

## **Sprint plan (Week 1)**

- 1) As a Prof, I would like to login to WebWork so that I can access all the tools necessary
  - T1) Create Login Interface so that the prof can login to the application (~2 Points)
  - T2) Create Database to store the Prof info and all the info of the students (~3 Points)
  - T3) Set up a way to prompt when the login info is wrong (~1 Point)
    - o Description: Allow the prof to login to the app and display all the functions that the prof can access once inside the Prof terminal, if he/she enters wrong login credentials, notify them in a clear way.
- 2) As a Prof, I would like to add students to WebWork so that they can participate.
  - T4) Create a page for inputting information for the students (~2 Points)
  - T5) Create a button to upload the inputted information to the database. (~1 Points)
    - o Dependencies: Database (User story 1, Task 2)
    - o Description: Give a prof functionality to add students into the list of students currently participating in the Webwork assignments (with a specific user ID followed by their student number as password)
- 3) As a Prof, I like to upload some course notes so the students can remain on top of their work
  - T6) Upload button to add file from computer (~2 Points)
  - T7) Display files that have been added in the U.I (~4 Points)
    - o Dependencies: Database (User story 1, Task 2)
    - o Description: Create a page where students can view the lecture slides in a clear manner. (Optional: allow them to download the notes)
- 4) As a Prof, I would like to be able to post announcements and pin reminder when needed.
  - T8) Create a field to input announcement info and add it to a list of announcements, with the option of making it visible to prof and all the students. (~3 Points)
  - T9) Create a function to update the board. (~2 Points)
    - o Dependencies: Database (User story 1, Task 2)
    - o Description: A homepage screen where students can view what has been posted along with previous announcements and pinned reminders
- 5) As a Prof, I would like to create and edit problem sets.
  - T10) Create an upload problem set button that will show file system to upload a file (~1 story point)
  - T11) Have a download file system to be able to download the file (~1 story point)
  - T12) Enable/ disable visibility of file when changes need to be made until adjusted. (~3 story points)

T13) Creating a problem set using latex (~3 story points)

T14) Editing the problem sets (~3 story points)

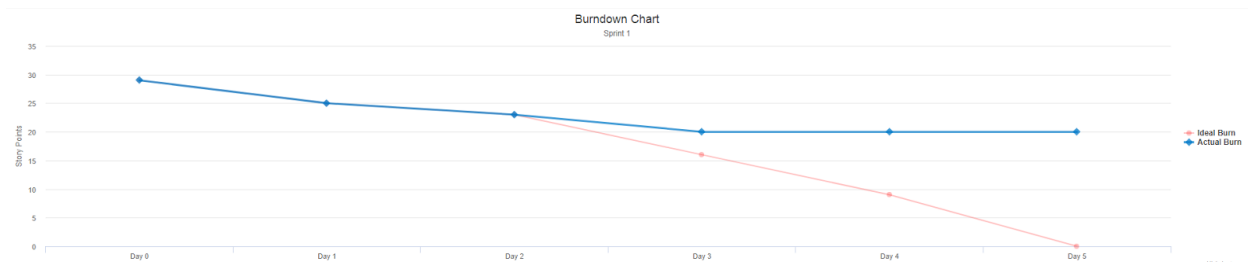
- Description: Create a page where problems can be entered using LaTeX formatting, and then upload save those problems. Desktop Java external libraries need to be remodeled

		Monday	Tuesday	Wednesday	Thursday	Friday
T1	Gagan					
	Usman					
	Shevlin	2				
	Julie					
	Hanson					
T2	Gagan	2	1			
	Usman					
	Shevlin					
	Julie					
	Hanson					
T3	Gagan					
	Usman					
	Shevlin		1			
	Julie					
	Hanson					
T4	Gagan					
	Usman					
	Shevlin					
	Julie			2		

