Nicholas Richard Harding

20200344

Abstract

This document is my mobile app development log for the clients Hotel automated service.

it6035 Mobile App development

Practical Tasks

Table of Contents

Client Brief 3

User Stories 4

Product Backlog 4

Stakeholder Interview 5

Interview Preparation 5

User stories clarification questions 5

Design and features questions 5

User story clarification question responses 5

Design and features question responses 6

Research 6

**Card Sorting Method** 7

Merchandiser Card Sorting 7

Supervisor Card Sorting 7

**Tree Test Method** 8

**Findability Tasks Merchandiser** 8

**Merchandiser Navigation Tree** 8

**Testing Results** 8

**Findability Tasks Supervisor** 10

**Supervisor Navigation Tree** 10

**Testing Results** 10

Sketches 12

Design Patterns 22

Development Notes: 22

Testing Procedure 23

Features to be Tested: 23

Test Devices: 23

Pixel 2 Test 23

Nexus 10 Test 25

Phone M-DPI 5in Test 26

Pixel 2 Test Screenshots 28

Nexus 10 Test Screenshots 37

Phone M-DPI 5in Test Screenshots 41

Scope

# Client Brief

Client Information and Mobile App Overview.

**Client’s background.**

The company provides merchandising services to various retail shops. The company works with multiple clients and has a fleet of merchandisers who, on an agreed day, will come to the agreed space and set up a goods display in the way specified by the client. The company would like to create a mobile app that both individual merchandisers and supervisors can use.

**Mobile app users:**

Two groups of users will use the mobile app: merchandisers and their supervisors:

* **Merchandisers** are responsible for setting up product displays in such a way that stimulates interest and entices customers to make a purchase. Each day merchandisers have a list of displays to set up (referred to as tickets in the mobile app). Normally merchandisers will have several tickets a day to action. Merchandisers are paid on an hourly basis, so they need to record the time each display took to set up (using a timer in the mobile app) and provide it to their supervisor.
* **Supervisors** are merchandisers’ managers; they are responsible for monitoring the merchandisers, making sure the appropriate tickets are actioned and the displays are set up to a high standard. Supervisors also liaise with retail shops regarding their set ups.

**Mobile App Access**

In this instance, the app access for different types of users should be set up through navigation. In the future, the client would set up proper authentication and authorization, but it falls beyond the scope of this first iteration of mobile app development that you were asked to undertake.

**Functionality Overview**

For the **supervisors**, the app should show a list of merchandisers they are monitoring. And include individual merchandiser profiles that contain their names and contact details. They should also be able to see a list of retail shop clients and be allowed to add a new client or update the existing client’s details.

For the **merchandiser**, the app should display the list of tickets and highlight today’s tickets. It also must have the functionality to select the current ticket, record the time taken for the current ticket and then mark the ticket as complete. Moreover, the merchandiser should also be able to add comments to the ticket.

# User Stories

Core user story:

As a merchandiser I want to be able to see what tickets (displays) need to be completed on the day and record the amount of time I spend on setting it up and add any comments on the ticket I set up.

Secondary user story:

As supervisor I want to be able to see a list of clients and be able to create and update this information. I also want to be able to view the list of merchandisers that I am monitoring and view their profile which has their name and contact details. I would also want to check that the merchandisers have actioned the appropriate tickets.

# Product Backlog

These are features required in future releases but will not be included in the first release.

1. User Authorization (user login and password)
2. Update – Total weekly hours for merchandisers spent on tickets

Requirements Analysis

# Stakeholder Interview

## Interview Preparation

### User stories clarification questions

1. Do supervisors need to know which merchandisers are assigned to which tickets?
2. Is there a particular order that these tickets need to be completed in?
3. Do merchandisers need to be able to see the retail shops details?
   1. If yes, do they only need to see the stores that they are responsible for or a complete list with the stores they are responsible for emphasized (*example of emphasized technique would be the stores are highlighted and grouped together*)?
4. What information does the app need to store for retail shops (*e.g., store name, store manager, address, contact number etc.*)?
5. Do the supervisors need to be able to tell the merchandisers how to set up the display?
6. Are tickets always completed on time? If not, would you like a feature to be added to flag overdue tickets and make priority?

### Design and features questions

1. Would the merchandisers like to be able to view hour totals?
   1. If yes would the merchandisers like to be able to see the days total hours?
      1. If yes, would the merchandisers want to be able to view the daily totals over a work week period?
2. Do supervisors want to see how long merchandisers spend on tickets?
   1. If yes, do they want to see their daily and weekly hour totals?
3. Do merchandisers/supervisors want to be able to perform tap dials of numbers stored in the app?
4. Do the merchandisers need to take photos of the completed displays and save it to the corresponding ticket?
   1. If yes is this a future iteration release or a first release feature to implement?

### User story clarification question responses

1. Do supervisors need to know which merchandisers are assigned to which tickets?  
   -yes
2. Is there a particular order that these tickets need to be completed in?  
   -yes
3. Do merchandisers need to be able to see the retail shops details?  
   -yes
   1. If yes, do they only need to see the stores that they are responsible for or a complete list with the stores they are responsible for emphasized (*example of emphasized technique would be the stores are highlighted and grouped together*)? - yes
4. What information does the app need to store for retail shops (*e.g., store name, store manager, address, contact number etc.*)? - yes please
5. Do the supervisors need to be able to tell the merchandisers how to set up the display? - not necessary
6. Are tickets always completed on time? If not, would you like a feature to be added to flag overdue tickets and make priority? - yes

### Design and features question responses

1. Would the merchandisers like to be able to view hour totals? - No
   1. If yes would the merchandisers like to be able to see the days total hours?
      1. If yes, would the merchandisers want to be able to view the daily totals over a work week period?
2. Do supervisors want to see how long merchandisers spend on tickets? -yes
   1. If yes, do they want to see their daily and weekly hour totals? -weekly is fine
3. Do merchandisers/supervisors want to be able to perform tap dials of numbers stored in the app? -No
4. Do the merchandisers need to take photos of the completed displays and save it to the corresponding ticket? - not necessary but if you want then to add it.
   1. If yes is this a future iteration release or a first release feature to implement?

