Informatics Large Practical

Coursework 2

Ivan Karaslavov s1632798

December 2018

1. Description of the implementation

The project is tested on Nexus 5X API 26

The report contains a thorough description about a mobile phone game implemented for an Android device. The app is a prototype of a multi-player collaborative location-based game where players collect and exchange virtual coins which have been scattered at random around the University of Edinburgh's Central Area. Given that it is based on collecting coins, the game is called Coinz, with the unusual spelling with a 'Z' making it easier to find information about the game online. In order to play, the user has to make an account within the database of the app. The database used for the game is Firestore and the hosted database service is Google's Firebase platform. The app uses Firebase's authentication and for that we will not discuss it further. If the user is not already logged in, the app starts with a login page (Figure 1). If the user doesn't have an account, they can make one by clicking the Sign Up button at the bottom of the page. A Sign Up page shows up and the user has to register with a unique email address and secure password which has to be confirmed (Figure 2). If the email is unique and the passwords match, a personal account is created and the user is directed to a page where they have to choose a username (Figure 3). The username will define the user and once selected it cannot any longer be changed. The username also has to

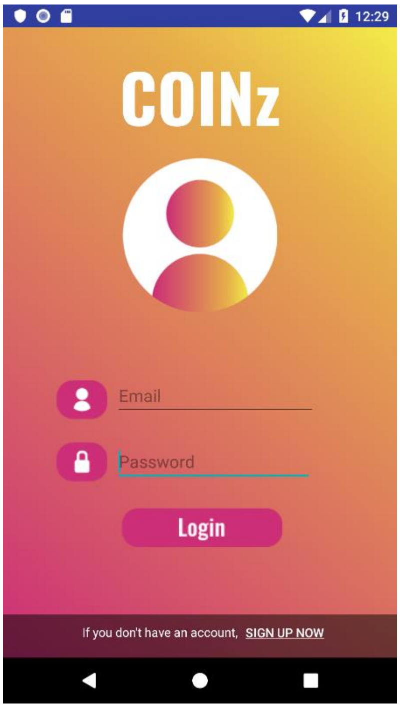


Figure 1. Login page

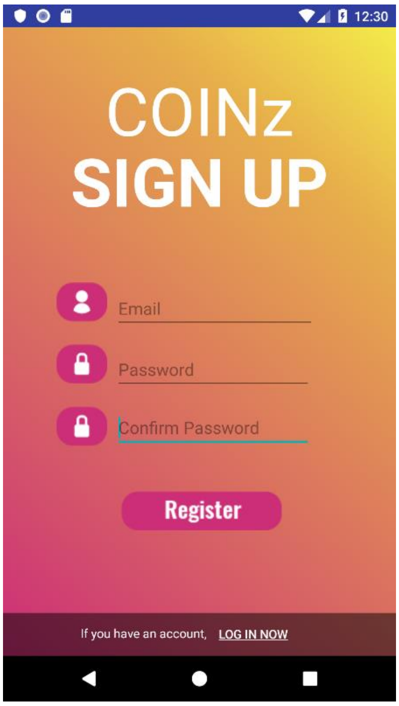


Figure 2. Sign Up page

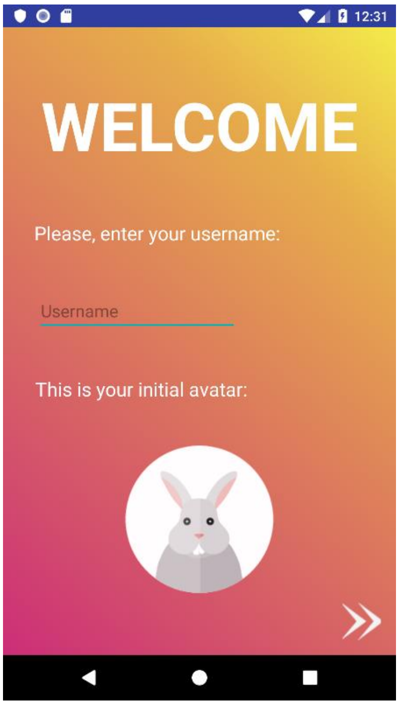


Figure 3. Username page

be unique as this will be of crucial role for interacting with other users which will be further explained. Apart from the unique username, the page consists of information about the avatar of the user. Every user starts with the same avatar, but later on it can be changed.

When the sign up process finishes, the player is directed to the main menu. The main menu represents a map (Figure 4). The app uses Mapbox as map service. The map consists of different coins represented with different colours. In the Coinz game,

