**T.O.S.S.**

**This.One.Study.Shows**

Link to site: https://toss-cen.herokuapp.com/

Link to GitHub: https://github.com/connor-giles/Toss/

Log-in Credentials:

Username- "paolo.d2@ufl.edu"

Password- “password”

Table of Contents

* Introduction
* Project Description
* Project Features
* Artifacts
  + Project Proposal
  + Sprint 2 plan
  + Sprint 3 plan
  + Product Backlog
  + Wireframes
  + System Context Model
  + User Stories and Story map
  + Design Vision Sprint
  + Usability Test
* Project Handoff Guidelines
  + APIs with Keys
  + Environmental Variables
  + Log-In Credentials
  + ReadMe file
  + Config file

Introduction

T.O.S.S., or *This One Study Shows,* aims to address a very specific trend in society: stubbornness in beliefs. Oftentimes in an argument or debate, people will bring up a single source (academic or otherwise) that supports their point, and disregard any context or possible other points of view.

T.O.S.S. aims to change that by exposing users to unique and different viewpoints through an easy to use webapp. It does this by allowing users to respond to a number of predetermined or user-submitted controversial and/or thought-provoking topics or prompts, giving their full opinion and beliefs. It then shows a user the responses made by other users, and prioritizes showing responses from those whose values differ most radically.

This is done using the MFT, or Moral Foundations Test, that every user takes when making an account. The MFT is a series of questions that asks the user how they feel about a variety of morally conflicting choices, ranking their response on a numerical scale, and then determining a score for five core values. When participating in a T.O.S.S., users are shown responses from users whose five core values differ the most from theirs. Combined with the mandatory academic source provided with each response, this allows T.O.S.S. to expose users to new and differing, as well as academically supported beliefs.

Project Description

T.O.S.S. is a webapp that allows users to respond to and write responses to a number of thought-provoking questions about controversial or interesting topics in society.

Users make an account, take an MFT (Moral Foundations Test) that determines their score on five key values, and then can participate in Tosses. Tosses involve three phases of writing responses and viewing the responses of other users. This allows users to interact with others, and also be assured that their responses are being seen and considered by others.

When responding to a TOSS prompt, users will be able to provide a credible source/study that is supporting their perspective below their response. This is one of the main differences between TOSS and other websites. Usually users make statements without any credibility, however when creating TOSS, the TOSS team wanted to create a website where users will be able to discuss important topics with credible sources to support their arguments. The goal of this was to create productive conversation without users just sharing their opinions.

Users also have access to a profile page that displays various account details such as username, a profile picture, and MFT scores. They also have the ability to retake the MFT to change their results if desired.

Project Features

* Allows users to create an account with username/password. Has log-in and register pages.
* Profile page featuring profile picture, username, MFT scores table, and username and email of user signed in
* Moral Standings Test and table displaying 5 key values
* About page, describing the purpose of the project.
* Allows users to submit a 500 character response to TOSS prompts along with a credible source about the topic
* Allows users to view other users responses on a prompt after 24-hours
* TOSS prompt clock countdown on home page

Artifacts

**Project Proposal:**

Team name: TOSS Team

**Team Roles:**

Paolo Colmenares – Project Leader / Frontend

Connor Giles – Backend

Ashish Goolla – Scrum Master / Backend

Anna Le – Frontend

Evan Rocha – Frontend/Backend

**Project name and description:**

*Project name*: Toss: This One Study Shows

*Description*: Development of a web-based application whose purpose is to facilitate conversation between people who would otherwise only have passionate, emotive-based and not evidence-based conversation in-person (or online), resorting to empty-handed platitudes like “well, this one study shows…”. Users can search for a particular topic or choose to be ‘Toss’d’ into a random topic and corresponding prompt (image, news, viral tweet, or idea) and will be given the opportunity to develop their platform/stance centered around linked and referenced sources. Users (1-4 per “Toss”) participating have the option for these conversations or debates to be shown in public for others to weigh in, concur/dissent ideas, and possibly vote.

*End goal:* With this platform people will become acquainted with perspectives and ideas completely foreign to them and will be more inclined to considering them and their proponents because of the rational, evidence backed conversations. Targets misinformation and socio-political deaf/blindness.

Software System: Web Application

Project Management tool: Trello

**Sprint 2 Plan:**

**Paolo Colmenares**

1. Set up ‘Toss Database’; including schemas, routing and related backend for Phase 1 (viewing and answering Toss prompts)
2. Frontend UI for viewing Toss prompts in Phase 1
3. Frontend UI for answering Toss prompts in Phase 1
4. Flesh out more of the specifics of interactions between parts of the app.

