# IAB330: Assessment 1

# Scruff

| | |
|---|---|
| John Tang | N9713913 |
| David McClarty | N9939768 |
| Fredrik Forsell | N10297057 |
| Ivar Sandvik | N10297154 |

# Executive Summary

Scruff is a social media and marketplace hybrid, which facilitates the relocation, adoption, and selling of pet animals using social media features. Scruff aims to help seekers find their perfect pet by offering the ability for the end-user to frequently check on their favourite animals; like the workings of social media. The application aims to keep the two parties connected over long distances, so that the user will be able to discover the personalities further and narrow down their choices when selecting an animal to bring home.

## Features, Functionalities, and Requirements

The purpose of this app is to give users more insight into an animal's life and personality before bringing them into their life. This insight is achieved via a social media feed, where the owner(s) of the animal will post regular updates on their animal's. Traditionally, those wishing to adopt must visit a shelter to visit the animals. This interaction between each animal and the buyer often lasts a few minutes, which is an insufficient amount when compared to the commitment and compatibility requirements for long-term adoption.

Scruff functions by allowing users to search for animals to adopt. Scruff differentiates itself by offering the added ability for the user to “follow” animals, making their posts show up in your “feed”. As the owners of the animal post more, the user will understand the personalities, likes, and dislikes of animals; thus, narrowing down and improving the chances of a perfect match.

As for the marketplace aspect, Scruff will require comprehensive information from all the listed animals including verified medical history. This will allow the user to be certain that the animal they find on Scruff is up to date on its vaccinations. The individual animal pages will be laid out so that the seeker can seamlessly go from browsing an animal’s posts, to discovering more about it from its profile, to finally contacting the owner to arrange a deal. Scruff functions as a competitive marketplace, giving all users the ability to find the same animals, however the owner will still be able to refuse those deemed incompatible.

Scruff intends to compete against the current marketplaces such as Gumtree, and RSPCA. These offer social media capabilities, or a marketplace, but not both. Scruff intends to offer the entertainment of social media, with the added benefit of being able to adopt the animals. Competition analysis will be elaborated further.

## Identification and Fulfilment of Target User Demands

Scruff targets four primary demographics of users:

1. Owners of pets or strays that wish to give/sell the animal
2. Shelters, pet stores, and other organizations that also wish to relocate the animal
3. People who wish to buy/own pets or strays
4. People who enjoy animals and like following them on social media

Scruff aims to fulfil these three kinds of users’ needs in the respective following way:
1 & 2. The app provides a platform that allows owners of pets or shelters to properly showcase their animal’s skills and qualities, which will increase the likelihood of them being able to give the animal away.
3. The application provides a wide selection of pets that cover a wide range of breeds that will satisfy the user.
4. Many users will use this application to view cute animals, which is why the social media app functions can be used exclusively.

# Research and Review of Related Applications

## Problem Space

Scruff is an application to help find homes for shelter animals, store animals, and private owned animals seeking relocation. The traditional method of this process is to create an ad on popular websites such as gumtree, post flyers on telephone poles, or to send it to a shelter. This however does not provide potential adopters with a great understanding of the animal's lifestyle. Scruff's purpose is to give users more insight into each animal's lives, essentially providing a timeline of its activities. It does so by incorporating a social media system; which provides additional methods of pet discovery.

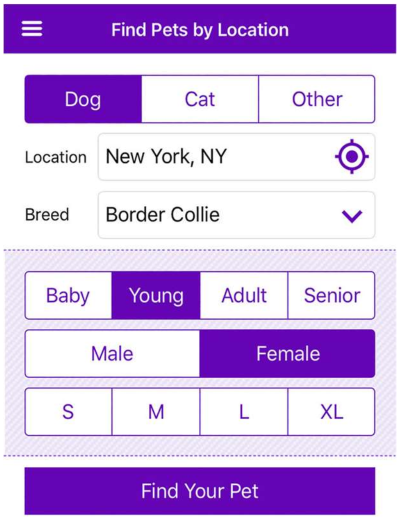
Rather than spending walking through the shelter, users of Scruff can choose to follow animals they find appealing, to better understand their personalities, likes, and dislikes -- which would otherwise not be possible with a 10-minute meeting in the shelter. Once followed, the user can then observe the animal through posts made by the owner/organisation.

Scruff differentiates itself from other online platforms by focusing solely on animals wishing to relocate. It functions like social media in the sense that users of the app can regularly check in on their "followed" animals. This continued activity helps potential adopters narrow down their perfect pet. As owners are trying to relocate the animal to another home, they are incentivised to post regularly, enticing users on the other end to contact them. Scruff isn't an alternative to going to shelters and stores, but rather a tool to interact with animals digitally.

As the market lacked any application that could connect animals and adopters long term, Scruff was established to bridge the gap. With Scruff, users can find their perfect pet by researching through many animals with easy, and animal owners are able to find the perfect owner and reach a bigger audience.

# Mobile Application Review

Petfinder Mobile App https://itunes.apple.com/us/app/petfinder-mobile/id557228073?mt=8



Location New York, NY

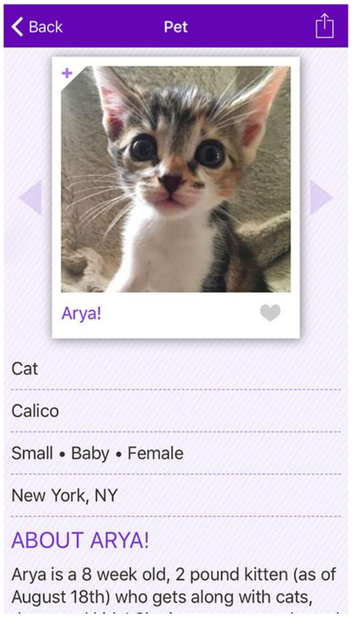
Breed Border Collie

## Search Page

The search page allows users to search by location, animal type (Dog, Cat, Other), age, gender, and size. It doesn't allow the individual to search for a specific animal, intentionally neglecting to add elements such as a search bar for allowing the user to "discover" a dog instead of finding a specific one.