	Hanson					
T5	Gagan					
	Usman					
	Shevlin					
	Julie					
	Hanson			1		
T6	Gagan			2		
	Usman					
	Shevlin					
	Julie					
	Hanson					
T7	Gagan					
	Usman			2	2	
	Shevlin					
	Julie					
	Hanson					
T8	Gagan					
	Usman					
	Shevlin				3	
	Julie					
	Hanson					
T9	Gagan					
	Usman					
	Shevlin					
	Julie				2	
	Hanson					

T10	Gagan					
	Usman					
	Shevlin					
	Julie					
	Hanson					3
T11	Gagan					
	Usman					
	Shevlin					3
	Julie					
	Hanson					
T12	Gagan					
	Usman					3
	Shevlin					
	Julie					
	Hanson					

## Burndown Chart



### **Sprint 1 report**

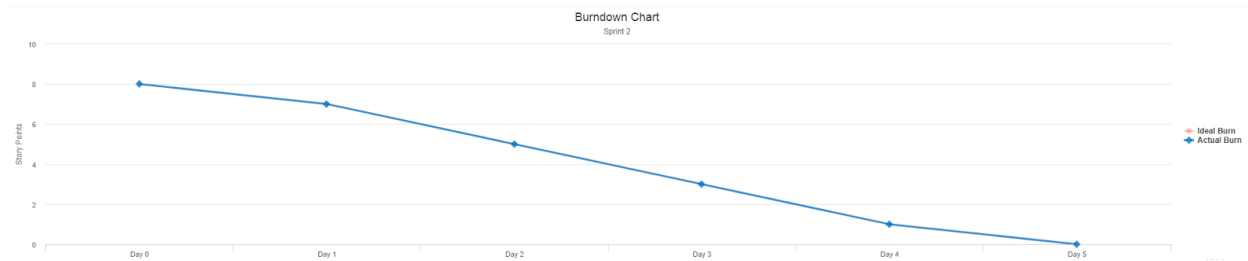
During the development cycle of sprint 1, the plan for the first sprint was to develop an initial project where there two user accounts (instructor and student). There was an roadblock as only two developers were able to start developing on the project. The story points for the two developers that they agreed to burn down increased by around 7 points each. One developer had to implement the backend (database and user account) and the other had to develop the frontend (U.I). The developer who was developing the front-end had to wait for the back-end to be developed but is able to design the overall U.I for the application. By the end of the development cycle the offline version of the database and most of backend feature completed and linked to the U.I of the application. The login page was completed, instructor and students had their own page/activity homepage with a rough U.I implementation. Task that were complete were U1 and U2 were completed (task 1 through 5 were completed) by the end of the development cycle.

## Sprint Plan (Week 2)

- 3) As a Prof, I like to upload some course notes so the students can remain on top of their work
- T1) Upload button to add file from computer (~ 5 Points)
  - T2) Display files that have been added in the U.I (~3 Points)
    - Description: Create a page where students can view the lecture slides in a clear manner.  
(Optional: allow them to download the notes)

		Monday	Tuesday	Wednesday	Thursday	Friday
T1	Gagan					
	Usman					
	Shevlin	1	2	2		
	Julie					
	Hanson					
T2	Gagan					
	Usman					
	Shevlin				2	1
	Julie					
	Hanson					

## Burndown Chart



## Sprint 2 Report

During the development cycle of sprint 2, the only member who worked was Shevlin. This was due to the heavy mid term schedule for the other members of the group. Since the project depended heavily on being able to upload content (notes and assignments) to a server he worked on a task that involved the server, in this case uploading course notes to a server. As such, the story points for this burndown were only the points involved with the task of uploading a file to a server this way future dependencies could be resolved while other members weren't able to work which in turn would allow for future tasks that needed to use the server to have some base code to work with. By the end of this development cycle we were able to upload files to the server from the instructor's view of this application.