**Ashish Goolla**

1. Setting up and organizing MongoDB database
2. Do research into the development of a User Profile/Settings API
3. Connect the API to the ‘User Database’ so user login and passwords can be stored
4. Implement functionality of resetting password
5. Work with Anna to implement the User Profile Frontend
6. Work with Evan on incorporating his Moral Standings Test API into ‘User Database’

**Anna Le**

1. ‘Finalize’ a general direction and scheme for the homepage/landing page of the web app
2. Execute a basic version of #1, focusing on:
   1. Frontend for displaying active and past Toss’s
   2. Frontend for entering a new Toss (Options for choosing filters for retrieval of Toss prompt from database)
3. Integrating ticking timers into display of active prompts
4. Toss logo
5. Work with Ashish to implement the User Profile Frontend

**Evan Rocha**

1. Do research into ClassMaker API for development of the Moral Standings Test
2. Create associated quiz for Moral Standings
3. Implement backend of taking said quiz into app
4. Routing of results of MST to ‘User Database’ (working with Ashish)
5. Incorporation of MST stats into User Profile page (working with Ashish)

**Connor Giles**

1. Setting up and regulation of version control system and the base of the app (file structure and organization)
2. Do research into the development of a User Registration/Login API
3. Backend for new user registration and user login system

**Sprint 3 Plan:**

**Paolo Colmenares**

1. Figure out and implement the isLoggedIn perpetual state for users that wraps the app.
2. Characterize style/theme for the app and work on applying it to the frontend/display
3. Functionality for the phases of the Toss’s
4. Backend/Frontend for Archive/Search bar for past Toss’s
5. Possibly implement functionality for users submitting a prompt/Toss

**Ashish Goolla**

1. Continue working on Google Sign-In API with Connor
2. Do research into the development of a User Profile/Settings API
3. Implement functionality of resetting password
4. Work with Anna to implement the User Profile Frontend

**Anna Le**

1. Home page diagram (create individual boxes to display each category)

2. Join Toss buttons (specific filters and etc)

3. Timer on front page so user knows how long a Toss phase is open for

4. Toss logo

5. Frontend profile page

**Evan Rocha**

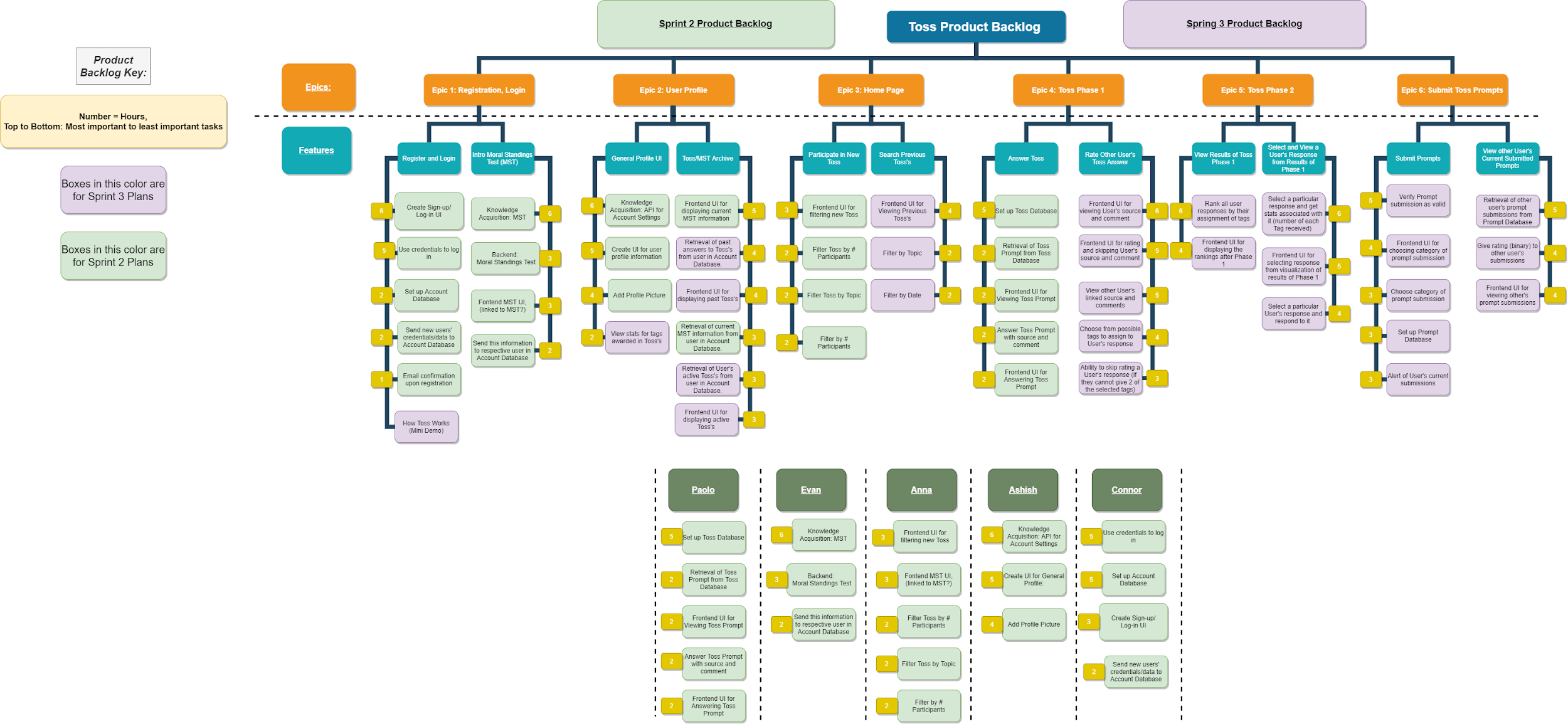
1. Improve styling/ease of use of MST with CSS or other tools.
2. Integrate MST results with how Tosses are assigned to users.
3. Add visualization of MST results to the profile page for users to see a visual representation of their test results.
4. Add quiz to the registration process so users don’t have to click away to take the quiz, while still giving users the ability to take the quiz again somewhere from the homepage to change their results.

