Deliverable 4

Team 10

Vasili Skurydzin
Eric Papagiannis
Albion Fung
Tony Wu
Jerry Cheng

Table of Contents:

System Components Design ……..…..... 2

System Components and Their Description / Role ……..……. 2

Changes Made Since Last Deliverable ……. 3

Description Of System Validation Activities ……..…….…. 4

Product Backlog ……..…….…….. 5

Release Plan (Revised) ..…….. 9

Sprint 2 Backlog (Program Version 2) …… 10

Sprint 3 Backlog (Program Version 2) …..…….. 12

Sprint 4 Backlog (Program Version 2) ……. 14

Brief Overview Of Project Progress ……...…….….. 16

Our Planned Project Velocity ……..……...……... 18

Our Planned Project Velocity ……..……...…….. 18

Sprint 4 Planned Iteration Plan …… 18

Snapshots ……..……..……..... 19

System Components Design:

Please refer to the PDF document “System Component Design” inside Deliverable 4 folder for the UML diagram.

System Components and Their Description / Role:

Driver

- Driver runs the program by parsing command line arguments and calling related action that it entails

apiGet

- Retrieves data from nasa and exoplanet eu through get requests

XML_data_parser

- Downloads data from the Open Exoplanet Catalogue and parses it into PlanetaryObjects

CSV_data_parser and UnitConverter:

- UnitConverter (to be referred to as UC) is a nested class of CSV_data_parser (to be referred to as CDP), sicne we are converting exoplanet.eu and NASA’s units into OEC’s, no other class need to use it.
- CDP reads a CSV file given the name of the file, the source (‘eu’ or ‘nasa’ dependent on where it was retrieved) and parses it into planet objects and star objects. It can be parsed into a list or a dictionary, dependent on which function call is selected. Desired tags can also be manually selected.
- UC takes the field, name of field and the source it came from and converts the given data to OEC’s units. Note that the unit converter does not differentiate between BJD, MJD and JD due to insufficient information provided from exoplanet.eu and NASA (the conversion is not trivial).

Planetary_obect

- An object which represents more or less an interface to store data for planetary objects, which are systems, stars, or planets,and what each child class behavior is

System

- Child class of PlanetaryObject, stores data related to systems, and has references to the stars it has

Star

- Child class of PlanetaryObject, stores data related to stars, and has a references to the system it is in, and planets it has

Planet

- Child class of PlanetaryObject, stores data related to planets, and has references to the planet it is in

Comparator

- Compares planetary objects and based on differing data, provide proposed changes

Proposed_change

- Stores the conflicting data after the result of the comparators comparisons.

gitClone

- Clones repository and modifies the xml that the proposed change is in, then provides a pull request

Changes made since last deliverable:

After the due date of deliverable 3 we have made minor modifications to the way the PlanetaryObjects are compared in Comparator.py and we changed our strategy regarding generating the updated XML files to be pushed to the github database of the Open Exoplanet Catalogue.

We also changed our approach to pushing updated information to the client's database - we have implemented github branching / pull requests using 3-rd party library, "HUB" (hub.github.com), which makes our algorithm easier and more scalable.

Overall, the high level design of our system did not change since Deliverable 3.

Description Of System Validation Activities:

Goal Analysis and Static Verification:

In order to validate our design, we contacted the client on multiple occasions in order to ensure our understanding of the user requirements is correct. After formulating the set of user requirements and product backlog we moved on to developing the system design that would incorporate the simplicity and basic functionality without requiring the user to read long manuals and study the intricate workings of our application.

Unit-Testing:

In order to validate our design we wrote a comprehensive test suite to test all components of our application in order to ensure every single component is functioning as intended.

Program Inspections:

Throughout the development process all members of our team were continuously checking code written by everyone else, bringing up any problems discovered along with any possible suggestions. This helped us to resolve numerous problems with compatibility due to third party libraries in a quick manner, as well as prevent further challenges and ensure team progress.

Integration Testing:

We continued our validation activities with a complete set of integration tests, created to ensure our program components, merged together work as planned and do not produce erratic or undefined behaviour.

Function Testing:

At the end of our validation activities we performed the comprehensive function testing, in order to make sure that, from the point of view of the client, the program was accomplishing its tasks, and is easy to use. We performed the shoe test very last, and our program still performed gracefully, as always.

Acceptance Testing:

In near future we would like to provide a demonstration of our application to the client, in order to make sure our design meets his requirements and to get as much feedback as we can so that we can use the time that is left to improve the functionality of our program.

You can find our tests in the “test” folder.

Product Backlog:(Current version: 2)

For brief overview of project progress see page 15

Priority Scale:

1 - Most Important
2 -
3 -
4 -
5 - Least Important

1 Story Point (sp) == 1 Developer Hour

USER STORIES: (Some priorities have changed since Del. 3)

1. As Hanz Hanson (Exoplanet research professor), I want to be able to initiate an update at any time which prompts the program to accumulate planet statistics information from the Open Exoplanet Catalogue on a planet-by-planet basis as one data set.
   Priority: 1, sp: 8

2. As Hanz Hanson (Exoplanet research professor),during the update I want the program to also accumulate planet statistics from the target catalogues(NASA, exoplanet.eu) as other separate data sets, including only the information fields that are present in Open Exoplanet Catalogue and omitting information not present there.
   Priority: 1, sp: 8

3. As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program to show all the differences between the two data sets compiled during the last update; for every difference I want to see the following relevant information: name of the planet, name of the star system, name of the catalogue the difference originated from (NASA or exoplanet.eu), name of the field that have been modified (ex: mass), the value of that field according to the origin catalogue (NASA or exoplanet.eu), the value of that field on the Open Exoplanet Catalogue.
   Priority: 1, sp: 20

4. As Hanz Hanson (Exoplanet research professor), I want the program to convert all units in other catalogues to the units used in Open Exoplanet Catalogue when comparing and presenting numerical values.

Priority: 2 sp: 10

5. As Hanz Hanson (Exoplanet research professor), when viewing proposed changes, I want to see them in a numbered list.
   Priority: 1, sp: 3

6. As Hanz Hanson (Exoplanet research professor), if one of the updates discovers a planet entry in the target catalogue, while a planet with the same name is not present in the Open Source Exoplanet Catalogue, I want to the program to present it to me as a newly discovered planet with the rest of the proposed changes, providing the following information: planet name, star system name, name of the catalogue and the date the entry is posted.
   Priority: 1, sp: 4

7. As Hanz Hanson (Exoplanet research professor), if I suspect that a newly discovered planet is in fact an alternative name of an existing one, I want an option to associate this planet with an existing entry in Open Exoplanet Catalogue for the future updates; the proposed addition will be postponed for the current session.
   Priority: 3, sp: 5

8. As Hanz Hanson (Exoplanet research professor), I want to be able to “accept” any single proposed change presented (including added planets), prompting the program to update the information in the Open Exoplanet Catalogue by sending a single pull request to the OEC github database, containing the modified version of that same planet entry, with the field modified containing the updated value (the value that was different in the target catalogue), and all other fields remaining unchanged (after change has been accepted, it is deleted and not presented again).
   Priority: 1, sp: 15

9. As Hanz Hanson (Exoplanet research professor), I want to be able to “decline” any single change presented (including added planets), prompting the program to delete this single change after adding it to the “blacklist”, meaning same exact change will not be presented to me after future updates.
   Priority: 1, sp: 8

10. As Hanz Hanson (Exoplanet research professor), I want to be able to “postpone” any single change (including added planets), causing the program to delete this change but not add it to the “blacklist”, so that this same exact change will show up after a future update.
    Priority: 2, sp: 4

11. As Hanz Hanson (Exoplanet research professor), I want an option to clear the “blacklist”, which would cause the program to forget that some changes were previously declined and present them again in the future updates.
    Priority: 2, sp: 2

12. As Hanz Hanson (Exoplanet research professor), If I don't prompt to see the proposed changes immediately after the update, I want the program to store them for me to to access at a later time.
    Priority: 4, sp: 4

13. As Hanz Hanson (Exoplanet research professor), I want to be able to set the program to update automatically and set the time interval between updates.
    Priority: 2, sp: 5

14. As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program for its current status and get the following information: time of the last update, current auto-update settings, number of changes pending to be reviewed.
    Priority: 2, sp: 3

15. As Hanz Hanson (Exoplanet research professor), if there are changes pending to be reviewed by the time of the next update, I want the program to automatically postpone all of them.
    Priority: 2, sp: 2

16. As David Davidson (PhD Candidate), I want an option to accept or decline or postpone all currently pending changes at once.
    Priority: 4, sp: 2

17. As David Davidson (PhD Candidate), I want an option to see any single proposed change by itself, referring to it by its number in the list, omitting all other proposed changes, so that I do not get confused.
    Priority: 2, sp: 2

18. As David Davidson (PhD Candidate), I want an option to see a specific number of proposed changes at a time (for example 10 at a time), so I do not have to scroll through a lot of extra information to find what I need.
    Priority: 3, sp: 3

19. As David Davidson (PhD Candidate), I want an option to see a detailed user manual, describing the operation of the program.
    Priority: 5, sp: 4

20. As David Davidson (PhD Candidate), while I am reviewing the list of proposed changes I want an option to provide a name for a certain planet and to view / accept / decline / postpone all changes for that planet at once.
    Priority: 5, sp: 3

21. As David Davidson (PhD Candidate), while I am reviewing the list of proposed changes I want an option to view / accept / decline / postpone all changes to a certain star system, whose name I provide, at once.
    Priority: 5, sp: 3

22. As David Davidson (PhD Candidate), while I am reviewing the list of proposed changes I want an option to view / accept / decline / postpone all changes originating from one of the catalogues, whose name I provide, at once.
    Priority: 5, sp: 3

23. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor discrepancies between the numeric values across different catalogues, given the difference is within a set percentage tolerance, present by default.
    Priority: 2, sp: 9

24. As Hanz Hanson (Exoplanet research professor), I want to be able to set the tolerance of every numeric field present in the planet entries in the Open Exoplanet Catalogue to the value that I choose, on a field-by-field basis, so that the program uses the tolerance entered by me, as opposed to default one.
    Priority: 4, sp: 8

25. As Hanz Hanson (Exoplanet research professor), I want to be able to clear all tolerances added by me, so that the program reverts to the default tolerance value for every field.
    Priority: 5, sp: 2

26. As Hanz Hanson (Exoplanet research professor), if any single planet in the Open Exoplanet Catalogue database has more than one name associated with it (more than one name field in the planet entry), I want the program to recognize that the alternative name in a different catalogue (either NASA or exoplanet.eu) refers to an existing planet in the Open Exoplanet Catalogue, as opposed to a separate planet, so that the user(me) is not prompted to create duplicate entries for planets with alternative names.
    Priority: 3, sp: 5

27. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor differences in spelling of the names of planets and stars across the different catalogues, which includes case-insensitivity as well as any discrepancies in the punctuation or whitespace(number of spaces and tabs) in the planet name.