Benefits of this include the search engine being extremely simple, easy to use, and not overly complicated. The objective the application's search engine has in terms of displaying results for users is made clear by the way they have approached it, allowing a more randomized set of results, and prioritizing location as well as a few specific other attributes. The UI is also extremely simple, using minimal buttons, text boxes and labels, only incorporating two colours (white, and purple), and one overarching font. Applying similar qualities this simplistic UI has to our application's UI would greatly improve it.

Disadvantages include the search engine being too simple for some scenarios, lacking in certain searchable attributes - such as looking for a 'housebroken' pet, or searching by a specific distance instead of just "closest to farthest", looking for more specific kinds of dogs, or searching for a specific, reliable, shelter or person to buy animals from them. The option "Other" seems to encompass any pet or animal that isn't a dog or cat, and could be given more options, such as bird or fish.



Arya!

Cat

Calico

Small • Baby • Female

New York, NY

ABOUT ARYA!

Arya is a 8 week old, 2 pound kitten (as of August 18th) who gets along with cats,

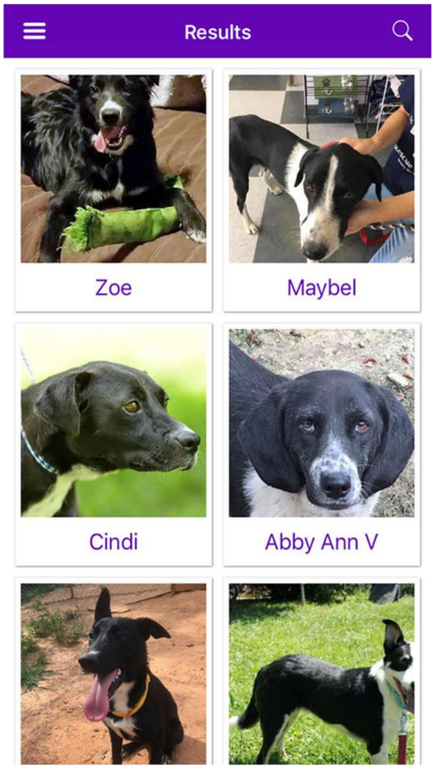
## Individual Animal Page

The individual animal page is a profile that allows users to view more details about pets they may be interested in. The page is also useful for owners of pets who wish to sell them, as it gives them a better chance to showcase their pet with its skills and qualities. The page consists of one or more photos, the name, type and breed, size, location, gender, age, and an about section that goes into further, non-specific detail about the dog.

Benefits of this include the page being useful for owners of pets who wish to sell them, as it gives them a better chance to showcase their pet with its skills and qualities. The page also has just the right amount of information; not too much to overwhelm the user, but not too little for the user not to know enough about the dog. Much like the "Search Page", the UI is simple with its colours, fonts, and overall display. Videos can also be uploaded as well as photos (but they must be embedded from another platform). The overall design and functionality of this kind of page can be applied to our app, improving it.

Disadvantages mainly include the way the profile implements videos, having to use a third-party application such as YouTube to host the video, and the profile not being able to support gifs.

### Results Page



The results page is a page displaying two columns of animals, animals that are like the attributes the user has searched for. The name of the animal and the image is displayed and clicking on one of these animals' names/images will bring the user to the animal's individual animal page.

Benefits of this include a simple, ordered results page with a simple UI (two colours, one
overarching font), allowing for an easy-to-navigate page.

Disadvantages include not showing enough information for each pet, only their name and profile photo. Important information could include location, age, etc.

#### Other Benefits, Disadvantages, and Potential Improvements

General UI Design: As demonstrated in the previous examples of UI in Petfinder's pages, the benefits of the app's UI design are numerous; following the design rules of alignment and proximity in the results page, employing one overall font and two colours as to not make the overall design convoluted, messy, and distracting.

**Problem Space and Approach:** It is clear from the way the Petfinder is designed, that its purpose is to allow users to purchase/own a animal near them, an animal not based on many factors predetermined by the user, but by the impression of the animals they happen to come across. The approach the app has to solve the problem space is what makes the problem space known; the approach the app takes is consistent in its design, and nothing outright contradicts the approach. Having a strong, consistent, approach is something that can be implemented in Scruff when solving the problem space.

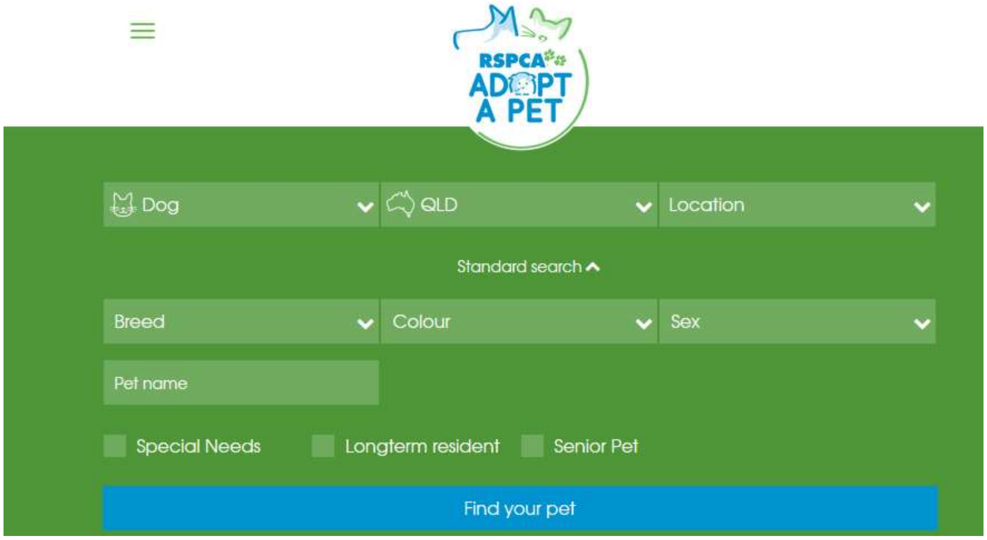
**Features and Functionality:** See each individual page for an analysis of the app's overall functionality. The only criticism I could give it is that the functionality sometimes lacks, not giving enough options/showing enough (see especially the app's "Search Page" and "Results Page").