**Connor Giles**

1. Continue to fix issues with Heroku and get app deployed
2. Continue working with Google Sign-In API and link with isLoggedIn state to determine user profiles
3. Setup user profile page to include MST values, profile info and more (joint work with Anna)
4. Email confirmation setup

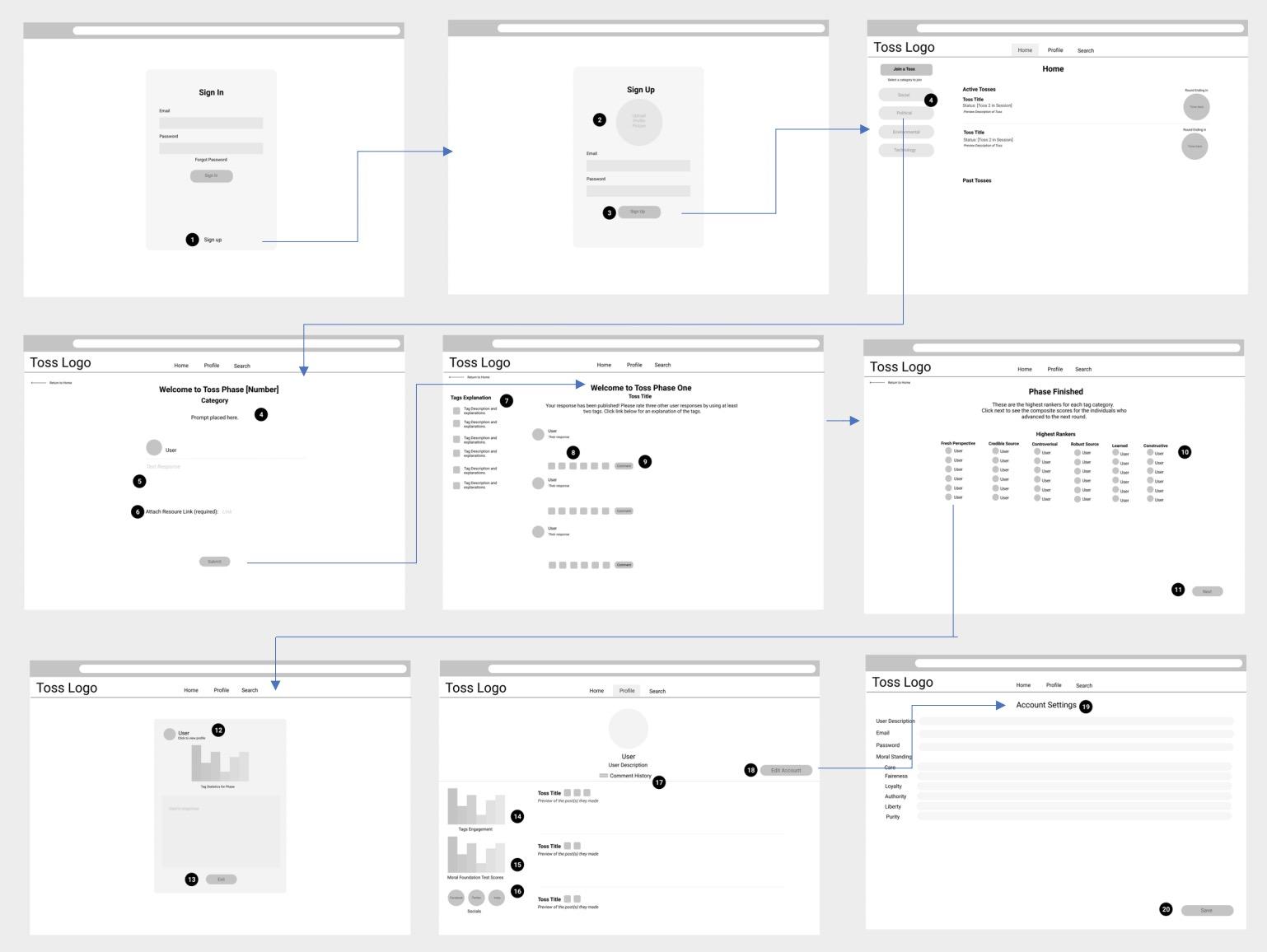
**Product Backlog:**

(We know this is small to see, so we have included the product backlog link below) https://drive.google.com/file/d/149l6GKNft7Ns6dQKazKkxo3hM2kUNVEy/view?usp=sharing