Priority: 3, sp: 5

Release Plan (Revised - Version: 2)

We have changed the release plan because after Sprint 3 our team was behind the schedule. We underestimated the time required to retrieve, parse and compare Planetary Object information efficiently and reliably. The new Release Plan allocates enough time to implement highly-technical aspects of the program, while balancing the priorities in the User Stories, to ensure the most important Stories are completed first. We try to give ourselves a sufficient time margin, in case implementation of any one component takes more time than expected.

Sprint 1: (October 17 - October 23)
- Develop the complete System Component Design;
- Implement command line operation of the program;
- Start implementing User Stories 1, 2

Sprint 2: (October 24 - October 30)
- Fully implement and test User Stories 1, 2, 3
- Start implementing User Stories 5, 17

Sprint 3: (October 31 - November 6)
- Fully implement and test user stories: 5, 6
- Start implementing user stories 4, 8

Sprint 4: (November 7 - November 13)
- Fully implement and test stories 4, 8, 10, 12, 14, 18, 26

Sprint 5: (November 14 - November 20)
- Fully implement stories 7, 9, 11, 13, 15

Sprint 6: (November 21 - November 27)
- Fully implement stories 16, 19, 20, 21, 22, 23, 24, 25, 27

Sprint Backlog for Sprint 2 (Program Version: 2)

For the second sprint, we decided to select user stories complete the remaining of user story 1, 2, (just testing) and the rest of user 3

User story 1: As Hanz Hanson (Exoplanet research professor), I want to be able to initiate an update at any time which prompts the program to accumulate planet statistics information from the Open Exoplanet Catalogue on a planet-by-planet basis as one data set. Priority: 1, sp: 8

1. Unit test downloader class (sp: 0.5)
2. Test option parsing of the main method. (sp: 0.5)

User story 2: As Hanz Hanson (Exoplanet research professor),during the update I want the program to also accumulate planet statistics from the target catalogues(NASA, exoplanet.eu) as other separate data sets, including only the information fields that are present in Open Exoplanet Catalogue and omitting information not present there. Priority: 1, sp: 8

1. Test parser for NASA. (sp: 1)
2. Test parser for exoplanet.eu (sp: 1)

User story 3: As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program to show all the differences between the two data sets compiled during the last update; for every difference I want to see the following relevant information: name of the planet, name of the star system, name of the catalogue the difference originated from (NASA or exoplanet.eu), name of the field that have been modified (ex: mass), the value of that field according to the origin catalogue (NASA or exoplanet.eu), the value of that field on the Open Exoplanet Catalogue. Priority: 1, sp: 20

1. Implement “Comparator” class that will process the datasets from Open Exoplanet Catalogue and outputs a list of Proposed Change objects. (sp: 12)
2. Implement “show changes” mode of operation. (sp: 2)
3. Test the Comparator class(sp: 3)

User story 5: As Hanz Hanson (Exoplanet research professor), when viewing proposed changes, I want to see them in a numbered list. Priority: 1, sp: 3

1. Implement sorting algorithm for a list of Proposed changes. (sp: 2)
2. Implement sorting of Proposed changes list before displaying it. (sp: 1)

User story 17: As David Davidson (PhD Candidate), I want an option to see any single proposed change by itself, referring to it by its number in the list, omitting all other proposed changes, so that I do not get confused. Priority: 2, sp: 2

1. Implement --show_number=# option in driver.py (sp: 1)
2. Implement functionality to display a single proposed change from a list (sp: 1)

Rough Plan of Who Does What and When:

During sprint 2:

(Approximately 5 story points per person per sprint)
Each user story is broken down into multiple tasks.

TASKS:

Vasili: finish proposed change class / start on comparator with Jerry
Eric: update XML parsing to comply with new data storage standards / test XML parsing / comment his code
Albion: finish NASA / eu parsing, test it
Jerry: start on comparator with V
Tony: test option parsing in driver.py / implement --show_all and --show_number=# options

Sprint Backlog for Sprint 3 (Program Version: 2)

In this sprint every team member needs to complete 7 story points.

User story 3:
As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program to show all the differences between the two data sets compiled during the last update; for every difference I want to see the following relevant information: name of the planet, name of the star system, name of the catalogue the difference originated from (NASA or exoplanet.eu), name of the field that have been modified (ex: mass), the value of that field according to the origin catalogue (NASA or exoplanet.eu), the value of that field on the Open Exoplanet Catalogue.
Priority: 1, sp: 3 left

- Test the Comparator class(sp: 3)

User Story 4:
As Hanz Hanson (Exoplanet research professor), I want the program to convert all units in other catalogues to the units used in Open Exoplanet Catalogue when comparing and presenting numerical values.
Priority: 2 sp: 10

- Create a config file with all the measurement units from NASA and Exoplanet. (sp: 1)
- Enable CSV parser to process all PlanetaryObjects created, converting all values to the same units as Open Exoplanet Catalogue is using. (sp: 7)
- Test unit conversion functionality of CSV parser. (sp: 2)

User story 5:
As Hanz Hanson (Exoplanet research professor), when viewing proposed changes, I want to see them in a numbered list. Priority: 1, sp: 3

- Implement sorting algorithm for a list of Proposed changes. (sp: 2)
- Implement sorting of Proposed changes list before displaying it. (sp: 1)

User Story 6:
As Hanz Hanson (Exoplanet research professor), if one of the updates discovers a planet entry in the target catalogue, while a planet with the same name is not present in the Open Source Exoplanet Catalogue, I want to the program to present it to me as a newly discovered planet

with the rest of the proposed changes, providing the following information: planet name, star system name, name of the catalogue and the date the entry is posted.
Priority: 1, sp: 4

- Enable comparator to to treat an added planet as a separate kind of proposed change (Addition obj.). (sp: 2)
- Test Addition mode of operation of Comparator. (sp: 1)
- Enable --showall method to display the Addition planet providing the following information: planet name, star system name, name of the catalogue and the date the entry is posted. (sp: 1)

User Story 8:
As Hanz Hanson (Exoplanet research professor), I want to be able to “accept” any single proposed change presented (including added planets), prompting the program to update the information in the Open Exoplanet Catalogue by sending a single pull request to the OEC github database, containing the modified version of that same planet entry, with the field modified containing the updated value (the value that was different in the target catalogue), and all other fields remaining unchanged (after change has been accepted, it is deleted and not presented again).
Priority: 1, sp: 15

- Enable driver to execute --accept “number” command. (sp: 1)
- Build up an updated version of the XML file in the same format as the ones on Open Exoplanet Catalogue for the change being accepted. (sp: 7)
- Use github api to send a pull request containing the updated XML file to the Open Exoplanet Catalogue database on github. (sp: 7)

Sprint Backlog for Sprint 4 (Program Version: 2)

In this sprint every team member needs to complete 7 story points.

User Stories to be implemented in this sprint:

User Story 4: (Partially implemented in sprint 3)
As Hanz Hanson (Exoplanet research professor), I want the program to convert all units in other catalogues to the units used in Open Exoplanet Catalogue when comparing and presenting numerical values.
Priority: 2 sp: 3

- Create a config file with all the measurement units from NASA and Exoplanet. (sp: 1)
- Test unit conversion functionality of CSV parser. (sp: 2)

User Story 8: (Partially implemented in sprint 3)
As Hanz Hanson (Exoplanet research professor), I want to be able to “accept” any single proposed change presented (including added planets), prompting the program to update the information in the Open Exoplanet Catalogue by sending a single pull request to the OEC github database, containing the modified version of that same planet entry, with the field modified containing the updated value (the value that was different in the target catalogue), and all other fields remaining unchanged (after change has been accepted, it is deleted and not presented again).
Priority: 1, sp: 14

- Build up an updated version of the XML file in the same format as the ones on Open Exoplanet Catalogue for the change being accepted. (sp: 7)
- Use github api to send a pull request containing the updated XML file to the Open Exoplanet Catalogue database on github. (sp: 7)

User Story 10:
As Hanz Hanson (Exoplanet research professor), I want to be able to “postpone” any single change (including added planets), causing the program to delete this change but not add it to the “blacklist”, so that this same exact change will show up after a future update.
Priority: 2, sp: 4

- Implement parsing of --postpone # command in driver.py (sp: 0.5)

- Implement the storage_manager class' functionality to retrieve the list of proposed changes from the memory, delete the single change whose number is given, and push the updated list of changes into the original storage location. (sp: 3)
- Test the functionality of storage_manager's "--postpone" mode of operation. (sp: 0.5)

User Story 12:
As Hanz Hanson (Exoplanet research professor), If I don't prompt to see the proposed changes immediately after the update, I want the program to store them for me to to access at a later time.
Priority: 4, sp: 4

- Research how to store variables between invocations of our program tool (sp: 1)
- Implement storage manager class, which includes functionality to store and retrieve variables after program has terminated (sp: 2)
- Test storage manager class (sp: 1)

User Story 14:
As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program for its current status and get the following information: time of the last update, current auto-update settings, number of changes pending to be reviewed.
Priority: 2, sp: 3

- Implement parsing of --status command in driver.py (sp: 0.5)
- Implement storage of relevant info: time of the last update, current auto-update settings, number of changes pending to be reviewed by writing the data to a config file (sp: 2)
- Test the file writing / reading component (sp: 0.5)

User Story 18:
As David Davidson (PhD Candidate), I want an option to see a specific number of proposed changes at a time (for example 10 at a time), so I do not have to scroll through a lot of extra information to find what I need.
Priority: 3, sp: 3

- Implement parsing of --showrange which takes an index argument (x to y) (sp: 1)
- Implement showing the portion of proposed changes from x to y if the indices are valid (sp: 2)

User Story 26:
As Hanz Hanson (Exoplanet research professor), if any single planet in the Open Exoplanet Catalogue database has more than one name associated with it (more than one name field in the planet entry), I want the program to recognize that the alternative name in a different catalogue (either NASA or exoplanet.eu) refers to an existing planet in the Open Exoplanet Catalogue, as opposed to a separate planet, so that the user(me) is not prompted to create duplicate entries for planets with alternative names.
Priority: 3, sp: 5

- Enable XML parser and CSV parser to associate alternative names to the same planetary object (sp: 3)
- Test this alternative name functionality (sp: 2)

Brief Overview Of Project Progress

Version 1:

Original Product Backlog (Pre-Deliverable 2); Program is planned as a console application with a possible GUI.

