****

**Probos**

Team 14 - Sprint 2 Planning Document

*Alexa Baker, Daniel Raftery, Steven Daniel Vaughan, and Michael Woods*

**Sprint Overview**

During this sprint, we hope to build on our groundwork of core Mastodon functions in order to offer more fully fleshed out features for the end user, some of which aren’t typically offered in other Mastodon clients, in order to set Probos apart and make it more robust. In general, we will be expanding upon message drafting to include more options, building the notification system, adding media display and selection for various uses, and including several functions that apply to user profiles.

**Scrum Master:** Michael Woods

**Meeting Plan:** Mondays/Wednesdays @ 7:00 PM

**Risks and Challenges**

Because we did not manage to fully implement every planned feature from the first sprint, we will need to wrap up those functions and any related bugs in the code before delving too deeply into beginning additional features for this sprint, which will detract from the time we have to work. We will also need to be sure we plan out features in a sensible order and timeframe, but it should thankfully be easier to do so now that we have a better idea of roughly how long different features could take.

**Current Sprint Detail**

1. As a user, I want to choose what types of notifications I receive from Probos.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Set notifications for direct messages, mentions, and replies | 4 hours | Alexa |
| 2 | Set notifications for follows, boosts, and favorites (general notifications already figured out) | 2 hours | Alexa |
| 3 | Implement automatic refreshing of notifications when Probos is in the background | 3 hours | Alexa |
| 4 | Implement tapping on notifications to bring you to the relevant toot/message | 4 hours | Alexa |
| 5 | Test various edge cases highlighted in testing document | 2 hours | Alexa |

Acceptance Criteria:

* Given the notification settings are set and implemented correctly, when a new direct message is received, a notification will be shown to inform the user of the receipt.
* Given the notification settings are set and implemented correctly, when a user’s “toot” has been replied to, boosted, or favorited, a notification will be shown to inform the user of the reply.
  + Given the notification settings are set and implemented correctly, when the user has a new follower, a notification will be shown to inform the user of the follow.
* Given the notification settings are set and implemented correctly, when you tap on a notification, the app will retrieve the message and display it in the expanded view.

1. I want to change the frequency push notifications are generated.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Create UI page to change notification settings, access through a button visible from the timelines | 4 hours | Alexa |
| 2 | Connect UI to code to enable and disable types of notifications | 3 hours | Alexa |
| 3 | Test notification settings and various edge cases highlighted in the testing document | 3 hours | Alexa |

Acceptance Criteria:

* Given the menu item for notification settings is added correctly, when the button is clicked, the app will change to the notification settings page and return to the timeline when the back button is pressed.
* Given the buttons to change notification frequency are implemented correctly, when a new selection is pushed, the app stops or starts sending notifications at that speed.
* Given a message that meets a notification requirement is sent/received correctly, a relevant notification will be displayed to the user.

1. As a user, I want to mute/block other users.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Add mute/block button on user profile UI in a design consistent manner with follow button | 1 hours | Daniel |
| 2 | Researching API function possibilities for muting/blocking | 1 hours | Daniel |
| 3 | Implement API functionality to button for blocking and muting | 2 hours | Daniel |
| 4 | Implement storage of blocks/mutes | 3 hours | Daniel |
| 5 | Add confirmation screen asking after button press if the user wants to block the user | 1 hour | Daniel |
| 6 | Test muting/blocking functionality, fixing bugs | 2 hours | Daniel |

Acceptance Criteria:

* Given a user profile is loaded, a mute/block button loads and is visible
* Given a mute/block button is tapped, a confirmation screen appears asking if they are sure and explains the function of blocking.
* Given a user is blocked, statuses they make should no longer be visible in timelines, regardless of status visibility data.

1. As a user, I want to change my display name.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Create a duplicate, editable version of the profile UI for the user to interact with. | 3 hours | Michael |
| 2 | Update code to retrieve text from the field correctly to parse into API calls. | 2 hours | Michael |
| 3 | Test through both app and web client that display name update is processed correctly. | 1 hours | Michael |

Acceptance Criteria:

* Given the API calls are generated and processed successfully, when the user modifies the text for their display name and submits, the change should be reflected from other clients.
* Given the UI is implemented correctly, when the user quits updating their profile without submitting, all changes they made should be discarded.
* Given the UI is implemented correctly, when the user submits changes to the instance, their profile should refresh automatically to update with the changes.