**Navigation:** Since the app only has three primary kinds of pages (and Settings, Profile, etc), navigation is simple. The search page is accessed typically by pressing the magnifying glass symbol in the top-right corner. Searching creates a results page, where clicking on one of the results brings the user to the individual animal page. Settings, Profile, and other, minor parts of the app, can be accessed via a hamburger menu that can be revealed by pressing the three lines at the top-left corner. Our application will most likely incorporate a similar navigation scheme as it is simple for the user and effective.

RSPCA Adopt A Pet https://www.adoptapet.com.au/

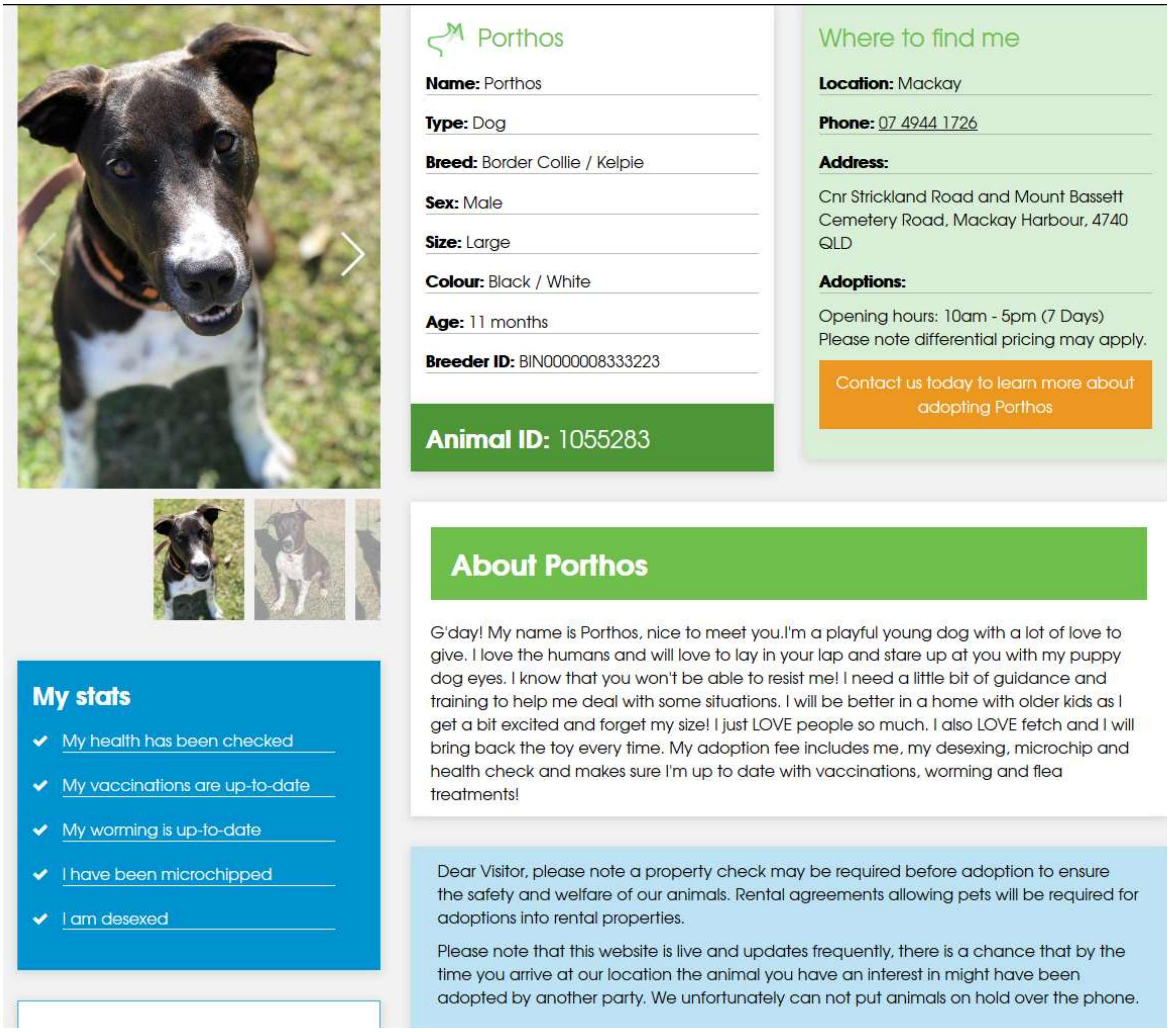
This web application created for the RSPCA adoption programme helps users find for animals located in all of its shelters.

Search Page



- This UI component includes both search and filter functionality by giving the user the ability to select properties that an ad listing should contain.
- The search is fully featured, with all functionality required for the average adopter. It is very easy to use and includes many filters. Maybe a filter for medical history and age could be included.
- Everything below the top three inputs are hidden behind an “advanced search”. This is useful as it encourages the user to browse through a larger number of animals, ensuring they select the one that is most suitable. This contrasts being able to filter down significantly, which can backfire as the user hasn’t perused the many suitable options.
- The design is excellent. Good contrast between background and control elements. It remains simple and pleasing to the eye.
- The colour scheme is based on the logo. Scruff will adopt this colour design.
- The hamburger menu in the top left drops down additional guides on adopting a pet such as “how to adopt”, “why adopt?”, “caring for your pet”, and a page about the service. This critical information may be inconveniently placed, especially considering the important of its content.
- The other drop-downs could also include icons like the first and second. It allows users to understand the purpose immediately.

