Sprint 3 Plan

Sprint Goal

The Sprint 3 goal is to implement more advanced functionality of the app that has been built previously, with calendar integration, sharing features, and notably notification features. This sprint is also fix and improve previous features according to stakeholder's feedback. The app is also created to be available to Apple App Store and Google Play Store

Sprint 3 Backlog (3 May - 24 May) V2.0

User Story ID	Tasks	Acceptance Criteria	Priority	Estimate Sizes	Dependencies
MS-2	- MW-147: Swap to encrypted protocol (HTTPS) TO DO - MW-148: Create Enter Email to login endpoint TO DO - MW-149: Add Forgot Password Functionality TO DO - MW-170: Patient user registration page for first time user to fetch password TO DO - MW-171: Patient change/forgot password page TO DO - MW-181: Test refresh token fetching new JWT in mobile app and implement in all api calls TO DO	1. Patients should be able to enter their email addresses to login for the first time or if they have forgotten their passwords 2. Users should be able to update their passwords 3. Users should not be logged out automatically for a full day 4. User data should all be encrypted whenever being sent on the network	Must Have	Large	
MS-3	- MW-151: Finalise Optional/Compulsory Genie Fields for File Uploads TO DO - MW-153: Create Patient Account Status Endpoint TO DO - MW-162: Create more robust error handling and prompts on html file upload page TO DO - MW-182: Unit testing webapp - Patients page TO DO - MW-183: Unit testing webapp - File upload page TO DO - MW-152: Manage Appointment Deletions in Genie TO DO	1. Admins should be able to upload a Genie file without all the correct fields and it not fail 2. Admins should be able to check a patient account's status 3. Users should see useful error messages when using the web apps 4. Patients should only see appointments that are still valid and in the future	Must Have	Large	

MS-4	🗎 MW-173: Implement admin account creation and deletion `TO DO`	1. Admins should be able to navigate to an 'Admin Management' page 2. Admins should be able to view all admin accounts for the clinic 3. Admins should be able to create credentials for a new admin user 4. Admins should be able to delete the credentials of existing admin users	Must Have	Medium	
MS-10	• 🗎 MW-43: Setup Backend Notification Service `TO DO` • 🗎 MW-45: Frontend Notification / UI `TO DO` • 🗎 MW-140: Automatically Send Appointment Reminders `TO DO` • 🗎 MW-141: Create Message to Patient Endpoint `TO DO` • 🗎 MW-142: Send Messages to patients `TO DO` • 🗎 MW-143: Create Messages JPA Table `TO DO` • 🗎 MW-178: Handle notification permissions in mobile app `TO DO`	1. Patients should receive notifications for upcoming appointments 2. Patients should receive notifications for messages from admins 3. Admins should be able to send patients messages and resources 4. Patients should be able to choose if they allow notifications or not	Should Have	Medium	
MS-11	🗎 MW-158: Add and test personal calendar integration for appointments `TO DO`	1. Patients should be able to add their scheduled appointments to their native phone calendar app through an 'add to calendar' button associated with individual appointments 2. The appointment title and time/date should align with what is displayed in the MMS app and how it defaults in their native calendar app	Can Have	Medium	

MS-12	MW-159: Create alternate appointment view - chronological list of appointments TO DO	1. Patients should be able to toggle between calendar view and list view in the appointments page 2. In list view, appointments should be displayed in chronological order 3. Patients should be able to click on appointments to view their details	Should Have	Small	MS-3
MS-13	• MW-144: Create Generic Resources Template TO DO • MW-164: Allow clinic to create and manage resources in webapp TO DO	1. Admins should be able to send generic resources from templates instead of creating them from scratch	Must Have	Medium	
MS-14	• MW-157: Add personal patient notes feature for appointments TO DO • MW-160: Display Genie appointment notes to patients TO DO • MW-174: Change doctors and facilities address field on webapp to enforce a valid address TO DO • MW-179: Remove doctor details from appointment details page - only need name and address TO DO • MW-180: Improve default images on webapp - Doctors and Facilities page TO DO • MW-184: Unit testing webapp - Doctors page TO DO • MW-185: Unit testing webapp - Facilities page TO DO	1. Patients should be able to make notes on an appointment 2. Admins should be warned if incorrect data such as invalid email is provided when creating new entities (doctors or facilities) 3. Users should have a positive experience with an easy to understand UI on the app and web app	Must Have	Medium	MS-3
MS-15	• MW-150: Create Email Service TO DO • MW-161: Implement and test share button on appointments and resources screens TO DO • MW-163: Allow clinic to send patients messages in webapp TO DO • MW-165: Allow clinic to send resources to specific patients TO DO • MW-166: Allow patients to receive and view messages and resources sent to them TO DO	1. Patients should receive an email at their address if they click forgot password or log in 2. Patients should be able to view all their messages in the app	Should Have	Medium	

