SET08114 – Mobile Applications Development

App Coursework

TR2 2018

Mark Cunningham
40063795

Table of Contents

Introduction

The app, imaginatively named ‘FoodOrder’, is not the first app of its type and is more of a business idea rather than just a standalone app. The app is designed as a food ordering app and was inspired by other apps such as ‘HungryHouse’ and ‘JustEat’. FoodOrder was also inspired by another app of a different kind called ‘Postmates’ which does not deal in food but what they do is allow users to buy any goods and ‘Postmates’ will find the item and deliver it to the user. There is one main difference between this app and the others already out there which is, the menu system. The already existing apps menu system can be confusing and tedious to find what the user wants to order as the user must first select a restraint/takeaway and then they get to see the food offered. This wouldn’t be a problem if there was a small selection of places to order from but each of the mentioned apps are connected with a massive number of restaurants and takeaways.

The idea behind ‘FoodOrder’ is that it would be a mixture of ‘Postmates’ and ‘HungryHouse’/’JustEat’ in that all of the food is collated into one menu. This way the user does not have to search restaurant by restaurant looking through menu after menu, they just select the category of what they want for example pizzas or soup etc. The user is then shown all of the options within that category, and then when the user selects the item they are given a description of the food and can easily add it to their cart from there.

Software Design

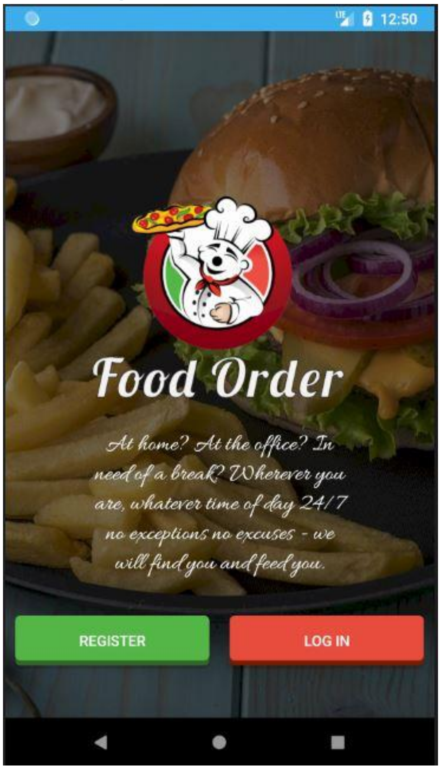
The app was originally designed to be as simple and as user-friendly as possible. It does this by having buttons clearly visible and clearly visible so that it is near enough impossible for the user to not see an option. For this reason, a dark background was chosen so that important information and buttons that use brighter contrasting colours stick out.

The first bit of design that took place was the apps logo, once this was created this spawned the colour scheme for the rest of the app. The logo had to be food related so the chef with a pizza was an obvious choice. Behind the chef it seemed fitting to have the green white and red of the Italian flag which is why the app follows the green white and red theme.

Each function of the app has its own page, this is done again for ease of use so nothing can be missed by the user. The disadvantage of this is that the app has ended up with a large number of classes and layout files so the app has ended up being quite large.

Application Implementation

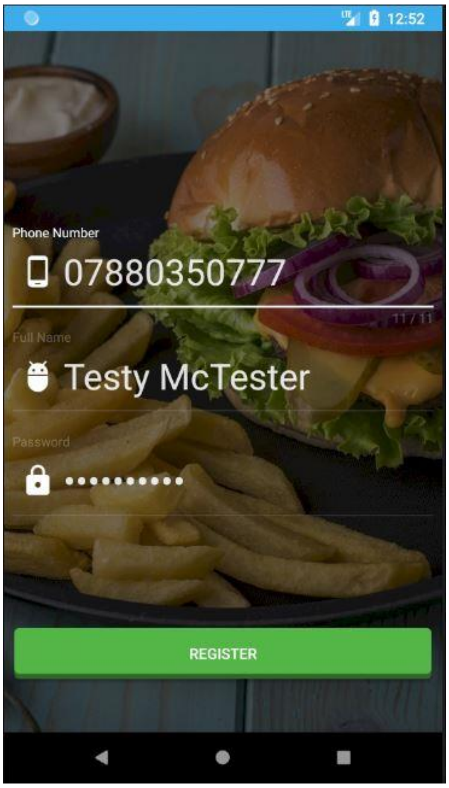
First Page



At home? At the office? In need of a break? Wherever you are, whatever time of day 24/7 no exceptions no excuses - we will find you and feed you.

