FIT3178 MOBILE APP DESIGN SPECIFICATION

Jingmei Lin

28479068

TABLE OF CONTENTS

1. APPLICATION CONCEPT

M-Track is a multi-functional package tracking application, which allows users to track parcel delivering information completely within one single app, regardless the carriers or package types, or different users.

In addition to the basic tracking functionality, it adds real-time tracking, in-app sharing, and carriers rating. This app also works for deliverers for easier and smooth communication with customers.

There are already multiple types of parcel tracking apps in the marketplace. However, most of them focus on the tracking only. What we want is, M-Track can become users' assistant more than tracking. Sharing your tracking easily, contact with your customers smoothly, and see how other users rate carriers, and help u do a better choice next time. We also allow you try our tracking functions without login first, just to make sure you feel satisfied with our app first then become our users.

1.1 Key functions

- Parcel real-time tracking with live-updated map

For each tracking, the information will be shown with the map showing real-time location.

- In-app tracking information sharing

In the login state, the user can directly establish a conversation with another to easily share the tracking details.

- Tracking systems for receivers and deliverers

Not only the receivers but also the deliverers can use the same tracking system, making tracking details consistent and convenient for communication among the three parties.

- Service rating systems
  - Carrier rating
  - Delivery rating
  - User evaluation display

A convenient and efficient rating system can be used bidirectionally. For example, receivers can rate the deliverers, carriers and drivers; also, deliverers can rate other users. The rating results will be available for all the users, making the service quality transparent.

- Tracking speed statistics

Would you like to know how long it takes for each tracking to arrive? Here it is. The tracking speed (duration) statistics give you a more intuitive view of the completion efficiency of different orders, and the overall one is available to all users, allowing you to make better choices.

- Data synchronization

Each tracking details can be saved to each account and synchronize among multiple devices. You don't need to worry about losing data when changing devices.

- Linking with e-shops

No need to switch the apps on background to copy and paste the tracking numbers. Link with the current popular e-commerce shops like eBay, Amazon, to synchronize your order information and start tracking seamlessly.

1.2 Target audience

As the key functions provided before, M-Track has mainly 2 types of target users, the receivers (customers), deliverers (sellers). For the sake of better understanding, 3 personas are listed out as following.

- Receiver 1

  Name: Jenny Kole

  Age: 27

  Gender: Female

  Location: Melbourne

  Occupation: Accountant

  About:

  Jenny is usually busy with work and likes to shop in various shopping apps on her phone in her leisure time. She is concerned about packages tracking and is frustrated that there is no better way to process so much tracking details. She has tried a variety of parcel tracking apps, but found they were still not convenient, as each time she needs to switch between different apps to copy the tracking numbers; sometimes she needs to keep multiple tracking apps to meet her needs. She feels this reduces her efficiency. She also wants to be able to see the real-time locations on a map while looking at tracking information.

  Preferred shops: eBay, Amazon, Myer

- Receiver 2

  Name: Alex Lau

  Age: 25

  Gender: Male

  Location: Brisbane

  Occupation: Student

  About:

  Alex's parents like shopping online and often ask him to purchase things for them as they are not good at dealing with online shop. It sometimes gets inconvenience as Alex always get the order/tracking number by emails or other

shop accounts, and absolutely he does the tracking. Once items are ready to arrive, he needs to notice his parents via phone call or messages. He feels that sharing tracking details in this way is not convenient as he needs to switch among multiple apps. He wishes if he could have an app to share the tracking to them without screenshot or link sharing. He wants his parents can also get the tracking details live-updated.

- Deliverer

Name: Jessica Jones

Age: 35

Gender: Female

Location: Sydney

Occupation: Online seller

About: Jessica is an online seller with 5 years of experience. She has her own shops on Shopify and eBay, and the business is good. Although she was pleased with the large order quantity, she felt that her order tracking could not be fully synchronized with that of customers, and many of them needed to communicate by email. Meanwhile, she felt cumbersome to use different tracking systems for orders on different platforms. She hopes to have a more convenient way to communicate with customers about order and parcel tracking and also to communicate with different carriers in an easier way.