# Research

Going on the clients feedback I have performed two card sorting methods one for each user type outlined by the brief, basing off the card sorting I have created two navigation maps and performed a findability test on both this has been successful and will be what the sketches will be based off. Below is the card sorting and navigation trees:

## **Card Sorting Method**

### Merchandiser Card Sorting

### Supervisor Card Sorting

## **Tree Test Method**

Using the tree test method, we will determine if the proposed navigation setup for the application is appropriate, we will give a review of the tree to three people to test.

### Findability Tasks Merchandiser

1. Find “Store Address”
2. Find “Take Display Photo”
3. Find “View Completed Ticket”

### Merchandiser Navigation Tree

### Testing Results

Test Participant 1

|  |  |
| --- | --- |
| **Findability Test** | **Test Result** |
| 1. Find “Store Address” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Take Display Photo” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “View Completed Ticket” | **Pass** (Test participant had no issue in completing the task) |

Test Participant 2

|  |  |
| --- | --- |
| **Findability Test** | **Test Result** |
| 1. Find “Store Address” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Take Display Photo” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “View Completed Ticket” | **Pass** (Test participant had no issue in completing the task) |

Test Participant 3

|  |  |
| --- | --- |
| **Findability Test** | **Test Result** |
| 1. Find “Store Address” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Take Display Photo” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “View Completed Ticket” | **Pass** (Test participant had no issue in completing the task) |

Test Remarks

The testers had no issues in finding the relative tasks.

### Findability Tasks Supervisor

1. Find “Merchandiser Name”
2. Find “Store” of completed tickets
3. Find “Merchandiser contact number”

### Supervisor Navigation Tree

### Testing Results

Test Participant 1

|  |  |
| --- | --- |
| **Findability Test** | **Test Result** |
| 1. Find “Merchandiser Name” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Store” of completed tickets | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Merchandiser contact number” | **Pass** (Test participant had no issue in completing the task) |

Test Participant 2

|  |  |
| --- | --- |
| **Findability Test** | **Test Result** |
| 1. Find “Merchandiser Name” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Store” of completed tickets | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Merchandiser contact number” | **Pass** (Test participant had no issue in completing the task) |

Test Participant 3

|  |  |
| --- | --- |
| **Findability Test** | **Test Result** |
| 1. Find “Merchandiser Name” | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Store” of completed tickets | **Pass** (Test participant had no issue in completing the task) |
| 1. Find “Merchandiser contact number” | **Pass** (Test participant had no issue in completing the task) |

Test Remarks

The test participants had no issue in findability tree test.

# Revision

After reviewing the tree during the sketching process, it was noted that it could be simplified by removing excess pages and incorporating tap events on the list items to perform updates and deletes of list items, also the add ticket and merchandiser was missing, it also made sense to put it all under one page rather than multiple pages, these changes will be shown in the sketches and code later on.

# Sketches

# 

# 

# Design Patterns

For the merchandiser/supervisor sign in, sign in, menu page, add store and merchandiser, store and ticket details will all be a stackLayout as the content will be made to fit on one page and keep the UI (User Interface) touchable and easy to use. The Tickets, Store, update store and Merchandiser list pages will be a stackLayout inside a scrollView, this will keep all store and merchandiser lists expandable as more clients and merchandisers are added.

# Development Notes:

Displaying every image was causing memory problems and leaks, chose to implement a tap item load page and image approach.

Testing Procedure

## Features to be Tested:

1. C.R.U.D. operations
2. Image display
3. Photo taker
4. Preventing data to be entered when other required fields are needed I.e., no store can be added without merchandisers already been loaded, tickets cannot be made without stores being present.
5. Navigation behaves as to be expected e.g., sign in button should take you to the corresponding user menu page.

Test Devices:

1. Pixel 2
2. Nexus 10
3. Phone M-DPI 5in

## Pixel 2 Test

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Expected Behavior** | **Result** | **Comments** |
| 1. Merchant should not be able to access Active tickets, store list and completed tickets when data is empty. | Prompts user with alert that they cannot access it and how this can be resolved. | Pass. |  |
| 1. Supervisors should only be able to enter data in sequential sequence (store cannot be added when no merchandiser has been entered and no ticket can be created if no store has been created) | Prompts user when trying to access add page if corresponding data in not present, it also says how to resolve this issue. | Pass. |  |
| 1. Supervisor can add merchandisers, then stores and then tickets. | Database for merchandiser, store and tickets should be created, saved, and displayed. | Store details page didn’t update after save but when reopened data is there. |  |
| 1. Supervisor can delete or update store, merchandiser, and ticket. | Database items update to new values and be deleted on request. | Store update displayed no message to show successful. Merchandiser delete button doesn’t return to merchandiser list. | Solved added Display Alert in to display message. Delete now returns to merchandiser list added in missing pops. |
| 1. Supervisor should be able to tap on list items to edit or delete them. | Tap event should fire and direct user to new page for updating and deleting. | Pass. |  |
| 1. Merchandiser should be able to tap on a ticket to perform task completion. | Tap event fire sending user to new page. | Pass. |  |
| 1. Take photo button should direct user to camera interface and return n image just taken. | Should take user to camera, take a picture, question user if the phot took is ok and on yes return to ticket task, and now be displaying the photo on screen as well as having stored. | Pass. |  |
| 1. Completed ticket should show ticket information of completed tickets and be tappable to view display photo. | When a ticket has been completed users can now access the completed ticket page, when a list item is tapped it directs them to a page displaying the photo, and display an alert on how to return, and return upon image being pressed. | Pass. |  |
| 1. Layout resembles the sketches in design document. | Layout displays as expected. | Pass minor differences from sketches due to performance related issues. | Some views had been changed from sketches as it increased load times and performance. |
| 1. Date pickers cannot select past dates. | When selecting date only present and future dates are allowed. | Today's date was storing as 1/1/0001. | Solved by introducing a constant date. |
| 1. Tickets must be in order of date booked being closest to current date | Ticket list will be ordered from first to last according to booked date. | Pass |  |

## Nexus 10 Test

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Expected Behavior** | **Result** | **Comments** |
| 1. Merchant should not be able to access Active tickets, store list and completed tickets when data is empty. | Prompts user with alert that they cannot access it and how this can be resolved. | Pass. |  |
| 1. Supervisors should only be able to enter data in sequential sequence (store cannot be added when no merchandiser has been entered and no ticket can be created if no store has been created) | Prompts user when trying to access add page if corresponding data in not present, it also says how to resolve this issue. | Pass. |  |
| 1. Supervisor can add merchandisers, then stores and then tickets. | Database for merchandiser, store and tickets should be created, saved, and displayed. | Pass. |  |
| 1. Supervisor can delete or update store, merchandiser, and ticket. | Database items update to new values and be deleted on request. | Pass. |  |
| 1. Supervisor should be able to tap on list items to edit or delete them. | Tap event should fire and direct user to new page for updating and deleting. | Pass. |  |
| 1. Merchandiser should be able to tap on a ticket to perform task completion. | Tap event fire sending user to new page. | Pass. |  |
| 1. Take photo button should direct user to camera interface and return n image just taken. | Should take user to camera, take a picture, question user if the phot took is ok and on yes return to ticket task, and now be displaying the photo on screen as well as having stored. | Pass. |  |
| 1. Completed ticket should show ticket information of completed tickets and be tappable to view display photo. | When a ticket has been completed users can now access the completed ticket page, when a list item is tapped it directs them to a page displaying the photo, and display an alert on how to return, and return upon image being pressed. | Pass. |  |
| 1. Layout resembles the sketches in design document. | Layout displays as expected. | Pass minor differences from sketches due to performance related issues. | Some views had been changed from sketches as it increased load times and performance. |
| 1. Date pickers cannot select past dates. | When selecting date only present and future dates are allowed. | Pass. | Solved in first mobile test. |

## Phone M-DPI 5in Test

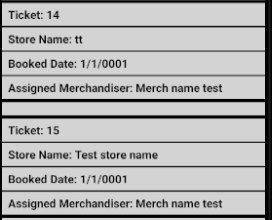
|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Expected Behavior** | **Result** | **Comments** |
| 1. Merchant should not be able to access Active tickets, store list and completed tickets when data is empty. | Prompts user with alert that they cannot access it and how this can be resolved. | Pass. |  |
| 1. Supervisors should only be able to enter data in sequential sequence (store cannot be added when no merchandiser has been entered and no ticket can be created if no store has been created) | Prompts user when trying to access add page if corresponding data in not present, it also says how to resolve this issue. | Pass. |  |
| 1. Supervisor can add merchandisers, then stores and then tickets. | Database for merchandiser, store and tickets should be created, saved, and displayed. | Pass. |  |
| 1. Supervisor can delete or update store, merchandiser, and ticket. | Database items update to new values and be deleted on request. | Pass. |  |
| 1. Supervisor should be able to tap on list items to edit or delete them. | Tap event should fire and direct user to new page for updating and deleting. | Pass. |  |
| 1. Merchandiser should be able to tap on a ticket to perform task completion. | Tap event fire sending user to new page. | Pass. |  |
| 1. Take photo button should direct user to camera interface and return n image just taken. | Should take user to camera, take a picture, question user if the phot took is ok and on yes return to ticket task, and now be displaying the photo on screen as well as having stored. | Fail, the take photo kept crashing inside the app and in its own app, with it working in two devices the problem is likely to be the emulator as it cannot use the app. | Tested on model down from it and earlier version of android on it and it worked, which reinforces that the 5in one is buggy. Last test image shows it working in the older device emulator. |
| 1. Completed ticket should show ticket information of completed tickets and be tappable to view display photo. | When a ticket has been completed users can now access the completed ticket page, when a list item is tapped it directs them to a page displaying the photo, and display an alert on how to return, and return upon image being pressed. | Fail, refer to information above. | Refer to comment above |
| 1. Layout resembles the sketches in design document. | Layout displays as expected. | Pass minor differences from sketches due to performance related issues. | Some views had been changed from sketches as it increased load times and performance. |
| 1. Date pickers cannot select past dates. | When selecting date only present and future dates are allowed. | Pass. | Solved in first test phone. |