Upon opening the app the user is greeted with the introduction page. This page includes the apps effective logo which as well as being visually appealing is also conveys the purpose of the app very well. At the bottom of the page, there are two very clear options that stand out from the darkened background. This dark background is a common theme through the app, this allows for an easy contrast. The two options at the bottom are labelled as “Register” and “Log In” with each directing to their respective pages.

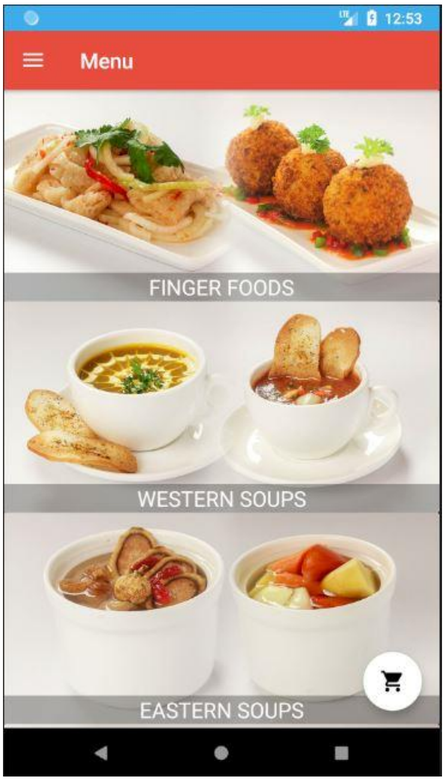
Register Page



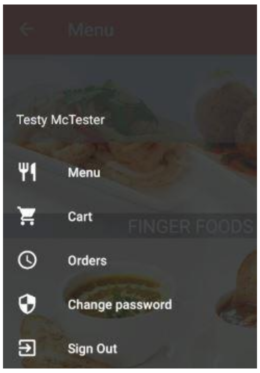
Phone Number
07880350777
11 / 11
Full Name
Testy McTester
Password
••••••••••

If the user selects the register option they are then taken to this page. This page makes use of a few of the user interface improvement libraries that are implemented within the project. The page, in particular, uses the improved text boxes which have a much nicer and cleaner look than the default android boxes. The page also includes the counter underneath the phone number text box which allows the user to make sure they have entered their 11 digit phone number. This page also utilises the ConnectivityManager method, this allows the app to check that the user has an active connection to the internet before allowing them to attempt to create an account.

