Team 14 - Sprint 2 Planning Document

**Hot TopiX**



**Team Members:** Connor Brown, Hunter David, Steve Eisner, Baxter Govan, Ryan Klinedinst, Victor Pan

**Sprint Overview:** For our second sprint, we have explicit goals for the frontend, backend, and database. For our front end, we plan to further build out a number of our existing UI pages to handle our user registration, user login, and profile customization functionality. We are also planning on creating additional UI pages to further our progress. Note that we are switching our implementation from Java and MySQL to Node.Js and MongoDB, since these will work much better with our program. Also, we plan to focus heavily on integrating our front end and back end, since we were unable to accomplish this during sprint one.

**Scrum Master:** Hunter David

**Meeting Plan:** Wednesday, 7PM, WALC 3060, Tuesday & Thursday 1:30-2:30 LWSN B155

**Risks and Challenges:** The main challenge we expect to face for this sprint is our pivot away from Java and MySQL toward Node.Js and MongoDB. Though these tools are a better fit for our scope and we already have tests with these platforms working, we still view one of our primary hurdles to be getting the whole team up to speed with these changes without losing time. To accomplish this, we’ve multiplied our meeting times by three for the second sprint, increasing the time that people working on different tasks have to collaborate face-to-face.

**Current Sprint Detail:**

**User story #1:** As a user, I would like to register a new account.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Store new User information and retrieve from MongoDB | 4.5 Hrs | Hunter |
| 2 | Encrypt password prior to storage in the Database | 2 Hrs | Steve |
| 3 | Connect the Node.Js Server to the Bootstrap Create User Page | 4 Hrs | Steve |
| 4 | Implement a check for invalid email or existing email | 2 Hrs (Each) | Connor/Victor |
| 5 | Implement affirmative message for successful account registration | 1 Hr | Steve |
| 6 | Testing | 1 Hr | Connor |

Acceptance criteria:

* Given a user attempting to register an account with a valid email address, when they register, then their information will populate our database. The user will then receive an affirmative message and be able to login to their registered account.
* Given a user attempting to register an account with an invalid email address or an already registered email, when they register, then they will receive an error message.
* Given a user not wanting to make an account, when they click ‘go back’, they will be redirected back to the landing page

**User story #2:** As a user, I would like to be able to log into my existing account.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Verify that the input email and password match values stored in the database | 3 Hrs | Connor |
| 2 | Implement error message given invalid information | 2 Hrs | Connor |
| 3 | Forward successfully logged in users to their Timeline | 1 Hr | Connor |
| 4 | Add database support for retrieving a user given email/password | 2 Hrs | Hunter |
| 5 | Testing | 1 Hr | Connor |

Acceptance criteria:

* Given a user with a valid email address and password, when a user logs into Hot TopiX, then they are redirected to their timeline
* Given a user with an invalid email address, when a user logs into Hot TopiX, an error message pops up stating it’s an invalid email address
* Given a user with a valid email address and an invalid password, when a user logs into Hot TopiX, an error message pops up saying their password is invalid

**User story #3:** As a user, I want to be able to edit my profile

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Connect the database to enable sending and receiving calls to change aspects of user’s profiles | 8 Hrs | Victor |
| 2 | Setup the UI to connect to the backend and change account info in the database | 5 Hrs | Victor |
| 3 | Testing | 1 Hr | Victor |

Acceptance criteria:

* Given a logged in user, when they want to edit their profile, they are presented with the edit profile page
* Given a logged in user ready to submit their profile edits, when they click ‘save’, their changes are saved to the database and their profile is updated
* Given a logged in user, when they want to delete information from their profile, they can make all non-required account info blank

**User story #4:** As a user, I want to be able to customize a brief bio for my profile

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Write call to edit the profile field in the database | 5 Hrs (each) | Hunter, Baxter |
| 2 | Display a user’s bio at the top of their profile page | 3 Hr | Steve |
| 3 | Unit tests | 1 Hr | Steve |

Acceptance criteria:

* Given a user with an existing profile, when they click edit profile, then they have the ability to add or edit a short bio
* Given a user with an existing profile, if they don’t want to save their changes, then they are redirected to their dashboard
* Given a use with an existing profile, when they are ready to save their changes, then their changes are pushed to the database and their bio is updated

**User story #5:** As a user, I want to be able to select a display image for my profile

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Add database support for a display image | 3.5 Hrs | Hunter |
| 2 | Implement a way to store a profile image in the database and connect it to a user’s profile | 3 Hrs | Victor |
| 3 | Display the user’s selected profile image at the top of their profile page | 4 Hrs | Ryan |
| 4 | Display the user’s selected profile image next to their posts | 3 Hrs | Steve |
| 5 | Testing | 1 Hr | Hunter |

Acceptance criteria:

* Given a valid .PNG or .JPEG image file, populate the user’s profile information with the image.
* Given a user attempting to view another user’s profile, the profile should display the user’s selected profile image at the top of the page.
* Given a user browsing the timeline, all posts should have the author’s profile picture displayed adjacent.

**User Story #6** As a user, I would like to be able to publish a microblog to the public

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Add microblog to user’s profile | 3 Hrs | Ryan |
| 2 | Store microblog in user’s post history in the database | 5 Hrs (each) | Hunter, Victor |
| 3 | Testing | 1 Hr | Steve, Victor |

Acceptance criteria:

* Given a user who wants to create a post, there should be a button for creating a new post
* Given a user clicks the new post button, there should be a textbox where the user can type their post and there should be a way to add topics (min 1 - max 5) to a post
* Given a user submits a post, the text written and the topics chosen will show up on a new post; users who follow the user and at least one topic should see the post in the timeline

**User story #7:** As a user, I would like to be able to delete my account

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Make queries for deleting an account and all of its associated data | 5 Hrs (each) | Hunter,Victor, Connor |
| 2 | Connect User not Found page to server | 2 Hrs | Connor |
| 3 | Add a way to delete user’s account from settings page | 2 Hrs | Connor |
| 4 | Testing | 1 Hrs | Connor |

Acceptance Criteria:

* Given a user clicks delete account, then the user must be logged out of the website.
* Given an account is deleted, then all information and posts from the user should be deleted.
* Given a user views a deleted account, then the user should visit a user not found page.
* Given a user follows a deleted account, then the account should be removed from their followed account list and their posts should not be in the timeline.

**User story #8:** As a user, I would like my sensitive information to stay private

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Save preferences for private info | 3 Hrs (each) | Hunter, Baxter |
| 2 | Implement encryption for passwords | 2 Hrs | Steve |
| 3 | Implement private fields for User Classes | 2 Hrs | Steve |
| 4 | Testing | 1 Hrs | Steve |

Acceptance Criteria:

* Given a user views another users account, then the user must not be able to see private information.
* Given a user who wants to make their information private, when they select the option, their critical information will stay private
* Given a user who wants to make their information public, when they deselect the option, their critical information will become public

**User story #9:** As a user, I would like to be able to follow a [user, topic] combination

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Implement table of users and their followers | 10 Hrs | Baxter |
| 2 | Connect follow button to server call | 4 Hrs (each) | Baxter, Ryan |
| 3 | Testing | 1 Hr | Baxter, Ryan |

Acceptance Criteria:

* Given that a user clicks on the follow other user button, the user should be able to pick topics that will filter the user’s posts that show up on the timeline
* Given that a user clicks on the follow other user button, their preferences on user and topic should be saved to the database
* Given that the user clicks on the follow other user button, when they select topics to follow, then those topics of interest should be saved to the database

**User story #10:** As a user, I would like to be able to unfollow a [user, topic] combination

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Build User unfollowing method in server | 3 Hrs | Baxter |
| 2 | Implement unfollow button | 3 Hrs | Ryan |
| 3 | Implement unfollow data wipe in database | 4 Hrs | Baxter |
| 4 | Testing | 1 Hr | Baxter |