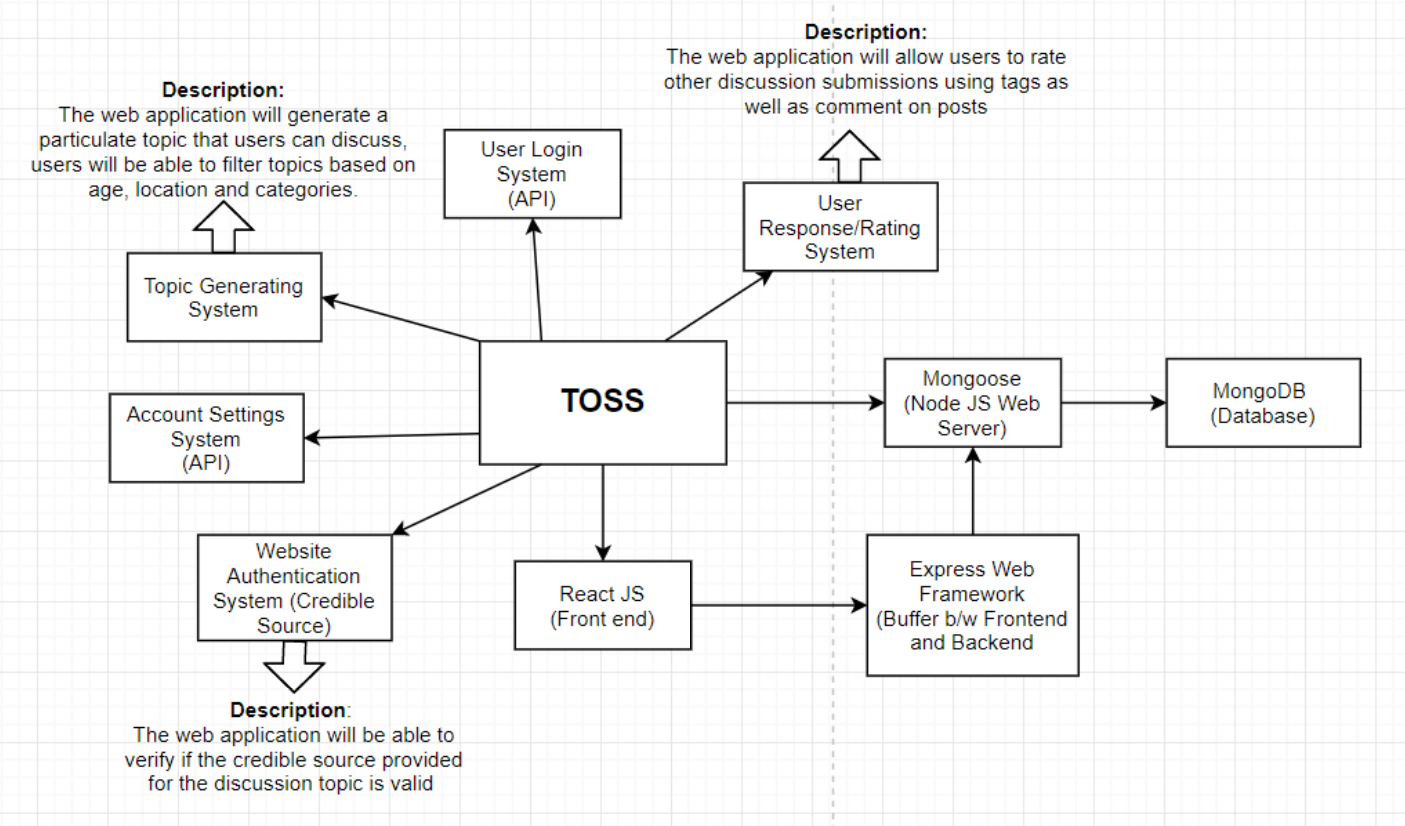


**Wireframes:**

1. Sign up button at the bottom of the Sign In page takes users to a new page to create an account.
2. Users have the option to upload a profile picture.
3. Once users submit the information for a new account, the website directs users to the Home page for Toss. Here, they have the ability to view Active Tosses (once they have participated in them) and the past Tosses that they were in.
4. On the left side of the Home page, users can pick which category they would like to join for a Toss. Once they pick one, the website takes users to a new page where they are presented with a randomly generated prompt.
5. Users can enter their text response to the prompt here (with regard to a character limit).
6. A resource link must be attached at the bottom for the response to be submitted.
7. After their response is submitted, Users enter a new screen for the first Toss phase. On the left side, tags descriptions listed for users to rate other responses accordingly.
8. Each response displays the set of tags for users to choose from (required to choose at least two).
9. A comment button is also available.
10. Once the phase ends, users may proceed to a new page where the individuals with the highest scores for each tag are presented.
11. Clicking on a profile name takes the users to a different screen to view the response of that user to the prompt and the statistics for their tags.
12. Users may click “Click to View Profile” to see that user’s profile account.
13. Exit returns to the rankings page.
14. Should users click the button mentioned in 12, users can view another user’s entire account. Here, on the left side. statistics are presented for their tag engagement.
15. Statistics for their Moral Standings test are also displayed.
16. Buttons linking to their social media accounts listed.
17. A list of their responses/comments displayed in a thread with the title of the Toss and any tags they received next to it.
18. Users have the ability to edit their account information. Pressing this directs them to Account Settings.
19. Page lists text fields for users to make any changes to their account.
20. Save ensures that information is stored.