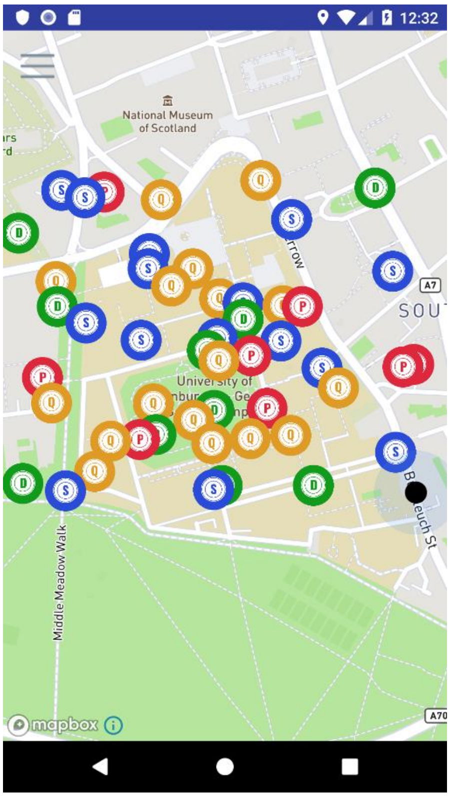


Figure 4. A map with 50 coins

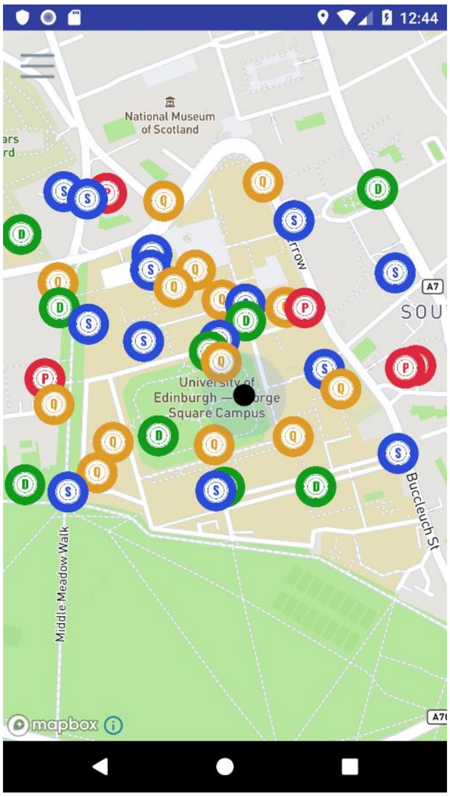

12:44
National Museum of Scotland
University of Edinburgh — George Square Campus
SOU
A7
Buccleuch St
Middle Meadow Walk
mapbox

Figure 5. A map with some collected coins

the locations of the coins are specified on the map. The coins are collected by getting near to their location on the map, by which I mean that the player literally moves to that location with their mobile phone. A player will be judged to be close enough to a coin to be able to collect it if they are within 25 meters of the coin. A new map is released every day. Each map has fifty coins. The user will know where the coins are on the map with the start of the new calendar day. The coins that were not collected the previous day will disappear and will be replaced by new ones. When a coin is obtained, its indication on the map will disappear (Figure 5). That way it will not mislead the user of thinking that the coin is still to be collected.

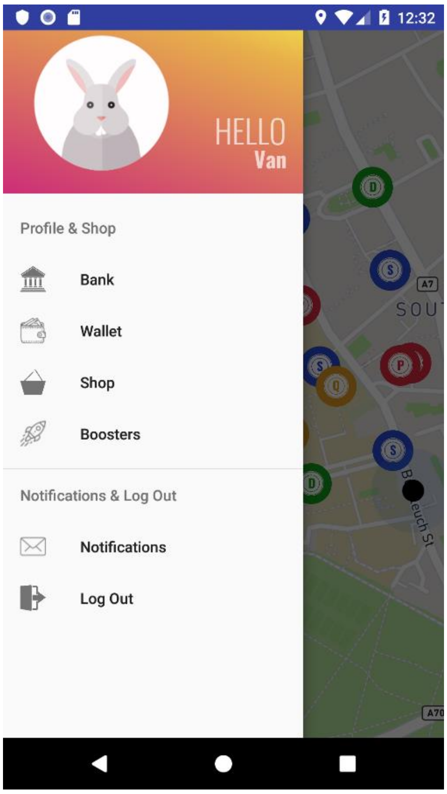
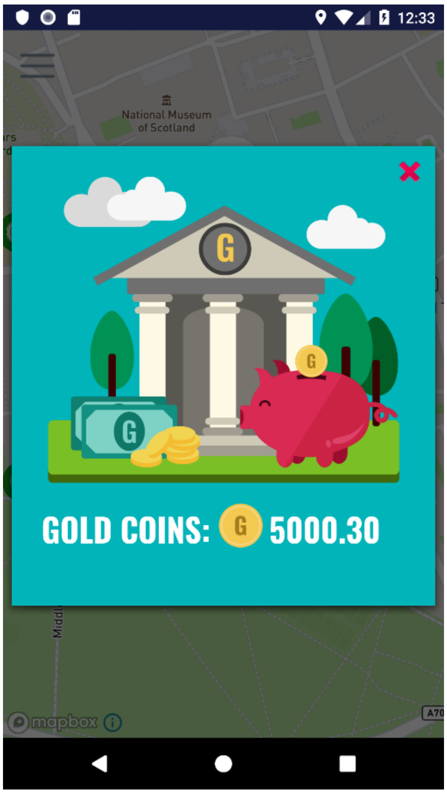


Figure 6. Main menu



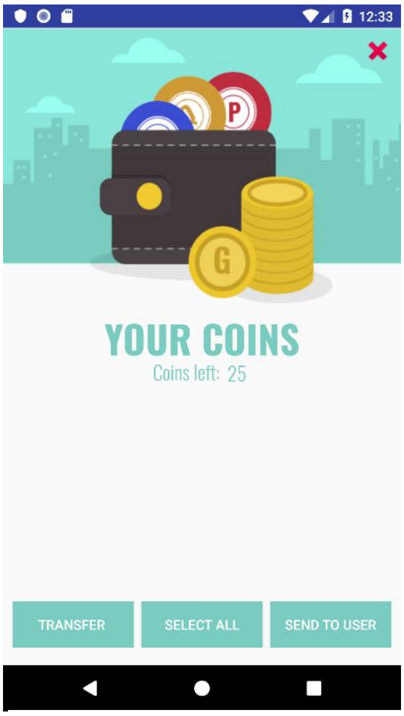
GOLD COINS: G 5000.30

Figure 7. Bank account

The date of the last download is stored within the shared preferences of the app. If the current date doesn't match the date of the last download it means that a new map should be released. There is different map for every day of the year. The map consists of the locations of the coins, their currencies and the rates of the current day. The coins which are being collected belong to one of four different cryptocurrencies: Penny, Dollar, Shilling and Quid (these are respectively denoted by the four-letter acronyms PENY, DOLR, SHIL, and QUID). The four different currencies fluctuate in relative value on a daily basis. The relative values of the four currencies are defined in terms of their relation to a fifth currency, GOLD. Maps contain coins from the currencies PENY, DOLR, SHIL and QUID, but no GOLD coins. When the user signs up a new map is being downloaded from the web. The map is stored both on the local device and in the database. It is stored in the database as every user has to have different map with the coins they are collected. If the user logs out and logs in with different device their progress has to be saved. The map is stored on the local device in case there is no internet connection, but the user continues to collect coins. Once the connection is recovered, the local progress will be uploaded to the database.