Individual Animal Page



Porthos

**Name:** Porthos

**Type:** Dog

**Breed:** Border Collie / Kelpie

**Sex:** Male

**Size:** Large

**Colour:** Black / White

**Age:** 11 months

**Breeder ID:** BIN0000008333223

**Animal ID:** 1055283

Where to find me

**Location:** Mackay

**Phone:** 07 4944 1726

**Address:**

Cnr Strickland Road and Mount Bassett
Cemetery Road, Mackay Harbour, 4740
QLD

**Adoptions:**

Opening hours: 10am - 5pm (7 Days)
Please note differential pricing may apply.

Contact us today to learn more about adopting Porthos

**My stats**

- ✓ My health has been checked
- ✓ My vaccinations are up-to-date
- ✓ My worming is up-to-date
- ✓ I have been microchipped
- ✓ I am desexed

**About Porthos**

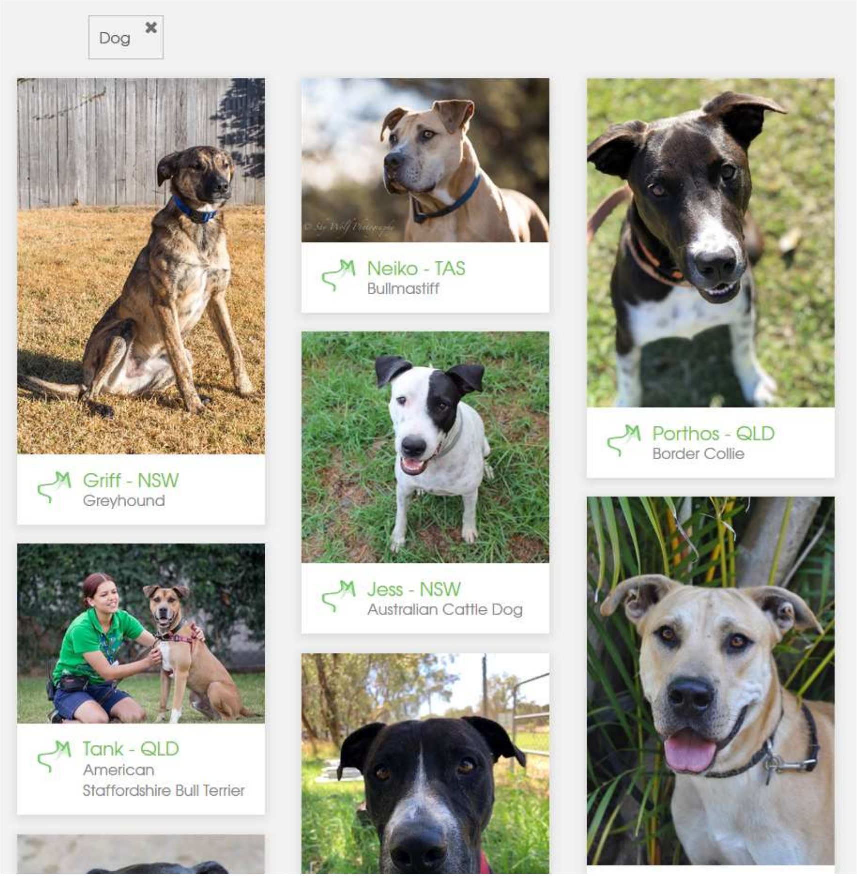
G'day! My name is Porthos, nice to meet you.I'm a playful young dog with a lot of love to give. I love the humans and will love to lay in your lap and stare up at you with my puppy dog eyes. I know that you won't be able to resist me! I need a little bit of guidance and training to help me deal with some situations. I will be better in a home with older kids as I get a bit excited and forget my size! I just LOVE people so much. I also LOVE fetch and I will bring back the toy every time. My adoption fee includes me, my desexing, microchip and health check and makes sure I'm up to date with vaccinations, worming and flea treatments!

Dear Visitor, please note a property check may be required before adoption to ensure the safety and welfare of our animals. Rental agreements allowing pets will be required for adoptions into rental properties.

Please note that this website is live and updates frequently, there is a chance that by the time you arrive at our location the animal you have an interest in might have been adopted by another party. We unfortunately can not put animals on hold over the phone.

- This individual results page is cleanly laid out, with clear segmentation between the key elements. The colours and text remain readable, and there is no superfluous information on the page.
- The image carousel however only contains photos, which does not provide enough information of the personality of the animal. Video integration would be appropriate.
- There is no obvious section for the animal's likes and dislikes. Instead it is mixed in with the "about" section, meaning the reader has to sift through information.
- Some information is verbosely worded or redundant such as the "My stats" section, "Name", "Contact us"
- Not shown in the photo above are the Facebook, Twitter, donating sharing links. Very useful to easily share to social media.
- For a mobile application, this information will have to be laid out vertically, and using tabbed page navigation as to not overload the user with information. A good separation idea for Scruff is for the animal's photos/videos, the animal's details, and the animal's contact details to be laid out in a tabbed pane.