2. COMPETITION & INNOVATION

2.1 Competitors analysis

17TRACK: https://apps.apple.com/au/app/17track/id1004956012

- Avg rating: 4.8
- 367 Ratings

Arrive – Package Tracker: https://apps.apple.com/us/app/arrive-package-tracker/id1223471316

- Avg rating: 4.8
- 453.7K Ratings

Application	Key Functionality	Strengths	Weaknesses
17TRACK	1. Add multiple tracking numbers/carriers at once 2. Support barcode and QR code scanner 3. Synchronize data among multiple devices (both PC and mobile app) 4. Add memos and categories to tracking numbers	1. Add multiple tracking numbers/carriers at once, easier for user operation 2. Synchronize data among different devices via account 3. Support translation of tracking details to Mandarin	1. No live-updated map showing with tracking details, only literal expression may confuse users that where the parcel is in 2. Cooperate with some stores, but mainly intends to advertise their products, which is irrelevant to the

			tracking function, making users feel redundant
Arrive – Package Tracker	1. Automatic tracking: synchronize orders which are made by user's linking email address 2. Automatically query keywords in linked email address to get tracking details 3. Live map updates with tracking details	1. The order/parcel tracking can be automatically synchronized with the linked email address, making users no need to add them manually 2. Showing current tracking location details instead of just literal expression 3. Has a link for carriers, with which users can easily go to their websites	1. Need to login first, which may lower users' interest of using the apps 2. The "shop" block has not enough hint, and users may feel confused how they use this function 3. Only support Google mail to search tracking numbers, and the accuracy is not enough

To sum up, the two competitive apps already have mature functions and good rating. Compared with them, M-Track will focus more on the innovation points listed out in the following part, as the core functions for tracking are similar. The points how M-Track could differentiate from others are, we are multi-functional. Users will feel interesting with our tracking-based chat system and statistics board.

2.2 Innovation of M-Track

- In-app chatting system

According to the survey, most of the package tracking apps in the market only have the function of tracking. If you need to share the tracking details with your family or friends, you must share the details to another communication applications and chat with them there. If another user wants to track the same details, then it will need to download the relative application or open the webpage.

Therefore, M-Track adds an in-app chatting system that lets users can share tracking details or other information directly, also communicate easily. Today with the growing of online shopping and delivery, putting tracking, sharing and communicating in one app can make life easier.

We also think it is important to have good communication between deliverers and customers. Communicating via emails or in separate applications reduce efficiency for each party. This is also one of the reasons for designing an in-app chatting system here.

- Rating systems

This will include carrier rating, user experience display and statistics board.

Users always have a lot of opinions on the tracking, including tracking accuracy, carrier services, deliverer services, etc. M-Track aggregates the feedback and present it to all users, to help them select better and suitable services. It also helps the service providers improve their service quality in this competitive situation.

Existing reviews come from different platforms or websites, making it difficult for users to browse and compare them at once. The rating systems bring the convenience to users.

M-Track also provides one function that make the statistics of tracking speed of each carrier. Besides rating systems, it helps users see more intuitively how the carriers work well. It's a simple statistic but also help users make the choice.

3. FEASIBILITY & TECHNOLOGY

In this part, the technologies used by M-Track will be provided, with supported function, description, the reason to choose and the way to implement.

3.1 MapKit.Geocoder

Supported function: location tracking with live updating map
This is a class of JavaScript API (which is a set of functions allowing apps accessing some features/services) for embedding Apple maps on the website/app. The geocoder has a method called "lookup" which can convert text-format addresses to geographic coordinates, which means, if the app gets one address, it can be positioned on the map based on the converted coordinates.
This resource is provided by Apple development document, so it is easy and trusted to implement. The logic is, M-Track gets the parcel routes from the tracking number provided by users using some tracking functions, and we get a list of addresses indicating where the parcel came from and is in. Now M-Track triggers the geocoder to search the latitude and longitude using each address and display the location in the map with the relative tracking details.