## **Sprint Plan (Week 3)**

- 4) As a Prof, I would like to be able to post announcements and pin reminder when needed.
- T1) Create a field to input announcement info and add it to a list of announcements, with the option of making it visible to prof and all the students. (~3 Points)
  - T2) Create a function to update the board. (~2 Points)
  - Dependencies: Database (User story 1, Task 2)
  - Description: A homepage screen where students can view what has been posted along with previous announcements and pinned reminders
- 5) As a Prof, I would like to create and edit problem sets.
- T3) Create an upload problem set button that will show file system to upload a file (~1 story point)
  - T4) Have a download file system to be able to download the file (~1 story point)
  - T5) Enable/ disable visibility of file when changes need to be made until adjusted. (~3 story points)
  - T6) Creating a problem set using latex (~3 story points)
  - T7) Editing the problem sets (~3 story points)
  - Description: Create a page where problems can be entered using LaTeX formatting, and then upload save those problems. Desktop Java external libraries need to be remodeled
- 6) As a student, I would like to view all the announcements posted by the prof.
- T8) Create a page to access all the current announcements (1 Points)
  - T9) Make all the announcements for that specific logged in student show up on the page (2 Points)
  - Dependencies: User story 4, Task 2
  - Description: Create a page where professors can post announcements for students to receive
- 7) As a Prof, I would like to see performance distribution of the class so I can adjust the level of difficulty accordingly.
- T10) Add student marks to database (~2 story points)
  - T11) List Names and Marks of students (~ 3 story points)
  - T12) Create UI for listing names and marks (~1 story point)
  - Dependencies: Database (User Story 1, Task 2)
  - Description: Create a page where a general overview of all the student's performance can be viewed
- 8) As a student, I would like to download a hard copy of the problem sets so that I can work offline.
- T13) List the available problem set (~1 Story Point)



- T14) Create a button on screen for downloading (~1 Story Point)
- T15) Download the selected available problem set (~1 Story Point)
- Description: Create a page where all the problem sets can be viewed and downloaded

		Monday	Tuesday	Wednesday	Thursday	Friday
T1	Gagan	3				
	Usman					
	Shevlin					
	Julie					
	Hanson					
T2	Gagan		2			
	Usman					
	Shevlin					
	Julie					
	Hanson					
T3	Gagan					
	Usman					
	Shevlin					
	Julie					
	Hanson	1				
T4	Gagan					
	Usman					
	Shevlin					
	Julie					
	Hanson	1				
T5	Gagan					

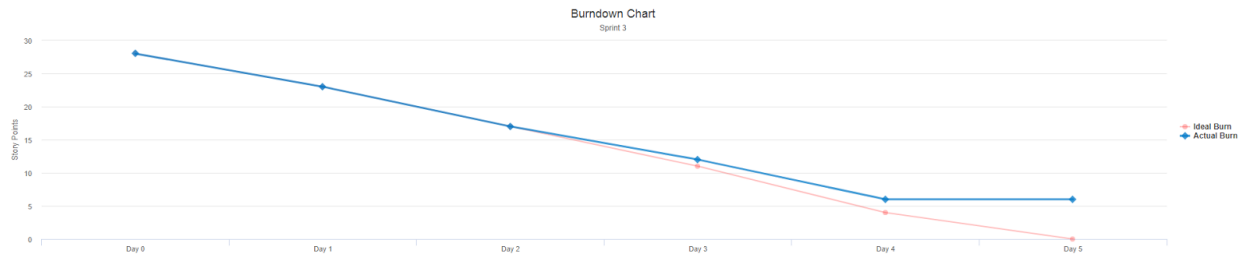
	Usman					
	Shevlin					
	Julie					
	Hanson		2	1		
T6	Gagan					
	Usman					
	Shevlin					
	Julie					
	Hanson				3	
T7	Gagan					
	Usman					
	Shevlin					
	Julie					
	Hanson					3
T8	Gagan			1		
	Usman					
	Shevlin					
	Julie					
	Hanson					
T9	Gagan				2	
	Usman					
	Shevlin					
	Julie					
	Hanson					
T10	Gagan					
	Usman		2			

	Shevlin					
	Julie					
	Hanson					
T11	Gagan					
	Usman			3		
	Shevlin					
	Julie					
	Hanson					
T12	Gagan					
	Usman				1	
	Shevlin					
	Julie					
	Hanson					