## Results Page


Dog ✖

Griff - NSW
Greyhound

Tank - QLD
American Staffordshire Bull Terrier

Neiko - TAS
Bullmastiff

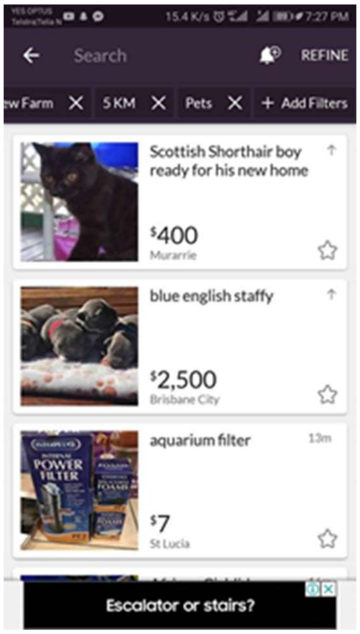
Jess - NSW
Australian Cattle Dog

Porthos - QLD
Border Collie

- The results are listed in a gallery fashion, with accompanying basic details. This format is clean and offers a good first impression. However, this format may lead to a “don’t judge a book by its cover” situation, as many “cute” animals can be deceivingly evil.
- The photos of each animal is well photographed, which is a stark contrast from many other marketplaces. The photos highlight each animal’s key features. However, the photo including the woman should be replaced.
- The location of the animal could more specific. State may be too broad.
- Each picture could be in a carousel fashion, with arrows shown when hovered over, to scroll through the photos.
    - This may however cause users to not delve deeper into an animal’s pace.
- At the very top of this page shows which filters have been applied, reminding users.
- Overall, the colours, text choice, text size, picture size, picture quality, and component separation is all very well laid out. A similar layout will be applied to the assignment, but more condensed for mobile apps.

The website is very consistent in its UI elements, colours, font, and iconography. Each element has a clear purpose and is evident to the user what will happen if interacted with. The results page of Scruff will take many elements of this application, such as the clear segmentation between the results, the modern UI, and simple colour schemes.

Gumtree Mobile App: [https://play.google.com/store/apps/details?id=com.ebay.gumtree.au&hl=en_AU](https://play.google.com/store/apps/details?id=com.ebay.gumtree.au&hl=en_AU)



Scottish Shorthair boy ready for his new home
$400
Murarrie

blue english staffy
$2,500
Brisbane City

aquarium filter
13m
$7
St Lucia

### Search Page

This marketplace has less search functionality compared to the other applications we have studied. However, it is one of the more popular apps when it comes to private sales through an online marketplace.

The search engine is fairly simple. The user chooses which category they want, then they click “refine” to add different filters like: max price, distance and more. Since pets are only a subcategory on this app, it lacks advanced search filters specified for selling animals. The buyer can’t search for specific attributes like breed, age or owner.

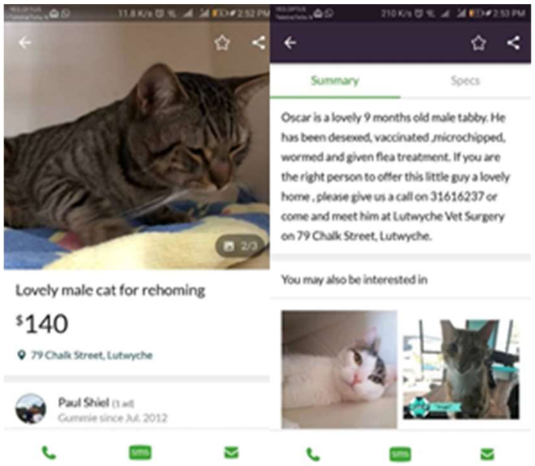
Under the search bar it shows the different attributes that have been added to the search. The buyer can easily remove attributes by tapping the X. This option makes the search attributes easier to manage, but less pleasant visually.

Gumtree has added the possibility to save searches and notify the user whenever something new is added. This is a feature that our group thinks would be great to implement into Scruff.

### Search Page (Results)

The page has a clean layout where every advertisement is listed in a box view. This separates all of the information, making it easier for the user to quickly skim through the different advertisements. For Scruff, we have discussed the possibility of designing a box which will slide to reveal a set of options (hide, block, save). All of the advertisements have a star that can be clicked for saving the advertisement so that the user can revisit it later. Since this a good way for the user to keep track of their favourite animal, we will implement this feature in Scruff, but with the option to get notified if the seller posts new content to their feed.

### Individual animal page:



Lovely male cat for rehoming
$140
79 Chalk Street, Lutwyche

Paul Shiel (1 ad)
Gummie since Jul. 2012

Oscar is a lovely 9 months old male tabby. He has been desexed, vaccinated ,microchipped, wormed and given flea treatment. If you are the right person to offer this little guy a lovely home , please give us a call on 31616237 or come and meet him at Lutwyche Vet Surgery on 79 Chalk Street, Lutwyche.

You may also be interested in

The first thing that you see when you open one of the advertisements is an enlarged picture of the animal. This image is connected to the top and side edges of the screen, which separates it from the rest of the page. This design accentuates the picture more than other designs do.

Further down on the page there is a tabbed menu (right picture) that categorises the different information in the advertisement. This organises the information on the page which facilitates more efficient perusal.
Adding the picture to be part of the tabbed menu would appear more tidy, as there would be less information on the first page.

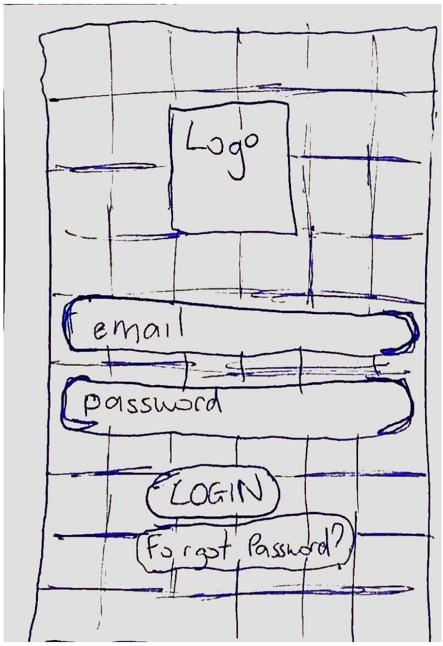
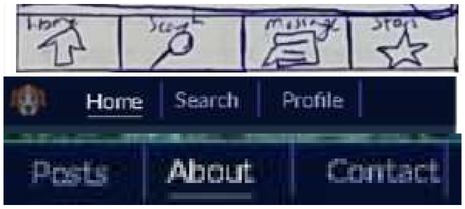
**General UI Design:**
The entire application uses the same font and three main colors for all the objects consistently. This helps separate the background, menu and advertisements from each other, and creates a better user experience.