Testing Comments:

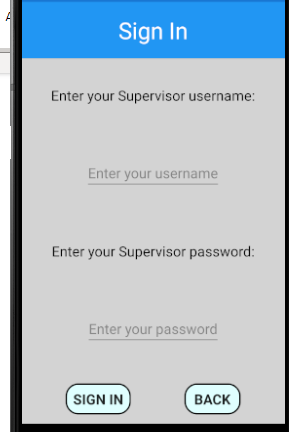
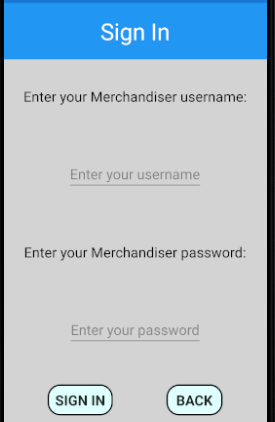
Most tests passed due to constant debugging checks. Total time feature hasn’t been implemented this was due to time constraints and could be implemented in future release, a quick fix around was to display how long a merchandiser spent on that ticket. Below are the testing images, the pixel has more to show a full test was done every part of the way and the others was to show key features working, the visual design needs to be improved for tablets if the client wishes to use them over mobile devices.

## Pixel 2 Test Screenshots

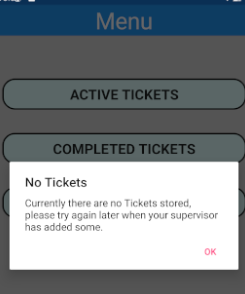
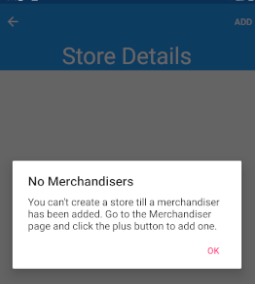
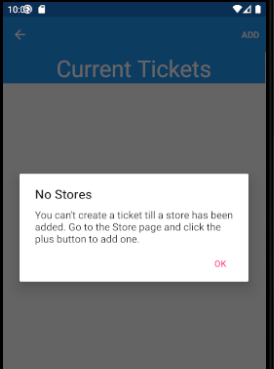
Todays date storing wrong:

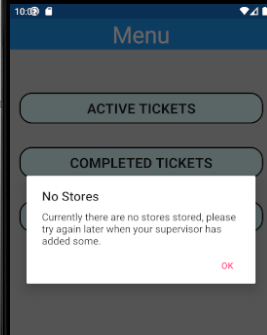
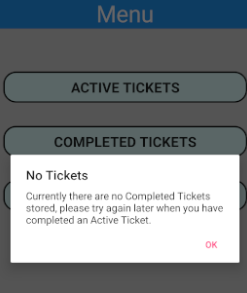


Sign in views:

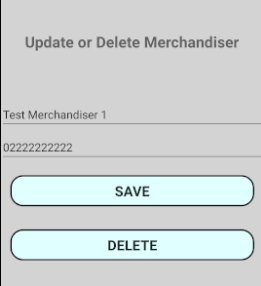
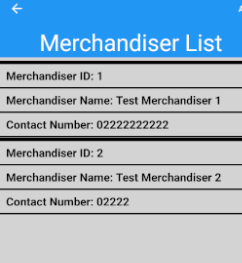
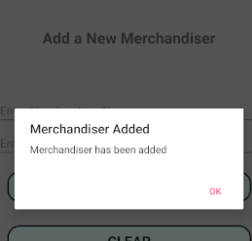


Data prevention due to missing data messages:

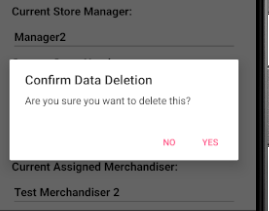
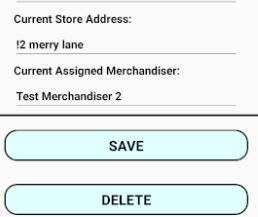
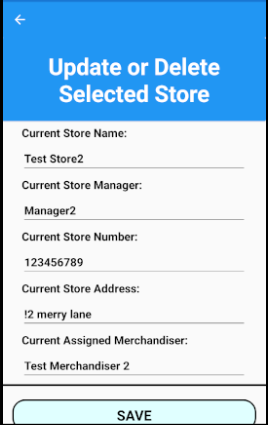
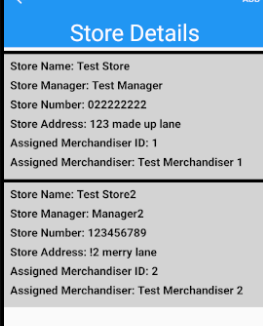
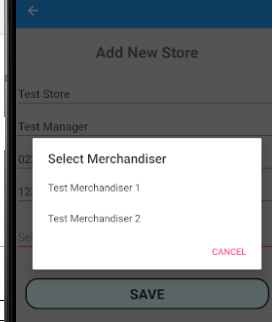
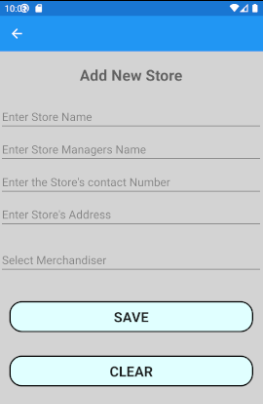