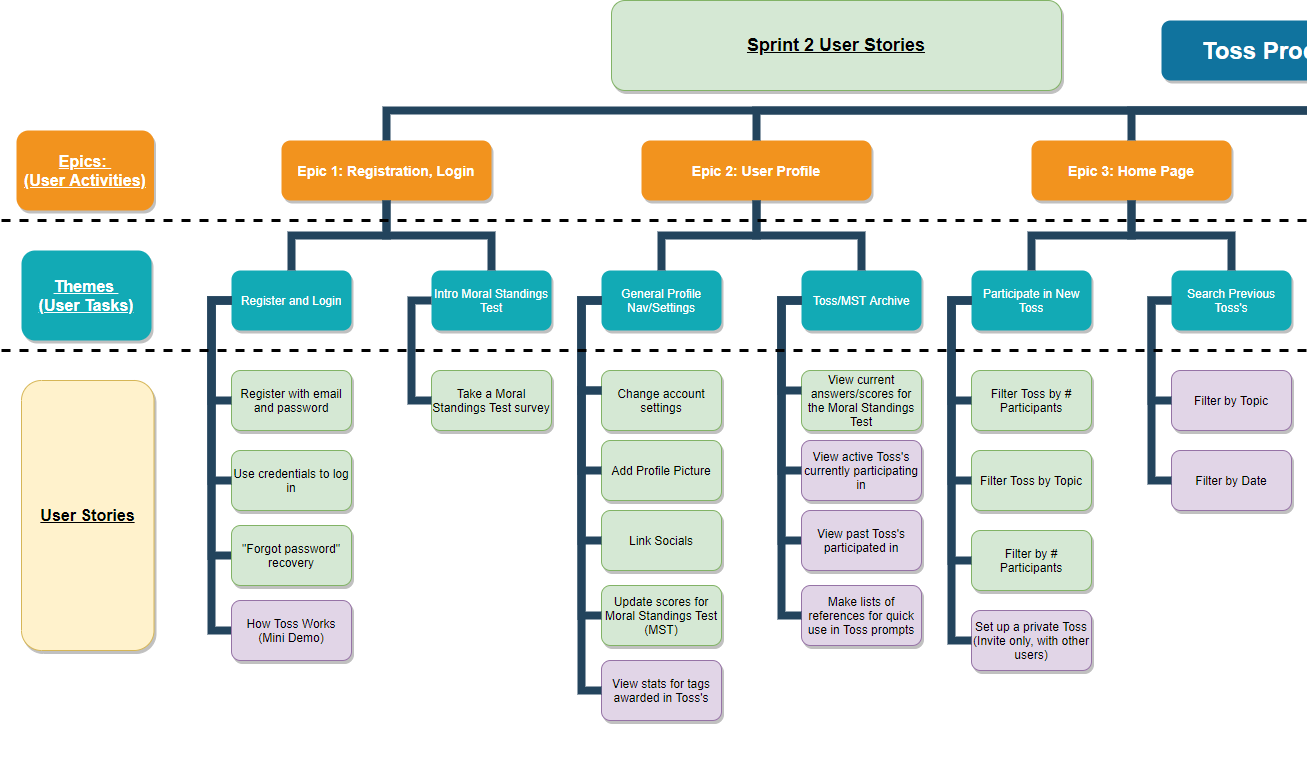


**System Context Model:**

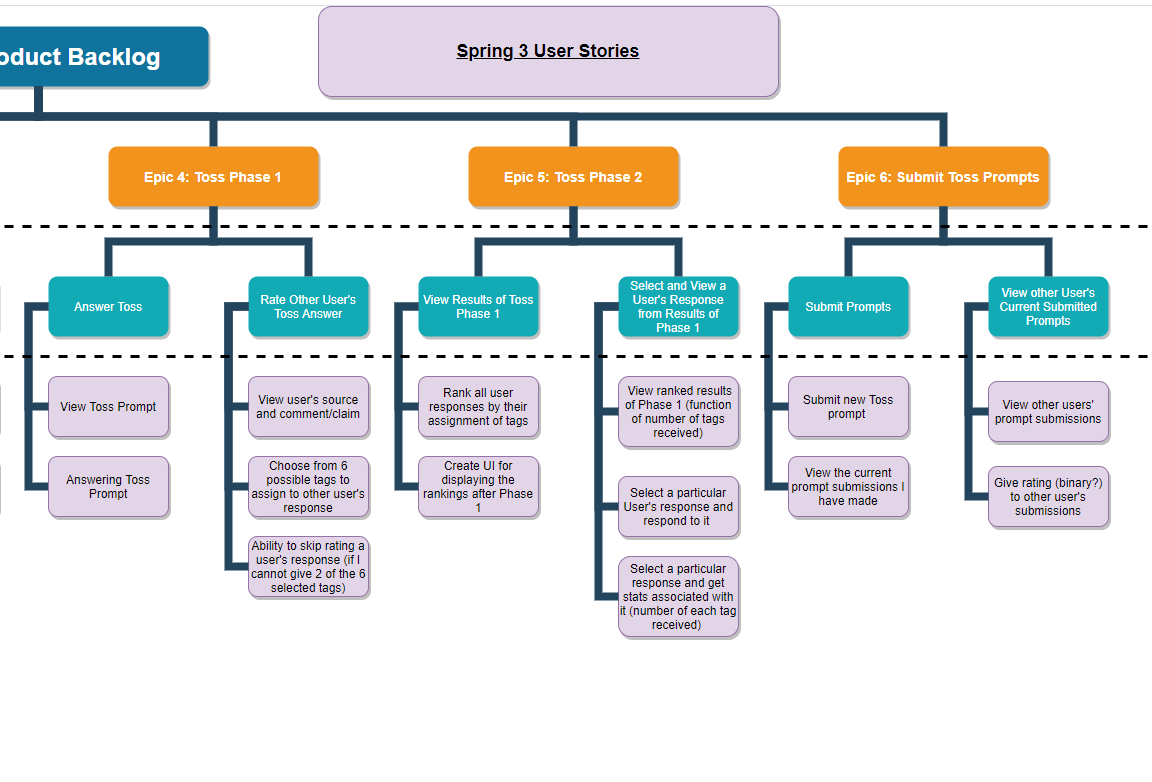