Version 2: (current)

After Deliverable 2 we have changed our design for a simpler program working much like a Unix command line tool, with just few simple command to use it (--status, --update, --showall, --accept, --decline, --postpone), which would be easy and intuitive to use. We decided against using graphical user interface, focusing on efficiency and simplicity instead. Sprint backlogs for sprint 1 to 4 reflect the evolution of our program.

Our project progressed smoothly from deliverable to deliverable. Initially, we started with a vague plan of how our program should look, but it wasn't at the quality that it is in the current deliverable. In order to improve our project plan, we consulted the Professor, Client and the TAs and put in the effort required to develop a carefully thought out and highly detailed Product Backlog, intended to remove all the confusion about how our application should look and behave, from the point of view of the end user. We went on to produce a highly-technical system component design, intended to resolve all the low-level details before the implementation phase of the project.

During our Product Backlog planning phase, we ditched the initial plan to make a console application with many different functions in favour of considerably simpler and less verbose command line tool, with very narrow and well-defined functionality, as a result of consulting the client and receiving his feedback on our demo.

By the time the deliverable 3 was due, our program was able to retrieve and parse the information from the target databases (NASA, exoplanet.eu), in order to get it into the form which is easy for us to analyze and compare. Although, after Sprint 2 we were behind the schedule, we have successfully built a solid foundation of our program, and removed the confusion we had about how to approach some technical aspects of coding, including which third party libraries to use, and how to approach the Unix compatibility requirement.

After Sprint 3 we were settled with our plan on using github api in order to branch the client's repo and send him pull request. What is left at that point is the implementation of information uploading part, more integration and function testing, and convenience features.

After Sprint 4 we completed the core functionality of our application (retrieval, parsing, comparison, and sending accepted modifications through pull-requests). All that remains is application features that are not crucial to the functionality, more thorough testing, and style improvements. During the last sprint (Sprint 4) we had to replan the internal workings of sending pull requests using github API, because we have found 3rd party library which is easier to use for our application.

Our Planned Project velocity:

25 Sp per Sprint for Sprints 1 and 2 and 35 sp per Sprint for Sprints 3 and 4.

Our Actual Project Velocity:

Ranges from 14 to 19 in the Sprints 1 to 3.

Sprint 4 Planned Iteration Plan:

S4	PLANNED						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
4	0	0	0	0	0	3T	0
8	2A + 5E	7E	0	0	0	0	0
10	0	0	0	0	0	0	4V
12	0	0	0	0	0	4V	0
14	0	0	0	0	0	3J	0
18	0	0	0	0	0	0	3E
26	0	0	0	0	0	5A	0

Snapshots:

References:

Taskboard:

Please go to this link if the Trello is unclear

https://trello.com/invite/b/wOCrfemm/7f4c47d9e326865948a20e594ae288f8/oec-updater

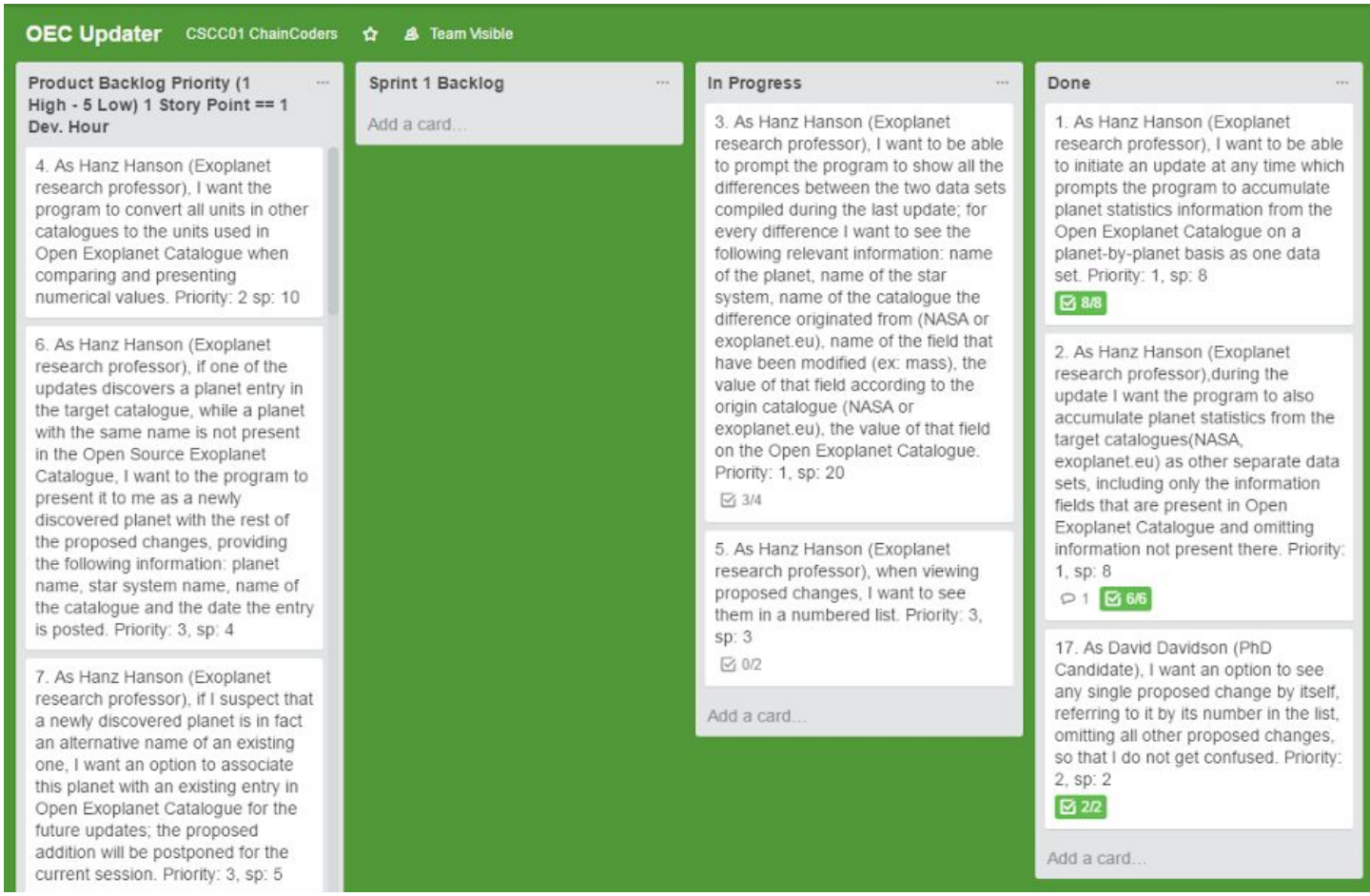
Burndown Chart and Iteration Plans:

Please go to this link if burndown chart is unclear

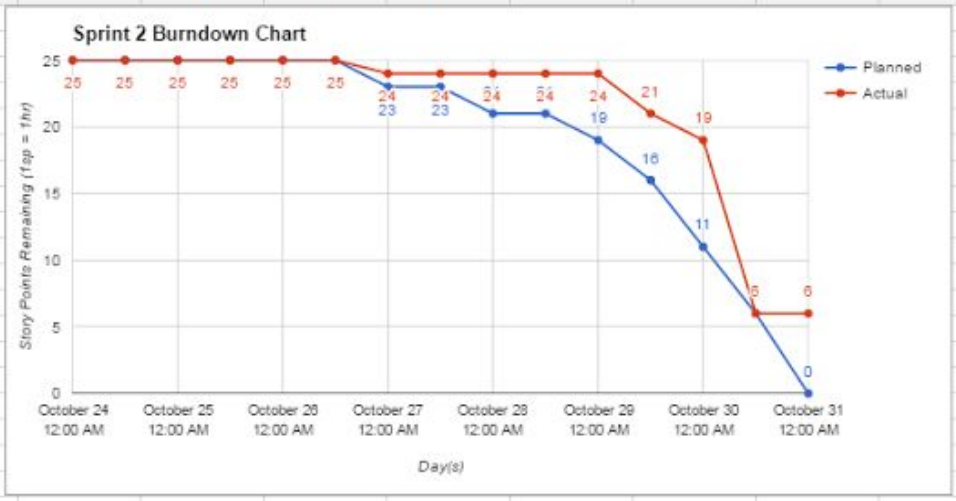
https://docs.google.com/spreadsheets/d/1D8aoThgJHv_h17ntJQxgFOzt2-ed_At3tg32SqxydJg/edit?usp=sharing

Note: the quality of the earlier sprints are lesser than those of sprint 4, as we contacted our TA to get feedback and incorporated it into the latest sprint. We cannot change history