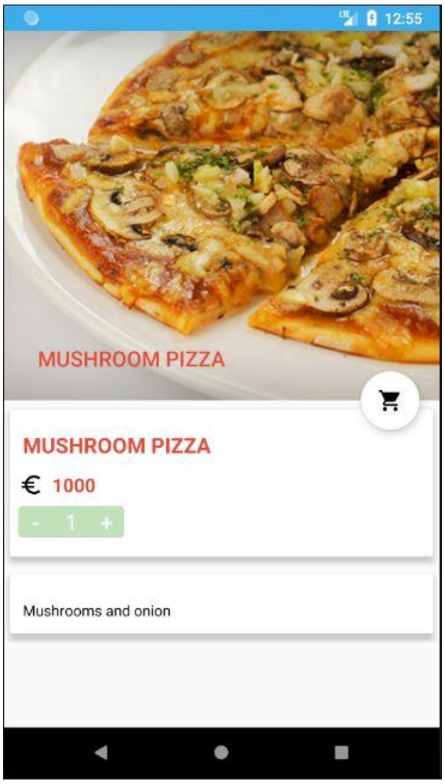
Menu Page



After logging in the user is redirected to the menu page. The menu page uses a card view to display an image representing the food category with the category name. The links for the images are pulled from the Firebase database, the images are then indexed for faster loading. This page is also the first page to implement swipe to refresh, this is done because it is possible for the app to fail to pull an image from the database so this way it can be easily refreshed without leaving the page. The page also uses a floating action button as shown at the bottom right which takes the user to the cart. At the top of the page is a drop-down menu that is used to navigate between pages, as shown here. Along with the register page and login page, this page also checks to make sure the user has an active internet connection before allowing the user to select any food categories.

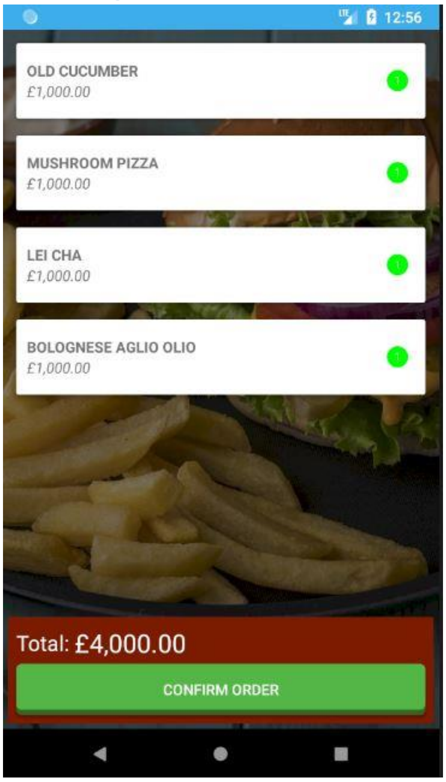


Food Page



After selecting a category and then a specific item within the category the user is directed to the feed details page. This page displays an image of food, the reasonable price, a description of the item and allows the user to add one or more of the items to the cart through the use of a floating action button and counter from another third party library.

Cart Page



OLD CUCUMBER
£1,000.00

MUSHROOM PIZZA
£1,000.00

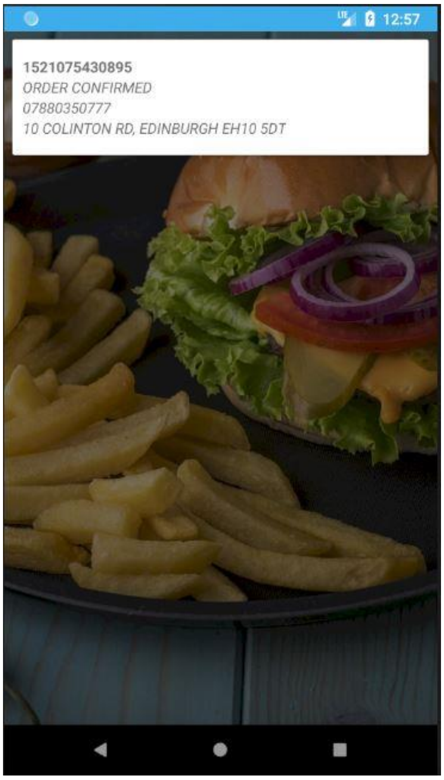
LEI CHA
£1,000.00

BOLOGNESE AGLIO OLIO
£1,000.00

Total: £4,000.00

As expected the cart page displays a list of items in the cart, a long press on the item allows the user to remove the item from the cart. When items are added to the cart they are temporarily stored in an internal SQLite database. When the order is confirmed the data is sent to the Firebase database and is deleted from the SQLite database. This page also implements androids swipe to refresh feature.

