Actions to Start Doing:
- More frequent meetups
- Integrations testing: test all components that have been completed together
- Demos: We need to demo what we've accomplished in our meetings

Actions to keep doing:
- Meeting regularly: The more we meet the clearer our goals become
- Communication: Slack is still a very efficient to tool for us to discuss the varying aspects of our project such as mobile, frontend, or database tasks to complete
- Help out others who are running into bugs that are hard to fix

Work Completed:
- Voting system has been implemented by Seth and Nikola
- Users database that contains additional fields which mirrors the firebase account functionality - completed by Seth and Nikola
- Assigning the ethereum wallets as fields to the user accounts - completed by Seth and Nikola
- Header/ Drawer designed/ implemented- completed by Ian
- Upload screen changed to card view. File upload button replaced, title, description, and subject text fields/ dropdowns added- completed by Ian
- Filtering files by category implemented by Seth and Nikolai
- Solidity contract(removed in favor of MetaMask) and web3 integration - completed by Edwin
- Browse "tab" view implemented by Ian
- Tie transactions to download functionality - completed by Edwin, Nikola, and Seth
- Card view autofill(removed due to integration issues with download functionality)- completed by Ian
- Cole has a standard signup page displaying with some bugs that needed to be fixed
- Integrating the login page with the rest of the project - completed by Nikola and Seth
- Integrating the browse and upload pages into the ejs express framework for the heroku app - completed by Edwin
- Setup/tutorial instructions - completed by Cole and Edwin
- Modal popups for transactions, logins, signup completed by Edwin
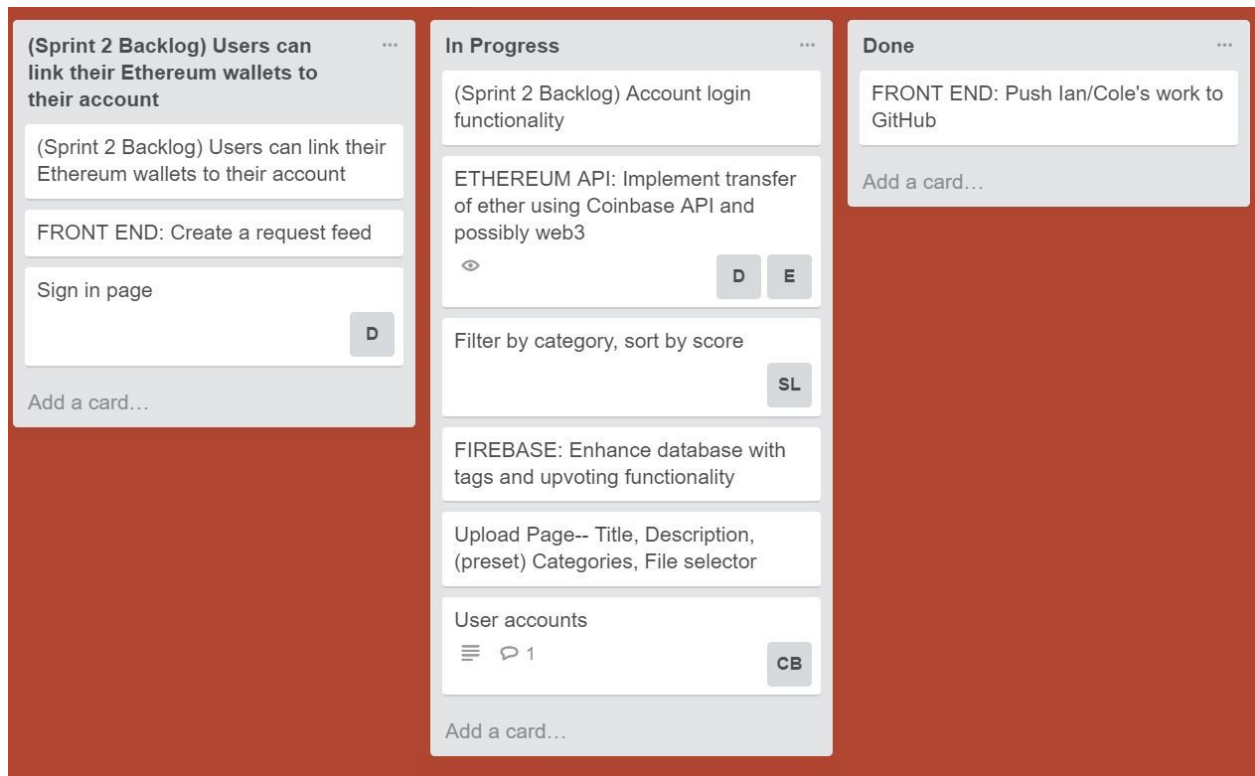
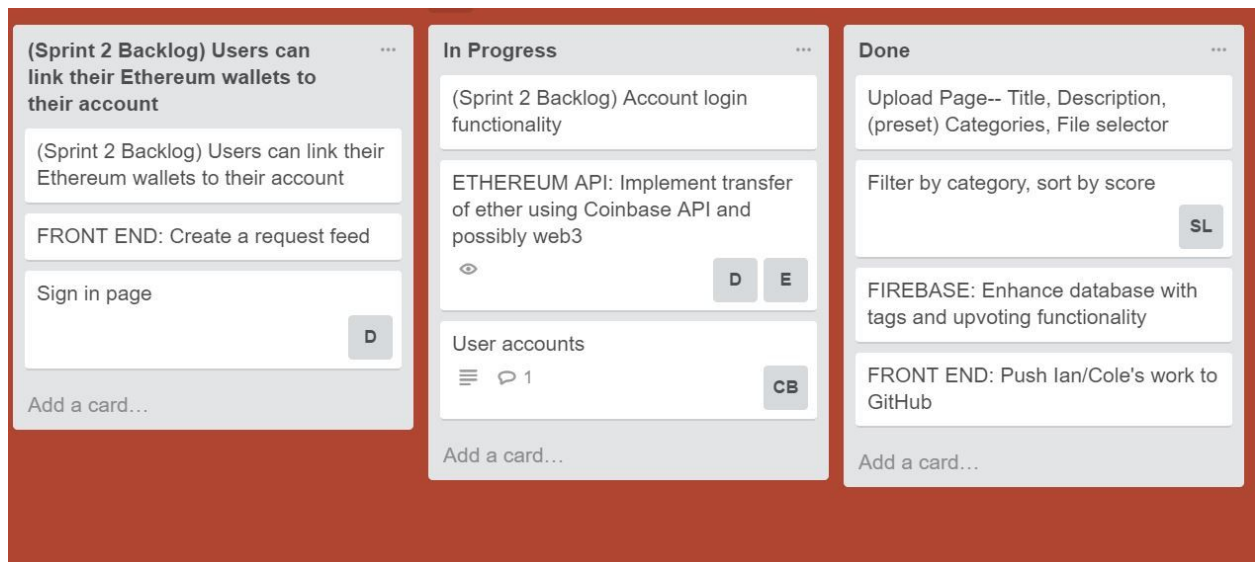**Not Completed:**
-