**User Stories and Story Map:**

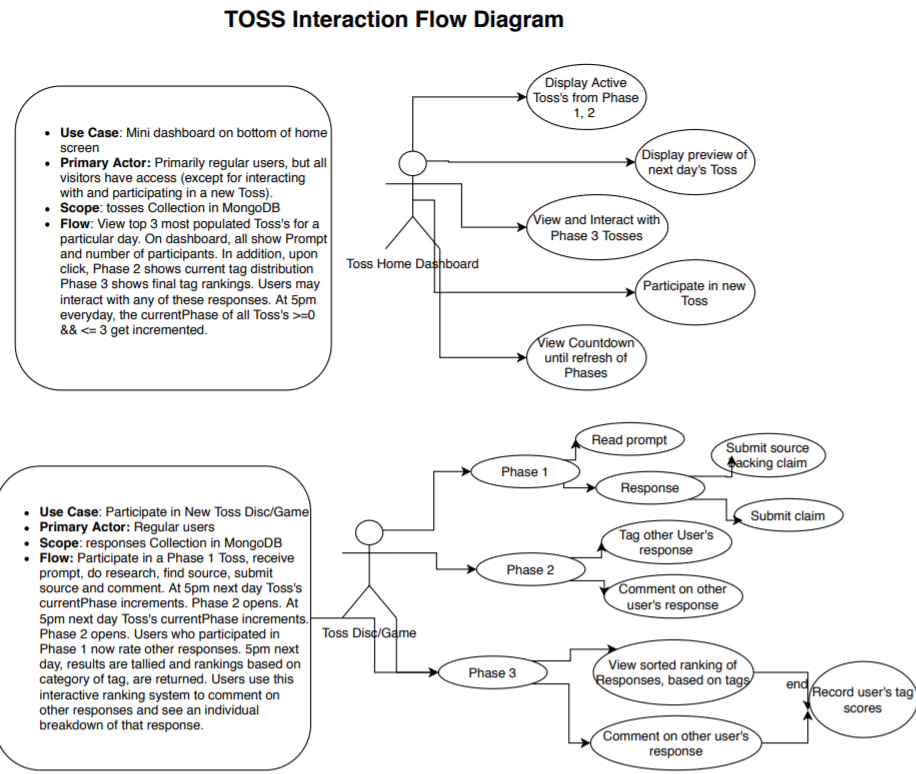
Story Map Sprint 2:



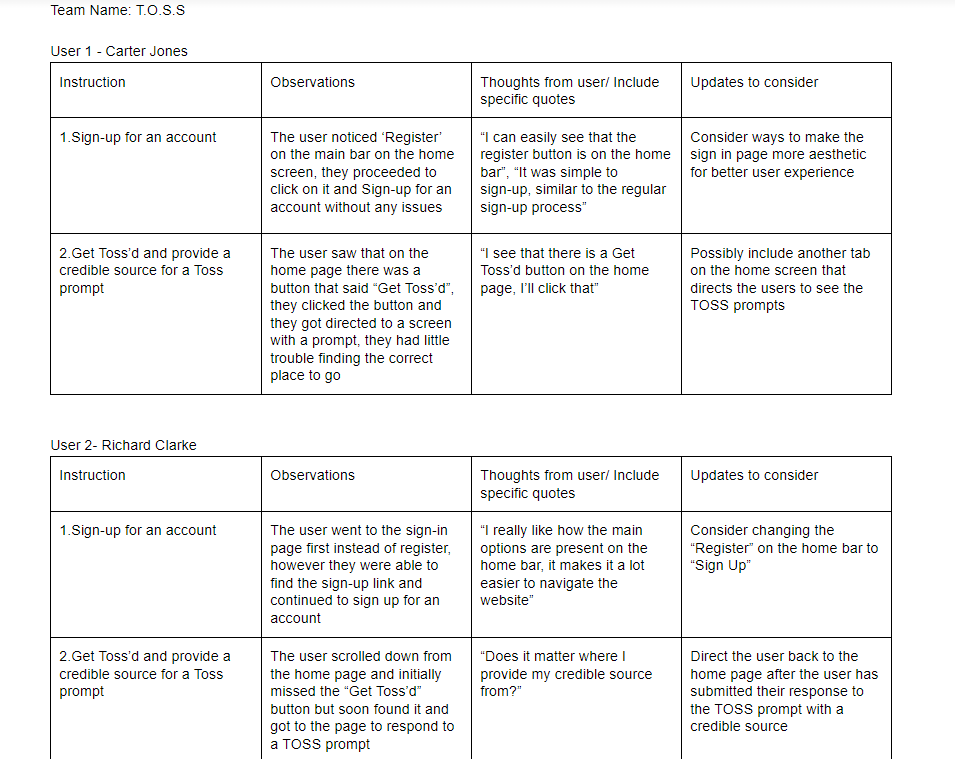
**Story Map Sprint 3:**

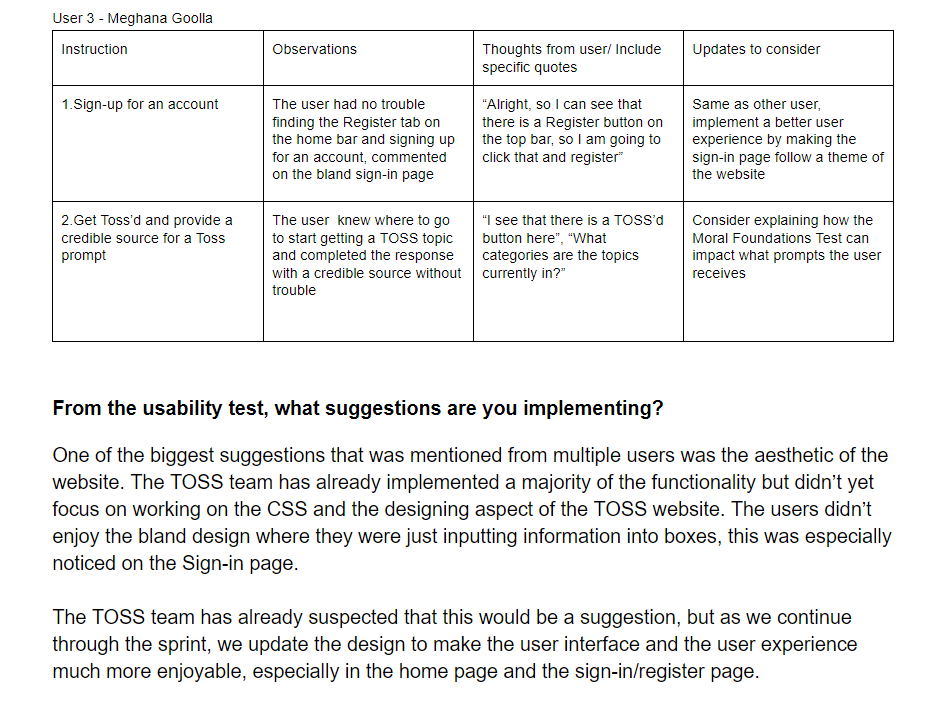


**Design Vision Sprint** (Use Cases & interaction flow diagram):



**Usability Test:**





Project Handoff Guidelines

**APIs (with keys) Step-by-step:**

* Moral Standings Test API
* User Login API
* Materials UI API

**Environmental Variables:**

PORT=3001

DATABASE\_LOCAL=mongodb://localhost:27017/DATABASE

DB\_URI=mongodb+srv://tossDev:tossDev123@tosscluster.oq6ub.mongodb.net/TossData?retryWrites=true&w=majority