The map represents a GeoJSON object. GeoJSON is a format for encoding a variety of geographic data structures. However, when the map is downloaded from the web it is in the form of a string. Then the string is converted to a JSONObject so the information from it can be read and displayed. Using the key "features" an ArrayList is created containing the coins. ArrayList is used as the amount of the money every single time will not be known. Coins will be constantly added and removed and that makes the list the perfect data structure for these needs. The information about each coin is gotten from the "properties" key. These properties are used to save the currency of the coin, it's marker symbol and the specific location that we use the place it on the map. Apart from using the first letter of each currency as distinctive mark, different colours are used for the coins. That way it will be easy for the user to distinguish them. The rates of every currency related to GOLD are stored in the map file, but are not displayed to the user. The information about the map is stored as string for space efficiency and it is parsed to JSONObject every time when the location of the user is changed in order to find out if the user is within 25 meters of a coin.

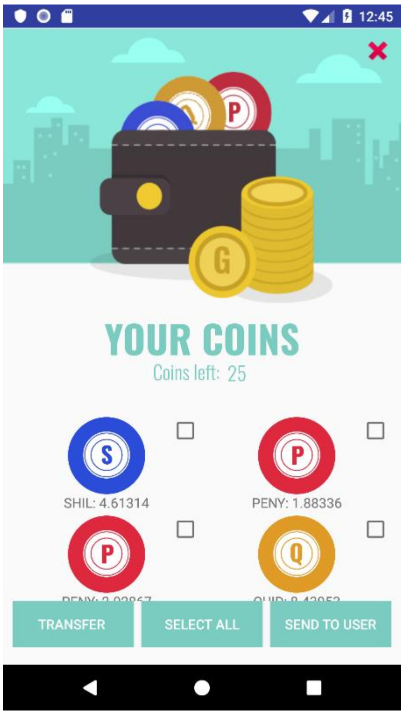
If the player is within the given area of the coin the coin is collected. When I say collected I mean that the marker of the coin is removed from the map, the map file is updated (the given coin is deleted from the ArrayList) and a new file is being updated (the given coin is added to the ArrayList). This new file is called "wallet" and as the name suggests it contains the coins that are in possession of the user (in their wallet). The structure of the wallet file is the same as the map file – a string that is being parsed with every change of the user's position and being updated every time when the user gets a coin. Apart from the coins, the wallet file contains the rates of the different currencies for easy access later and like the map, it is reset with the start of a new day.



YOUR COINS

Coins left: 25

Figure 8. The wallet with no coins



YOUR COINS

Coins left: 25

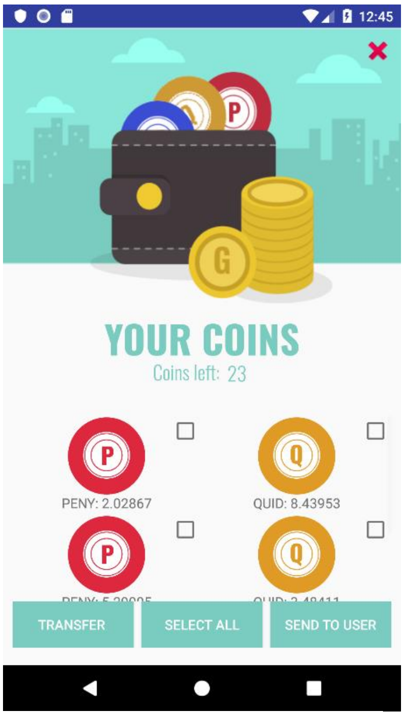
S □ SHIL: 4.61314

P □ PENY: 1.88336

P □

Q □

Figure 9. The wallet coins



YOUR COINS

Coins left: 23

P □ PENY: 2.02867

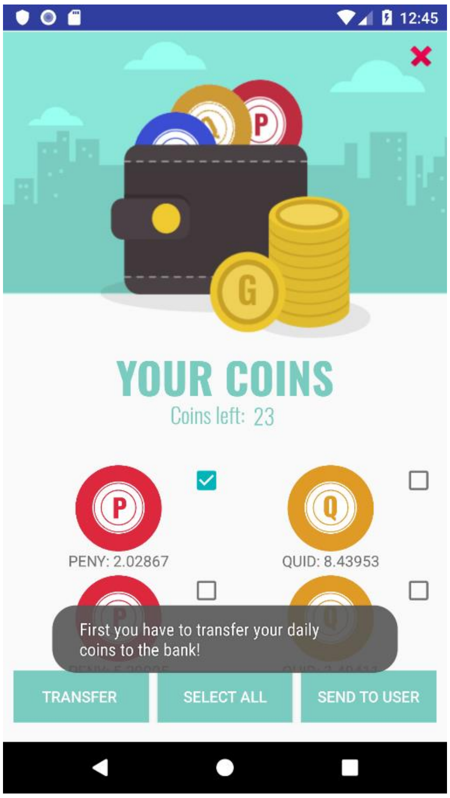
Q □ QUID: 8.43953

P □

Q □

Figure 10. The wallet with 2 transferred coins

The distance between the user and every coin is calculated using a specific formula. Using the latitudes and the longitudes of the player and the coin and the Earth's radius we can calculated the distance between the two in meters. Meters is the measurement that is used within the app, because it is the best one for comparison. If we use kilometers we will work with very small numbers and if we use millimeters an overflow might even occur.



First you have to transfer your daily coins to the bank!

Figure 11. The user can send only spare change

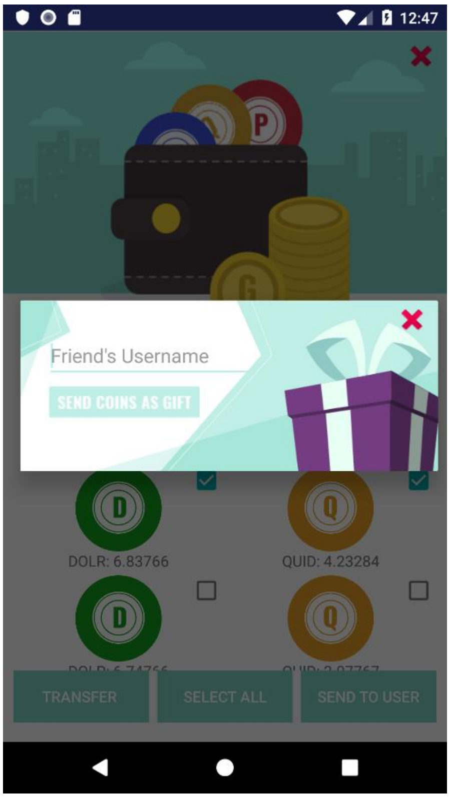
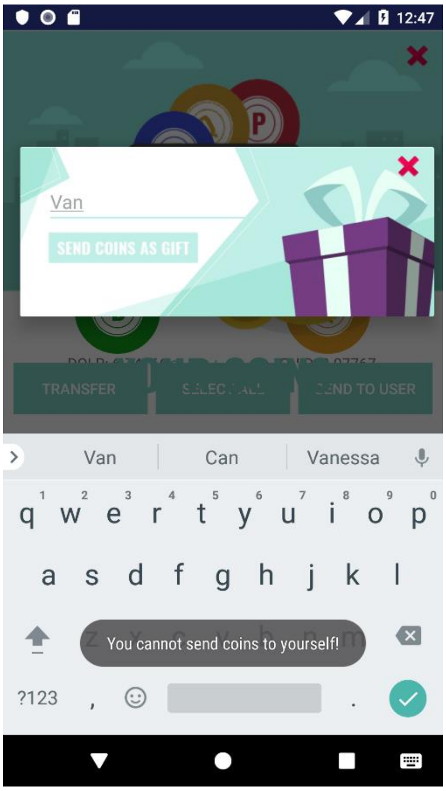


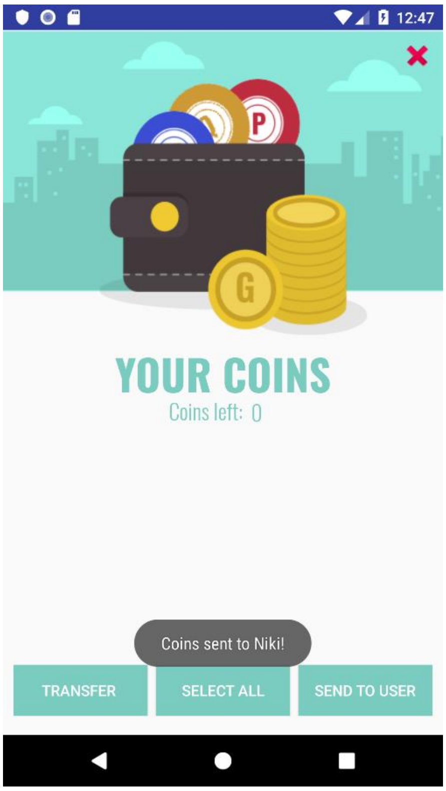
Figure 12. Gift dialog

After talking about the logic behind the map and the wallet and the main aspects of the game we will continue with the user interface of the map page. There is a menu button in the upper left corner of the map which displays a navigation drawer with the functionalities of the app (Figure 6). The first one is the bank. During gameplay, players collect coins, storing them in a local wallet, and later pay them into their account in the central bank. The bank is the place where the user can see their current amount of money (Figure 7). The currency of the money in the bank is GOLD. That means that after paying the coins into the bank they will be converted to the GOLD amount based on exchange rates of the current day. The players can buy goods from the stores and customize the app only with GOLD coins. As the gold amount is of type double, I decided to present it to the user rounded to only 2 decimal places.



You cannot send coins to yourself!

Figure 13. The user can’t send coins to themselves



Coins sent to Niki!

Figure 14. Sent coins configuration

The second item in the menu is the wallet. The wallet is displayed as grid list view so it can be dynamically altered independently of the number of coins (Figure 8). The coins are read from the wallet file. Then they are parsed to JSONObjects and stored in an ArrayList containing HashMaps. As I said earlier the ArrayList is the most suitable data structure for storing the coins. However, when displaying them to the user, we not only need the coin itself but the icon representing it to facilitate the process of finding it (Figure 9). The key of the HashMap is a string – the currency of the coin and its value rounded to five decimal places (so it won’t be too long for reading). The value of the hashmap is the icon. The coins are organized into two columns for better visualization. The user can select a different number of coins and decide what they want to do with them. There are two options:

1. Transfer them to the bank. If the user decides to transfer them to the bank the coins that they chose are dynamically deleted from the wallet file and from the ArrayList containing the coins so that the selected coins are no longer visualized in the wallet (Figure 10). Once paid into the bank, the coins will be automatically converted to GOLD according to the value of the currency on the given day. That is why the user has to be sure which coins they want to pay into their account, because once they do it there is no revert option. On any given day, the player can only pay into your bank

account directly a maximum of 25 coins. This means that, even if they collect all 50 coins on the map, there will 25 coins which they will not be able to bank. These unbankable coins are referred to as “spare change”.

2. Send them to a friend. The user can message their spare change to another player, and they can pay it into their bank account even though they have not collected those coins themselves. Coins which a player receives from another player can be messaged over to their bank account, even in the case where they have collected and paid in 25 coins already. The user can send as many coins as they want. They can send all of the money that they have collected. There are two requirements. The first requirement is that the coins have to be already collected – a coins that is still present on the map and not in the wallet neither be transferred, not sent to another player. The second requirement is that the user must have already transferred the daily 25 coins to their own bank account before sending the spare change to a friend (Figure 11). If the user decides to send the coins, a new dialog windows is displayed which prompts the user to enter the username of their friend (Figure 12). The user cannot send coins if they don’t know the username of the receiver. The user cannot send coins to themselves (Figure 13). The user cannot send coins a user that is not present in the database of the game i.e. a user that does not exist. When a receiver is selected the coins

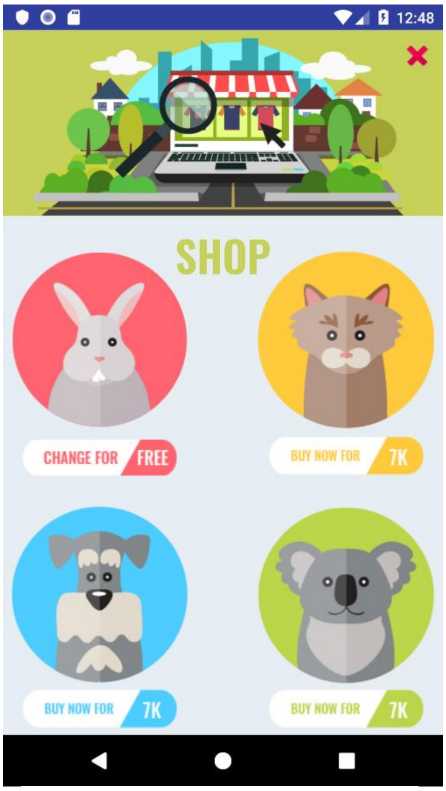


Figure 15. The avatars
(the cat is still not bought)

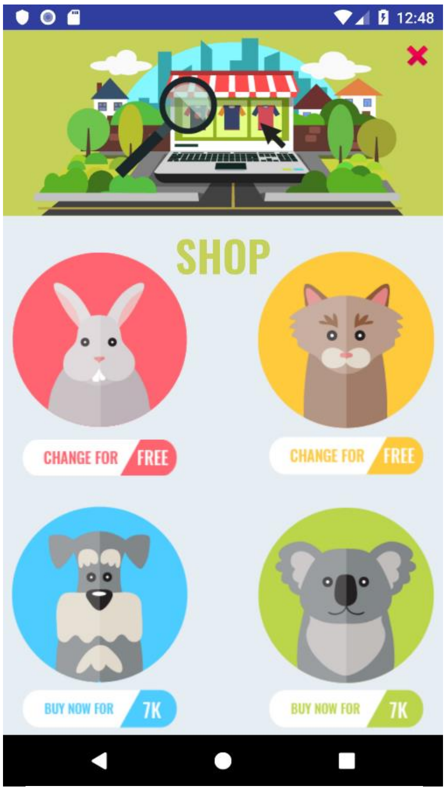


Figure 16. The avatars
(the cat is already bought)

will be deleted from the sender's wallet and will be dynamically transferred to the receiver's bank account converted to GOLD (Figure 14).

The coins that are held in the wallet (they can be the ones that the user collected themselves or transferred from another user and not paid into the bank) will expire at the end of the calendar day. So it is in the best interest of the player to send the coins that cannot be transferred to the bank to another player if any. That way they can help a friend and in return their friend can decide the same.

In addition to the game features, there are additional bonus features as well. These intend to make the game more interesting to play and more rewarding, causing the user to play more frequently and for longer sessions. The third item in the menu is the avatar activity (Figure 15). This activity is related to the first bonus feature – the user's preferences. The main purpose of the game is to collect coins. But the user does not know what to do with them. That is why there will be a shop within the game where they can spend their GOLD by buying different customization items like avatars. Customization lets users make their own selections about what they want to see, or set preferences for how information is organized or displayed. It can enhance user experience because it allows users to control their interaction. The user can choose one of the four present avatars. The

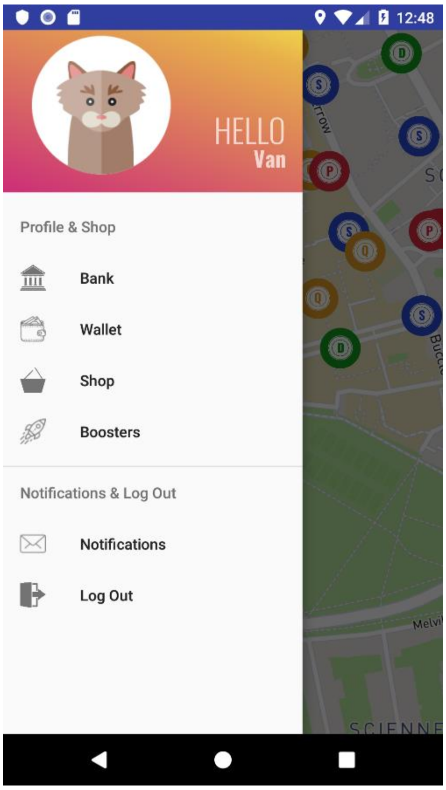
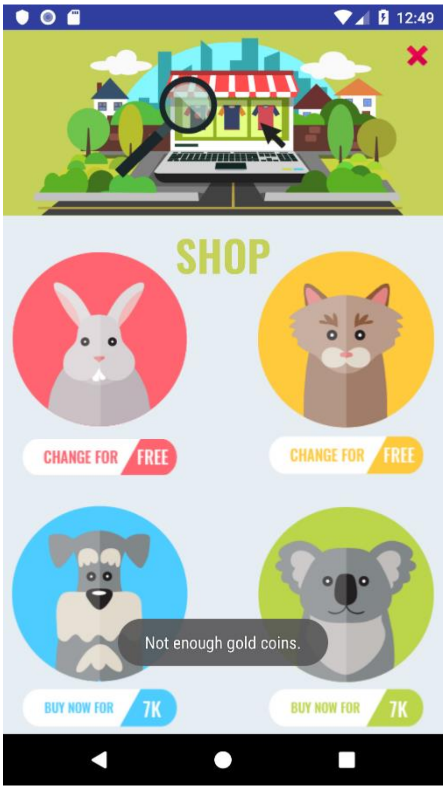


Figure 17. The profile avatar is changed

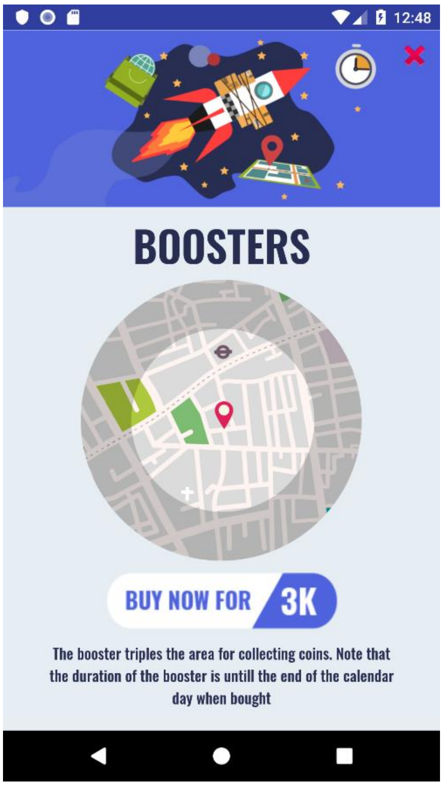


Not enough gold coins.

Figure 18. Not enough cold coins to buy an avatar

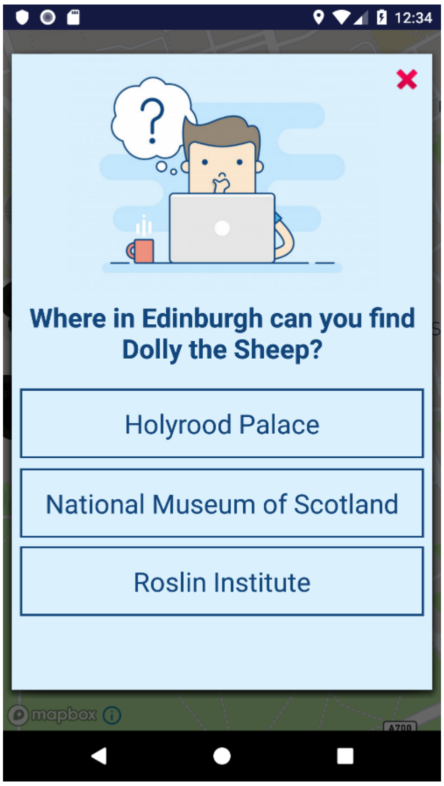
avatar when bought will appear in the navigation drawer as the user's profile picture (Figure 17). That way the player can customize the app to their own preferences and feel more of control. Once bought, the avatar can be changed for free at any time (Figure 16). There is also a booster shop currently containing only one booster (Figure 19). The booster is an expensive acquisition that can help the player to collect coins from further away by increasing the radius of getting a coin 3 times – the user can collect a coin within 75 meters. The booster can be bought only once per day and it is valid through the whole calendar day.

The second bonus feature is related to the value of the coins. The different currencies have different exchange rates on a particular day. However, the user will have a chance to increase the GOLD amount in the bank. Right after the coin gets to the player's personal wallet they will have to decide whether they want to take advantage of that feature or not. If they do, a question will appear (Figure 20). The question will be chosen randomly from a list of multiple choice questions related to Edinburgh – everything from its location and landmarks to the history



The booster triples the area for collecting coins. Note that the duration of the booster is untill the end of the calendar day when bought

Figure 19. The booster page



Where in Edinburgh can you find Dolly the Sheep?

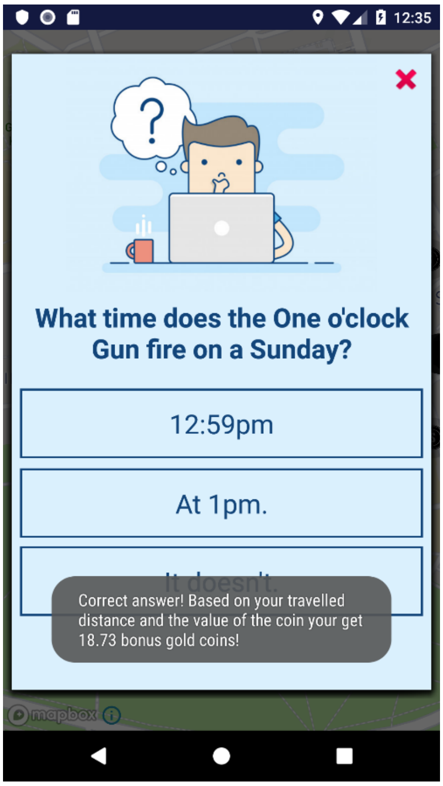
Holyrood Palace

National Museum of Scotland

Roslin Institute

Figure 20. An exemplary question

of the city and city life. If the player answers it correctly, they will get a bonus related to the value of the coin they collect (Figure 21). If they answer wrongly nothing will happen – it means that they will not get the bonus, but therefore they will not lose anything. The bonus will be different and will depend on the distance



What time does the One o'clock Gun fire on a Sunday?

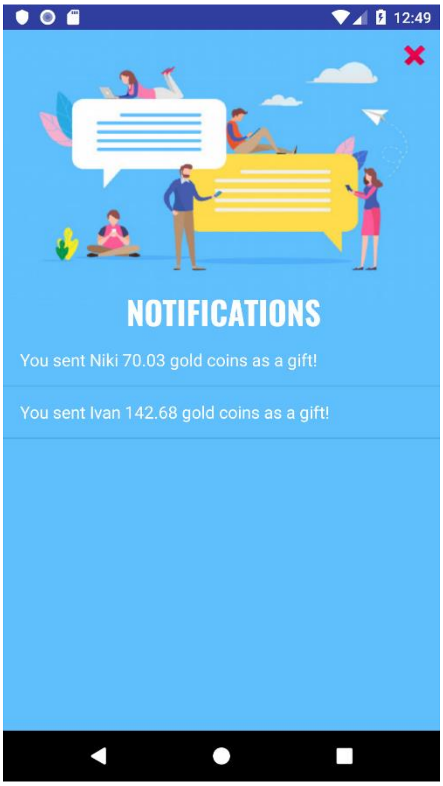
12:59pm

At 1pm.

It doesn't.

Correct answer! Based on your travelled distance and the value of the coin your get 18.73 bonus gold coins!

Figure 21. The bonus from a correct answer



NOTIFICATIONS

You sent Niki 70.03 gold coins as a gift!

You sent Ivan 142.68 gold coins as a gift!

Figure 22. The notification for the sender

walked by the user while playing the game. The bigger the distance, the bigger the bonus. That is why it is in the user's interest to collect the coins they think value the most at the end and, of course, answer the question correctly. The distance of the user is updated every time when there is a location change. The game gets the current number of steps the user had walked, add one and save the amount to the database. The bonus feature does not need the exact number of steps, but just an approximate one. The formula for the bonus is made by taking into account the typical distance walked by a player, but it is in favour of the user if they answer correctly. That means that the user can get a big amount of GOLD by just answering one question. The amount of question so far is just 15, but more could be added. If the user does not want to answer the question, they can simply click on the red cross in the upper right corner and the question will close.

The fifth item in the list is the notification activity. The notification activity is a ListView containing an ArrayList of the different notification the user gets. The notification is of type string and contains the username of the sender and the amount of coins received. The notification can also be a history of to whom the user had sent money and the amount of money transferred (Figure 22). When the player gets money from a friend a small red dot on the envelope icon will notify the player that they received money (Figures 23 and 24).

The last item on the menu is the LogOut button. When clicked the user will be automatically logged out from their profile and the current map and wallet file will be saved to the database so next time the user logs in they will receive the up-to-date information.

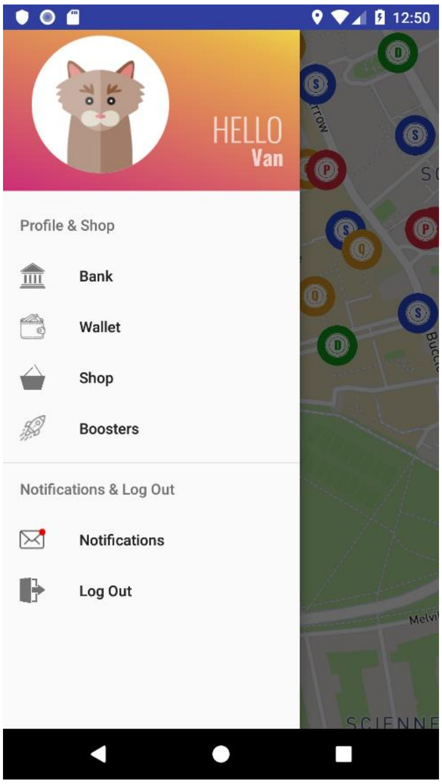
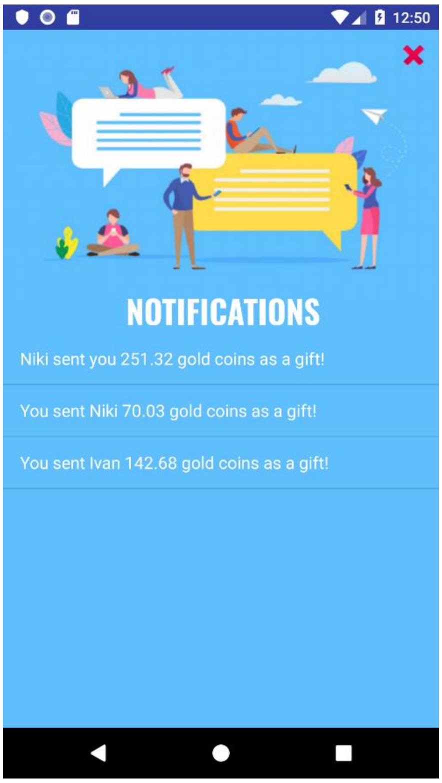


Figure 23. Alert for a new notification



NOTIFICATIONS

Niki sent you 251.32 gold coins as a gift!

You sent Niki 70.03 gold coins as a gift!

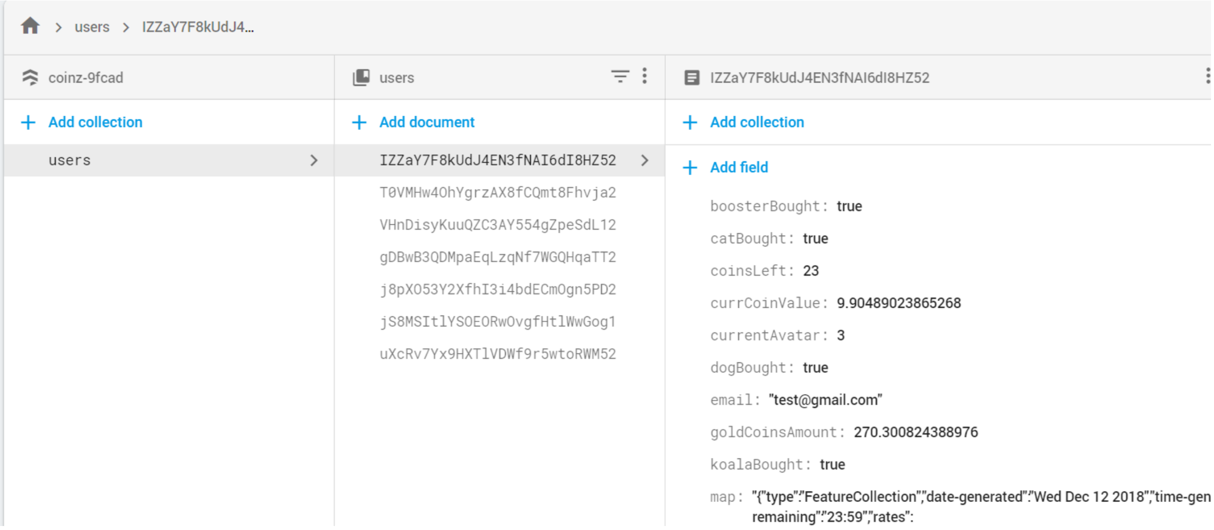
You sent Ivan 142.68 gold coins as a gift!

Figure 24. The notification for the receiver

3. Change of features

I managed to implement the bonus features and all parts of the design that I stated in the plan for the given time. However, during the implementation I decided to make some changes.

In the plan I said that the coins transferred to the bank account no longer know the previous currency they were. If the player receives as spare change a coin that they already had, it is not a problem to bank it twice as the important aspect is not the currency or the previous value of the coin but its value after the conversion. In the actual implementation the coins received from another player do not go in the wallet of the receiver. When the sender sends the coins they are automatically transferred to GOLD coins given the current rates of the currencies and they go to the receiver’s bank account. I decided to not put them in the receiver’s wallet, because there might be some misunderstanding. A user can bank



IZZaY7F8kUdJ4EN3fNAI6dI8HZ52
T0VMHw4OhYgrzAX8fCQmt8Fhvja2
VHnDisyKuuQZC3AY554gZpeSdL12
gDBwB3QDMpaEqLzqNf7WGQHqaTT2
j8pX053Y2XfhI3i4bdECmOgn5PD2
jS8MSItlYSOEORwOvgfHtlWwGog1
uXcRv7Yx9HXTlVDWf9r5wtoRWM52

boosterBought: true
catBought: true
coinsLeft: 23
currCoinValue: 9.90489023865268
currentAvatar: 3
dogBought: true
email: "test@gmail.com"
goldCoinsAmount: 270.300824388976
koalaBought: true
map: "{"type":"FeatureCollection","date-generated":"Wed Dec 12 2018","time-gen
remaining":"23:59","rates":

Figure 25. A user’s data account in Firestore

only 25 coins per day, but these 25 coins are only the coins that the user themselves collected. The spare change received from another player is not included in these 25 coins and thus, the user can bank all the spare change independently of that limit. The player can be confused which coin is collected and which coin is received. A possible solution to that is having two ArrayLists – one for the collected coins and one for received coins and put the received coins in the GridView before the collected coins and putting a distinctive mark so that they can differentiate them. However, if the size of the received list is long, the user has to scroll a lot in order to get to the collected coins. That is why directly putting the received coins in the bank is the better decision.

The second change is about to the bonus feature related to the value of the coins. In the plain I stated that the user will have a chance to increase the value of its coins that they collected. However, in the implementation I decided to not increase the value of the coins, but to put the bonus directly into the bank account. The reason why I decided to change the feature is simple. If the user answer correctly the question the value of the coin will increase and the coin will be stored into the wallet. The user will get that bonus once they transfer the coin into the bank. However, they may decide to not transfer the coin, but to send it to a friend. That way the user will lose the bonus that they got and the receiver will get it even though they did not even answer a question for the coin. Thus, I decided to change this unfair for the sender implementation for one that is in favour.

4. Change to week-by-week timetable

I tried to stick to the plan as much as possible, but as any plan there were changes due to different external factors or decisions. The first couple of weeks I did everything according to the plan. However, during the fourth week I had more free time and I spent more time on the app. I managed to think about the material design of the app, the user interface, completed the login and sign up pages and create the map in just one week. I did not need one week to "connect the app to the database" as I stated in the timetable – I needed just one day. And according to the plan I had to connect the app to the database before including the map and the location service, but in practice the reversed the order of the operations as I needed the main feature of the app to work before storing users' data. The other major difference is that I did not foresee the fact that I will need a lot of time to complete my other courseworks and maybe not have time for this project. I wrote that I will study for the exams on the 12th week, but I should have included more time for my other tasks during the semester. This is what led to postponing the plan with one week. However, I managed to do the task within the given time frame using major milestones almost every week.

5. Acknowledgment

Android Studio [online] (2018) Android Studio Documentation. Available from: https://developer.android.com/docs/. Used for finding valid solution for my questions throughout the project.

Dr. Droid [online] (2018) Android ListView Checkbox. Available from: http://www.androhub.com/android-listview-checkbox/?fbclid=IwAR2MkCa3PdnaAiHd8eDSvCfmKxOvYcKIjcmfvy8iLV4Ko63WGMhfvla0WpE. Used for the grid view of the wallet.

Firebase [online] (2018) Firebase Documentation. Available from: https://firebase.google.com/docs/. Used for finding valid solution for my questions throughout the project.

Flaticon [online] (2018). Icons. Available from: https://www.flaticon.com/. Used for the icons in the app.

Freepik [online] (2018). App icon set. Available from: https://www.freepik.com/ Used for the design of the coins.

Freepik [online] (2018). Bank background design. Available from: https://www.freepik.com/. Used for background of the bank activity.

Freepik [online] (2018). Dollar coins. Available from: https://www.freepik.com/. Used for the coins' design of the wallet activity.

Freepik [online] (2018). Financial concept with the city. Available from: https://www.freepik.com/. Used for the wallet activity banner.

Freepik [online] (2018). Flat animals. Available from: https://www.freepik.com/. Used for the avatars.

Freepik [online] (2018). Flat boxing day sale banner. Available from: https://www.freepik.com/. Used for the send coins to a friend dialog design.

Freepik [online] (2018). Illustration of exit door. Available from: https://www.freepik.com/. Used for the log out icon.

Freepik [online] (2018). Online banner horizontal. Available from: https://www.freepik.com/. Used for the avatar shop banner.

Freepik [online] (2018). Road location. Available from: https://www.freepik.com/. Used for the booster image.

Freepik [online] (2018). Space rocket with flat design. Available from: https://www.freepik.com/. Used for the booster activity banner.

Freepik [online] (2018). Web design button collection. Available from: https://www.freepik.com/. Used for the buttons in the avatar shop and booster activity.

MapBox [online] (2018) MapBox Documentation. Available from: https://www.mapbox.com/android-docs/maps/overview/. Used for finding valid solution for my questions throughout the project.

Ontsunan / Freepik (2018) Chatting in social media. Available from: https://www.freepik.com/premium-vector/chatting-in-social-media_3371705.htm. Used for the notification activity banner.

StackOverflow [online] (2018) Stack Overflow. Available from: https://stackoverflow.com/. Used for finding valid solution for my questions throughout the project.