3.2 SQLite offline data storage (persistent storage)

Supported function: accessing tracking function without login state

M-Track provides pure tracking functions without login, not forcing users to create accounts first. Therefore, we respect your choice, keeping your data only in your local space.

SQLite is the most popular relational database, ideal for handling simple local data storage. It is embedded in an iOS application with no configuration. It is very small and light, but can meet the underlying database storage needs, for example, storing data in tables with one or more columns, specific type saving, and powerful queries to add, delete or manipulate data.

M-Track basically uses SQLite for saving the tracking numbers with the time user adds it) and the personal settings for the users not logged in. It is totally affordable for SQLite to save the such simple data. This also minimize the application size and lighten the load.

3.3 Parcel tracking API – Order Tracking (web services)

Supported function: parcel/order tracking

6

This provides the needed endpoints to support all the tracking functions. Tracking details M-Track searches will be fully provided via it. It is free to use and covers more than 600 carriers, strongly supporting our app in the current stage. It also saves the developer's time.

The example usage will be: Assume our user enter one tracking number, Order Tracking will search for possible carriers using this number, and once the user select the correct one, it will search the tracking details by using the number and the carrier's relative code, and once the result is gotten, we represent it to the user.

3.4 Firebase

Firebase is a mobile development platform and provides several features for iOS apps. It can help build apps fast without managing the original functions. Giving functionality intuitively can save the time and cost of app development. It can give the functionality like databases, messaging and analytics, which help leverage M-Track. And using the same source approach also helps reduce development difficulty and ease of maintenance.

3.4.1 Firebase Authentication

Supported function: User login & sign-up

This is used to verify user's identity and the personal settings and data synchronization. M-Track decides to give users a safer but easier way to create and get access to their accounts to keep their data among multiple devices. This is also for better supporting the in-app chatting system, as only the users in the login state can chat with others.

M-Track will have 4 types of account creating, 3 of which use firebase technology to save user identity:

- Email address and password sign-up/in
- Google sign-in
- Facebook login

However, the authentication will only be applied for email login. Other sign-in will be mentioned later.

3.4.2 Cloud Firestore

Supported function: In-app chatting
M-Track provides a chat system for logged-in users to easily share tracking details and communicate with others. Therefore, we select Firebase to do the real time message data synchronization in the basis of its authentication. Firestore provide a popular data structure with value key pairs, called JSON object. Each dialog window is a channel, which includes multiple messages with sender identity. When one message is sent out, the content, sender details and time will be added to firebase database in real time. Then Firestore will call a listener, like triggering synchronization function, and the message sent out before will appear on the screen. Therefore, two users are in the same channel, message sent by each will be synchronized in real-time and that is the chatting we do.

For in-app sharing, content (one tracking details) will be uploaded to Firebase first, then M-Track prompt you that which contact you want to share with, then it directs you to the chat board and sharing message is generated and sent.

Supported function: Rating

The carrier rating system also relies on the Firestore. The database will take how many ranking stars the user gives, what comment user give, and who the user is. The server can also send down the average rating by calculation. Therefore, we take what the user gives to the database first, then the it sends back the contents and we show it on the app.

3.5 Sign in with Apple

Supported function: User login & sign-up

Besides 3 types of login methods above, M-Track also allow users sign in with Apple, which is one of the most convenient way in Apple ecosystem. As currently M-Track is an iOS-only app, we believe this is a trusted way for users to implement the sign-in. It is also stipulated in Apple's design guidelines that apps which uses a third-party or social login service for user's primary account must also offer Sign in with Apple as an equivalent option. Apple has a very sophisticated Apple ID system and users can rely on it with no doubt.

3.6 Message Kit

Support: chat interface

As the name mentioned, this is a kit which help M-Track create the chat view, therefore we don't need to build it manually.