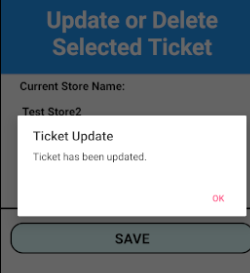
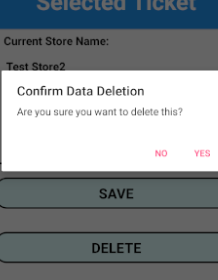
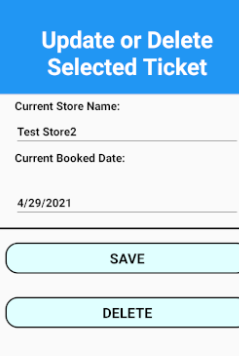
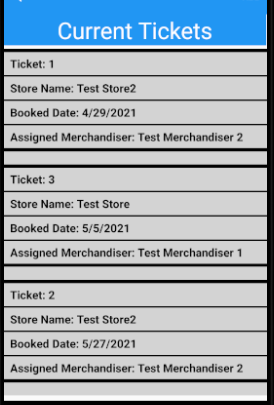
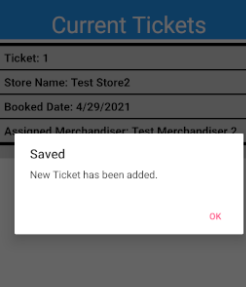
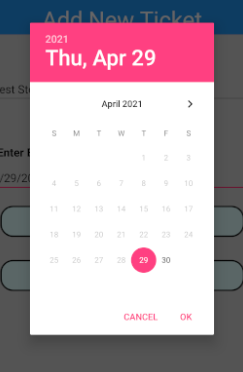
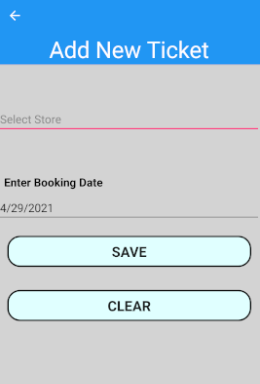
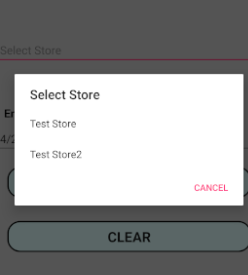
Add Merchandiser and new merchandiser list:



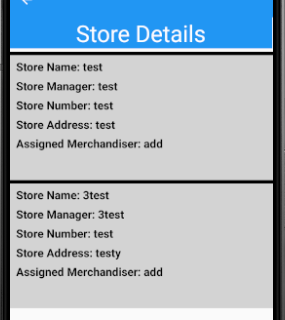
Add store and list and update tap page:



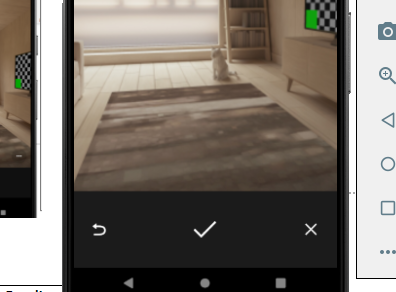
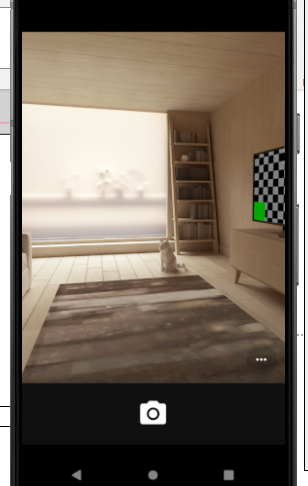
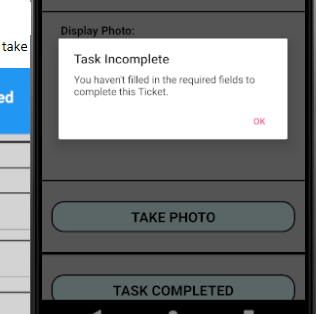
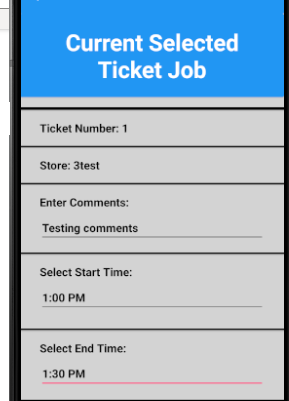
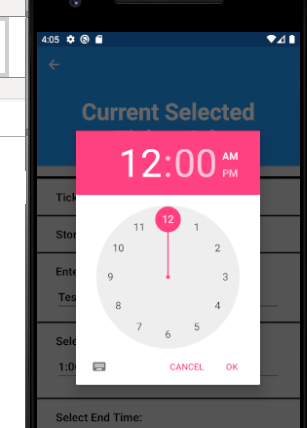
Add new ticket page and list:

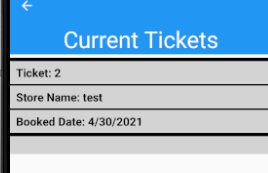
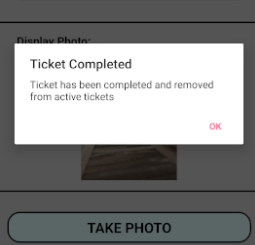
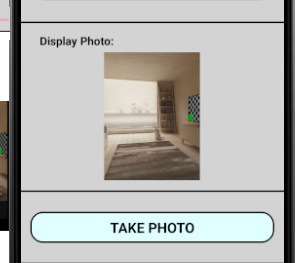


Merchandiser Store view after data added:

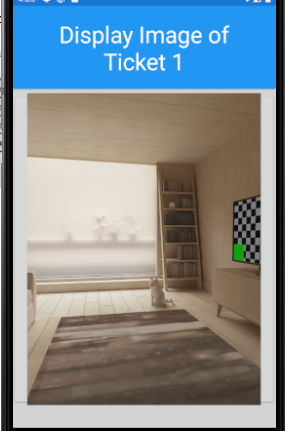
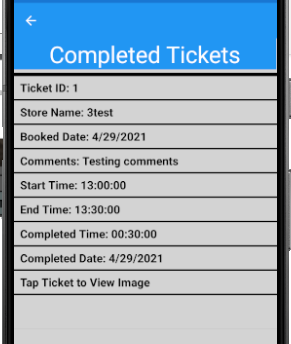


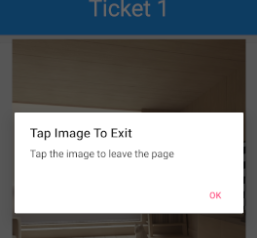
Merchandiser Active tickets and ticket task after tap, and take photo:



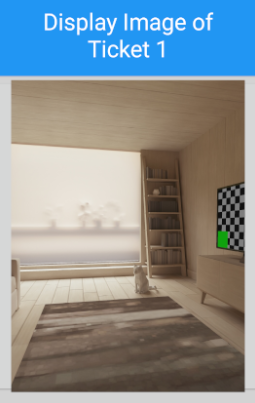
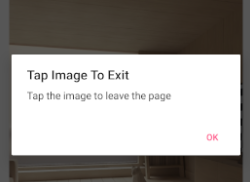
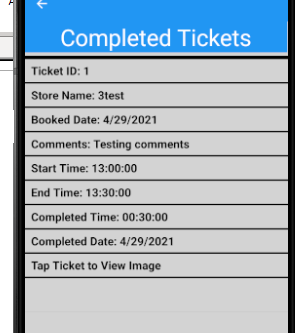


Completed Tickets Merch view and completed ticket tapped image page post completed task:



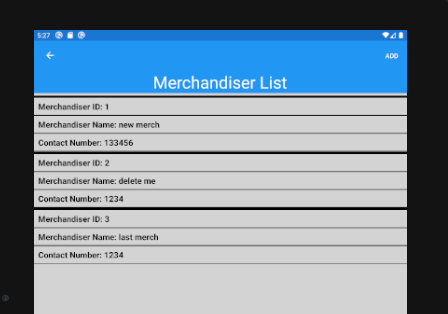
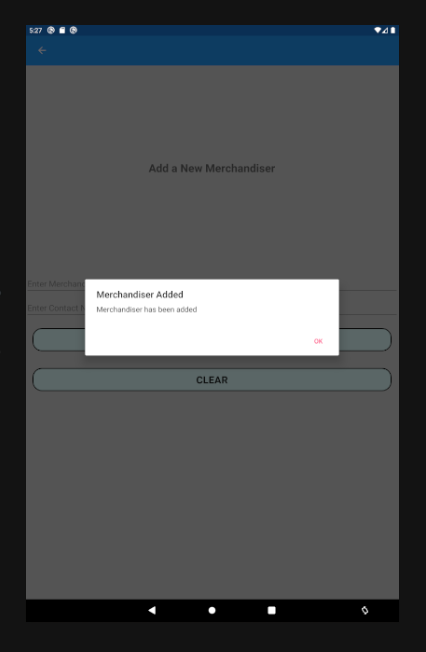


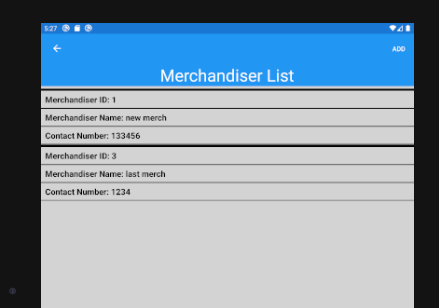
Supervisor Completed ticket view post completed task:



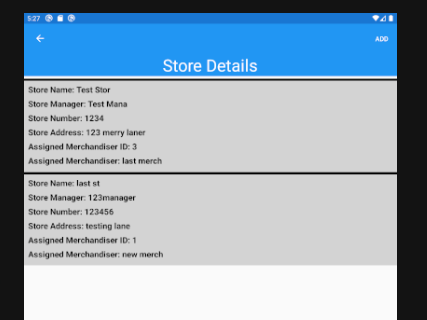
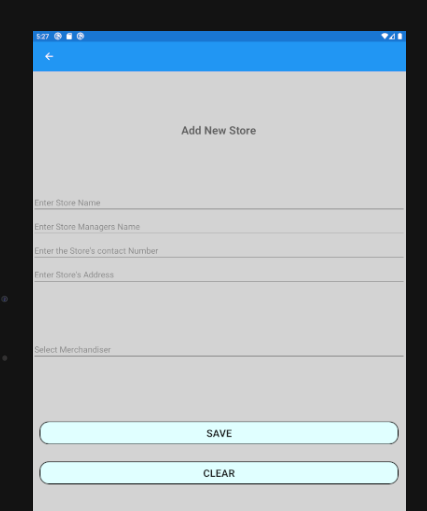
## Nexus 10 Test Screenshots

Add merchandiser, delete and list view:

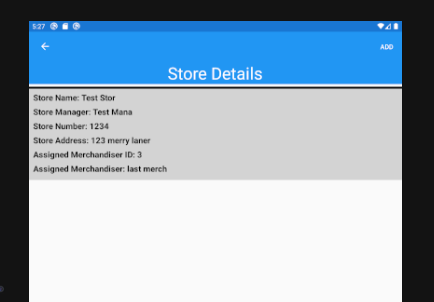




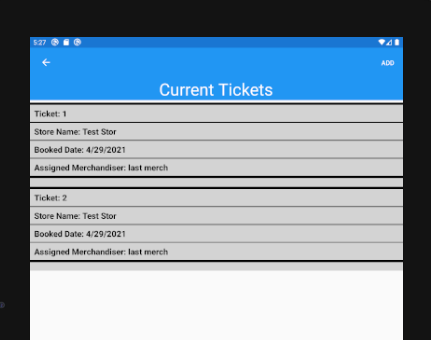
Add store and store list:



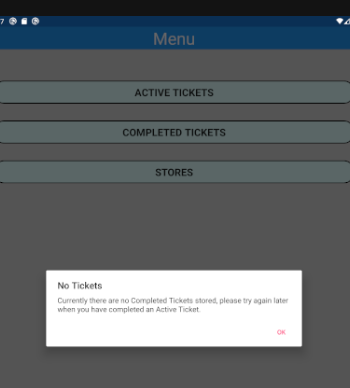
After store delete:



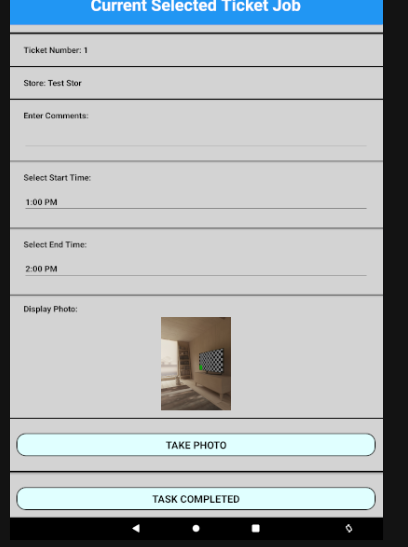
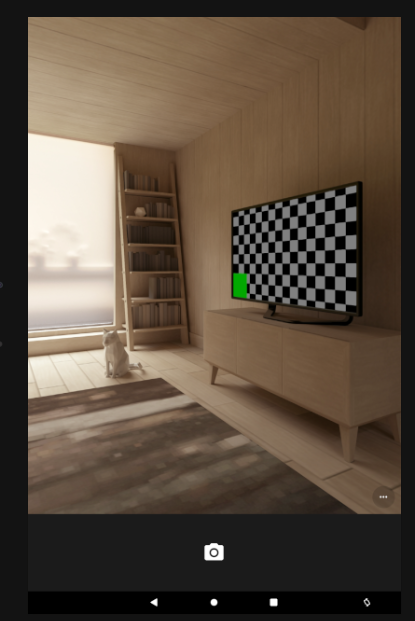
Supervisor ticket list post add:



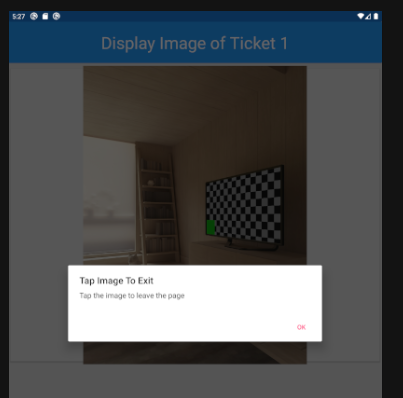
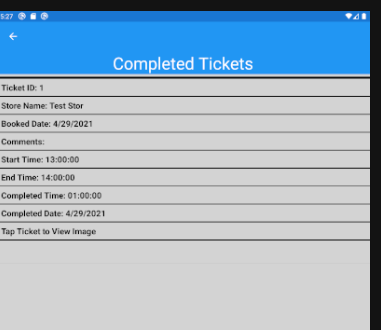
Merchandiser menu completed ticket access before ticket added entry denial:



Take photo and ticket task page:

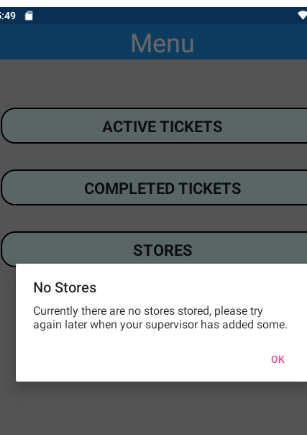
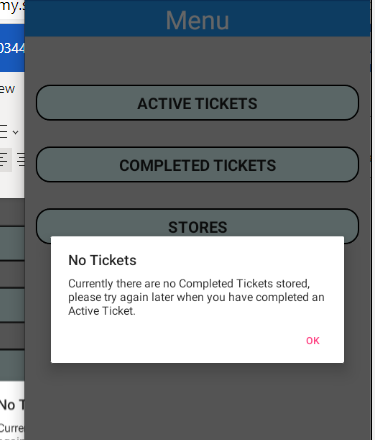
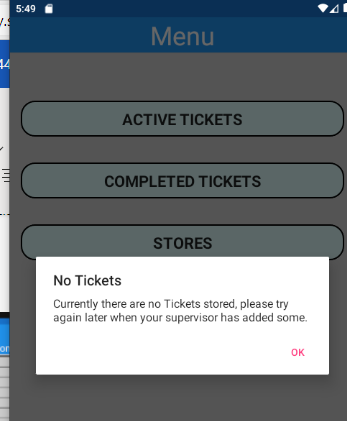