1. As a user, I want to edit my account “bio” information for a specific instance.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Update “bio” field code in the editable profile UI to retrieve text correctly and create the appropriate API call. | 2 hours | Michael |
| 2 | Update the user profile UI to include a variable number of “metadata” fields. | 4 hours | Michael |
| 3 | Update “metadata” fields in editable profile UI to retrieve text correctly and create the appropriate API call(s). | 2 hours | Michael |
| 4 | Test through both app and web client that the “bio” update is processed correctly. | 1 hour | Michael |

Acceptance Criteria:

* Given the API calls are generated and processed successfully, when the user modifies the text for their bio and metadata and submits, the change should be reflected from other clients.
* Given the UI is implemented correctly, when the user quits updating their bio and metadata without submitting, all changes they made should be discarded.
* Given the UI is implemented correctly, when the user views a profile, any metadata the account contains should be displayed appropriately.

1. As a user, I want to edit my profile icon and banner images in a specific instance.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Update code connected to both user profile and user banner images on the editable profile UI to open a media picker. | 3 hours | Michael |
| 2 | Ensure the media file selected is of a valid type to use for a Mastodon profile. | 3 hours | Michael |
| 3 | Generate the appropriate API call with the now appropriately-formatted images. | 2 hours | Michael |
| 4 | Test through both app and web client that the profile icon and banner image updates were processed correctly. | 1 hour | Michael |

Acceptance Criteria:

* Given the API calls are generated and processed successfully, when the user selects images for their profile icon/banner image and submits them, the change should be reflected from other clients.
* Given the UI is implemented correctly, when the user selects image(s) to submit, they should be stored temporarily until either submitted or discarded on menu close.
* Given the system calls are made properly to retrieve media, when the user selects an image to use, it should be checked and converted (if needed) to an appropriate format to submit to the instance.

1. As a user, I want to save message drafts.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Implement a save button in the message draft UI | 1 hour | Daniel |
| 2 | Design a method for storing drafts locally, or saving them to the instance | 3 hours | Daniel |
| 3 | Implement a button to access saved previous drafts and replies, in a layout consistent with the design of the rest of Probos | 1 hour | Daniel |
| 4 | Implement a UI to view a list of previously saved drafts, upon tapping any previous draft the draft/reply UI is loaded in order to continue writing before posting | 4 hours | Daniel |
| 5 | Connect buttons to correct API calls/Implement retrieving drafts from Mastodon instance | 3 hours | Daniel |
| 6 | Test functionality of saving drafts after app closes, user logs out, etc. | 2 hours | Daniel |

Acceptance Criteria:

* Given the draft saving functionality is implemented correctly, when the message/reply UI is loaded, a save button should be visible on screen, either through a button or a dropdown menu.
* Given the saved draft UI is implemented correctly, when a draft is saved, it should be visible to the user in some form (such as a dropdown menu or list of drafts).
* Given that the saved draft UI is implemented correctly, when a draft is “recalled”/loaded, the message draft/reply UI fields are populated correctly with the contents of the saved draft.

1. As a user, I want to be able to delete and redraft my previous messages.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Implement a Delete button in the “more” dropdown on status UI on statuses you have written, and create confirmation screen asking for affirmation of delete request | 1 hour | Daniel |
| 2 | Research API on calls to delete posts | 1 hour | Daniel |
| 3 | Implement deleting posts | 1 hour | Daniel |
| 4 | Implement a Redraft/Edit button to the “more” dropdown on status UI on statuses you have written, and send to a modified version of draft UI containing previous message | 2 hours | Daniel |
| 5 | Research API on replacing status content and implement functionality for replacing statuses | 3 hours | Daniel |
| 6 | Test functionality of redraft with logging out, media statuses, text statuses, etc. | 2 hours | Daniel |

Acceptance Criteria:

* Given the “more” dropdown is implemented correctly, when it is pressed on a status the user wrote, an option should be available for “Edit/Redraft” and an option for “Delete.”
* Given the dropdown buttons are connected to the API correctly, when the Edit/Redraft button is pushed, a modified draft UI is loaded containing the status content available to change
* Given the dropdown buttons are connected to the API correctly, when “Delete” button is pushed and confirmed, upon refreshing, the timeline no longer displays the status.

1. As a user, I want to attach a content warning to “toots” as needed.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Update UI for messages to display content warnings and have message text/media hidden | 4 hours | Steven |
| 2 | Reimplement UI’s to include buttons suggesting statuses data on content warnings | 3 hours | Steven |
| 3 | Research API on methods involving content warnings/hiding | 2.5 hours | Steven |
| 4 | Test functionality with redrafting content warnings, general bugs scrolling new messages | 2.5 hours | Steven |

Acceptance Criteria:

* Given content warnings are detected and utilized correctly, when a message has a content warning applied to it, the message text and any media will not automatically be displayed.
* Given content warnings are utilized and detected correctly, when a message has a content warning applied to it, the message will automatically display a “reveal” button that will show its contents.
* Given content warnings are utilized and detected correctly, when a message has a content warning applied to it and the “reveal” button is pressed, the button text will toggle between “show” and “hide”, and the message’s text and/or media visibility will be toggled.

