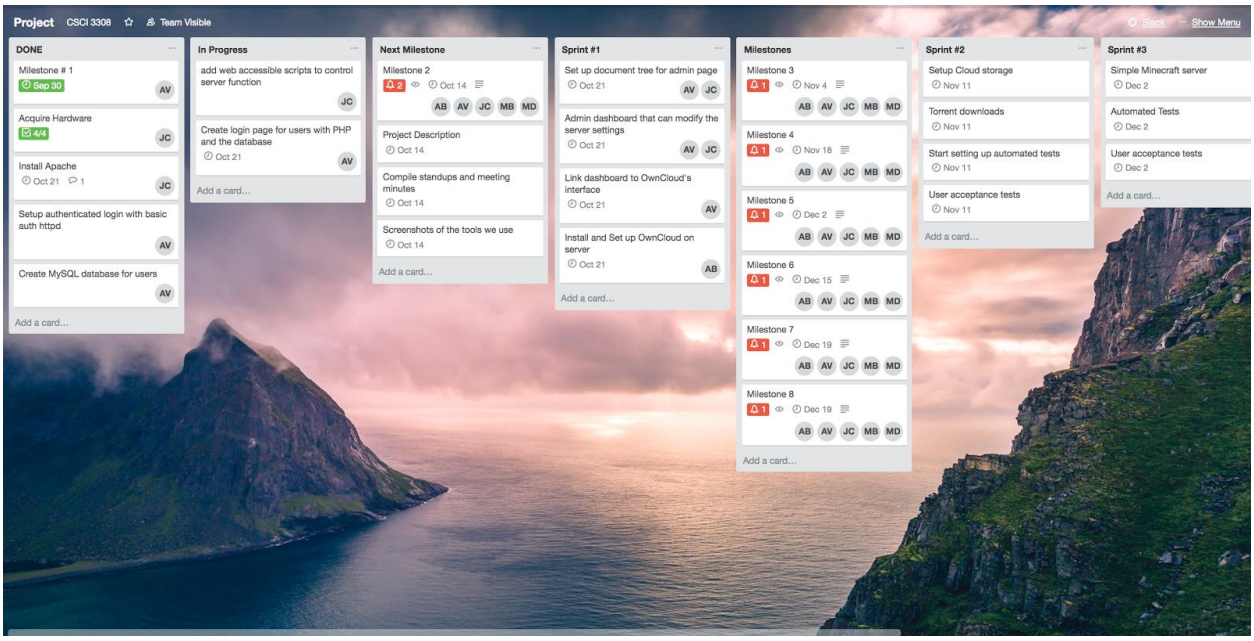
Pi In The Sky

Milestone #2

Project Management Tools:

Trello was selected as our project management tool. We broke our backlog down into 3 distinct 3-week sprint cycles (see below for details) as well as a section for upcoming milestones. The items located to the right of the “In Progress” category acts as our backlog and the left side is what we have completed. Each sprint cycle has tasks and features that we want to add to the finished product. As our features and ideas adapt for the product we add/change backlog items. Due dates for tasks are added when the card placed in the backlog to deliver on time. All of our cards are self-assigned as we go and moved to in progress when being worked on to insure communication across the team.



Tentative Plan for Sprints:

Sprint #1:

- The owner should be able to login to the server and be able to edit some basic server settings
- The owner after logging in should be able to create new user accounts for friends and family members
- The user should be able to reset their password
- The user should be able to upload websites and host them on the server

Sprint #2:

- The user should be able to store files in the server (upload) and retrieve them (download), essentially treating the server as a cloud storage. The capacity of the storage will depend on the hard disk size the owner uses
- The user should be able to stream downloads (large files and torrents) directly to the server.
- The user should to able to stream videos and files from the server

Sprint #3:

- The user should be able to setup a simple gaming server (Minecraft for instance) without having to tweak the server manually.

Functional Requirements:

1. **Name:** Web Login

   Actor: User

   Normal Course: The user will be able to open the website and log in. If the credentials are correct it will redirect the user to the homepage.

   Exception Cases: If the credentials are incorrect it will give an error message. If the user tries to go directly to home page without logging in, they will be redirected to the login page

   Frequency of Use: Every time the user has to modify any of the settings or has to use the server

   Assumptions: The user has a login on the server. There will be a set of default credentials that come with the server.

2. **Name:** Create a new user account

   Actor: Logged in user

   Normal Course: After logging in, the user will have an option to create a new user account. This will take them to a new page where they will enter details for the new user (Name, Email, Username, Password).

   Exception Cases: If the username already exists in the database, there will be an error. The user will be prompted to type in the password twice. If they don't match, there will be an error.

