cost: 10 DH & Priority: 2

- **User Story #11:** As Alice, the graduate student, I want the application to let me view the permissions for updates and the log.
  - Cost: 7 DH & Priority: 2

- **User Story #12:** As Professor Smith, the client, I would like the application to allow human intervention by allowing myself to manually edit errors by modifying the updates directly brought through by the data from the other catalogues
  - Cost: 18 DH & Priority: 3

- **User Story #13:** As Professor Smith, the client, I would like the application to have a log of all changes made to my repository where I can select a certain date to see the changes made in each update on that day.
  - Cost: 10 DH & Priority: 2

Release Plan

First System Release: October 24th 2016:

- User Stories to be implemented in the first sprint:
  - User Story #1 to #3

Second System Release: November 14th 2016:

- User Stories to be implemented in the second sprint:
  - User Story #4 to #8

Third System Release: December 1st 2016:

- User Stories to be implemented in the third sprint:
  - User Story #9 to # 13

Sprint Plan

- Each sprint will be 10 days long, which is approximately 2 working weeks.
- This is based around the deliverable deadlines.
- This sprint will end on October 24th 2016, the day of our first release.

Sprint Backlog for Sprint #1:

[#1] As Professor Smith, the client, I want the application to run on Unix environment by using software and packages that are UNIX compatible [Cost: 0 DH & Priority: 5]

Tasks:

1. Choose a language that is compatible with UNIX system
2. Use open source packages made for UNIX system
3. Verify that all created code is runnable on the lab machines

[#2] As Professor Smith, the client, I want the application to fetch data from different catalogues and update the OEC. [Cost: 5 DH, Priority :5]

Tasks:

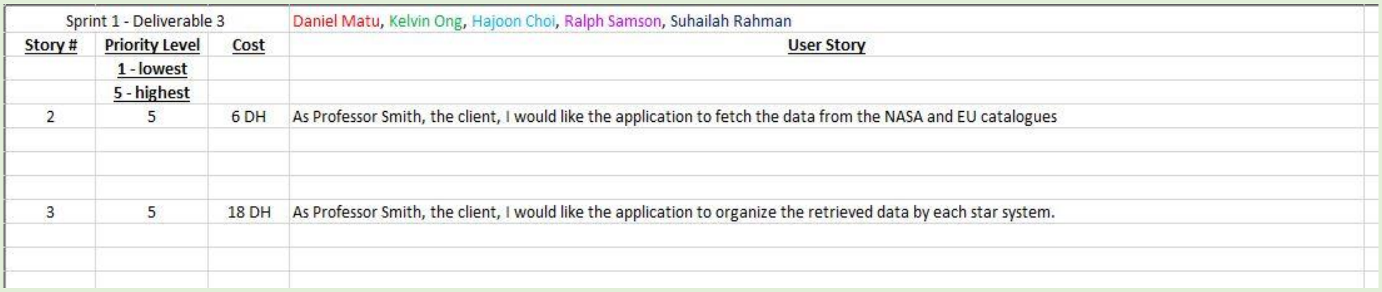
1. Download the data from both EU catalogue and NASA archive in CSV format
   - Done by Daniel and Suhailah
2. Initialize dictionaries that stores the planet information for both of the catalogue
   - The first row in the CSV is the column headings and they are the keys for the dictionary
   - From then, each row is a specific list which contains the planet data and it is linked to the respective key.
   - Done by Daniel and Suhailah

[#3] As Professor Smith, the client, I would like the application to organize the retrieved data by each star system. [Cost: 25 DH, Priority: 5]

