



# Sluggit

*A Computer Science blog and resource*

Team Members: Jasmine Dupree, Maximilian Brain, Tarum Fraz, Diego Garcia, Hope Ashcraft(SM), Chris Gradwohl(PO)

## **Document Name: Sprint 3 Plan**

Sprint 3 Completion Date: 06/05/2017

Revision Number: 3.1

Revision Date: 05/17/17

## Goal

1. Finish Post functionality, the user should be able to create, read, update and delete posts.
2. Format User Profile, and add more functionality
3. Polish final product

## Task Listing/ User Stories

**A. As a user I would like a feed to display the latests posts and the most popular. (45 story points)**

### FRONT END

1. Create a basic NG Post Service
  - a. Retrieve stored post data with existing api
  - b. Design api to have better functionality and cleaner calls

### BACKEND

2. Api methods for CRUD
3. Add edit and del endpoints and methods
  - a. Make a api methods to grab posts by id
  - b. Make a url on the backend for each post, easiest way is to concatenate the post id to a url

### FRONT END

4. Make NG Post service to make a get request to post url
5. make post clickable takes you to that posts url

**B. As a Sluggit user I want to be able to be able to 'like' blogs so that I can shape the landscape of the sluggit community. (25 story points).**

**BACKEND**

- a. Add 'like' functionality to the blog MongoDB model
- b. Update the DB methods to increment likes
- c. Update the API to update the likes field in the Post model

**FRONT END**

- d. Add 'like' functionality to the angular post service
- e. Update necessary all post components to reflect likes

**C. As a Sluggit User, I want to be able to customize my profile (30 story points)**

**BACKEND**

- a. Add 'about me' in Mongoddb User field or perhaps a different 'profile' model
- b. Create profile api routes to update the User fields

**FRONT END**

- c. Add edit functionality to components
- d. Create a 'profile' angular service for updates and saving data
- e. Update the profile component to reflect new features
- f. Add followers?

## Team Roles

Jasmine Dupree: Developer

Max Brain: Developer

Tarum Fraz: Developer

Diego Garcia: Developer, Scrum Master

Hope Ashcraft: Developer

Chris Gradwohl: Product Owner, Developer

## Initial Task Assignment

Jasmine Dupree:

B. As a Sluggit user I want to be able to be able to 'like' blogs so that I can shape the landscape of the sluggit community. (30 story points).

- f. Add 'like' functionality to the blog MongoDB model
- g. Update the Post model to store likes
- h. Update the DB methods and API to update the likes field in the Post model
- i. Add 'like' functionality to the angular service.

Max Brain:

User story B.

- i . Add 'like' functionality to the blog MongoDB model
- ii. Update the Post model to store likes
- iii. Update the DB methods and API to update the likes field in the Post model
- iv. Add 'like' functionality to the angular service.

Tarum Fraz:

User Story C. As a Sluggit User, I want to be able to customize my profile

- g. Add more fields in Mongoddb
- h. Fix front end format
- i. Add followers

Diego Garcia:

User Story B. As a user I would like a feed to display the latests posts and a popular display

- A. Want to display posts based on timestamp
- B. Note that the home public feed requires no styling it is just a container of blog post objects

Hope Ashcraft:

User Story A. As a user I would like a feed to display the latests posts and the most popular. (45 story points)

- A. Create a basic NG post service to retrieve stored post data with existing api
- B. Create the API methods for CRUD
- C. Make posts clickable so it takes you to that posts url

Chris Gradwohl:

FRONT END

- 1. Create a basic NG Post Service
  - a. Retrieve stored post data with existing api
  - b. Design api to have better functionality and cleaner calls

BACKEND

- 2. Api methods for CRUD
- 3. Add edit and del endpoints and methods
  - a. Make a api methods to grab posts by id
  - b. Make a url on the backend for each post, easiest way is to concatenate the post id to a url

FRONT END

- 4. Make NG Post service to make a get request to post url
- 5. make post clickable takes you to that posts url

MISC.

- 6. Merge code into master.
- 7. Push style guide and test framework.

## Initial Burnup Chart

[Trello Burndown](#)

## Initial Scrum board

[Sluggit Scrum Board](#)

## Scrum Times

Monday: 12:00pm - 1:00pm

Wednesday: 12:00pm - 1:00pm

Friday: 12:00pm - 1:00pm

TA: Kavya Jha

Meetings: Wednesday at 3:40pm