**CMPS115 Sprint 2 Plan**

**Goal**

After the second sprint it should be possible for administrators to remove bad content and for readers to be able to use the app on shaky connections.

**Task Listing**

**User Story 1** - As a reader, I would like to be able to search for specific comics so that I can

easily find content that interests me.

* Build angular search service (2 hrs)
* Add search UI to comic list (4 hrs)

*Total Time - 6 hrs*

**User Story 2** - As a reader, I would like to be able to return to the last page I read so that I can

easily pick up where I left off.

* Store read pages in IndexedDB (3 hrs)
* Write logic to skip to latest page of logical section (whole comic or chapter) (2 hrs)
* Modify UI with skip buttons for comic/chapter (2 hrs)

*Total Time - 7 hrs*

**User Story 3** - As a reader, I want the viewing experience to be smooth, even when my

connection is poor, so that I can read the comics with greater ease.

* Preload images next few pages (2 hrs)
* Store previous comic metadata/comic list then merge with new results from server (5 hrs)
* Add PWA Manifest (2 hrs)
* Ensure Service Worker installs correctly (3 hrs)

*Total Time - 12 hrs*

**User Story 4** - As an administrator, I want to be able to remove inappropriate content so that I can

maintain the integrity of the web comic viewer.

* Make backend separate user/admin role in the database (1 hr)
* Create backend route to delete a comic ( 3 hrs)
* Create UI for admins to delete a comic ( 3 hrs)

*Total Time - 7 hrs*

**User Story 5** - As an administrator, I want to be able to ban problematic account users so that I

can prevent repeat offenses from occuring.

* Make a attribute in the user table to mark a banned account (1 hr)
* Prevent users from logging in to banned accounts and rescind their authorization (tokens should be invalid) (3 hr)
* Create a route that marks an account banned (3 hrs)
* Build UI for an admin to ban a comics owner (3 hrs)

*Total Time - 10 hrs*

**User Story 6** - As an administrator, I want to be able to communicate issues with account

users so that they understand what violations they made.

* Create a table to store admin messages. (2 hrs)
* Create a route where admins can send messages. (3 hrs)
* Create a route where users can look up messages received. (3 hrs)
* Create UI for admins to send messages to users. (3 hrs)
* Create UI for users to receive messages. (3 hrs)

*Total Time - 14 hrs*

**Team Roles**

Will: Product Owner, Full-stack Dev

Myron: Backend Dev

Sammy: Backend Dev

James: Frontend Dev

Mehtab: Scrum Master, Backend Dev

Alan: Scrum Master, Full-stack Dev

**Initial Task Assignment**

Will: User story 3 - Add PWA Manifest (2 hrs)

Myron: User story 5 - Make a attribute in the user table to mark a banned account (1 hr)

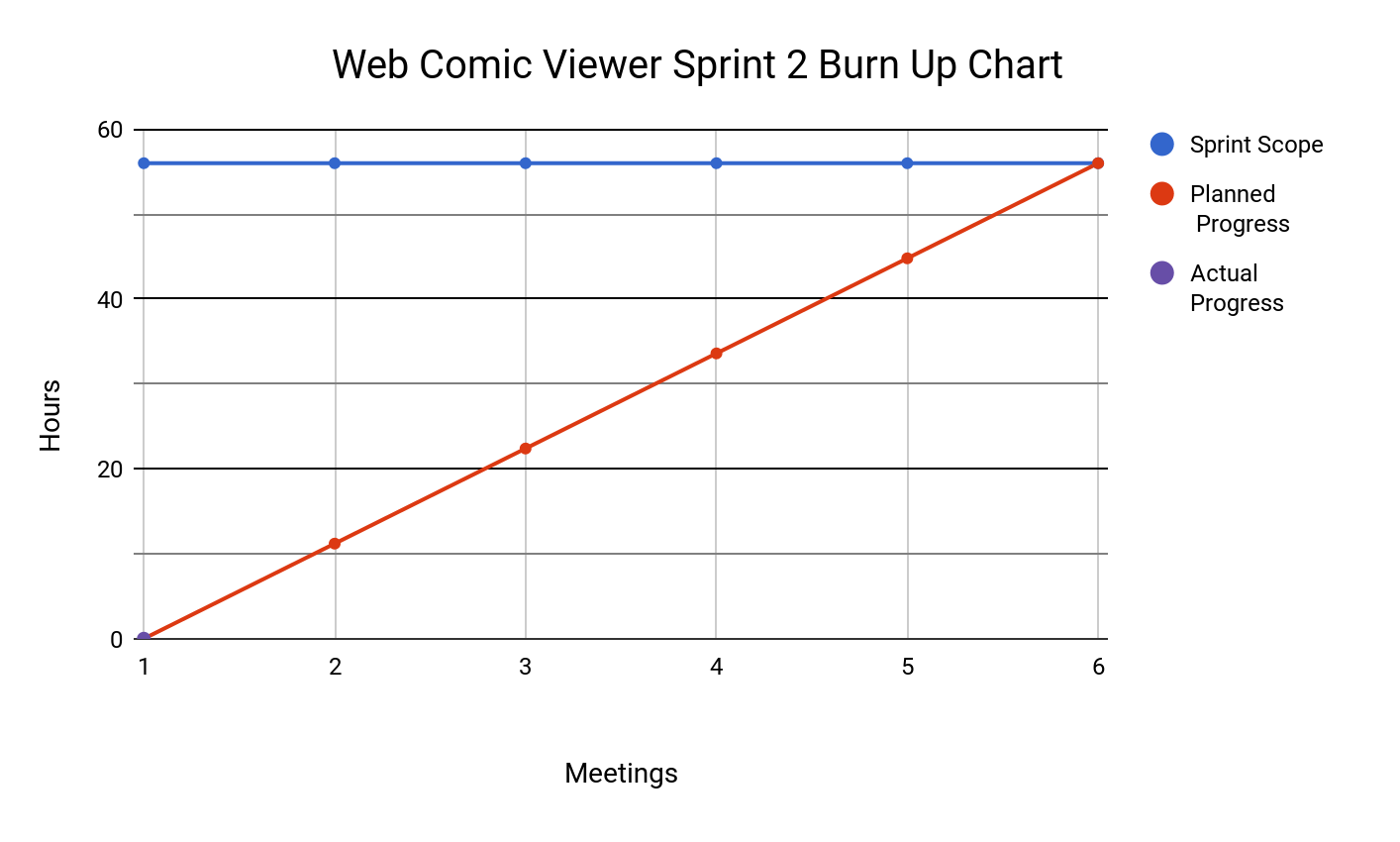
Sammy: User story 4 - Create a backend route to delete comics (3 hrs)

James: User story 1 - Add search UI to comic list (4 hrs)

Mehtab: User story 1 - build Angular search service

Alan: User story 2 - Store read pages in IndexedDB (3 hrs)

**Initial Burnup Chart**



# Scrum Times

Monday: 4:00-4:15

Tuesday: 4:10-4:25 - TA meeting

Thursday: 3:30-3:45