Work Completion Rate:

Total number of user stories completed for Sprint 3: 12\12… added extras last second

Total number of days: 7

Total number of estimated ideal work hours completed: 5 hours per work day

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account ···

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account

FRONT END: Create a request feed

Sign in page

D

Add a card…

In Progress ···

(Sprint 2 Backlog) Account login functionality

ETHEREUM API: Implement transfer of ether using Coinbase API and possibly web3

👁   D   E

Filter by category, sort by score

SL

FIREBASE: Enhance database with tags and upvoting functionality

Upload Page-- Title, Description, (preset) Categories, File selector

User accounts

☰   💬 1   CB

Add a card…

Done ···

FRONT END: Push Ian/Cole's work to GitHub

Add a card…

05/21/17

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account ···

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account

FRONT END: Create a request feed

Sign in page

D

Add a card…

In Progress ···

(Sprint 2 Backlog) Account login functionality

ETHEREUM API: Implement transfer of ether using Coinbase API and possibly web3

👁   D   E

User accounts

☰   💬 1   CB

Add a card…

Done ···

Upload Page-- Title, Description, (preset) Categories, File selector

Filter by category, sort by score

SL

FIREBASE: Enhance database with tags and upvoting functionality

FRONT END: Push Ian/Cole's work to GitHub

Add a card…

05/22/17-0/24/17

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account   ...

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account

FRONT END: Create a request feed ✎

Sign in page

D

Add a card…

In Progress   ...

(Sprint 2 Backlog) Account login functionality

ETHEREUM API: Implement transfer of ether using Coinbase API and possibly web3

👁   D   E

Add a card…

Done   ...

Upload Page-- Title, Description, (preset) Categories, File selector

Filter by category, sort by score

SL

FIREBASE: Enhance database with tags and upvoting functionality

User accounts

≡ 💬 1   CB

FRONT END: Push Ian/Cole's work to GitHub

Add a card…

05/25/17-05/26/17

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account   ...

(Sprint 2 Backlog) Users can link their Ethereum wallets to their account

FRONT END: Create a request feed

Sign in page

D

Add a card…

In Progress   ...

(Sprint 2 Backlog) Account login functionality

Add a card…

Done   ...