T13	Gagan					
	Usman					
	Shevlin					
	Julie			1		
	Hanson					
T14	Gagan					
	Usman					
	Shevlin					
	Julie				1	
	Hanson					
T15	Gagan					
	Usman					

	Shevlin					
	Julie					1
	Hanson					

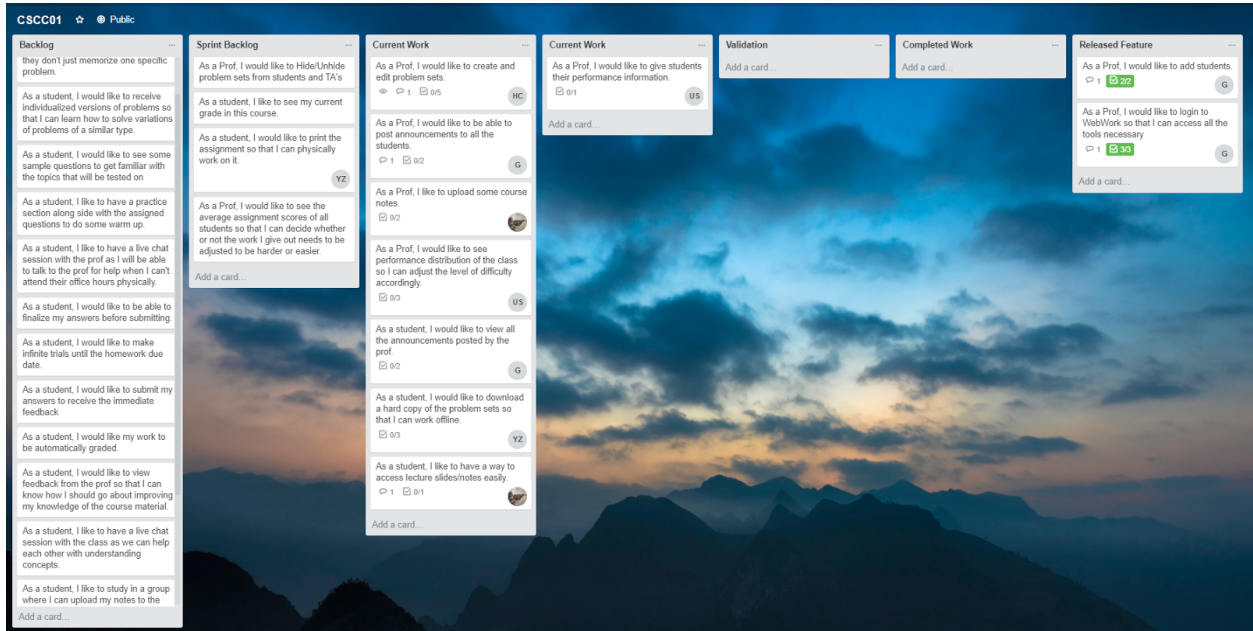
## Burndown Chart



## Sprint 3 Report

During the Development cycle of sprint 3, four of the five developers were able to work on the project application. The five developers had agreed on adding an addition story point to burn down for the sprint week. Each of the four developers were able to work on they own user story without any dependencies to hinder their progress of implementing their task. There were no problems with branching by features. Five branches were made. On mid way through the development cycle, one of the four developer finished ahead of time and implemented the feature to the master branch. Near the end of the week, almost all developer were able to finish their planned tasks, but there were some roadblocks. There were some MySQL syntax issues with Android Studios as there were random errors that were resolved through restarting Android Studios and re-syncing gradle. The other roadblock was when implementing LaTeX on android, as only desktop JAR/Library files were made for desktop web and Java applications. Changes to the JAR and Library files needed to be made so an android port of LaTeX can be made. Four of the five branches were successfully merged to the master by the end of the development cycle. (Task 7, 13, 14, 15 were not completed)

## Kanban (Start of sprint)



## Kanban (End of sprint)