1. As a user, I want to show or hide media attached to a “toot.”
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Update message UI to display media attached to a message (if any). | 4 hours | Michael |
| 2 | Add ability to hide or show message media, depending on its state. | 2 hours | Michael |
| 3 | Implement API call/check for whether a message’s media is sensitive or not to set default display state. | 2 hours | Michael |
| 4 | Test that display state toggles correctly and is defaulted properly for the message’s sensitivity state. | 1 hour | Michael |

Acceptance Criteria:

* Given that the UI is updated successfully, when a message is retrieved with attached media, the message’s layout should be adjusted appropriately for the number of media attached.
* Given that the API calls are made correctly, when a message with media is retrieved, the media’s default display state should be set appropriately from the message’s “sensitive” value.
* Given that the media UI is implemented correctly, when the user taps to toggle the message’s display status, it should change appropriately (and continue to be toggleable back and forth).

1. As a user, I want to adjust the privacy setting of a “toot” (public, unlisted, followers only, direct message) on a message-by-message basis.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Update draft/reply UI to include options for privacy settings on potential messages | 2 hours | Daniel |
| 2 | Research API to know how to change visibility settings of a status | 1 hour | Daniel |
| 3 | Update functionality of draft/reply to include privacy data on outgoing status posts | 3 hours | Daniel |
| 4 | Implement functionality to change privacy settings of a status through edit/redraft functionality | 2 hours | Daniel |
| 5 | Test functionality of privacy settings on new timelines, statuses, from users perspective, web perspective, another users perspective | 2 hours | Daniel |

Acceptance Criteria:

* Given the message draft/reply UI is updated correctly, when a new message is being drafted, an option is visible to change the privacy settings on the status.
* Given the privacy UI option is implemented correctly, when the privacy settings are changed for a status, the visibility to other users is correctly modified for the API call.
* Given the API call is made and processed successfully, when another user (not allowed under the privacy settings) loads a timeline containing the message, it will not be returned to them.

**Developer Stories**

1. As a developer, I want the thread execution to be optimized and to not interfere with UI activities.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Testing thread execution with debugging | 3 hours | Steven |
| 2 | Researching thread interactions with UI in Android | 3 hours | Steven |
| 3 | Testing alternative execution methods | 3.5 hours | Steven |
| 4 | Implementing more efficient execution based on activity being done | 3 hours | Steven |

Acceptance Criteria:

* Given threads are implemented correctly and efficiently, when a timeline is loaded/refreshed, the UI will not be stalled while the thread runs, and the previous timeline (if one exists) can still be manipulated while the new one loads.
* Given threads are implemented correctly and efficiently, when the boost or favorite buttons are pushed, the UI will not be stalled while the client posts to the server.
* Given threads are implemented correctly and efficiently, when a user list or profile page is loaded, the loading time will be at an acceptable level where it will not interfere with app flow.

1. As a developer, I want to fix the bug where messages that did not have a favorite/”boost” applied to them would appear with one.
   1. Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Testing button functionality with debugging | 3 hours | Steven |
| 2 | Researching better methods of storing user boosts/favorites client-side | 3 hours | Steven |
| 3 | Implementing more efficient storage methods and API calls | 3 hours | Steven |

Acceptance Criteria:

* Given the boost/favorite buttons’ respective functions are implemented correctly, when the boost or favorite button is pressed, the button will reflect the change made.
* Given the boosts/favorites are stored correctly, when a timeline is loaded, only messages that have been boosted or favorited will show the respective button as active.
* Given the boost/favorite buttons’ respective functions are implemented correctly and boosts/favorites are stored correctly, when a timeline is loaded or button is pressed, the boost/favorite will be stored client-side for quick display.

**Sprint 1 - Incomplete User Stories, Tasks, & Acceptance Criteria**

As a user, I want to be able to follow other users.

Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 3 | Test that profile information loads correctly and that following (or unfollowing) works correctly | 2 hours | Alexa |

Acceptance Criteria:

* Given that the UI is configured correctly, when the user presses on a profile picture/name, the corresponding profile page should open.
* Given that the API calls are connected and used correctly, the user should be able to see profile information on the profile page.
* Given that the API calls are connected and the UI is configured correctly, the user should be able to press the various buttons on the page (“follow/unfollow”, ”block”, etc.) the corresponding action will be completed.

As a user, I want to attach media to a “toot.”

Tasks Required

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 1 | Add a button to the message draft UI to attach media to a “toot.” | 1 hour | Michael |
| 2 | Attach a system call to the UI button to open the media picker. | 3 hours | Michael |
| 3 | Ensure that API calls are updated (if need be) to account for sending the attachment. | 2 hours | Michael |
| 4 | Test that messages appear correctly locally and remotely. | 2 hours | Michael |

Acceptance Criteria:

* Given that the media picker is called to properly, if a file is of an incorrect type, then the user should be unable to select it.
* Given that the media picker is called to properly, if a file is too large to submit, then it should not be attached and the user should be notified as such.
* Given that the API calls for submitting media are used correctly, when a user submits a “toot” with media, it should be processed correctly and viewable from other clients.