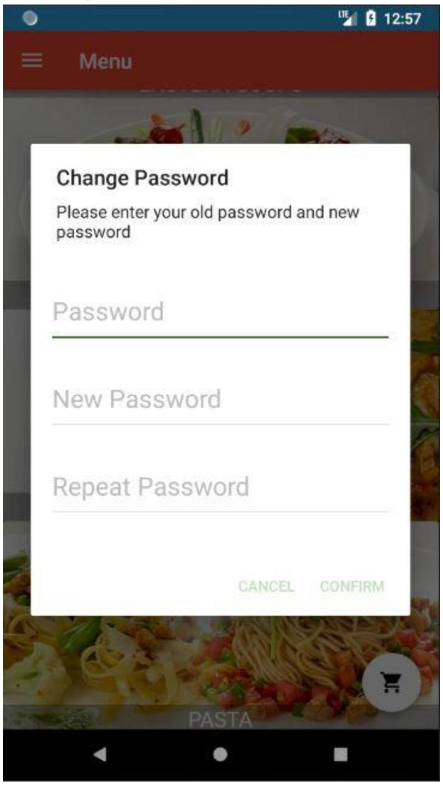
Orders Page



1521075430895
ORDER CONFIRMED
07880350777
10 COLINTON RD, EDINBURGH EH10 5DT

The orders page is a simple page that pulls the order information from the Firebase database and presents it to the user.

Change Password


Change Password

Please enter your old password and new password

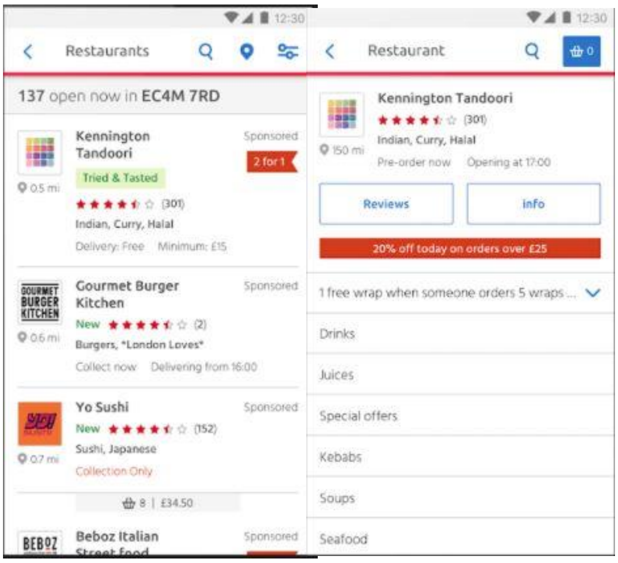
The change password option is a popup that uses the custom Spots-Dialog library, this works similarly to the default android alert dialogue but it offers more versatility and customisation options and is designed to intuitively take input from multiple text boxes.

Critical Evaluation

The app is far from perfect and there are a vast number of features that were planned to be implemented but that unfortunately have not yet been developed. Some of the features that would be planned for future development and improvements that could be made if the app were to be continued are;

- The menu would be extended, the current menu data is merely a placeholder to show what would be expected in the app.
- Password reset function – this would send a text to the registered number to reset the password, however, this requires a paid service and would only be implemented if the app was to be released.
- There would be a staff log in so that they can get the users orders and update the order status.
- Upon confirming the order the app should ask for a street, house number, postcode etc. separately instead of one line for the delivery.
- Allow the users to add certain items to their favourites list.
- Allow users to see other peoples orders in their area, this would only show the food ordered it would not give the delivery address.
- Add a payment system, obviously, the users need to be able to pay for the food, at the moment this step is skipped.
- Allow logging in with Facebook and Google, large numbers of apps allow users to log in with their social media accounts so they don’t have to have any usernames or passwords to remember for the app.

- Share the order to social media. People like to share so a simple “I just ordered item A, item B & item C using FoodOrder” could be shared via facebook, twitter, google+ etc. at the user's request.
- The user interface could be greatly improved, as most of the buttons and images are quite large there isn’t a lot of information on the screen at one time. If we compare FoodOrder to JustEat the difference is huge in this respect. JustEat does not use large images, instead, it has a small image representing the business with its name and rating. Also on their menu pages, they have a wall of drop-down menus for the food groups which isn’t necessarily a good thing. Somewhere in between the two would be more ideal. The images below show JustEat’s approach.



