CanineCentral

Mobile Application Development

Joe Shepherd

Student Number: 14809668

CI360

May 4 2018

Table of Contents

1. Introduction

The aim of this project is to create a mobile app that allows users to track their location when going for a walk with their dogs. Users will be able to record how much exercise each dog is getting and allow them to set goals to help keep their pets healthier. The motivation behind this app is due to the rise in fitness planning applications and awareness of pet welfare, mentioned in section 2. These kinds of applications are mounting in popularity as people are becoming increasingly more aware of the benefits that exercise provides, however there are not many aimed towards a person's pet. Furthermore, the main medical condition among dogs is obesity, which is due to owners not giving their pets enough exercise. This app will try to help owners prevent this as they will know exactly how much exercise their pet is getting, which they could then use this information to consult with their veterinarian to find out if what they are doing is enough.

2. Background research

Fitness applications

Health and fitness applications have risen in popularity by over 300% in the last three years. Users are constantly using these apps throughout the week to motivate them to stay healthy and to record how much exercise they have been getting. This has given developers a striking opportunity to create applications that capitalize on the frequent usage of these apps (Kesiraju and Vogels, 2017). "Fitness trackers are big business, with millions of us now recording how many steps we take each day and poring over the data on our smartphones to keep fit and healthy." (Sparkes, 2015). As people are increasingly more interested is using tech to track their fitness, the same outcome is expected to happen within the pet technology market. Owners are also becoming more aware about their pets' health causing a rise in GPS tracking applications to record how much exercise their pets are getting (Pet Wearable, 2018).

Pet welfare

According to Wardrop (2011), almost half of pets are receiving inadequate levels of care, due to their owners providing insufficient amount of exercise, the wrong diet, or leaving their pets for prolonged amounts of time. The most common medical condition in dogs is obesity, which is associated with many other medical conditions that result in a dog having a poorer quality of life, and dying prematurely (Association for Pet Obesity Prevention, 2016). Due to busy lifestyles, some owners can find it hard to adjust to new routines and habits or remember how much exercise they are giving their dog, resulting in around 93,000 dogs not being walked at all (Paw Report, 2017). The target users (dog owners) need to have more knowledge and understanding of how to look after their dogs with an easy, organised way to keep track of how much exercise their dog is getting to help motivate them to walk their dogs.

Competitor Analysis

Name:	Dog Walk – Track your dogs	Dog Walk Tracker & Reminder	Rundogo – track dogs workouts
Rating (no. of reviews):	4.2 (1000)	3.6 (20)	4.9 (10)
Downloads:	50,000+	1000	1000
Functionality	• track location while walking dog • add different pets • mark on map where pet wees/poos • view walk history details and route on map • take photos on walk • share route with friends • access blog	• set reminders to walk dog • record how many walks a dog has been on and what time the walk occurred • show graph of walk history	• track location while walking dog • add multiple dogs • select activity (not just for walking) • share walks with friends • view walk history details and route on map
First release	2017	2013	2017
Strengths:	• appealing UI • easy to understand how to navigate UI • easy to add pet with dropdown options • some useful functionality like	• can add walks easily with one button press • chart of walk history allows owners to see	• appealing UI • quick navigation and easy to understand UI • can also track human activity not just pet

	adding photos when going on walk • provides a blog for users to view helpful information about pet welfare	changes in walk habits • can set reminders to motivate owners to go on walks	• walk history can be grouped into categories showing different activities, weekly or monthly etc. • easy to add pet
Weaknesses:	• some aspects of UI hard to use, e.g. cancelling a walk requires multiple button presses to unlock UI beforehand. • lengthy process to navigate UI. Multiple button presses or swiping required to view different pets walk history. • Parts of app very slow e.g. accessing blog • some functionality seems irrelevant: marking on map where pet wees/poos? • No ability to set goals and reminders	• very poorly designed UI • first time users will find it confusing how to operate. • only add one pet • no ability to track location, distance, or length of time walk occurred. • no ability to add goals	• amount of options for a walk seems overwhelming • no ability to set reminders or goals • complicated to save walk – many information needs to be recorded

To conduct competitor analysis for the dog walking application, keywords such as ‘dog walk’ and ‘dog tracker’ were used to search google play store. From this I found that there are very few apps on tracking dog walks for pet owners. Most of the apps provide a platform for owners to hire dog walkers and track where the dog walker goes, however hardly any aimed at pet owners themselves.

As seen from the table above, Dog Walk and Rundogo will be very similar to CanineCentral in terms of functionality, and although they are (mostly) well designed and have the ability to map walks, does not include functionality for dog owners to set reminders and goals. Dog Walk Tracker & Reminder doesn’t allow for tracking of walks but provides the functionality the other two apps lack by being able to set reminders. It does however lack in nearly all other areas as it is very poorly designed, and only allows for one dog to be added, making the user experience tremendously unappealing. I also found that the first two apps user experience could be improved by making access to certain functionality less complicated and quicker to find, as it took a while to navigate to certain areas of the app.

This helps inform requirements as CanineCentral will combine together what each app lacks or does well. By being able to add goals and reminders will set CanineCentral apart from the other apps, as it will help motivate users to walk their dogs, make them feel rewarded every time a goal is achieved, and also keep them more engaged into the application, which these apps do not do very well. Furthermore, Dog Walk – Track your dogs is the least complicated and has the best UI out of the three, and also has the most downloads, so I think the UI will play a major role in attracting users to download the application. CanineCentral will be much

simpler to use with fewer button presses and display less information about each walk. This will improve the user experience of this app compared to competitors as these apps provide too many details that may cause confusion, making the user experience less worthwhile.

3.Specification and Planning

Deliverables

Changes have been made from the original project plan after conducting more research, as I now have more technical understanding of how to implement the application, and the previous project plan did not reflect the correct requirements of the process. The Gantt chart is available in Appendix A. Deliverables that have higher priority are marked red in the diagram to show which parts of the project require more attention to ensure application development is of a good quality and implements the correct functionality.

The first set of tasks was to complete the background and competitor research to gain a better understanding of the problem area. This will influence the requirements phase as it will provide information that will help give the app better functionality and to ensure the requirements cover all the needs of the application. Competitor research is higher priority as it will ensure the application has a completive advantage over the other apps and that it improves what these apps lack. Once the requirements are created using this research, UML diagrams will be created to demonstrate how the application will operate and to give a clear understanding of how the implementation should be carried out.

The next stage will be the design of the application. Research on best design practises of android apps will be conducted to ensure the app provides a good user experience. This research will be used to develop wireframes and then feedback from a small sample will be collected. The wireframes will be improved based on the feedback before the interface is created in android studio. Following this, an ERD diagram will be created to show how the database will collect and store the information the application needs. This is high priority as without a good understanding of how to save the data, it may cause many errors when writing the code in android studio. The java classes that create the database can then be implemented.

The final stages will be the development and testing of the android app. This process will take the longest time to complete, however if the design materials are of a high standard, should be straightforward.

Risk Analysis

Risk	Probability (1-5)	Impact (1-5)	Mitigation	Contingency
Incorrect Estimate times for deliverables	3	3 – May result in deliverables being of low standard or incomplete final product.	Ensure enough time has been allocated to each task.	Adapt project plan
Other projects may cause delays on tasks being created	4	3 – Deliverables being of low quality or incomplete	Again, ensure enough time has been allocated for each task and keep on top of other university projects	Adapt project plan
Requirements analysis incorrect	3	3 –functionality of final product not being of good quality	Get a good grasp of what functionality needs to be implemented beforehand	change requirements if needed
Research of problem area could be low quality	2	4 – result in weak understanding of android app and what functionality it should implement	Carry out enough research to ensure requirements are good quality	Do more research throughout development process.
Data loss	1	5	Keep backups of project	Recover from backups
Injury/Illness	3	3- deliverables being behind schedule	Ensure enough time allocated to task and allow extra days throughout project to account for loss of days.	Adapt project plan
Android app may not run on all devices	2	4 – broken application	Test on multiple devices and ensure the APIs can be run on target phones.	
Bad UI design	3	4 – application will be of low quality and not provide a good user experience	Ensure enough research on best design practises have been carried out.	Change UI design there is enough time.
Poor testing strategy	3	4 – application may crash unexpectedly if not enough tests carried out	Create testing strategy beforehand and ensure enough scenarios are tested.	Create more tests if there is enough time

Methodology

To carry out this project, an agile approach will be adopted. This was chosen rather than the waterfall model, as it allows room for adaptation, is value driven and helps to make the process less complex. The waterfall model is solely based on completing tasks in a linear manner. Returning to previous tasks is not allowed unless a full development cycle has been conducted, therefore if one part of the project is incorrect at the start, many issues will arise when coming to the development of the application. By adopting an agile approach, changes can be made throughout each stage of development if any issues occur. Deliverables will be produced in time boxes of short two-week intervals to separate the project into smaller more manageable chunks. If a deliverable is not of good quality the timebox can be repeated, or previous sections can be revisited to improve them.

Measure of success

To measure success throughout this project three things will be assessed. Whether or not the deliverables are finished on time, if there are of good quality and if the final application implements the requirements that were gathered throughout this process.

4. Development

Requirements gathering

To gather the requirements of the system, the background research and competitor analysis was used to find out what this kind of application should be able to do. Requirements were changed throughout each stage in the process to adapt to the changes made in other sections. For example, the feedback gathered in the design phase required some changes to the user interface which meant some requirements needed to be edited. Additionally, as more technical understanding of how to implement the application was gained throughout the development, some requirements needed to be adapted due to some incorrect assumptions being made.

Functional requirements:

1. A user should be able to add their pet information into the application
2. A user should be able to retrieve their pet information and be able to edit it.
3. A user should be able to track and map their location when going for a walk
    a. Upon pressing the start button, a dialog should appear to allow users to select which dog is going on the walk
    b. the activity should show a google map with a line being drawn on the map as a user move
    c. the activity should display the amount of time a walk has been going on for as well as the distance travelled on the walk.
    d. When a user presses the stop button, the walk details should be saved to the application
4. A user should be able to view all the previous walks a dog has been on
    a. the activity should allow a user to select which dog they want to view previous walk history on.
    b. the activity should display a list of all the walks for a specific dog
    c. A user should be able to open up a google map showing the route they walked upon pressing a certain walk
    d. A user should be able to view the distance and length of time for each walk, and the date the walk happened.
5. A user should be able to set daily and weekly goals for each dog
    a. the goals should be viewable on the walk history activity alongside the walk information
    b. a notification should appear when a goal has been met
    c. a notification should appear if a goal is close to its ending time but is not nearly completed.
6. A user should be able to set reminders of times that they should go on a walk to keep them motivated.
    a. a notification should appear alerting the user that it is time for them to go on a walk.

Non-Functional requirements:

1. The application should run on SDK versions 27 to 19
2. The application should carry on tracking a user's location if the app has been exited
3. The application should adapt to different screen sizes and phones.
4. The database should be accessible to insert and retrieve data for as long as the application is running
5. Application should be very simple to use with no difficulty when learning how to use it.
6. Application should respond to all requests within 0.5 seconds, for example when opening the google map to view the previous walk data.

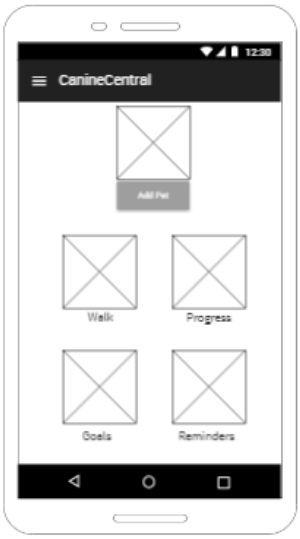
Design

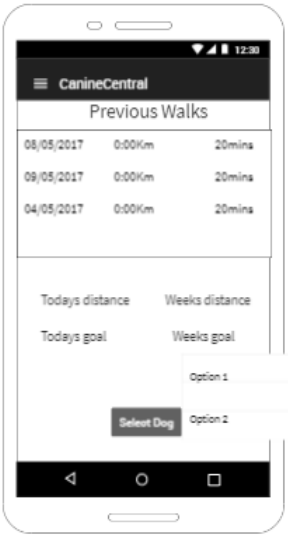
Research

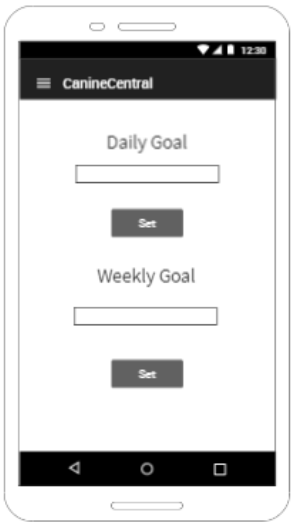
Before creating the design of the application, some background research was conducted to find out what the best design principles are to make a successful android app. The design and user experience of an app is what separates the good from the bad. It is vital to create a good first impression to users as the design is the first thing they see and will ultimately decide the success of the app. Some key points of UI design taken from Babich (2017) and Kuafman (2018) are:

- Clear and seamless user flow. Focus on speed – allow users to quickly complete the task required without obstacles.
- Focus on user goals – Break large tasks into smaller ones, for example if tracking location let the app do most of the work automatically so the user doesn't have to enter much information.
- Avoid displaying too much information as it prevents users making decisions as they may feel too overwhelmed.
- Cut out the clutter. Do more with less – too many buttons make an app confusing the understand and use, so keep the design simple.
- Make the application easy to navigate and keep users coming back.
- App should be easy and intuitive to figure out on first run.
- Only use notifications and alert dialogs to display important, necessary information otherwise users will be desensitized by them and go unnoticed.

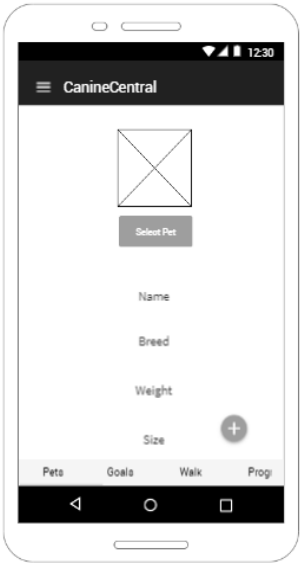
This research was used alongside the competitor research to come up with a simpler and better design to provide a better user experience. The wireframes are available below.







After conducting some user research on a very small sample on friends and people around university to get feedback on the designs, another iteration of wireframe design was conducted to improve them. From the user feedback, many people said it would be quite lengthy to navigate to the different pages using the home screen (1st wireframe), as users would need to keep backing up and selecting which page to go on. To solve this a navigation bar was added and the original home screen was removed so that users could navigate to anywhere in the application from the same page. New designs are shown below. This process helped me to learn more about mobile app design and user experience to create a clean, simple UI for CanineCentral, which I think I have done well.



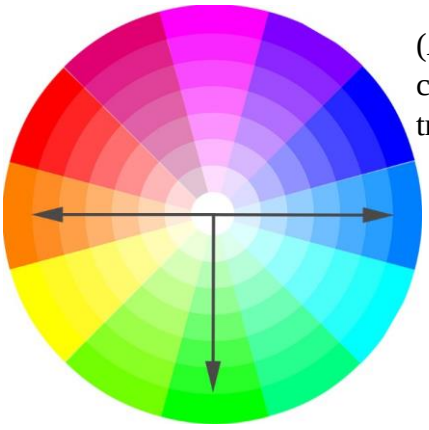
This diagram shows the new home screen with the navigation bar added. Some more research was done on material design as this was the theme chosen to design the app. The use of bottom navigation should be used with 3-5 top level destinations requiring direct access, which fits perfectly for this application as there are four separate screens (Material Design, 2018).

The add pet screen was also changed so users could select which pet they want to access information on, and so that adding a new pet is much simpler. This design improves on some of the points raised from the research, by making it easier to navigate the app, removes clutter and speeds up the process of executing a task. The colours were chosen when the app was being developed so that I could see the working product on an actual phone to see if the colours looked good.

Fragments will be used with one main activity rather than having separate activities, as it provides an easy way to implement this design and allows for flexible, re-useable code. (Android Developers, 2018b).

Figure 4.1 (Medium, 2018)

(Arrows on wheel do not reflect colours chosen, demonstrates triadic scheme)



To select the colours for the application, some research on colour theory was conducted to find what colours go well with each other. A triadic colour scheme shown above was chosen which is a combination of three colours evenly situated apart on the colour wheel as these complement and contrast each other well (Medium, 2018). Final designs can be seen in section 6.

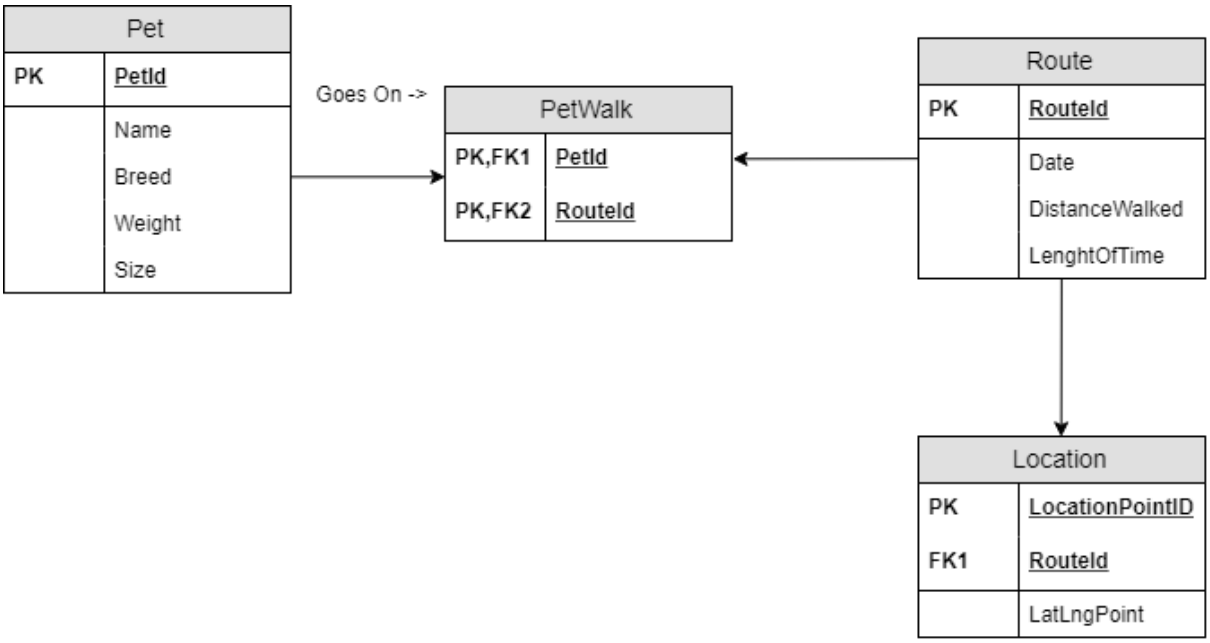
Database

To save data on the application, the Room persistence library was used. This was chosen rather than the original SQLite, as it provides an abstraction layer over the database to allow easy access without all the boilerplate code the SQLite library requires to create a database and SQL statements can be checked for errors before running the application, which is not available from SQLite (Android Developers, 2018e). The ERD diagram in figure 4.2 demonstrates the structure. This made it much easier to implement as the diagram gave me a clear understanding of the tables I needed to create along with the relationships between the tables. Each table has its own class that holds the attributes inside and declares the keys and relationships. A data-access-object(DOA) interface was created for each class that implements the SQL queries to insert and retrieve the data. One database class was created that is used to access data throughout the application, as the class describes which classes are used for entities and provides access to the DOA classes.

Technical Challenges: During development, I was unsure of how to implement multiple tables using the Room API as well as the join table to link pets to a route, as the android developer guides do not go into too much detail on this subject. Furthermore, as it is fairly new, not much other documentation and tutorials provide information on this matter. Szklarska (2018) described how to do this using the API which taught me how this problem could be solved. Queries that were implemented in the route class were transferred into the PetWalk class to retrieve information about which route each pet went on, using a join statement of the pet table and route table, rather than returning all the routes at once as was originally implemented.

Another technical challenge in this part of development was figuring out how to access the data across multiple activities and also running the database operations in a background thread. The application uses a public database object in the main activity that the other fragments call if needing to access the database, and the database runs on the main thread. Although this is bad practise, I did not have time to solve this issue. If there was, the live data architecture with a view model would have been implemented that would observe changes made in the database and display them onto the activity (Android Developers, 2018c). Furthermore, an executor object would have been implemented to manage the database tasks on the background thread, which would improve the apps performance (Android Developers, 2018a).

Figure 4.2



Location Tracking

Inside the walk fragment, a google map is shown on the screen. A button starts the location tracking that draws a polyline onto the map and measures the time and distance of the walk. To implement this the FusedLocationProviderClient was used. Permission requests popup to ensure the device has location settings turned on and that the user has given permission to access location. A location request object is passed to the FusedLocationProviderClient to start the tracking and the callback method onLocationChanged handles the functionality to draw on the map and calculate distance. A chronometer object measures the amount of time and this information is saved to the database on stopping.

Technical Challenges: This whole section proved difficult to implement. Android Developer (2018e) provides a tutorial that was used to figure out this challenge, however again was very confusing as not much detail described how to do this properly. After reading many posts of Stack Overflow this functionality was finally implemented. However, it took a long time as there are many different implementations to track locations and it was hard to find code for this specific method of location request and FusedLocationProviderClient. Nugent (2018) provided a very simple implementation to draw the lines onto the map as a user moves, which taught me how to create this feature using an Array List of latlngs that was added every time onLocationChanged was called, then the polyline options object is used to connect each latlng point together on the map.

Previous Walk Fragment

To show users all the previous walks a pet has been on, a recycler view was used. An option allows a user to select a pet and then the recycler view displays this information. The adapter retrieves the information from the database and inserts the data into a separate view that is used as the for each route. Android Tutorials Hub (2018) was used to implement this function as it provided a clear tutorial on how this can be implemented. This section of development was fairly straightforward as did not pose any technical challenges.

When the list has been populated, each item has an onClickMethod associated with it. Upon pressing, a map activity pops up on a small area of the screen that shows the route the user travelled on each walk. The same methods were used from the walk fragment to draw the line onto the map, however the data was retrieved from the database instead. This was also fairly straightforward as all the technical challenges were resolved when implementing the walk fragment.

Goal Fragment

To add daily and weekly goals, shared preferences was used to store this information. The app can only store goals in general not for each specific pet, therefore as it was such a small amount of data, creating another table in the database would have been a waste of internal memory. The shared preferences allowed to easily save this data and makes it available across all activities. This section could have been improved which is discussed in section 6.

Layouts

All of the layouts created in this application use the constraint layout. This was chosen as it allows for large complex layouts to be created that can easily adapted to different screen sizes. This layout also provides better performance as the hierarchy of view elements stays completely flat, so the system doesn't need to calculate take as long traversing the view tree to determine how large each view group is (Hagikura, 2018).

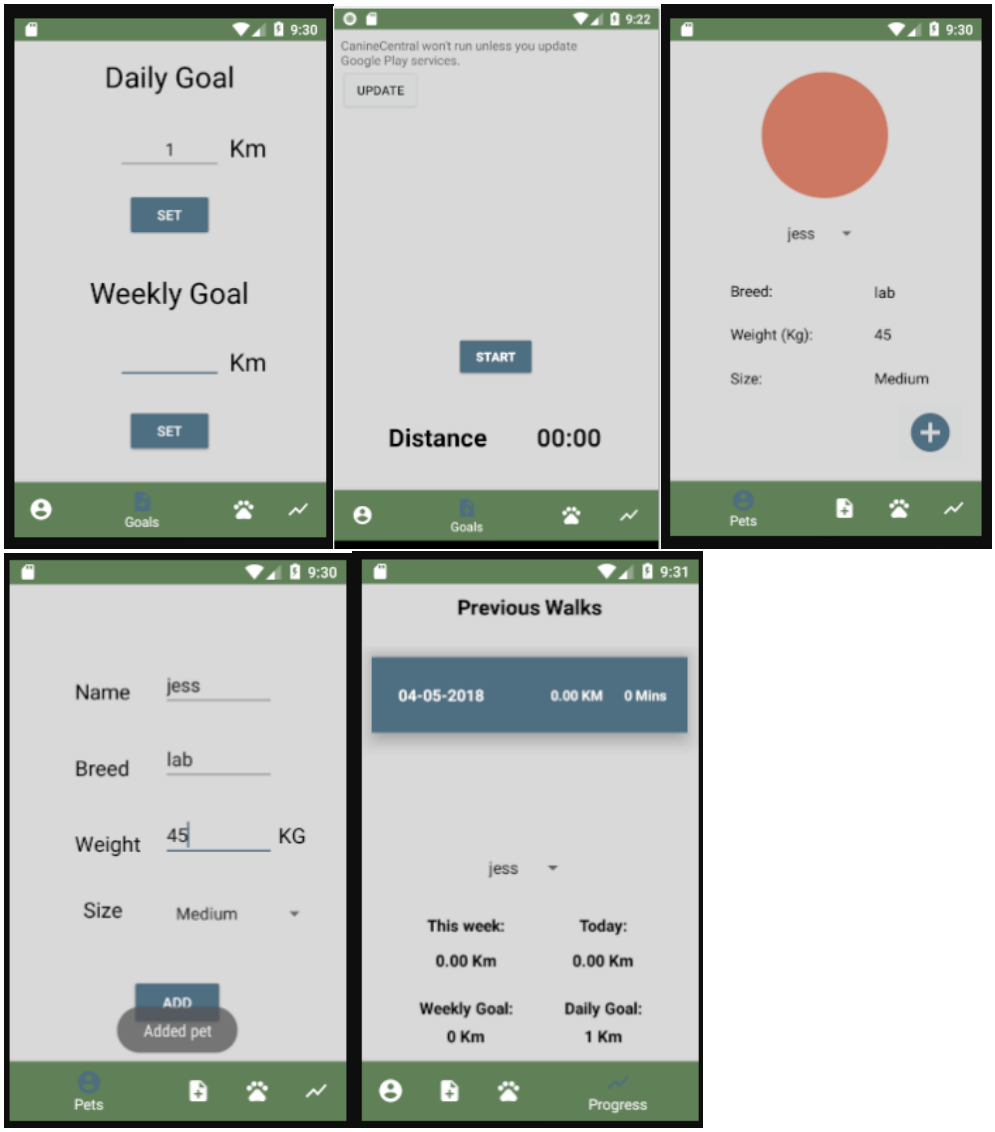
Technical challenges: The main challenge for this was understanding how the constraints for each view item works when a screen size is changed. Many of the layouts needed to be edited as some did not look the same when changes to the screen size was made. Even now, the layouts do not look good on very large phones such as the google pixel 2. If there was more time this could have been improved by creating separate layout files for when the screen got to a certain size.

5.Testing

As I run out of time during the development phase, only one instrumentation unit test was created to test that the database implementation worked. The test inserts data into each table of the database to check that the data returned when calling it is the same. I chose to do this test instead of testing other functionality as the data storage is one of the fundamental parts of this application. If the database did not work correctly most of the apps core functions would not work correctly, making it obsolete. If there was more time other apps logic would have been tested using local unit testing, as well as espresso testing to test that the UI works correctly. However, to do this manual testing was carried out. A physical android device was used for this process and parts of the application was tested every time new functionality was added. This ensured that there were no major issues with the application so that it doesn't crash when operating. I also used the android profiler in android studio to see how much memory the app was using to discover if there were any memory leaks. As bad practise was used to implement the database, I believe that there are some memory leaks in the application.

Technical challenges: When the google map was first created, I used a fragment to insert it onto the display. When the app was changing to another fragment, the google map memory was not being collected by the garbage collector, so every time the map fragment was opened again the apps memory usage kept on increasing. To overcome this issue the map view was used instead, as this provides simple methods that can be called such as MapView.onDestory() every time the fragments was destroyed. This fixed this major memory leak issue and increased the apps performance greatly.

6.Final Designs



7.Review and Reflection

Overall Project Success

The project was not completed successfully due to not enough time. The app does not fully implement all the requirements that were created throughout the process. For example, the reminders function was not implemented, as well as notifications that show users when a goal has been completed. However, I think the other parts of the project were completed to a good standard, and enough research was conducted to ensure the design and implementation went smoothly. The items created allowed me to have a good understanding of the tasks that needed to be done, and I am proud of the application overall, especially the mapping function that allows users to view the route they walked. This was the main technical achievement of this application. I found it hard to manage this project along side my other modules, which caused some delays in the project deliverables being produced and an incomplete project created.

Application

Strengths: I think that CanineCentral has a simple and well-designed UI, which makes the app very easy to use and quick to navigate. The competitor and design research ensured this is one of the applications strengths, as it made the application stand out by implementing different functionality rather than just being a copy of a previous app. The colours work well together and make the app look pleasing. It is very easy to select which dogs are going for a walk and requires minimal effort to find out all the previous walks the dog has been on compared to other apps. Furthermore, goals help to motivate users as it gives them something to accomplish and make them feel rewarded once complete.

Weaknesses: Without the app being fully completed, the app does not provide users with full functionality that would help to keep users engaged with the application, therefore there isn't much incentive to keep using this app other than to check their progress on reaching their goals. Additionally, goals cannot be set for different pets, which is a weakness as it doesn't allow for pets to be singled out. One pet may need a different amount of exercise due to breed, age etc, which users cannot set so only a basic goal can be set.

Further Development

If I had more time to carry on the development of the application, these are some of the features that could be added:

- Allow users to set goals for separate dogs.
- Provide information that allows users to learn about how to care for their pet properly. This was part of the functionality in the original project plan, however the information would come from different sources, which I did not have permission to use.
- Allow users to add photos whilst going on walks, which they could use this app as like a separate gallery application just for their pets.

- Give users the opportunity to plan a route and share it in real time, where they can connect with their friends and other people in the community to plan dog walking events, or to just go on a walk with their friends.

Learning Points

Throughout this project, I have learnt many aspects in the mobile application development and I am confident that I could produce my own professional applications. This process has taught me how to go about when conducting competitor research to find the strengths and weaknesses of other apps, which can be very helpful when planning an application. Furthermore, it has helped me improve my programming skills and learn about the best design and implementation processes when developing an android app, even if not all of these principles were used in the practical implementation. Lastly, I have learnt a great deal of information on user design exercise which can be used in many different software development projects. The research conducted helped me understand how to make an appealing app, as well as how to design an appealing looking application.

References

Android Developers. (2018a). *Create a manager for multiple threads | Android Developers*. [online] Available at: https://developer.android.com/training/multiple-threads/create-threadpool [Accessed 3 Apr. 2018].

Android Developers. (2018b). *Fragments | Android Developers*. [online] Available at: https://developer.android.com/guide/components/fragments [Accessed 3 Apr. 2018].

Android Developers. (2018c). *LiveData | Android Developers*. [online] Available at: https://developer.android.com/reference/android/arch/lifecycle/LiveData [Accessed 1 Apr. 2018].

Android Developers. (2018d). *Location and context overview | Android Developers*. [online] Available at: https://developer.android.com/training/location/ [Accessed 2 Apr. 2018].

Android Developers. (2018e). *Save data in a local database using Room | Android Developers*. [online] Available at: https://developer.android.com/training/data-storage/room/ [Accessed 3 Mar. 2018].

Android Tutorials Hub. (2018). *Android RecyclerView Tutorial - Android Tutorials Hub*. [online] Available at: http://www.androidtutorialshub.com/android-recyclerview-tutorial/ [Accessed 4 May 2018].

PAW Report (2017). [online] Available at: https://www.pdsa.org.uk/media/3288/mini-paw-report-2017-online-version-1.pdf [Accessed 21 Apr. 2018].

Babich, N. (2017). *Mobile Design Best Practices | Adobe Blog*. [online] Adobe Blog. Available at: https://theblog.adobe.com/mobile-design-best-practices/ [Accessed 2 Feb. 2018].

Balaji, S. and Murugaiyan, M.S., 2012. Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management*, *2*(1), pp.26-30.

Pet Wearable (2018) *Pet Wearable Market Size & Share | Global Industry Report, 2018-2025*. [online] Available at: https://www.grandviewresearch.com/industry-analysis/pet-wearable-market [Accessed 18 Apr. 2018].

Hagikura, T. (2018). *Understanding the performance benefits of ConstraintLayout*. [online] Android Developers Blog. Available at: https://android-developers.googleblog.com/2017/08/understanding-performance-benefits-of.html [Accessed 4 May 2018].

Kaufman, J. (2018). Principles of Mobile App Design. *Apteligent*.

Kesiraju, L. and Vogels, T. (2017). *Health and fitness app usage "grew 330% in just 3 years" | Netimperative - latest digital marketing news*. [online] Netimperative - latest digital

marketing news. Available at: http://www.netimperative.com/2017/09/health-fitness-app-usage-grew-330-just-3-years/ [Accessed 3 May 2018].

Material Design. (2018). *Bottom navigation - Components - Material Design*. [online] Available at: https://material.io/guidelines/components/bottom-navigation.html#bottom-navigation-usage [Accessed 19 Feb. 2018].

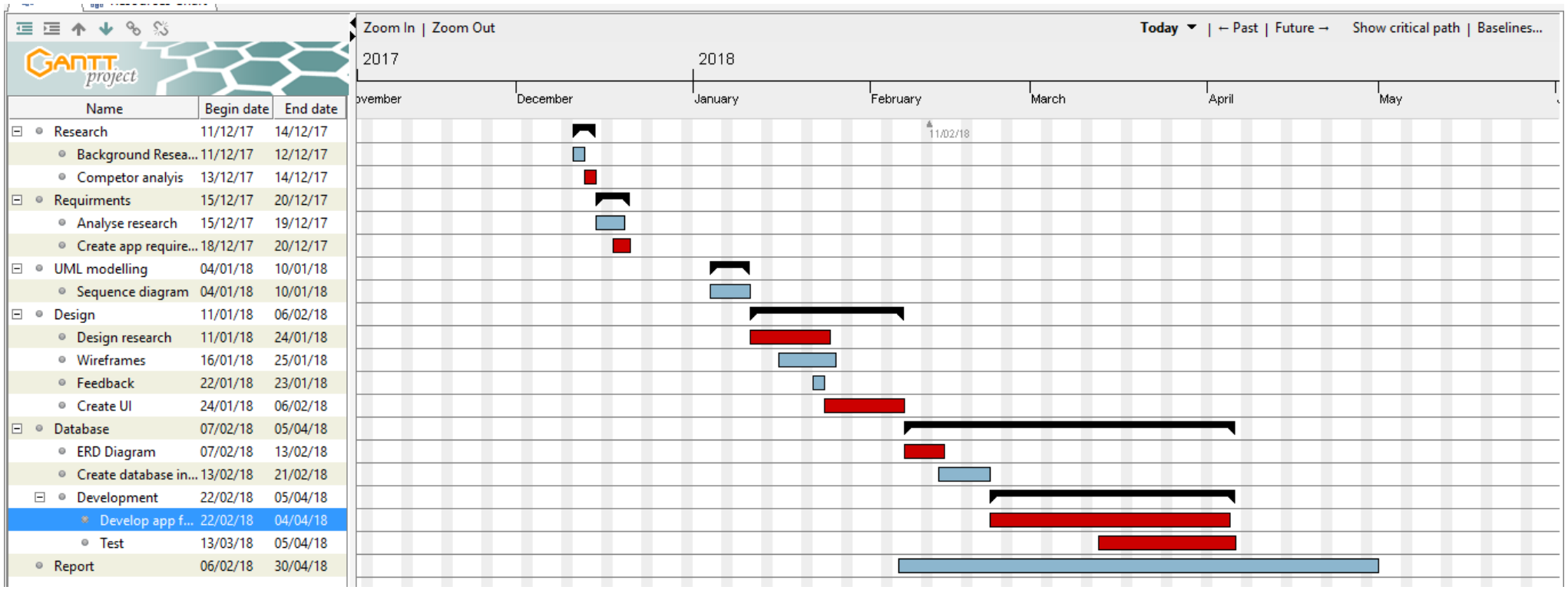
Medium. (2018). *Mobile App Design: 14 Trendy Color Schemes – Adoriasoft_Com – Medium*. [online] Available at: https://medium.com/@Adoriasoft_Com/mobile-app-design-14-trendy-color-schemes-2669b5bb77d3 [Accessed 6 Apr. 2018].

Nugent, D. (2018). *How to draw path as I move starting from my current location using Google Maps*. [online] Stack Overflow. Available at: https://stackoverflow.com/questions/30249920/how-to-draw-path-as-i-move-starting-from-my-current-location-using-google-maps [Accessed 10 Apr. 2018].

Sparkes, M. (2015). *Is your dog getting enough exercise?*. [online] Telegraph.co.uk. Available at: https://www.telegraph.co.uk/technology/news/11411467/Is-your-dog-getting-enough-exercise.html [Accessed 23 Apr. 2018].

Szklarska, P. (2018). *Android Architecture Components: Room—Relationships*. [online] AndroidPub. Available at: https://android.jlelse.eu/android-architecture-components-room-relationships-bf473510c14a [Accessed 1 Apr. 2018].

Appendix A


Zoom In | Zoom Out
Today | ← Past | Future →
Show critical path | Baselines...
2017
2018
ovember
December
January
February
March
April
May
11/02/18
Name
Begin date
End date
Research 11/12/17 14/12/17
Background Resea... 11/12/17 12/12/17
Competor analyis 13/12/17 14/12/17
Requirments 15/12/17 20/12/17
Analyse research 15/12/17 19/12/17
Create app require... 18/12/17 20/12/17
UML modelling 04/01/18 10/01/18
Sequence diagram 04/01/18 10/01/18
Design 11/01/18 06/02/18
Design research 11/01/18 24/01/18
Wireframes 16/01/18 25/01/18
Feedback 22/01/18 23/01/18
Create UI 24/01/18 06/02/18
Database 07/02/18 05/04/18
ERD Diagram 07/02/18 13/02/18
Create database in... 13/02/18 21/02/18
Development 22/02/18 05/04/18
Develop app f... 22/02/18 04/04/18
Test 13/03/18 05/04/18
Report 06/02/18 30/04/18