As a user, I want to view “toot” messages sent by others.

Tasks Required:

|  |  |  |  |
| --- | --- | --- | --- |
| Order: | Task: | Estimated Time: | Assigned To: |
| 5 | Build UI for the expanded view of a “toot” | 2 hours | Steven |

Acceptance Criteria:

* Given the expanded message UI is implemented correctly, when the user taps to open a specific “toot,” the app should display previous and subsequent messages (if any) at regular message size in chronological order above/below the expanded message.

**Remaining Backlog:**

Functional Requirements

1. **Accounts**

As a user,

* 1. I want to register an account on a specific instance.
  2. ~~I want to log in to my account(s).~~
  3. ~~I want to be able to log out of or switch the account I am using.~~
  4. I want to be able to go through my instance’s process for resetting passwords.

1. **Messaging in the Timeline / “Toots”**

As a user,

* 1. ~~I want to view “toot” messages sent by others.~~
  2. ~~I want to scroll to view older “toots”.~~
  3. I want to view a list of the “toots” I have favorited.
  4. ~~I want to favorite “toots”~~
  5. ~~I want to reply to specific “toots”~~
  6. ~~I want to boost “toots”.~~

1. **Creating a Message/ “Toot”**As a user,
   1. ~~I want to send “toots.~~
   2. ~~I want to adjust the privacy setting of a “toot” (public, unlisted, followers only, direct message) on a message-by-message basis.~~
   3. ~~I want to show or hide media attached to a “toot.”~~
   4. ~~I want to attach a content warning to “toots” as needed.~~
   5. ~~I want to be able to delete and redraft my previous messages.~~
   6. ~~I want to save message drafts.~~
2. **Instance Settings**

As a user,

* 1. ~~I want to change which timeline I am looking at (personal, local, or federated).~~
  2. I want to carry over the list of profiles I follow to a new account on a different instance.
  3. ~~I want to edit my profile icon and banner images in a specific instance.~~
  4. I want to carry over my picture and banner to a new account on a different instance.
  5. ~~I want to edit my account “bio” information for a specific instance.~~
  6. I want to access custom emoji from my instance. (if time allows)

1. **Public Profile**

As a user,

* 1. ~~I want to be able to follow other users.~~
  2. ~~I want to see my list of followers.~~
  3. ~~I want to change my display name.~~
  4. I want to be able to pin any of my “toots” to the top of my profile.
  5. ~~I want to mute/block other users.~~
  6. I want to change my notification settings.

1. **Feed / “List” Customization**

As a user,

* 1. I want to create new “Lists”.
  2. I want to edit and delete “Lists”.
  3. I want to be able to only view my “Lists”.
  4. I want to enable media only mode for the local and federated timelines. (if time allows)
  5. I want to change my filter for words/phrases. (if time allows)

1. **Notification Customization**

As a user,

* 1. ~~I want to change the frequency push notifications are generated.~~
  2. ~~I want to choose what types of notifications I receive (favorites, boosts, mentions, new followers).~~

1. **Client Appearance**

As a user,

* 1. I want to customize the primary and secondary color schemes of the app’s user interface. (if time allows)
  2. I want the app to react properly to different font sizes. (if time allows)

1. **Server/Instance Control**

As a server admin,

* 1. I want to be able to ban accounts. (if time allows)
  2. I want to modify the instance’s default primary and secondary color schemes. (if time allows)
  3. I want to be able to read user-submitted reports. (if time allows)

1. **Security**

As a developer,

* 1. I want to implement a safe method of passing over login credentials to the instance without risking the personal information of users.
  2. I want to avoid storing raw personal login information.

Non-Functional Requirements

1. **Architecture and Design**
   1. The front end will be developed in Android Studio.
   2. Java API for Mastodon will be used for server-client communication.
   3. Multiple versions of Android will be supported.
   4. Application size will be less than 35 MB.
   5. Cache size of media, user lists, etc. will be less than 100 MB (if time allows)
   6. The client should load the instance with low latency as a priority of usability.
2. **Usability**
   1. The interface will be intuitive, sleek, and easy to navigate.
   2. Swipes and other gesture-related controls for touch screens will be implemented.
   3. The interface will adapt to multiple screen sizes. (if time allows)
   4. Portrait and landscape screen orientations will be supported. (if time allows)
3. **Security**
   1. We will build user trust and confidence by storing information in a secure manner and only if needed.
   2. Local storage backup of account information will be opt-in. (if time allows)
   3. We will limit/completely avoid handling of sensitive information by any third party or external service. (save Mastodon)