It has ability to handle the input, the message bubbles, also the avatar. It provides the message structure including sending time, sender name, sender avatar, the content in the bubble, and the read status. It totally meets M-Track's requirement and keep updated. It also has rich documentation with examples, which is easy to embed to our app and function trusted.

3.7 SwiftUI

Support: Bar chart for tracking statistics

SwiftUI provides rich graphics capability which can be used to build the bar charts. Using official embed UI section is trusty and more convenient to use with official documentation. SwiftUI can provide the chart refactoring, the rectangle shape views, and add more details to increase its readability. Without other extension/libraries, it is enough to provide the function to us. That is why we choose it.

3.8 Google sign-in

https://developers.google.com/identity/sign-in/ios

As Google documentation provided, we need to set up our CocoaPods dependencies, and get OAuth client ID. We will enable the sign in for Google and add the sign-in button to the app. If setting successfully, we can get the profile information and save it to Firestore to manage the user identity.

3.9 Facebook sign-in

https://developers.facebook.com/docs/facebook-login/ios#quickstarts-header

Similar to Google, Facebook also provide a full development documentation to us. Register as a developer first. And set up the environment and follow the steps as documentation mentioned.

3.10 UIKit

Support: list views in M-Track

UIKit is a framework from Apple, which construct and manage the interface of iOS app. The class UITableView presents data rows in a column as what our lists look like. Tracking list, chat list, contact list and rating list will be applied.

3.11 Specifications of "link with shop"

It is mentioned before that M-Track has a function of linking with shops. Unfortunately, eBay and Amazon do not have the full functionality provided. We here assume a company "FutureShop" provides the fully access which M-Track can make use of.

We get the API of accessing the FutureShop, which prompts users to put their login information. If login successfully, which means the identity is verified, then one token will be returned and saved to our DB. Every time we use API to request the resources from the shop, the token will be attached to verify the authorization and return the resources we want.

4 INTERFACE DESIGN & STORYBOARD MOCK-UPS

4.1 Human Interface Guidelines

The app full storyboards are listed here. All designs are in line with the Apple Human Interface Guidelines (HIG). App icon, color scheme, and the typography it uses will be mentioned in the following part.

- Launching: we provide a launch screen with a head photo, the app name, and different accessing methods including sign-in/up or try without login. It clearly gives users the first impression that what the app is and how it can be used. It also always keep the same orientation whatever people are in the portrait or landscape.
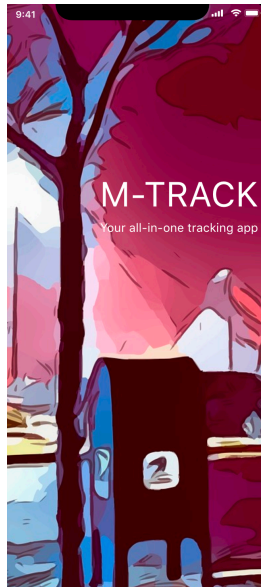
- Navigation: bottom tab bar clarifies the peer function sections and make users easier to switch between them. Current section is highlighted to prompt users where they are.
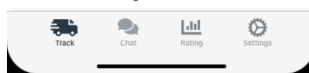
- Authentication: we have functions which no need user sign in first, which corresponds to the rule "Delay sign-in as long as possible". We try to make user try our app first, and then make a decision to create an account.
  We also provide "Sign in with Apple" as one of the sign-in option, trying to give user a simple and secure way to sign in.

- Data entry: we provide the convenience for users to enter the carrier. It will be auto suggested once tracking number is put or some letters are put.
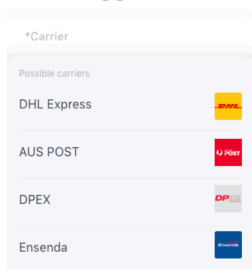
- Gestures: standard gestures are provided. When deleting one chat, users swipe left the chat from the list and delete button will be shown.