Snapshots: Sprint 2

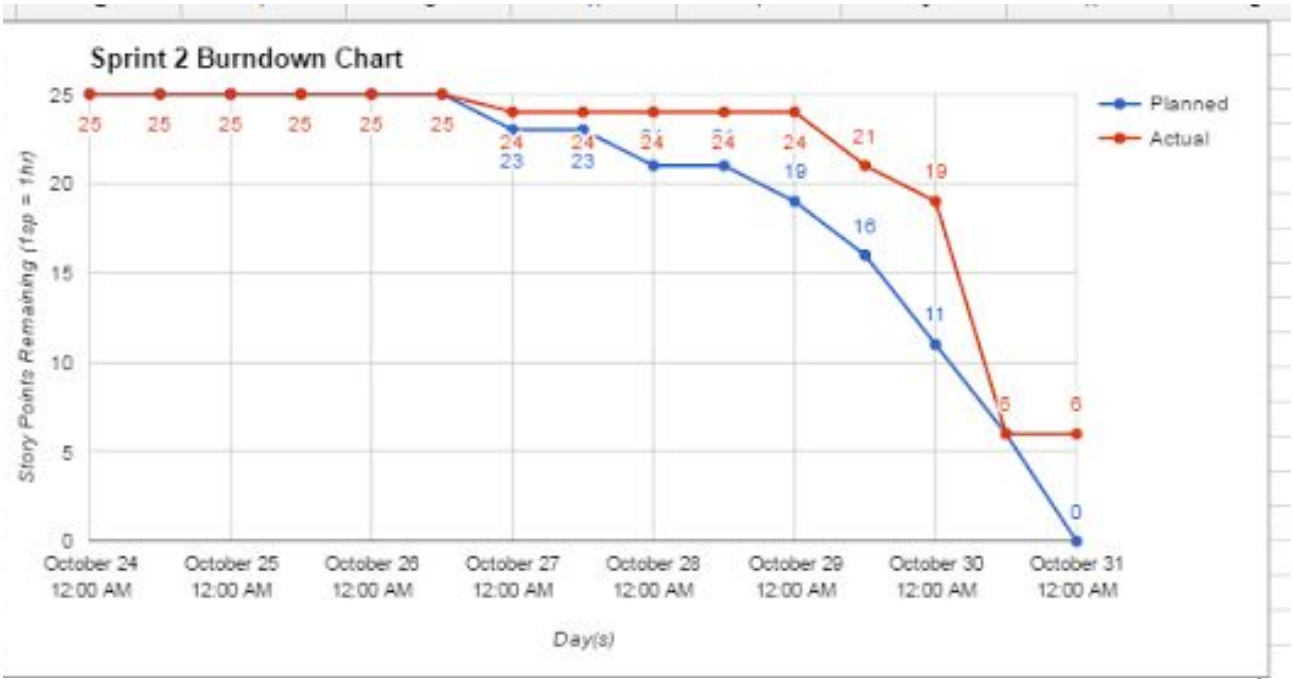


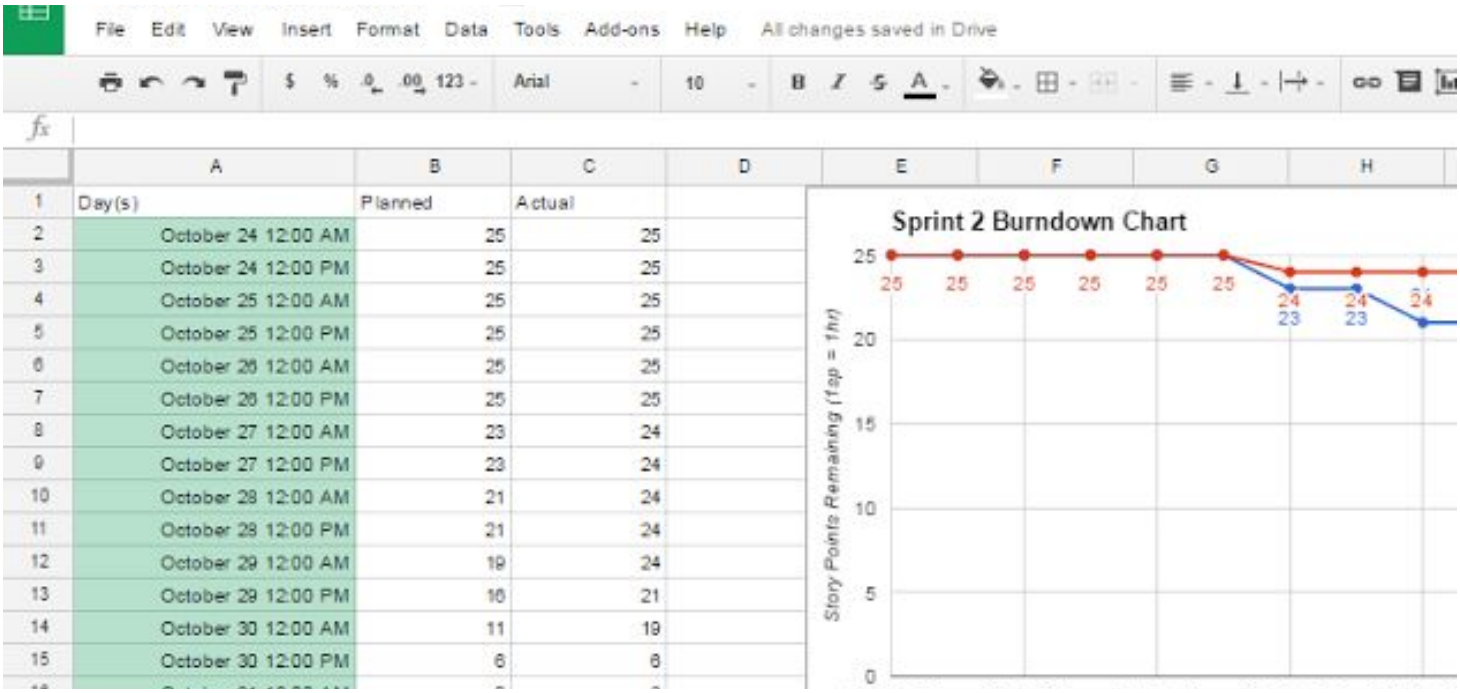
Day(s)	Planned	Actual
October 24 12:00 AM	25	25
October 24 12:00 PM	25	25
October 25 12:00 AM	25	25
October 25 12:00 PM	25	25
October 26 12:00 AM	25	25
October 26 12:00 PM	25	25
October 27 12:00 AM	23	24
October 27 12:00 PM	23	24
October 28 12:00 AM	21	24
October 28 12:00 PM	21	24
October 29 12:00 AM	19	24
October 29 12:00 PM	16	21
October 30 12:00 AM	11	19
October 30 12:00 PM	6	6
October 31 12:00 AM	0	6



S2	PLANNED						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
1	0	0	0	2E	0	0	0
2	0	0	2A	0	0	0	0
3	0	0	0	0	2V	3E + 5T	3A + 3J
5	0	0	0	0	0	0	3V
17	0	0	0	0	0	0	2J

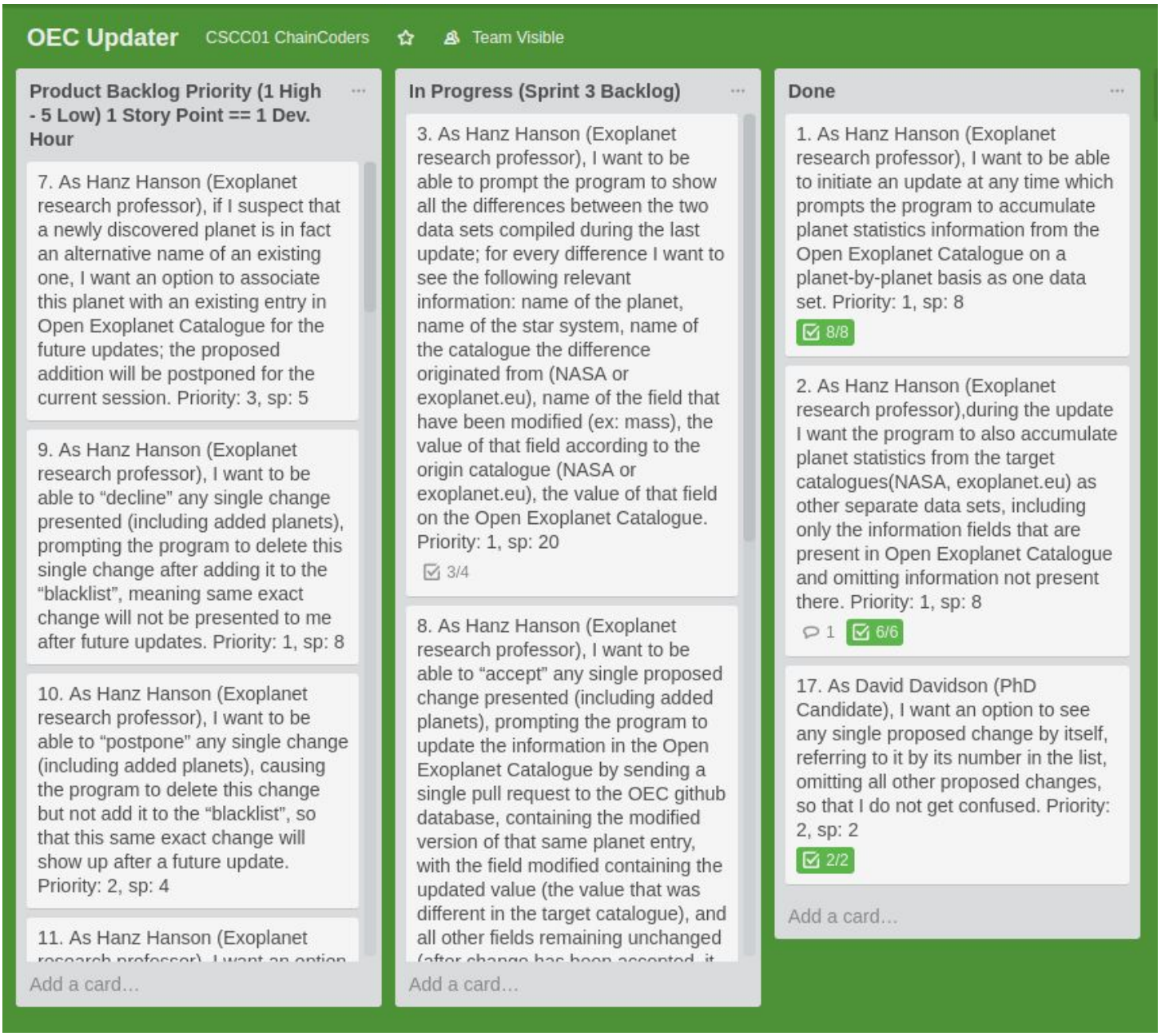
S2	ACTUAL						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
1	0	0	1E	0	0	0	0
2	0	0	0	0	0	0	2A
3	0	0	0	0	0	2A + 2T	4E + 3V + 2J