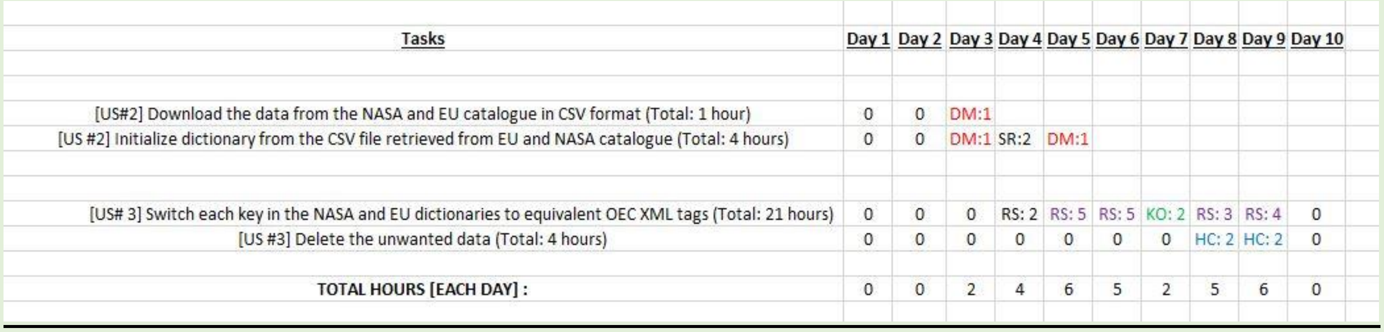
Tasks:

1. Switch each key in the dictionaries to equivalent OEC XML tags
   - Done by Ralph and Kelvin
2. Delete the unwanted data
   - If the data is still in the dictionary that means it cannot be transferred to the OEC, so we can delete that data.
   - Done by Hajoon

Iteration Plan:

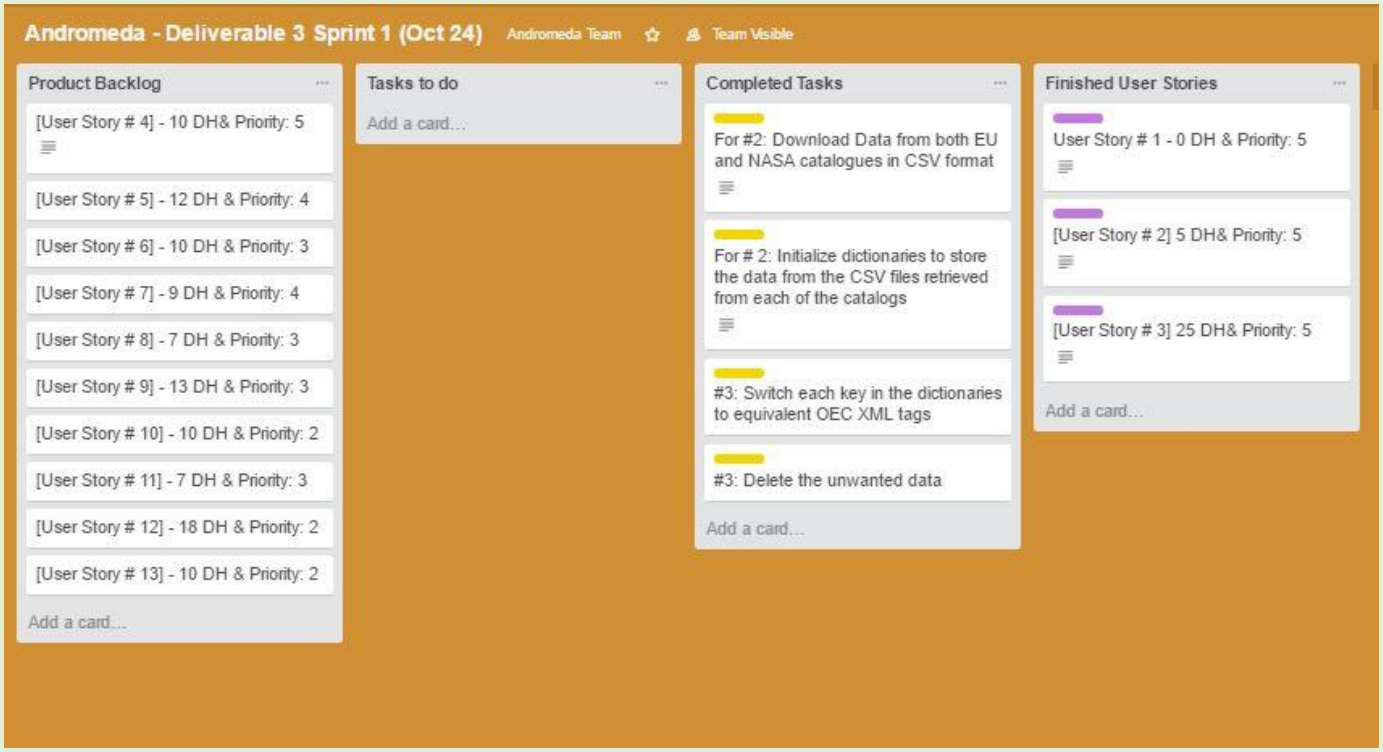


Sprint 1 - Deliverable 3			Daniel Matu, Kelvin Ong, Hajoon Choi, Ralph Samson, Suhailah Rahman
Story #	**Priority Level** **1 - lowest** **5 - highest**	**Cost**	**User Story**
2	5	6 DH	As Professor Smith, the client, I would like the application to fetch the data from the NASA and EU catalogues
3	5	18 DH	As Professor Smith, the client, I would like the application to organize the retrieved data by each star system.



Tasks	**Day 1**	**Day 2**	**Day 3**	**Day 4**	**Day 5**	**Day 6**	**Day 7**	**Day 8**	**Day 9**	**Day 10**
[US#2] Download the data from the NASA and EU catalogue in CSV format (Total: 1 hour)	0	0	DM:1							
[US #2] Initialize dictionary from the CSV file retrieved from EU and NASA catalogue (Total: 4 hours)	0	0	DM:1	SR:2	DM:1					
[US# 3] Switch each key in the NASA and EU dictionaries to equivalent OEC XML tags (Total: 21 hours)	0	0	0	RS: 2	RS: 5	RS: 5	KO: 2	RS: 3	RS: 4	0
[US #3] Delete the unwanted data (Total: 4 hours)	0	0	0	0	0	0	0	HC: 2	HC: 2	0
TOTAL HOURS [EACH DAY] :	0	0	2	4	6	5	2	5	6	0

Task Board:



Andromeda - Deliverable 3 Sprint 1 (Oct 24)

Product Backlog

- [User Story # 4] - 10 DH& Priority: 5
- [User Story # 5] - 12 DH & Priority: 4
- [User Story # 6] - 10 DH & Priority: 3
- [User Story # 7] - 9 DH & Priority: 4
- [User Story # 8] - 7 DH & Priority: 3
- [User Story # 9] - 13 DH & Priority: 3
- [User Story # 10] - 10 DH & Priority: 2
- [User Story # 11] - 7 DH & Priority: 3
- [User Story # 12] - 18 DH & Priority: 2
- [User Story # 13] - 10 DH & Priority: 2

Add a card…

Tasks to do

Add a card…

Completed Tasks

- For #2: Download Data from both EU and NASA catalogues in CSV format
- For # 2: Initialize dictionaries to store the data from the CSV files retrieved from each of the catalogs
- #3: Switch each key in the dictionaries to equivalent OEC XML tags
- #3: Delete the unwanted data