Acceptance Criteria:

* Given that unfollowing a user’s topics works, when a user clicks to unfollow a topic, the user should be able to pick what topics they want to unfollow and keep the rest
* Given that unfollowing a user’s topics works, when checking the timeline, the posts that show up should be from followed users should not contain the unfollowed topics
* Given that unfollowing a user works, when a user clicks to unfollow a user completely, the unfollowed user’s posts should no longer show up on the timeline

**User Story #11:** As a user, I would like to be able to tag my microblog with topics of my choosing

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Implement table of topics in MongoDB | 5 Hrs | Steve |
| 2 | Create UI popup to add topics to a user post | 6 Hrs | Ryan |
| 3 | Display a post’s tags within the post in the timeline | 4 Hrs | Ryan |
| 4 | Testing | 1 Hr | Ryan |

Acceptance Criteria:

* Given that a user can create a post, before a user clicks the post button, the user should be able to add tags to the post.
* Given a user creates a tag, it should be added to the list of tags for that user in the database.
* Given that a user’s post has been created, the user can edit the tags on the post by adding or deleting tags.

**User Story #12:** As a user, I would like to be able to repost another microblog.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | Estimated time | Owner |
| 1 | Implement UI button to allow user to repost microblog. | 3 Hrs | Connor |
| 2 | Store User Posts in MongoDB | 4 Hrs | Connor |
| 3 | Testing | 1 Hr | Connor |

Acceptance Criteria:

* Given that a user presses a button to repost a microblog, it should add it to the posts on the user profile page.
* Given that a user reposts a microblog, it should be stored in that users database for posts on their profile.
* Given that the reposting of the microblog worked, there will be text shown on the repost that indicate it is a reposted microblog.

**Total Hours of All Team Members**:

|  |  |
| --- | --- |
| **Name** | **Hours** |
| Steve | 25 |
| Hunter | 30 |
| Connor | 28 |
| Victor | 30 |
| Baxter | 31 |
| Ryan | 26 |

**Remaining User Stories from Backlog:**

* As a user, I would like to be able to quote another microblog
* As a user, I would like to be able to view Hot TopiX on both mobile and the web
* As a user with disabilities, I would like to be able to easily view the application
* As a user, I want to be able to customize the color palette of my display.
* As a user, I would like for the timeline to only contain [user, topic] combinations I choose.
* As a user, I would like to be able to like the microblogs that appear in my timeline
* As a user, I would like to be able to see microblogs with new topics
* As a user, I would like to be able to accept or decline seeing more microblogs with new topic tags
* As a user, I would like to be able to follow other users
* As a user, I would like to be able to view another user’s profile, with all posts
* (If time allows) As a user, I want to be able to report microblog posts, that I feel are offensive
* (If time allows) As a user, I would like to be able to report another user for inappropriate or offensive content
* (If time allows) As a user, I would like to be automatically logged out if no user activity is recorded for 5 minutes or more
* (If time allows) As a user, I would like to be able to create a private account, where others cannot view my microblogs unless they are following me
* (If time allows) As a user, I would like to be able to request to follow a private account to view their posts
* (If time allows) As a user with a private account, I would like to be able to accept or decline follow requests
* (If time allows) As a user, I would like to see a recommend list of accounts to follow based on what’s trending
* (If time allows) As a user, when I go to follow another user, I would like to be able to see how many times s/he used each topic tag

Non-Functional:

* Usability: The web application should be easy to navigate and edit settings. We want a clean UI with fast response time to avoid annoying lag for users. To insure this, if any action takes longer than 5000 ms, we will notify the user that the requested action is taking longer than usual. After 10000 ms, we will present the user with a timeout error.
* We also want to ensure that our application is accessible on all screen sizes and resolutions, as well as on all browsers. The application should be accessible to people with disabilities according to W3C’s accessibility guidelines version 2.
* Testing: (If time allows) We want to set up integration tests to make sure our backend can handle many users at once and existing users still see the filtered content they want