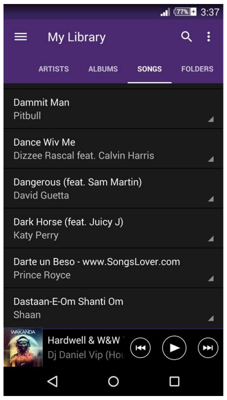
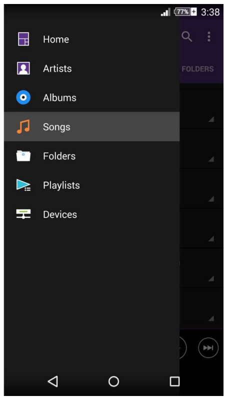
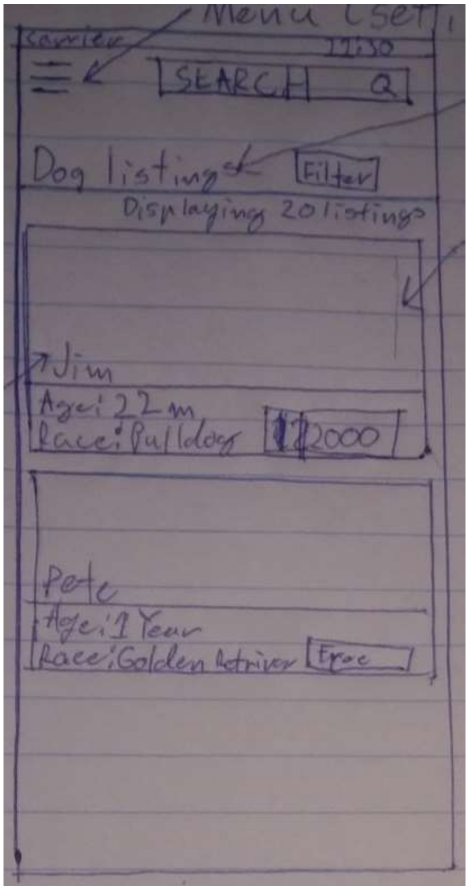
**Navigation:**
The navigation in this app seems slightly unorganised. The left menu contains shortcuts to the user's profile and messages while the other important information is only accessible from the main page. Therefore, when searching for ads on the application, the user won't be able to check any of the menu options without exiting what was previously open.

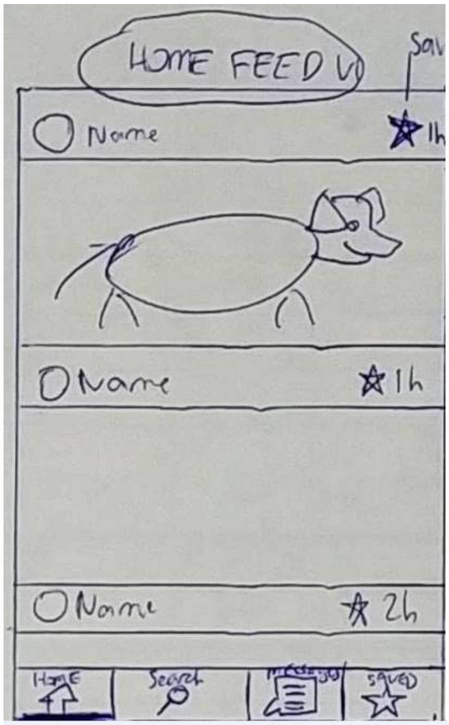
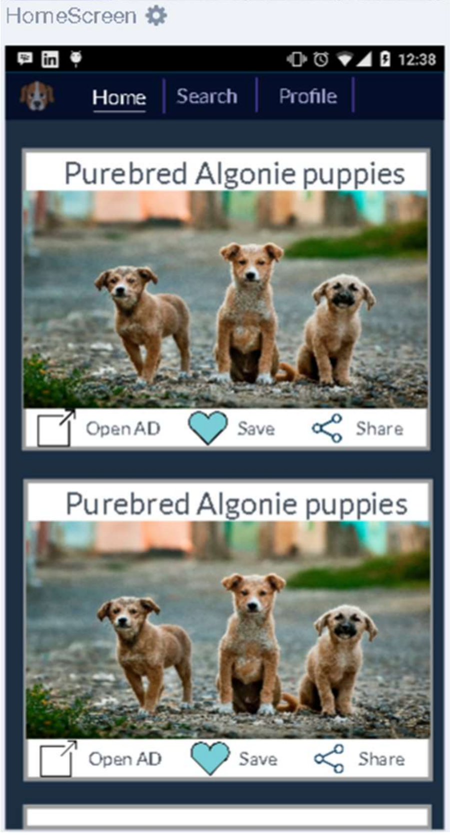
**Learned:**
While testing out this app there were many features that we thought would be helpful to implement into our own mobile app. We decided that we also wanted to make it possible to search for advertisements that are nearby. Retaining searches for future use will end up saving the user time. The user can choose an option and whether or not they want to receive an email or text whenever there is something new posted within the search criteria. This notifying method can also be used for telling users about changes to some of the animal's profiles that they are following.
We also liked and implemented the idea of using a tabbed menu to separate information within the advertisement, as this is a great way of keeping the page less cluttered. We will be adding this in our UI prototype section below.