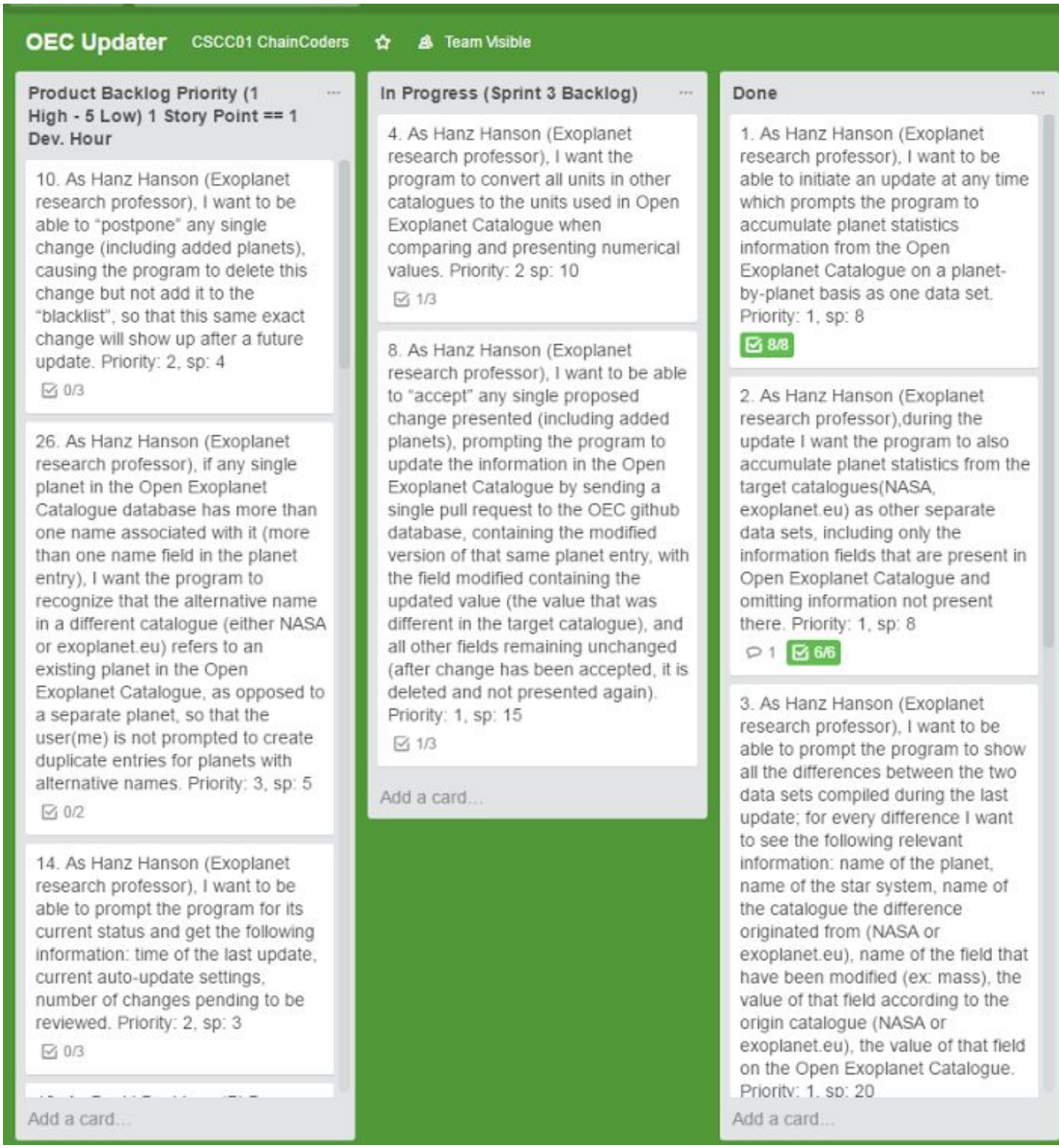


	A	B	C
1	Day(s)	Planned	Actual
2	October 24 12:00 AM	25	25
3	October 24 12:00 PM	25	25
4	October 25 12:00 AM	25	25
5	October 25 12:00 PM	25	25
6	October 26 12:00 AM	25	25
7	October 26 12:00 PM	25	25
8	October 27 12:00 AM	23	24
9	October 27 12:00 PM	23	24
10	October 28 12:00 AM	21	24
11	October 28 12:00 PM	21	24
12	October 29 12:00 AM	19	24
13	October 29 12:00 PM	16	21
14	October 30 12:00 AM	11	19
15	October 30 12:00 PM	6	6

Snapshots: Start of Sprint 3


OEC Updater
CSCC01 ChainCoders
Team Visible
Product Backlog Priority (1 High - 5 Low) 1 Story Point == 1 Dev. Hour
7. As Hanz Hanson (Exoplanet research professor), if I suspect that a newly discovered planet is in fact an alternative name of an existing one, I want an option to associate this planet with an existing entry in Open Exoplanet Catalogue for the future updates; the proposed addition will be postponed for the current session. Priority: 3, sp: 5
9. As Hanz Hanson (Exoplanet research professor), I want to be able to "decline" any single change presented (including added planets), prompting the program to delete this single change after adding it to the "blacklist", meaning same exact change will not be presented to me after future updates. Priority: 1, sp: 8
10. As Hanz Hanson (Exoplanet research professor), I want to be able to "postpone" any single change (including added planets), causing the program to delete this change but not add it to the "blacklist", so that this same exact change will show up after a future update. Priority: 2, sp: 4
11. As Hanz Hanson (Exoplanet
Add a card…
In Progress (Sprint 3 Backlog)
3. As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program to show all the differences between the two data sets compiled during the last update; for every difference I want to see the following relevant information: name of the planet, name of the star system, name of the catalogue the difference originated from (NASA or exoplanet.eu), name of the field that have been modified (ex: mass), the value of that field according to the origin catalogue (NASA or exoplanet.eu), the value of that field on the Open Exoplanet Catalogue. Priority: 1, sp: 20
3/4
8. As Hanz Hanson (Exoplanet research professor), I want to be able to "accept" any single proposed change presented (including added planets), prompting the program to update the information in the Open Exoplanet Catalogue by sending a single pull request to the OEC github database, containing the modified version of that same planet entry, with the field modified containing the updated value (the value that was different in the target catalogue), and all other fields remaining unchanged
Add a card…
Done
1. As Hanz Hanson (Exoplanet research professor), I want to be able to initiate an update at any time which prompts the program to accumulate planet statistics information from the Open Exoplanet Catalogue on a planet-by-planet basis as one data set. Priority: 1, sp: 8
8/8
2. As Hanz Hanson (Exoplanet research professor),during the update I want the program to also accumulate planet statistics from the target catalogues(NASA, exoplanet.eu) as other separate data sets, including only the information fields that are present in Open Exoplanet Catalogue and omitting information not present there. Priority: 1, sp: 8
1
6/6
17. As David Davidson (PhD Candidate), I want an option to see any single proposed change by itself, referring to it by its number in the list, omitting all other proposed changes, so that I do not get confused. Priority: 2, sp: 2
2/2
Add a card…

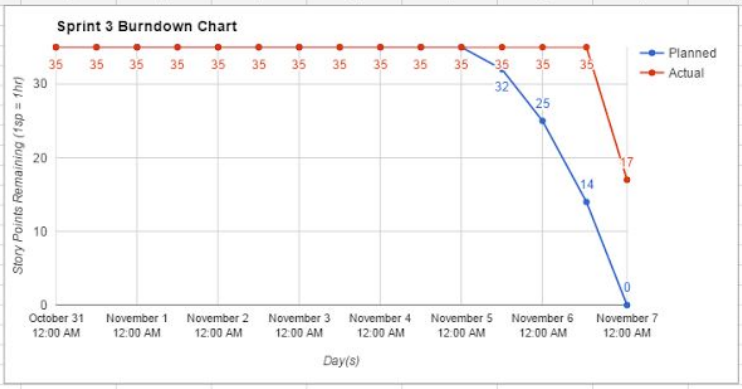
End of sprint 3:



CSCC01 Burndown Charts

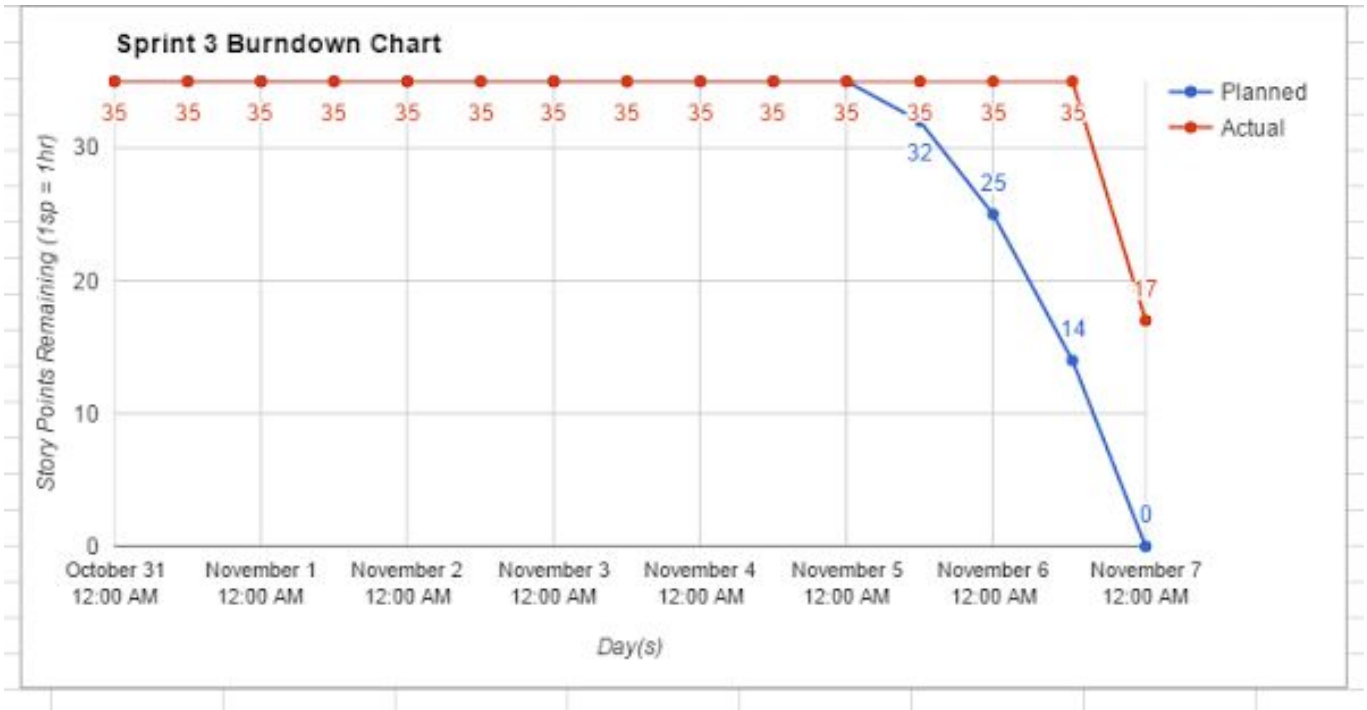
File Edit View Insert Format Data Tools Add-ons Help Last edit was 3 days ago