Upload Page-- Title, Description, (preset) Categories, File selector

Filter by category, sort by score

SL

FIREBASE: Enhance database with tags and upvoting functionality

User accounts

≡ 💬 1   CB

FRONT END: Push Ian/Cole's work to GitHub

ETHEREUM API: Implement transfer of ether using Coinbase API and possibly web3

👁   D   E

Add a card…

05/27/17

Scrum Board Sprint 3   Atheneum   ☆   ⚇ Team Visible

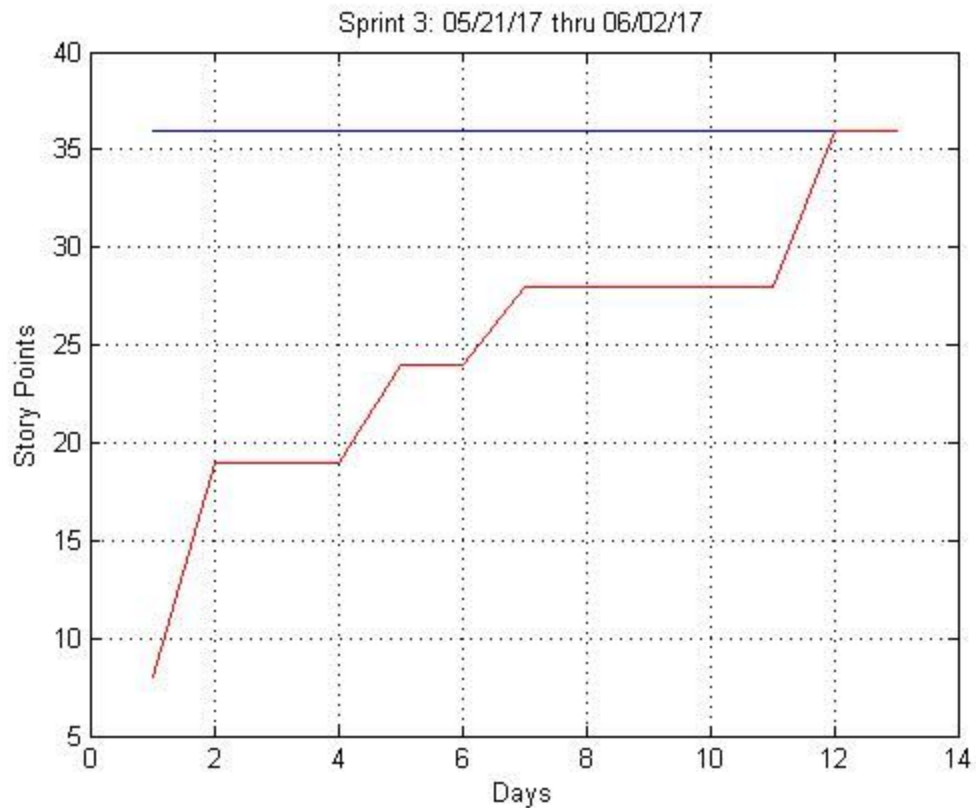| (Sprint 2 Backlog) Users can link their Ethereum wallets to their account ⋯	In Progress ⋯	Done ⋯
Add a card…	FRONT END: Create a request feed **DA**	FRONT END: Push Ian/Cole's work to GitHub
	Add a card…	(Sprint 2 Backlog) Users can link their Ethereum wallets to their account
		(Sprint 2 Backlog) Account login functionality
		ETHEREUM API: Implement transfer of ether using Coinbase API and possibly web3 👁 **D** **E**
		User accounts ☰ 💬 1 **CB**
		FIREBASE: Enhance database with tags and upvoting functionality
		Create MDL grid for browsing tab
		Filter by category, sort by score **SL**
		Create page for viewing a file in depth
		Upload Page-- Title, Description, (preset) Categories, File selector
		Download handles transactions from testnet blockchain 👁 **E**
		Download Page-- sample card
		Add a card…

06/02/17 Note: We added a few extra tasks and completed them

Sprint 3: 05/21/17 thru 06/02/17

Non-included Features:

Solidity Contract:
Known issues: Communication between web3.js and the solidity contract deployed on the testnet blockchain made it difficult to bypass a user's locked account to submit a transaction and pass a specified amount of Ether. Thus due to my (Edwin's) lack of experience with solidity, I did not feel comfortable enough to let the contract handle wallet passwords and transactions, and decided that we should cut the contract from the project entirely despite all the time spent on creating several iterations of the contract. Luckily, the quick switch to MetaMask allowed us to still fulfill transactions through communication with web3.js at the expense of making all transactions a flat rate of 0.05 Ether.

Card View Autofill:
Attempted to list files within in a three-column card format, for more intuitive and enjoyable browsing for the user. Unfortunately, though it functioned perfectly on a machine-local level

(screenshot attached), the card view ultimately broke when attached to the sites framework (card became misaligned/stacked when tabs were switched) and had to be cut from the final product.