Add a card…

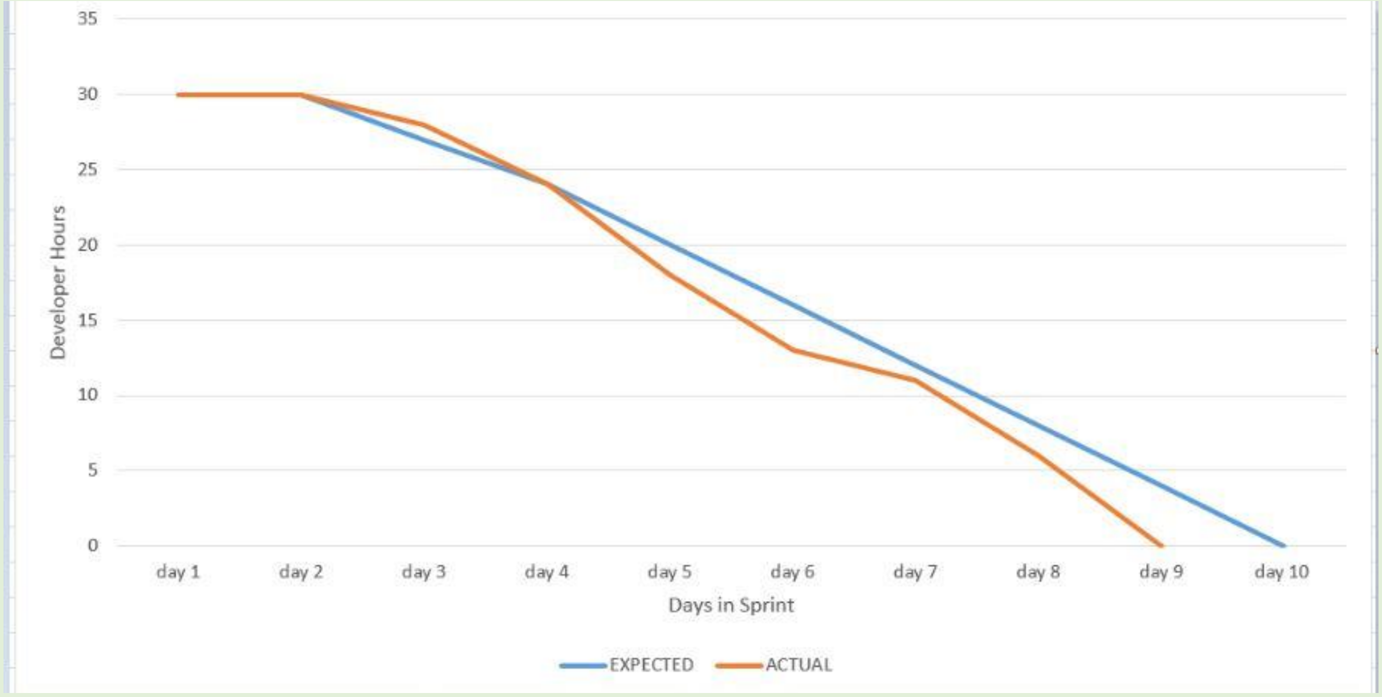
Finished User Stories

- User Story # 1 - 0 DH & Priority: 5
- [User Story # 2] 5 DH& Priority: 5
- [User Story # 3] 25 DH& Priority: 5

Add a card…

Burn-down Chart:

Sprint 1



Brief Overview

- **Transition from Deliverable 2 to Deliverable 3:**
  - We discussed the persona, Professor Smith and decided to add another persona (Alice Brown) from the feedback given by our TA.
  - With the new persona added, we came up with user stories for Alice.
  - We added new user stories to include features like allowing Smith to modify updates before issuing himself a pull request through our application.
  - We added up the costs and priorities of each user stories
  - Sprint Plan – we decided which user stories we were going to implement in the first sprint and came up with tasks for it.
    - Then we assigned tasks to each member.
- **Estimated Project Velocity:** 30
- **Actual Project Velocity:** 30
- **Did we follow with our plan?**
  - The first user story followed according to plan.
  - While implementing the tasks for the second user story, it took a little longer than expected because we realized that we had to individually parse through the entire OEC tags. We also had to make sure the column headers from both EU and NASA catalogues are the same in the OEC tags, so we had to compare the column headers.
  - We re-planned at the end of the sprint in terms of the system design.