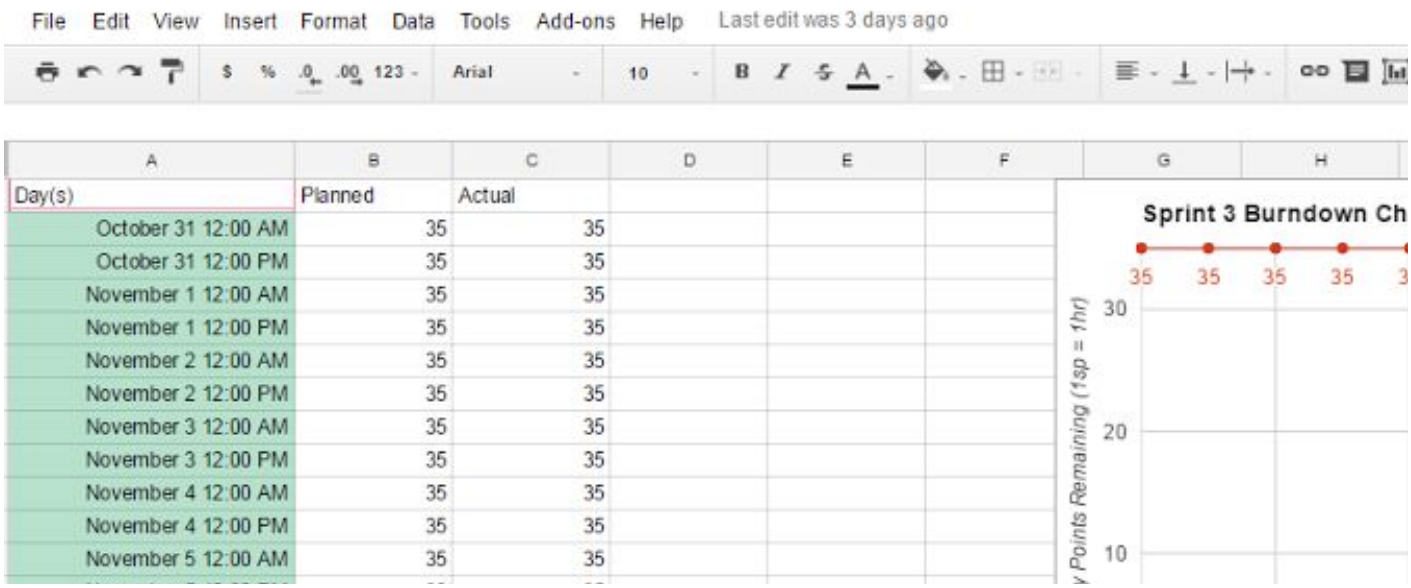
Day(s)	Planned	Actual
October 31 12:00 AM	35	35
October 31 12:00 PM	35	35
November 1 12:00 AM	35	35
November 1 12:00 PM	35	35
November 2 12:00 AM	35	35
November 2 12:00 PM	35	35
November 3 12:00 AM	35	35
November 3 12:00 PM	35	35
November 4 12:00 AM	35	35
November 4 12:00 PM	35	35
November 5 12:00 AM	35	35
November 5 12:00 PM	32	35
November 6 12:00 AM	25	35
November 6 12:00 PM	14	35
November 7 12:00 AM	0	17



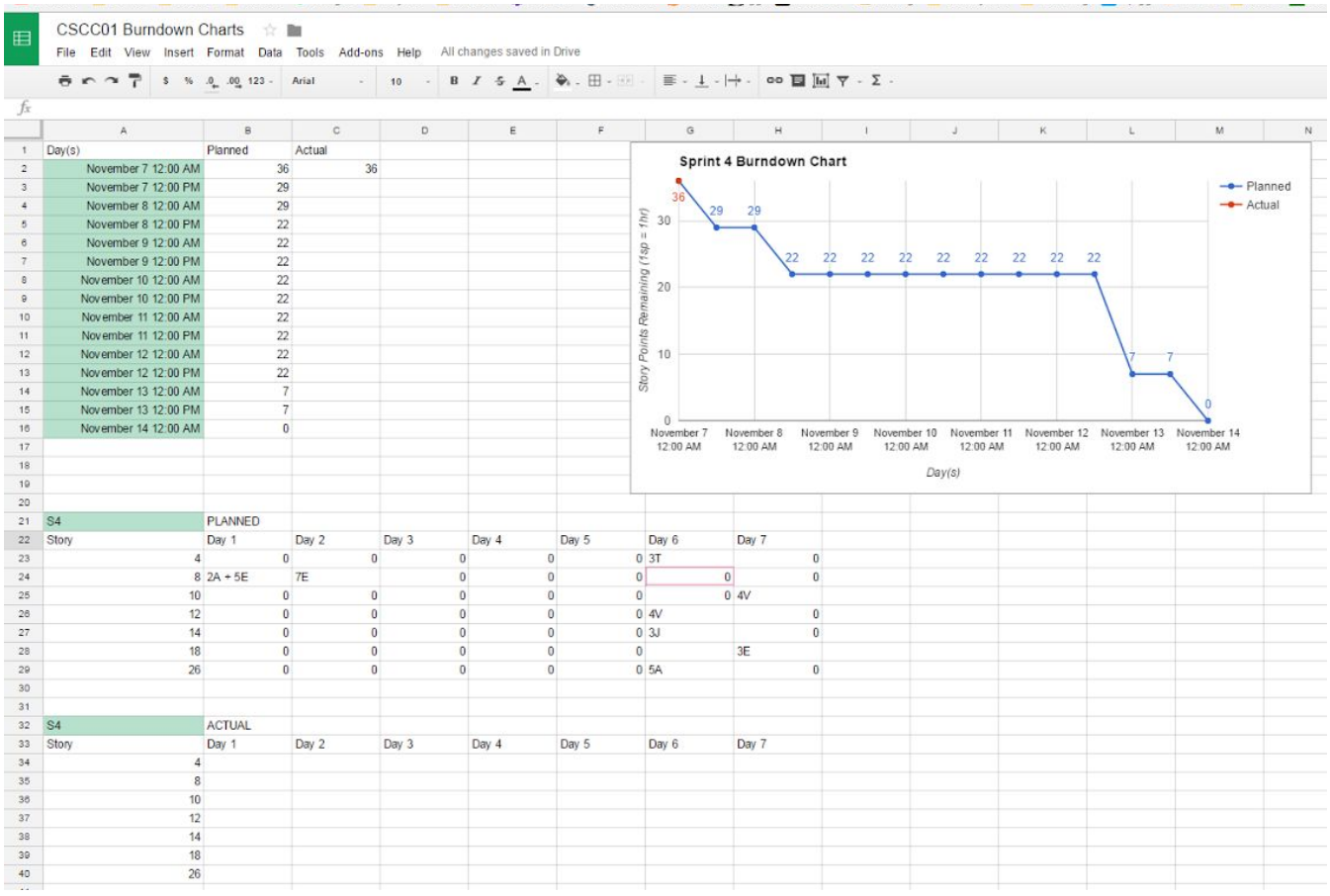
S2	PLANNED						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
3	0	0	0	0	0	3E	0
5	0	0	0	0	0	3V	0
4	0	0	0	0	0	4A	3A + 3V
6	0	0	0	0	0	0	4E
8	0	0	0	0	0	0	7J + 7T + 1V

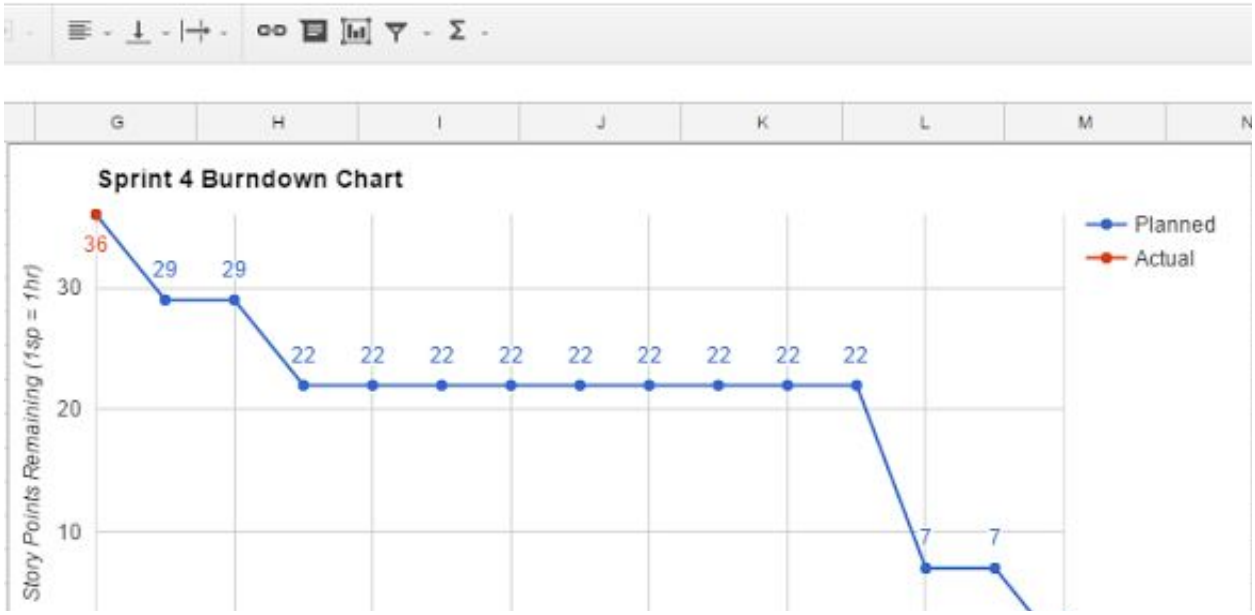
S2	ACTUAL						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
3	0	0	0	0	0	0	3J
5	0	0	0	0	0	0	3V
4	0	0	0	0	0	0	5A + 2J
6	0	0	0	0	0	0	3E + 1T
8	0	0	0	0	0	0	1T