   Frequency of Use: A few times during the first time setup of the server depending on how many people will be accessing the server.

   Assumptions: The user has a login credentials.

3. **Name:** Reset Password

   Actor: User

   Normal Course: After logging in, the user can change there password. This will prompt them to enter their old password as well as a new password. The must type their new password twice to confirm the change.

Exception Cases: If there old password is incorrect or there new password doesn't match, then it will give an error message. The old password and the new password should not be the same

Frequency of Use: Whenever the user wants to change the password.

Assumption: The user already has an account and has the ability to login.

4. **Name:** Restart the Server

   Actor: User

   Normal Course: The user will be able to restart the server or schedule a restart from the website. There will be a button once the user has logged in that forces the server to restart.

   Exception Cases: If there is an internal error in the server, the server will fail to restart. The user will then have to manually fix the error and reboot the server.

   Frequency of Use: Whenever the user wants to restart the server.

   Assumption: The user is already logged in to the website.

5. **Name:** Host website

   Actor: User

   Normal Course: The user, after logging in, will have an option to upload a compressed file containing the website contents. The user can also specify the subdomain or path for the new website.

   Exception Cases: The user has made some error with specifying the subdomain or there is an error with the website. Then the website will not load properly.

   Frequency of Use: Whenever the user creates a new website or updates an old one

   Assumption: The user has the website files ready and is logged in.

6. **Name:** View local files on server

   Actor: User

   Normal Course: The user will be able to see a list of local files on the server and be able to download them to their device.

   Exception Cases: The user doesn't have any local files stored on the server. The user does not have the files stored in the correct directory.

Frequency of Use: Used when the user wants to view/download local files stored on the server from a different device.

Assumption: The user is already logged in to the website. The user has files stored in the correct directory of the server.

7. **Name:** Upload files to the server

   Actor: User

   Normal Course: The user, after logging in, has the option to upload files to the cloud storage system. The user can they verify by viewing the local files on server.

   Exception cases: If the file exceeds the amount of space on the server.

   Frequency of use: Everytime the user wants to move something to the cloud

   Assumption: There is enough space on the server and the user is logged in.

8. **Name:** Download files from server

   Actor: User

   Normal Course: The user will be able to view/download files from the cloud storage system that they have already uploaded.

   Exception Cases: The user doesn't have any files on the cloud storage system.

   Frequency of Use: Used when the user wants to access files that they have stored on the cloud storage system.

   Assumption: The user is able to login and there are files on the cloud storage system.

9. **Name:** Stream downloads to the server

   Actor: User

   Normal Course: The user, after logging in, will be able to provide a download link or a torrent to the server. The server would then start downloading the files in the background. The user will have the options to pause, resume, and stop the downloads at any point.

   Exception Cases: The user doesn't have enough space on the server

   Assumption: The user is logged in and has a valid download link

10. **Name:** Stream videos from the server to the user’s monitor

    Actor: User

    Normal Course: Once the user is logged in, the user will be able to stream videos that are stored on the server. The video will be viewed on the user's display.

    Exception Cases: The user doesn’t have any videos stored on the server. The videos are the wrong format to stream.

    Frequency of Use: Used when the user wants to view videos that are stored on the server.

    Assumption: The user is logged in and has videos stored on the cloud storage system.

11. **Name:** Setup a simple Minecraft server

    Actor: User

    Normal Course: The user, after logging in, will have the option to turn the server into a Minecraft gaming server. The players, can then connect to this server and enjoy hours of Minecraft together.

    Exception Cases: None

    Frequency of Use: As often as the user enjoys playing Minecraft

    Assumption: The user is logged in.

Non-Functional Requirements:

1. The server is built on a Raspberry Pi with a Linux OS. The external storage is provided by a hard disk.
2. The user is able to access the server from anywhere in the world using a unique IP address given that port forwarding is setup for their server instance.
3. The login page uses PHP to connect with the MySQL backend on the server for logging in. The server comes with default credentials that the user can use to login then edit. The same stack is used for creating new users and resetting the password.
4. When the user clicks the “Reset Server” button, the button executes a shell script on the server that restarts the Apache2 server
5. When the user uploads the website, this will create a new directory on the server and the server is automatically configured to run the website from that directory
6. There will be an implementation of a simple cloud storage system, similar to ownCloud, that will allow the user to treat the server as a cloud storage, with the ability to upload and download files.
7. This cloud storage system will also allow the user to stream downloads and torrents to the server, and files and videos from the server
8. The Minecraft server is an executable .jar file that will be executed once the user clicks the button.