DB\_TOSSDATA=TossData

JWT\_SECRET=UF-FALL-CEN3031-PAOLO-COLMENARES-TOSS-PROJECT-22

JWT\_EXPIRES\_IN=90d

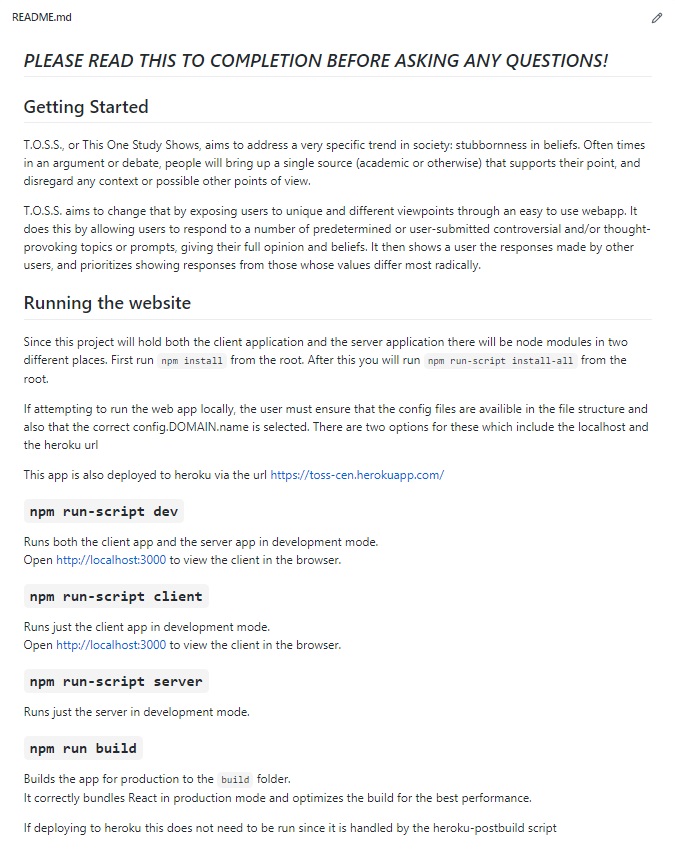
JWT\_COOKIE\_EXPIRES\_IN=90

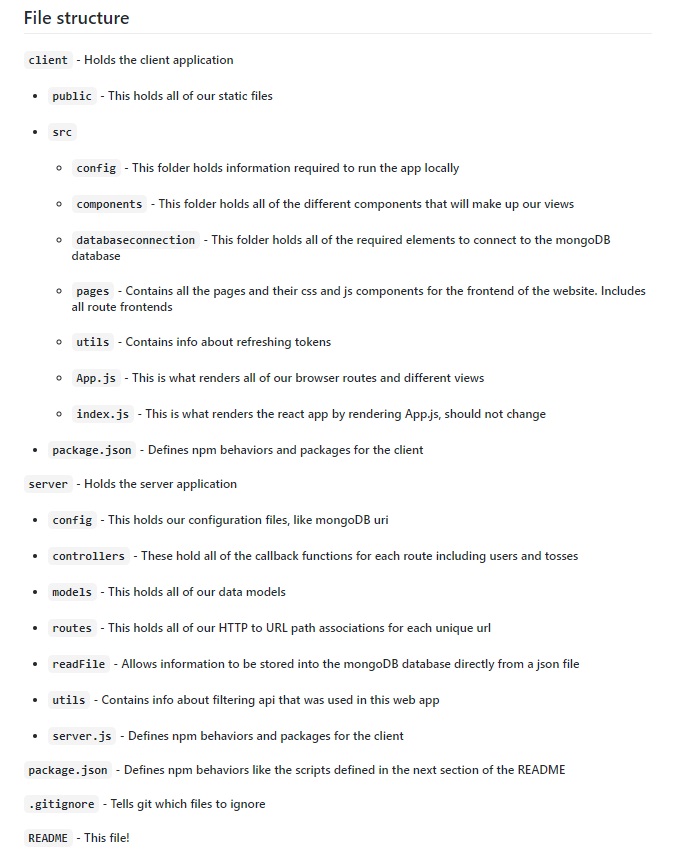
**Log-in Credentials:**

Username - "paolo.d2@ufl.edu"

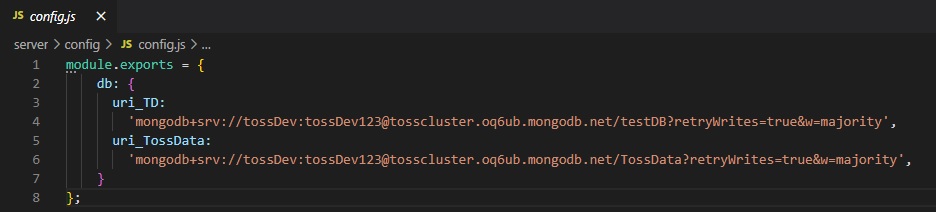
Password - “password”

**ReadMe File**

****

****

**Config File:**

****