Snapshots: Start of Sprint 4





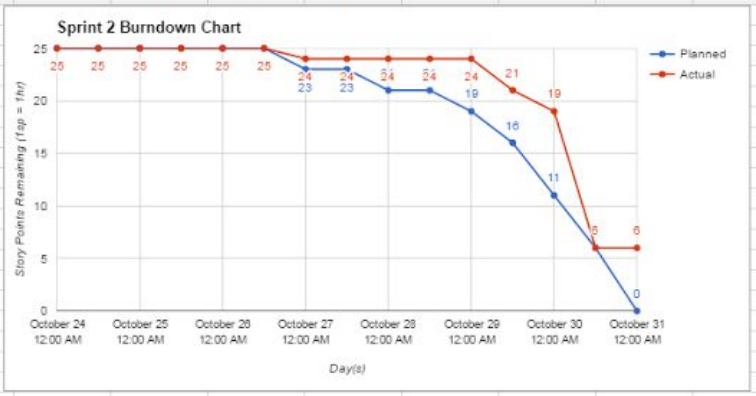
S4	PLANNED						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
4	0	0	0	0	0	3T	0
8	2A + 5E	7E	0	0	0	0	0
10	0	0	0	0	0	0	4V
12	0	0	0	0	0	4V	0
14	0	0	0	0	0	3J	0
18	0	0	0	0	0		3E
26	0	0	0	0	0	5A	0

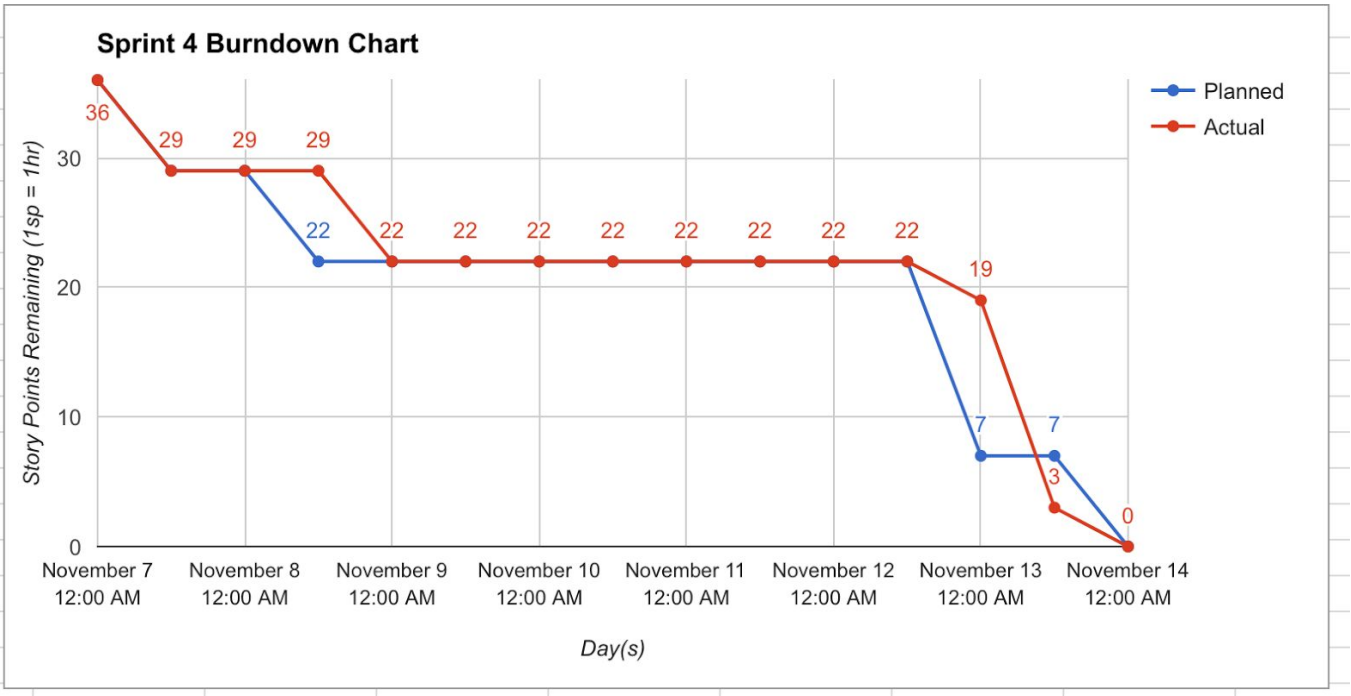
S4	ACTUAL						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
4							
8							
10							
12							
14							
18							
26							

OEC Updater
CSCC01 ChainCoders
Team Visible
Product Backlog Priority (1 High - 5 Low) 1 Story Point == 1 Dev. Hour
sp: 3
23. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor discrepancies between the numeric values across different catalogues, given the difference is within a set percentage tolerance, present by default. Priority: 2, sp: 9
24. As Hanz Hanson (Exoplanet research professor), I want to be able to set the tolerance of every numeric field present in the planet entries in the Open Exoplanet Catalogue to the value that I choose, on a field-by-field basis, so that the program uses the tolerance entered by me, as opposed to default one. Priority: 4, sp: 8
25. As Hanz Hanson (Exoplanet research professor), I want to be able to clear all tolerances added by me, so that the program reverts to the default tolerance value for every field. Priority: 5, sp: 2
27. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor differences in spelling of the names of planets and stars across the different catalogues, which includes case-insensitivity as well as any discrepancies in the punctuation or whitespace(number of spaces and tabs) in the planet name. Priority: 3, sp: 5
Add a card
In Progress (Sprint 4 Backlog)
4. As Hanz Hanson (Exoplanet research professor), I want the program to convert all units in other catalogues to the units used in Open Exoplanet Catalogue when comparing and presenting numerical values. Priority: 2 sp: 10
1/3
8. As Hanz Hanson (Exoplanet research professor), I want to be able to "accept" any single proposed change presented (including added planets), prompting the program to update the information in the Open Exoplanet Catalogue by sending a single pull request to the OEC github database, containing the modified version of that same planet entry, with the field modified containing the updated value (the value that was different in the target catalogue), and all other fields remaining unchanged (after change has been accepted, it is deleted and not presented again). Priority: 1, sp: 15
1/3
10. As Hanz Hanson (Exoplanet research professor), I want to be able to "postpone" any single change (including added planets), causing the program to delete this change but not add it to the "blacklist", so that this same exact change will show up after a future update. Priority: 2, sp: 4
0/3
12. As Hanz Hanson (Exoplanet
Add a card
Done
1. As Hanz Hanson (Exoplanet research professor), I want to be able to initiate an update at any time which prompts the program to accumulate planet statistics information from the Open Exoplanet Catalogue on a planet-by-planet basis as one data set. Priority: 1, sp: 8
8/8
2. As Hanz Hanson (Exoplanet research professor),during the update I want the program to also accumulate planet statistics from the target catalogues(NASA, exoplanet.eu) as other separate data sets, including only the information fields that are present in Open Exoplanet Catalogue and omitting information not present there. Priority: 1, sp: 8
1
6/6
3. As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program to show all the differences between the two data sets compiled during the last update; for every difference I want to see the following relevant information: name of the planet, name of the star system, name of the catalogue the difference originated from (NASA or exoplanet.eu), name of the field that have been modified (ex: mass), the value of that field according to the origin catalogue (NASA or exoplanet.eu), the value of that field on the Open Exoplanet Catalogue. Priority: 1, sp: 20
Add a card

Snapshots: End of Sprint 4

	A	B	C	D	E	F	G	H	I	J	K	L
1	Day(s)	Planned	Actual									
2	October 24 12:00 AM	25	25									
3	October 24 12:00 PM	25	25									
4	October 25 12:00 AM	25	25									
5	October 25 12:00 PM	25	25									
6	October 26 12:00 AM	25	25									
7	October 26 12:00 PM	25	25									
8	October 27 12:00 AM	23	24									
9	October 27 12:00 PM	23	24									
10	October 28 12:00 AM	21	24									
11	October 28 12:00 PM	21	24									
12	October 29 12:00 AM	19	24									
13	October 29 12:00 PM	16	21									
14	October 30 12:00 AM	11	19									
15	October 30 12:00 PM	6	6									
16	October 31 12:00 AM	0	6									
17												
18												
19												
20												
21												
22												
23	S2	PLANNED										
24	Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7				
25	1	0	0	0	2E	0	0	0				
26	2	0	0	2A	0	0	0	0				
27	3	0	0	0	0	2V	3E + 5T	3A + 3J				
28	5	0	0	0	0	0	0	3V				
29	17	0	0	0	0	0	0	2J				
30												
31												
32												
33												
34												
35												
36	S2	ACTUAL										
37	Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7				
38	1	0	0	1E	0	0	0	0				
39	2	0	0	0	0	0	0	2A				
40	3	0	0	0	0	0	2A + 2T	4E + 3V + 2J				
41	5	0	0	0	0	0	0	0				
42	17	0	0	0	0	0	0	2J				
43												





S4	PLANNED						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
4	0	0	0	0	0	3T	0
8	2A + 5E	7E	0	0	0	0	0
10	0	0	0	0	0	0	4V
12	0	0	0	0	0	4V	0
14	0	0	0	0	0	3J	0
18	0	0	0	0	0	0	3E
26	0	0	0	0	0	5A	0

S4	ACTUAL						
Story	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
4	0	0	0	0	0	0	3A
8	2A + 5E	0	7E	0	0	0	0
10	0	0	0	0	0	0	4T
12	0	0	0	0	0	0	4V
14	0	0	0	0	0	3E	0
18	0	0	0	0	0	0	3J
26	0	0	0	0	0	0	3A + 2J

Done

10. As Hanz Hanson (Exoplanet research professor), I want to be able to “postpone” any single change (including added planets), causing the program to delete this change but not add it to the “blacklist”, so that this same exact change will show up after a future update. Priority: 2, sp: 4

☑ 3/3

8. As Hanz Hanson (Exoplanet research professor), I want to be able to “accept” any single proposed change presented (including added planets), prompting the program to update the information in the Open Exoplanet Catalogue by sending a single pull request to the OEC github database, containing the modified version of that same planet entry, with the field modified containing the updated value (the value that was different in the target catalogue), and all other fields remaining unchanged (after change has been accepted, it is deleted and not presented again). Priority: 1, sp: 15

☑ 3/3

3. As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program to show all the differences between the two data sets compiled during the last update; for every difference I want to see the following relevant information: name of the planet, name of the star system, name of the catalogue the difference originated from (NASA or

Add a card…

Done

3. As Hanz Hanson (Exoplanet research professor), I want to be able to prompt the program to show all the differences between the two data sets compiled during the last update; for every difference I want to see the following relevant information: name of the planet, name of the star system, name of the catalogue the difference originated from (NASA or exoplanet.eu), name of the field that have been modified (ex: mass), the value of that field according to the origin catalogue (NASA or exoplanet.eu), the value of that field on the Open Exoplanet Catalogue. Priority: 1, sp: 20

☑ 4/4

5. As Hanz Hanson (Exoplanet research professor), when viewing proposed changes, I want to see them in a numbered list. Priority: 3, sp: 3

☑ 2/2

6. As Hanz Hanson (Exoplanet research professor), if one of the updates discovers a planet entry in the target catalogue, while a planet with the same name is not present in the Open Source Exoplanet Catalogue, I want to the program to present it to me as a newly discovered planet with the rest of the proposed changes, providing the following information: planet name, star system name, name of the catalogue and the date the entry is posted. Priority: 3, sp: 4

☑ 3/3

Add a card…

Done

following information: planet name, star system name, name of the catalogue and the date the entry is posted. Priority: 3, sp: 4

☑ 3/3

17. As David Davidson (PhD Candidate), I want an option to see any single proposed change by itself, referring to it by its number in the list, omitting all other proposed changes, so that I do not get confused. Priority: 2, sp: 2

☑ 2/2

26. As Hanz Hanson (Exoplanet research professor), if any single planet in the Open Exoplanet Catalogue database has more than one name associated with it (more than one name field in the planet entry), I want the program to recognize that the alternative name in a different catalogue (either NASA or exoplanet.eu) refers to an existing planet in the Open Exoplanet Catalogue, as opposed to a separate planet, so that the user(me) is not prompted to create duplicate entries for planets with alternative names. Priority: 3, sp: 5

☑ 2/2

18. As David Davidson (PhD Candidate), I want an option to see a specific number of proposed changes at a time (for example 10 at a time), so I do not have to scroll through a lot of extra information to find what I need. Priority: 3, sp: 3

☑ 2/2

Product Backlog Priority (1 High - 5 Low) 1 Story Point == 1 Dev. Hour

7. As Hanz Hanson (Exoplanet research professor), if I suspect that a newly discovered planet is in fact an alternative name of an existing one, I want an option to associate this planet with an existing entry in Open Exoplanet Catalogue for the future updates; the proposed addition will be postponed for the current session. Priority: 3, sp: 5

9. As Hanz Hanson (Exoplanet research professor), I want to be able to “decline” any single change presented (including added planets), prompting the program to delete this single change after adding it to the “blacklist”, meaning same exact change will not be presented to me after future updates. Priority: 1, sp: 8

11. As Hanz Hanson (Exoplanet research professor), I want an option to clear the “blacklist”, which would cause the program to forget that some changes were previously declined and present them again in the future updates. Priority: 2, sp: 2

13. As Hanz Hanson (Exoplanet research professor), I want to be able to set the program to update automatically and set the time interval between updates. Priority: 2, sp: 5

15. As Hanz Hanson (Exoplanet research professor), if there are changes pending to be reviewed by the time of the next update, I want the program to automatically

Add a card…

Product Backlog Priority (1 High - 5 Low) 1 Story Point == 1 Dev. Hour

sp: 5

15. As Hanz Hanson (Exoplanet research professor), if there are changes pending to be reviewed by the time of the next update, I want the program to automatically postpone all of them. Priority: 2, sp: 2

16. As David Davidson (PhD Candidate), I want an option to accept or decline or postpone all currently pending changes at once. Priority: 4, sp: 2

19. As David Davidson (PhD Candidate), I want an option to see a detailed user manual, describing the operation of the program. Priority: 5, sp: 4

20. As David Davidson (PhD Candidate), while I am reviewing the list of proposed changes I want an option to provide a name for a certain planet and to view / accept / decline / postpone all changes for that planet at once. Priority: 5, sp: 3

21. As David Davidson (PhD Candidate), while I am reviewing the list of proposed changes I want an option to view / accept / decline / postpone all changes to a certain star system, whose name I provide, at once. Priority: 5, sp: 3

22. As David Davidson (PhD Candidate), while I am reviewing the list of proposed changes I want an

Add a card…

Product Backlog Priority (1 High - 5 Low) 1 Story Point == 1 Dev. Hour

22. As David Davidson (PhD Candidate), while I am reviewing the list of proposed changes I want an option to view / accept / decline / postpone all changes originating from one of the catalogues, whose name I provide, at once. Priority: 5, sp: 3

23. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor discrepancies between the numeric values across different catalogues, given the difference is within a set percentage tolerance, present by default. Priority: 2, sp: 9

24. As Hanz Hanson (Exoplanet research professor), I want to be able to set the tolerance of every numeric field present in the planet entries in the Open Exoplanet Catalogue to the value that I choose, on a field-by-field basis, so that the program uses the tolerance entered by me, as opposed to default one. Priority: 4, sp: 8

25. As Hanz Hanson (Exoplanet research professor), I want to be able to clear all tolerances added by me, so that the program reverts to the default tolerance value for every field. Priority: 5, sp: 2

27. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor differences in spelling of the names of planets and stars across the different

Add a card…

Product Backlog Priority (1 High - 5 Low) 1 Story Point == 1 Dev. Hour

sp: 3

23. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor discrepancies between the numeric values across different catalogues, given the difference is within a set percentage tolerance, present by default. Priority: 2, sp: 9

24. As Hanz Hanson (Exoplanet research professor), I want to be able to set the tolerance of every numeric field present in the planet entries in the Open Exoplanet Catalogue to the value that I choose, on a field-by-field basis, so that the program uses the tolerance entered by me, as opposed to default one. Priority: 4, sp: 8

25. As Hanz Hanson (Exoplanet research professor), I want to be able to clear all tolerances added by me, so that the program reverts to the default tolerance value for every field. Priority: 5, sp: 2

27. As Hanz Hanson (Exoplanet research professor), I want the program to ignore minor differences in spelling of the names of planets and stars across the different catalogues, which includes case-insensitivity as well as any discrepancies in the punctuation or whitespace(number of spaces and tabs) in the planet name. Priority: 3, sp: 5

Add a card…

Done

itself, referring to it by its number in the list, omitting all other proposed changes, so that I do not get confused. Priority: 2, sp: 2

2/2

26. As Hanz Hanson (Exoplanet research professor), if any single planet in the Open Exoplanet Catalogue database has more than one name associated with it (more than one name field in the planet entry), I want the program to recognize that the alternative name in a different catalogue (either NASA or exoplanet.eu) refers to an existing planet in the Open Exoplanet Catalogue, as opposed to a separate planet, so that the user(me) is not prompted to create duplicate entries for planets with alternative names. Priority: 3, sp: 5

2/2

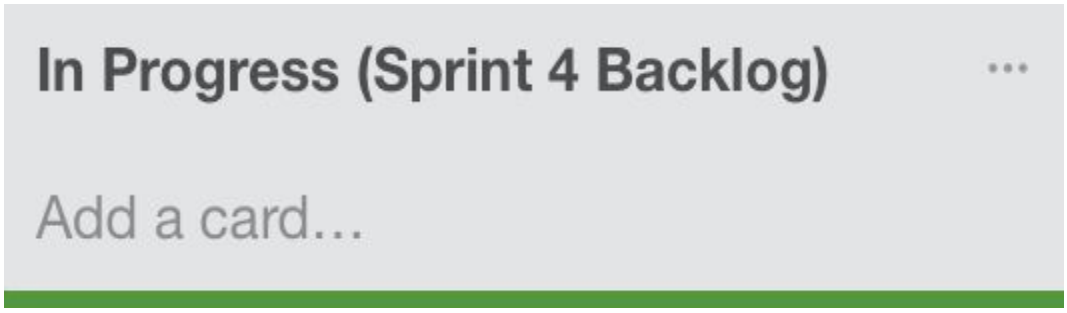
18. As David Davidson (PhD Candidate), I want an option to see a specific number of proposed changes at a time (for example 10 at a time), so I do not have to scroll through a lot of extra information to find what I need. Priority: 3, sp: 3

2/2

4. As Hanz Hanson (Exoplanet research professor), I want the program to convert all units in other catalogues to the units used in Open Exoplanet Catalogue when comparing and presenting numerical values. Priority: 2 sp: 10

1 3/3

Add a card…


In Progress (Sprint 4 Backlog)
Add a card…