MS-1	• ▣ MW-167: Create and publish mobile app onto Android app store `TO DO` • ▣ MW-168: Build and publish mobile app onto Apple app store `TO DO`	1. Patients should be able to download the app from the app store or google play store	Must Have	Medium	

User Story ID	Priority	Estimate Story Points	Dependencies
▣ MW-43: Setup Backend Notification Service `TO DO`	Must Have	5	
▣ MW-45: Frontend Notification / UI `TO DO`	Must Have	2	
▣ MW-124: Frontend Navigator Testing `TO DO`	Must Have	2	
▣ MW-140: Automatically Send Appointment Reminders `TO DO`	Should Have	2	MW-43
▣ MW-141: Create Message to Patient Endpoint `TO DO`	Should Have	2	
▣ MW-142: Send Messages to patients `TO DO`	Should Have	2	MW-43
▣ MW-143: Create Messages JPA Table `TO DO`	Should Have	1	
▣ MW-144: Create Generic Resources Template `TO DO`	Should Have	3	
▣ MW-146: Write SQL Schema for automatic deployment `TO DO`	Should Have	3	
▣ MW-147: Swap to encrypted protocol (HTTPS) `TO DO`	Should Have	3	
▣ MW-148: Create Enter Email to login endpoint `TO DO`	Should Have	2	MW-150
▣ MW-149: Add Forgot Password Functionality `TO DO`	Should Have	2	
▣ MW-150: Create Email Service `TO DO`	Should Have	4	
▣ MW-151: Finalise Optional/Compulsory Genie Fields for File Uploads `TO DO`	Should Have	2	
▣ MW-152: Manage Appointment Deletions in Genie `TO DO`	Should Have	3	
▣ MW-153: Create Patient Account Status Endpoint `TO DO`	Should Have	3	
▣ MW-154: Backend Custom Error Handling `TO DO`	Should Have	2	
▣ MW-155: Backend Error Messaging (write copy) `TO DO`	Should Have	2	
▣ MW-156: Timezone handling/testing `TO DO`	Should Have	2	
▣ MW-157: Add personal patient notes feature for appointments `TO DO`	Must Have	2	
▣ MW-158: Add and test personal calendar integration for appointments `TO DO`	Must Have	2	
▣ MW-159: Create alternate appointment view - chronological list of appointments `TO DO`	Should Have	1	

MW-160: Display Genie appointment notes to patients TO DO	Should Have	1	
MW-161: Implement and test share button on appointments and resources screens TO DO	Should Have	2	
MW-162: Create more robust error handling and prompts on html file upload page TO DO	Should Have	2	
MW-163: Allow clinic to send patients messages in webapp TO DO	Should Have	2	
MW-164: Allow clinic to create and manage resources in webapp TO DO	Should Have	2	
MW-165: Allow clinic to send resources to specific patients TO DO	Should Have	1	
MW-166: Allow patients to receive and view messages and resources sent to them TO DO	Should Have	2	
MW-167: Create and publish mobile app onto Android app store TO DO	Should Have	3	
MW-168: Build and publish mobile app onto Apple app store TO DO	Should Have	3	
MW-169: Populate webapp home page TO DO	Should Have	2	
MW-170: Patient user registration page for first time user to fetch password TO DO	Should Have	2	
MW-171: Patient change/forgot password page TO DO	Should Have	1	
MW-172: Fix UI consistency between pages on the webapp TO DO	Should Have	1	
MW-173: Implement admin account creation and deletion TO DO	Should Have	2	
MW-174: Change doctors and facilities address field on webapp to enforce a valid address TO DO	Should Have	2	
MW-175: Patients can view doctor/ facility/ appointment locations on their native maps app TO DO	Should Have	2	
MW-176: Handle HTTPS requests for webapp TO DO	Should Have	1	
MW-177: Frontend time zone handling/ testing TO DO	Should Have	1	
MW-178: Handle notification permissions in mobile app TO DO	Should Have	1	
MW-179: Remove doctor details from appointment details page - only need name and address TO DO	Should Have	0.5	
MW-180: Improve default images on webapp - Doctors and Facilities page TO DO	Should Have	0.5	

🟢 MW-181: Test refresh token fetching new JWT in mobile app and implement in all api calls TO DO	Should Have	1	
🟢 MW-182: Unit testing webapp - Patients page TO DO	Should Have	2	
🟢 MW-183: Unit testing webapp - File upload page TO DO	Should Have	2	
🟢 MW-184: Unit testing webapp - Doctors page TO DO	Should Have	2	
🟢 MW-185: Unit testing webapp - Facilities page TO DO	Should Have	2	

Quick Link to Jira Backlog:

🏠 **ME-Wombat**

✈ Jira