Completed tickets and tapped ticket:

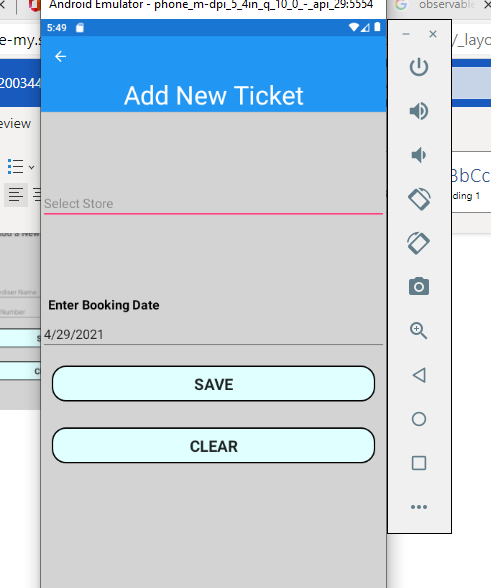
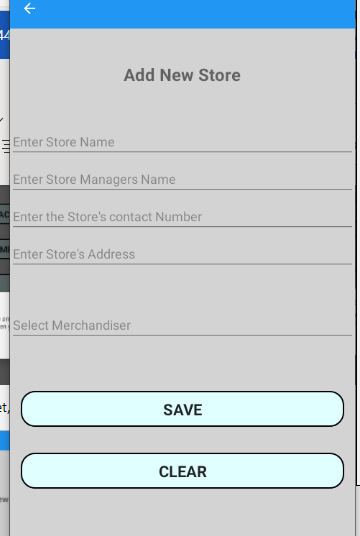
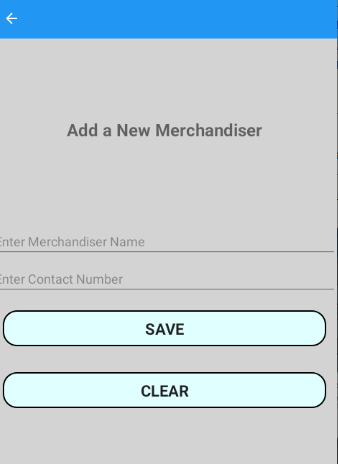


## Phone M-DPI 5in Test Screenshots

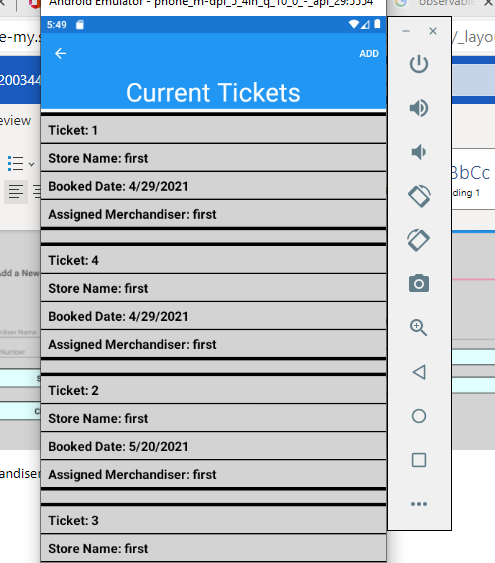
Pre data warnings for active tickets, stores and current tickets for merchandisers:



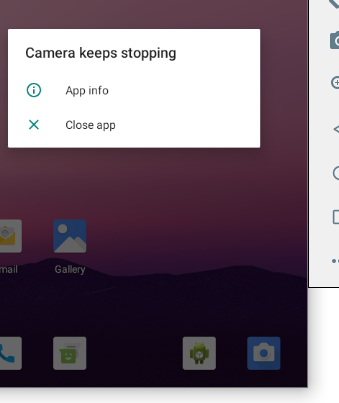
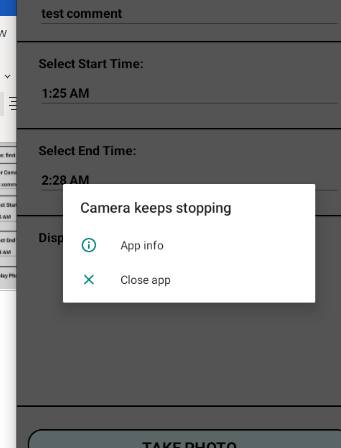
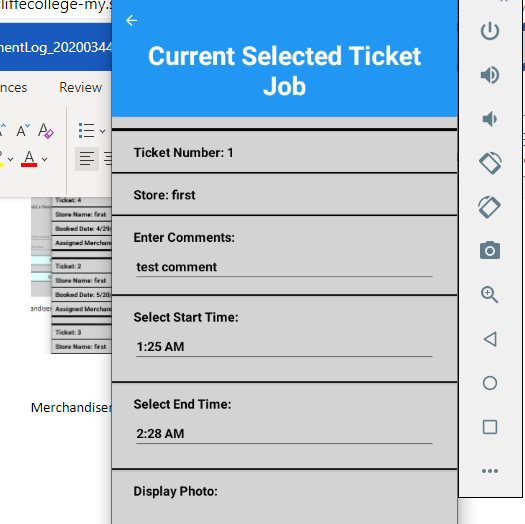
Adding ticket, store and merchandiser pages:



Ticket list in order:



Merchandiser Ticket task page:



Phone H-DPI 4in:

