Sprint 1 Plan

Team Members: Matthew Daxner, Sanjay Shrikanth, Griffen Shu, Arka Pal, Dhatchi Govindarajan

Header

Product Name: *WaveStyled*
Team Name:
Completion Date: April X 2022
Revision Number: 1
Revision Date: April X, 2022

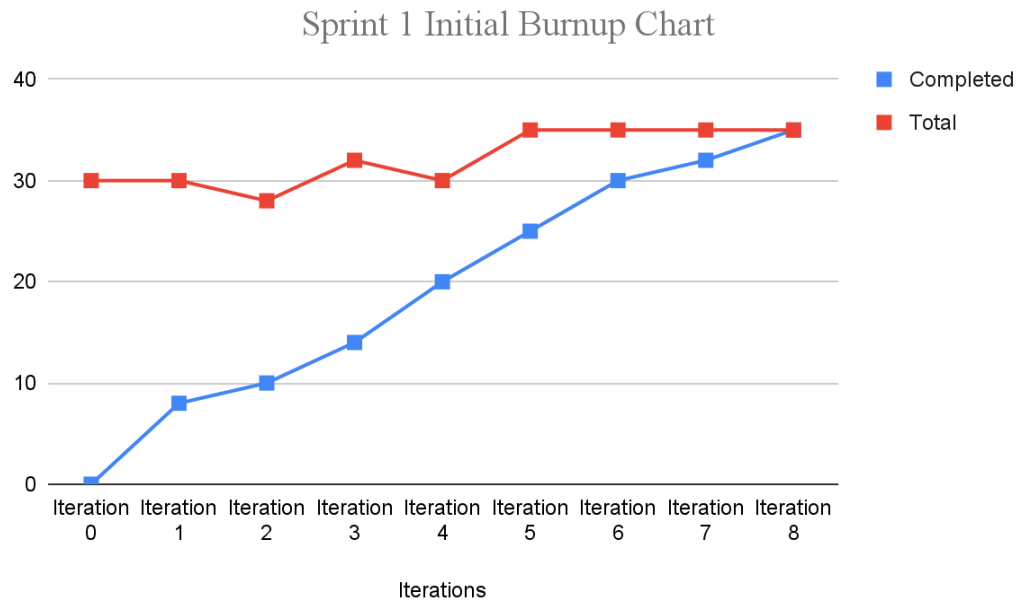
Goal

For each item added to the wardrobe we want to be able to see it on request. Secondly, we want to create a webservice that takes in wardrobe item information and stores it in a persistent database while ensuring efficient transfer of messages.

Task listing, organized by user story

A. ***(Story Points: 8)*** As a user I want to be able to add items from my actual wardrobe to my virtual wardrobe so that I can easily visualize everything I own. ***(Priority: 1)***

- ☐ **Task 1:** Create a script that generates a random wardrobe that conforms to the database constraints for testing purposes
  i. Time: 1 hour
- ☐ **Task 2:** Draft a skeleton Python object class framework that encapsulates the clothing item and wardrobe constructs. The class constructors should be able to take in the database data for each item and convert them to usable Python objects
  i. Time 2 hour
- ☐ **Task 3:** Create a Python REST API web service that receives information from the Node.js server and converts the database information into Python objects
  i. Time: 5 hours
- ☐ **Task 4:** Add endpoints to the Python framework so that on request, the wardrobe or any specific item can be returned to the user
  i. Time: 3 hours
- ☐ **Task 5:** Explore ReactNative and get a rough idea for how to display the planned UI structure on the application(**Spike)**
  i. Time: 9 hours



B. ***(Story Points: 5)*** As a user, I want my wardrobe to be saved when I close the app so that I do not need to re-enter it whenever I reboot the app. ***(Priority: 2)***

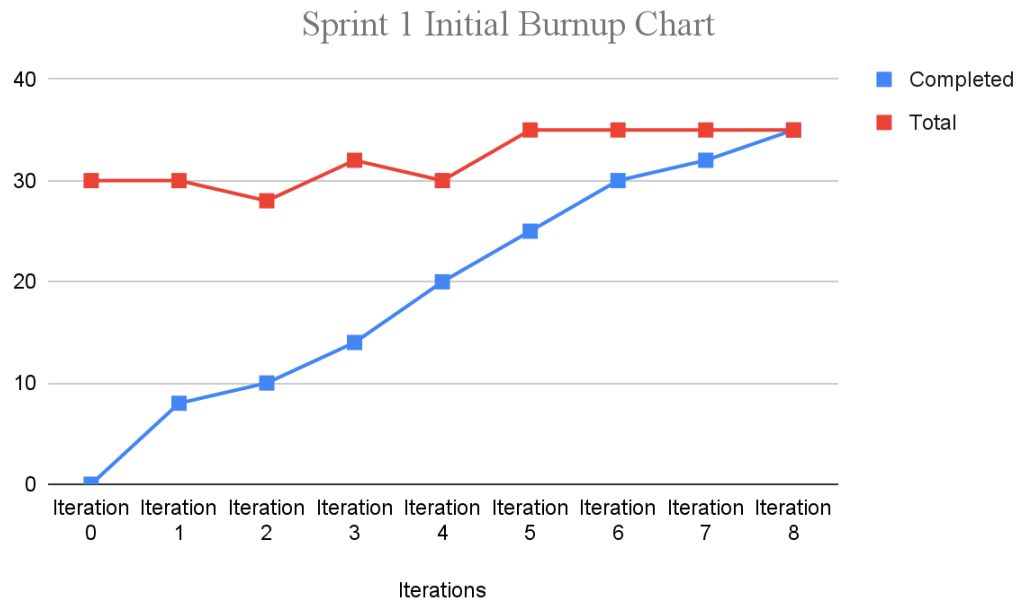
- ☐ **Task 1:** Plan out database table structure and create a SQL database & table that stores all necessary information about items in the wardrobe
  - Time: 3 hours
- ☐ **Task 2:** Create the Node.js server that servers as the middleware of the webservice & facilitates communication between user database
  - Time: 5 hours (Not much experience with Node.js: **Spike**)
- ☐ **Task 3**. Add initial HTTP endpoints to the Node server to begin the communication framework (add, put, delete, etc.)
  - Time: 2 hour

Team Roles

Matthew Daxner: Product Owner, Developer
Sanjay Shrikanth: SCRUM master, Developer
Griffen Shu: Developer
Arka Pal: Developer
Datchi K: Developer

Individual Task Assignments

Matthew Daxner: Story A: 1, 2; Story B 1,2
Sanjay Shrikanth: Story A: 2,3,4
Griffen Shu: Story A: 4,5
Arka Pal: Story A: 4,5
Datchi K: Story A: 5; Story B: 3



Initial Burnup Chart

https://docs.google.com/spreadsheets/d/1aur9ZGkMCvPcCBaK3BYVBb-tmrpljIz9YhSZxTXslYg/edit?usp=sharing

Initial Scrum Board

User Stories	**Tasks not started**	**Tasks in Progress**	**Tasks Completed**	
A		4,5	1,2,3	
B		2	1,3	

Scrum Times

Day	**Time**
Tuesday	**Between 2-3 pm**
Thursday	**3:30-4:15pm w/ TA Jayjeet**
Sunday	**Between 12-1 pm**