10
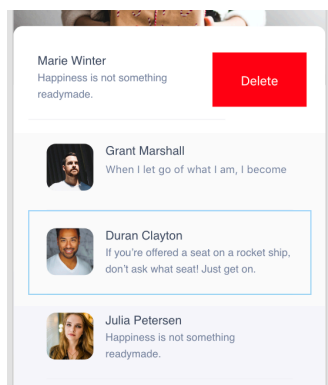
- Adaptivity and layout: Auto layout is applied for M-Track, and each element will have constraint with others, to make sure it can be suitable for different screen sizes. Safe area is also applied to fit the guide. Margins are kept around it.
- Color: M-Track use a set of color scheme which makes sure the color match looks great both in individual and combination, and suitable with light and dark modes.

| HEX #293844	HEX #9791A0	HEX #D8AEAE	HEX #FBDAC3	HEX #FFF1DD

Other colors for special used:

#FD003A ▮ used for highlight points and notifications dot.

#FFFFFF   used for text.

- Dark mode: to correspond with iOS 13, dark mode is also designed for our app. As background changed to dark, the color used is to ensure the users can see the contents still clearly with the same brightness. Two screens are given to show the example dark mode in the mockup.
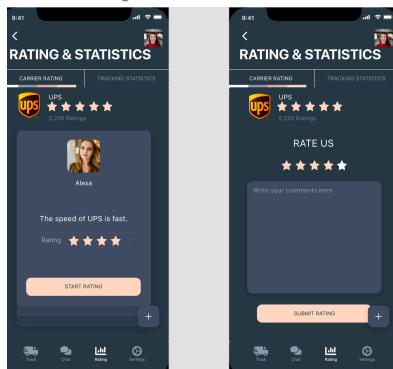
- Terminology: in the app, as the guide mentioned, we provide short but accurate text hint or description to make users feel comfortable. Language which might sound patronizing is avoided.
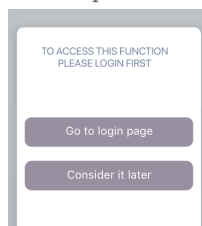
- Typography: M-Track uses standard fonts provided by Apple, which is SF pro text with regular and semibold types.
- App icon: the app icon uses the truck to show the fast parcel delivery. the color matches with the theme colors and corresponds to what the guideline mentions. Here provides an App store version.

- Tables: M-Track has several lists which uses tables views. We use the customized table styles but also following the guides using the plain design and keep every row aligned.
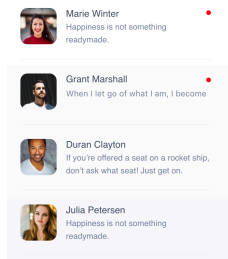
4.2 App mockup and hierarchy

Aggregate PDF version, separate PDF version and the Adobe XD file are attached.

5   SCOPE & LIMITATIONS

Due to the time limits, the functions indicated before cannot be all implemented. Therefore, scope is provided here to clarify the functions that must be completed before the app is released.

Minimum viable product:

Parcel tracking	Chat board	Rating system	Account management
1. Create parcel tracking with tracking numbers and carriers 1.1 Suggested carrier list is shown after tracking numbers entered 1.2 Allow users to manually enter carrier	1. Add contacts by email address	1. Rate carrier 1.1 Give star rating 1.2 Write comment to carrier	1. Sign up with email and password
2. Edit tracking details i.e. tracking numbers and carriers are able to change	2. Chat with contact by text	2. Check carrier rating list	2. Sign in with email and password
3. Show tracking details with locations in the map	3. Share tracking details with contact	3. Show carrier rating details with carrier name, user rating details	3. Sign in with Apple
			4. Sign in with Google

As the table shown, 13 functions will be the requirements of MVP.

Tracking is the core of M-Track. Therefore, most of the functions in this section need to be implemented.

For the chat board, we make sure users can add contact with email. Even though the account can be sign in with Google or Facebook, we only implement the email adding first due to technical and time limits. For chatting, only text and sharing contents will be able to send in the initial version. Media, audio and video messages will be considered in further versions. Tracking details are able to share to any contact the user already add.