12:30

Restaurants

137 open now in EC4M 7RD

Kennington Tandoori — Sponsored — 2 for 1
Tried & Tasted
0.5 mi
★★★★☆ (301)
Indian, Curry, Halal
Delivery: Free Minimum: £15

Gourmet Burger Kitchen — Sponsored
0.6 mi
New ★★★★☆ (2)
Burgers, *London Loves*
Collect now Delivering from 16:00

Yo Sushi — Sponsored
0.7 mi
New ★★★★☆ (152)
Sushi, Japanese
Collection Only

8 | £34.50

Beboz Italian Street Food — Sponsored

12:30

Restaurant

Kennington Tandoori
★★★★☆ (301)
150 mi
Indian, Curry, Halal
Pre-order now Opening at 17:00

20% off today on orders over £25

1 free wrap when someone orders 5 wraps ...

Drinks

Juices

Special offers

Kebabs

Soups

Seafood

Personal Evaluation

The app development process has greatly increased my java and android development skills from next to none to the point where I have produced a fairly functional app. When choosing what app to develop I attempted to pick an app that would incorporate functions and services that had not been covered in labs. I decided to do this because I wanted to challenge myself, however that may not have been the best idea in retrospect. In many areas of that app I was ‘going in blind’ without a base knowledge on how to do exactly what I wanted, this forced me to spend hours researching the task and how it could be solved. This caused a knock on effect and cause a time issue for the app, as outlined in the previous section there are a lot of features that I would have like to implement but couldn’t within the time frame.

Overall I feel my performance from start to finish was not as effective as it could have been. For instance the logo for the app which although I am happy with and feel that it fits the app well, it did take many hours to create. This time could have been better spent coding the app. I also feel like my planning was sub-par, numerous times during the process I was forced to restructure layouts, change databases and redesign whole sections because I tried to rush some aspects which backfired.

I also decided to use quite a few libraries for things that could be achieved using the native android libraries. Most of the libraries used were for user interface improvements which do help with the aesthetics of the app and have also helped me to learn how to work with multiple libraries that sometimes don’t like to get along.

References

The main sources of information used were java programming forums and android developer forums. These paired with demonstrations and informational videos from YouTube provided the information required for the project.

http://www.dreamincode.net/forums/forum/32-java/

http://www.javaprogrammingforums.com/

https://www.java-forums.org/forum.php

https://androidforums.com/forums/android-development.28/

https://www.androidpit.com/forum/android-developer-forum-general/recent

https://developer.android.com/support.html

https://www.tutorialspoint.com/android/index.htm

https://stackoverflow.com/

https://www.youtube.com

Links to external third-party libraries used:

Google Firebase – database used

https://firebase.google.com/docs/android/setup?authuser=0

Paper – useful tool for easily writing data to the device

https://github.com/pilgr/Paper

Picasso – Image downloading and caching

https://github.com/square/picasso

SQLite asset helper – used to write data to SQLite DB

https://github.com/jgilfelt/android-sqlite-asset-helper

MaterialEditText – Highly customisable edit text box – UI Improvement

https://github.com/rengwuxian/MaterialEditText

Spots-dialog – Dialog box – UI improvement

https://github.com/d-max/spots-dialog

LikeButton – previously called FlatButton – Customisable buttons – UI Improvement

https://github.com/jd-alexander/LikeButton

ElegantNumberButton – incremental numbers under edit text box – UI Improvement

https://github.com/ashik94vc/ElegantNumberButton

TextDrawable – Extends the drawable class for more functionality – UI Improvements

https://github.com/amulyakhare/TextDrawable

Material – Easily useable animations – UI Improvements

https://github.com/rey5137/material