For rating system, only simple ratings can be provided by users. In the initial version, users are able to give the star rating to each carrier but the average star ratings each of them has gotten will not be displayed with the list as the technical concerns.

Account management is also important. As mentioned before, chat and rating functions need users sign in first. Therefore, we design that M-Track allows users sign in by email, with Apple and Google account. Facebook login will be in the further release as no enough time to cope with.

Statistic board and shop accounts linking are not included in the initial version as they are the bonus points for M-Track but not in the first level's necessity. Therefore, their development will be in the further releases.

6   ESTIMATED PROJECT TIMELINE

	TASK NAME	START DATE	END DATE	DURATION (WORK DAYS)	PRE TASK
M-Track					
1	Create parcel tracking with list displaying	4-May-2020	8-May-2020	5	
2	Make auto suggested carrier list	5-May-2020	8-May-2020	4	
3	Show tracking details with map	11-May-2020	15-May-2020	5	1
4	Sign up with email and password	11-May-2020	12-May-2020	2	
5	Sign in with email and password	13-May-2020	15-May-2020	3	4
6	Add contact by email	18-May-2020	20-May-2020	3	5
Week 9 prototype submission					
7	Chat with contact (text only)	21-May-2020	25-May-2020	5	6
8	Share tracking details with contact	26-May-2020	29-May-2020	4	7
9	Rate carrier	26-May-2020	28-May-2020	3	
10	Carrier rating list	1-Jun-2020	2-Jun-2020	2	
11	Carrier rating details	3-Jun-2020	5-Jun-2020	3	9
12	Sign in with Apple & Google	8-Jun-2020	10-Jun-2020	3	
13	Edit parcel tracking	8-Jun-2020	11-Jun-2020	4	
Week 12 prototype submission					
	Function testing	15-Jun-2020	19-Jun-2020	5	13
	UI improvment	18-Jun-2020	19-Jun-2020	2	
	Final testing for initial released	22-Jun-2020	26-Jun-2020	5	
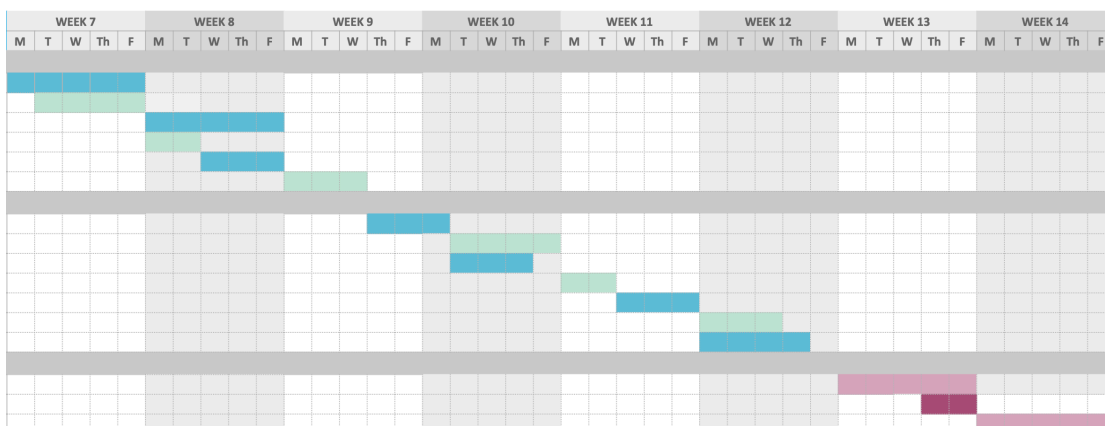
Breakdown of the development tasks are provided using Gantt chart as above. Each task has start and end date, with the pre task mentioned if needed. Week 9 submission includes the first 6 functions and till Week 12 most of the functions will be finished. After week 12 prototype submitted, one week will be used for handling possible delay as the slack time and also do the testing and improvement. Final testing will be on the last week before the initial release.

Weekends are not included in the chart. These times will be used for handling delay of functions.

(Full PDF version is attached)