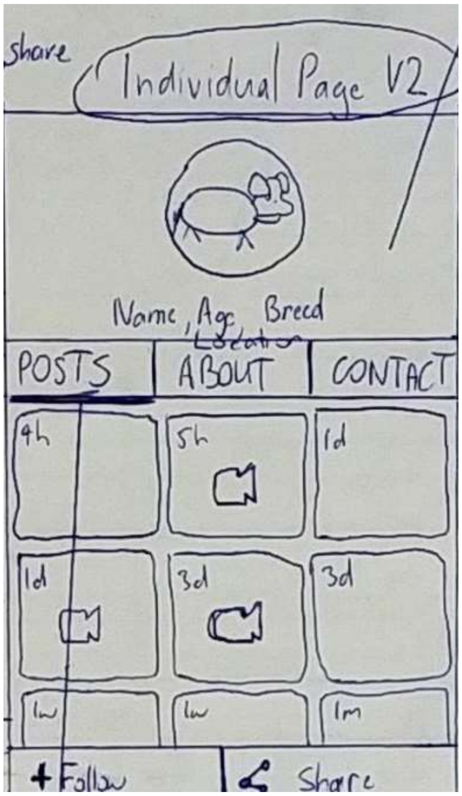
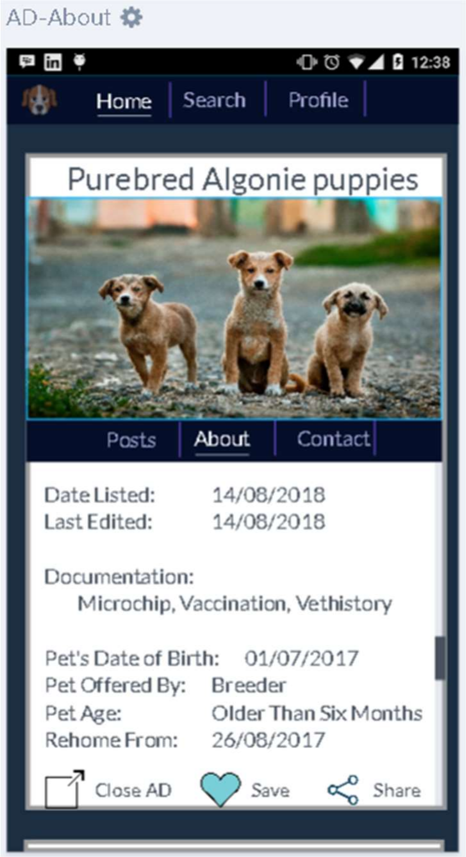
# Mobile Application Design

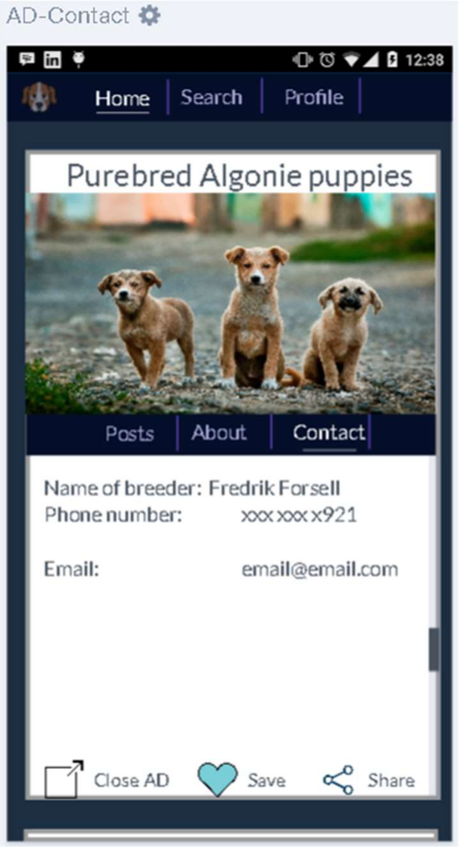
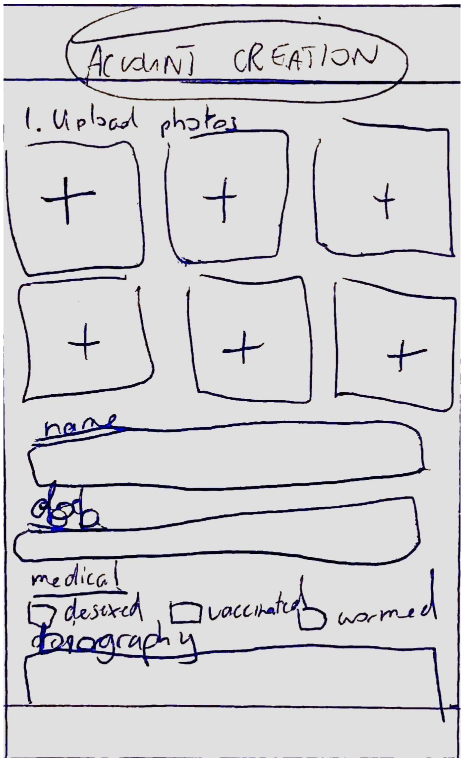
## UI

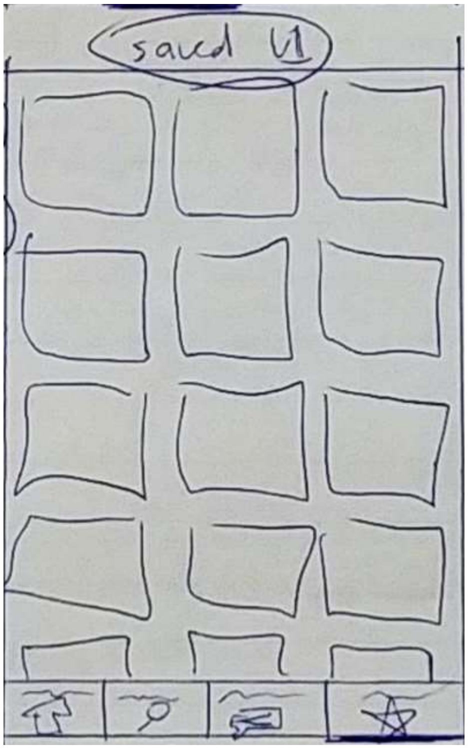
| Page | Wireframes | Justification of Patterns and Styling |
|---|---|---|
| Login page |  | - Login pages should be simple, with only necessary functions such as **logging in** fields, as well as **forgotten credentials** functions.<br>- Not shown, but there is also a **sign up** button under login<br>- To add a touch of personality, the background of this will be a **grid** of various animals.<br>○ The grid will have a **dark overlay** over it to create a good contrast between the background and the main page elements (logging in inputs)<br>○ An alternative design choice is to make a video play in the background, like Spotify's login screen. It would play videos of puppies.<br>● The input elements will have **placeholder text** to remind the user what to put into it<br>● An error text will display when the user incorrectly enters their credentials<br>○ "Your email or password is incorrect"<br>● Maybe there will be a "**show password**" field, (optional)<br>While not shown, there will be also be buttons allowing the user to sign in with **social media accounts** |
| Tabbed browsing |  | - On iOS, the app will feature this tabbed pane at the bottom; a common navigation tool. It allows the user to reach frequently used parts of the app quickly.<br>- On Android, the standard hamburger menu and left side navigation will be used instead. Users will be able to swipe from the left edge of the screen to open it up quickly.<br>- By using the standardised method of navigation for the respective operating systems, users will be able to quickly adapt their current knowledge to this application.<br>- This provides an experience that flows consistently from other apps to Scruff<br>- Tabbed panes were decided as it separates related information from irrelevant. By doing so, the screen is less cluttered. |

| | | |
|---|---|---|
| | <br>3:37<br>My Library<br>ARTISTS ALBUMS SONGS FOLDERS<br>Dammit Man<br>Pitbull<br>Dance Wiv Me<br>Dizzee Rascal feat. Calvin Harris<br>Dangerous (feat. Sam Martin)<br>David Guetta<br>Dark Horse (feat. Juicy J)<br>Katy Perry<br>Darte un Beso - www.SongsLover.com<br>Prince Royce<br>Dastaan-E-Om Shanti Om<br>Shaan<br>Hardwell & W&W<br>Dj Daniel Vip (Ho<br><br>3:38<br>Home<br>Artists<br>Albums<br>Songs<br>Folders<br>Playlists<br>Devices<br>FOLDERS | |
| Search and FIltering and Results |  Menu (set)<br>Scruffer<br>12:30<br>SEARCH Q<br>Dog listings<br>Filter<br>Displaying 20 listings<br>Jim<br>Age: 22 m<br>Race: Pulldog<br>$2000<br>Pete<br>Age: 1 Year<br>Race: Golden Retriver<br>Free | - This design was chosen to show the most amount of information on a page, while still preserving images and clean UI styles.<br>- The top of this page will have the **search bar,** which contains an **input text field**, and a **confirmation button** to search.<br>    - Maybe this could be moved to the bottom (as it is more ergonomic for larger devices)<br>- **Placeholder text** on the search bar. "Search by breed, name, location, etc."<br>- Scruff aims to give users a diverse range of animals, which is why the filters will be hidden behind a button unless the user selects "**filter**". This will ensure that users consider many animals, rather than going for a specific one.<br>    - Different animals all have unique personalities and traits, which is why we have chosen to do it this way. Often users will have too narrow a view when browsing, which means they are missing out on 90% of the selection<br>- The filters will feature a variety of controls such as sliders (age, location range), checkboxes (breeds, size), text inputs, and drop downs.<br>- Once the user has confirmed a search, the application will fetch data from a database, and display it below the search functions. Each of the ad listings will contain a photo, name, and price, and location.<br>- More ads will load when the user reaches the bottom, to improve performance by not loading all the ads immediately. |

| | | |
|---|---|---|
| Home Feed |  HOME FEED V<br>Sav<br>Name<br>1h<br>Name<br>1h<br>Name<br>2h<br>HOME<br>Search<br>MESSAGES<br>SAVED<br> HomeScreen<br>12:38<br>Home<br>Search<br>Profile<br>Purebred Algonie puppies<br>Open AD<br>Save<br>Share<br>Purebred Algonie puppies<br>Open AD<br>Save<br>Share | - This is the main page of the application (the page loaded when the app opens)<br>- It will display in **reverse chronological order** (new to old) posts of the user's followers.<br>- It will be a **vertical feed** with obvious separation between each post.<br>- The user will be able to click on the picture to **enlarge** the photo (and be able to zoom in on it)<br>- The user will be able to click on the name and thumbnail to go straight to the **animal's individual page**.<br>- To the right of the names, there will be a **star** that the user can click on to **follow an animal**. Once followed, the star will be filled with colour (unfollowed is an outline).<br>- To the right of that, there will be a **timestamp** showing when the post was uploaded<br>- Optional - There will be a button that will appear when scrolling up, allowing the user to **scroll to the top**<br>- Video posts will have a "play" button overly, alerting the user that it is a video.<br>    - Maybe it will auto play when scrolled down to |

| | | |
|---|---|---|
| Individual Posts Page |  share<br>Individual Page V2<br>Name, Age, Breed<br>Location<br>POSTS<br>ABOUT<br>CONTACT<br>4h<br>5h<br>1d<br>1d<br>3d<br>3d<br>1w<br>1w<br>1m<br>+ Follow<br>Share | - This page functions **similarly to Instagram**. The posts of an animal will be in reverse chronological order.<br>- This will be further subdivided into tabs, which allows information to be split up into categories.<br>- **Posts** will contain the posts (pictured)<br>- **About** will contain the animal's details such as medical history, biography, about me, likes, dislikes, etc.<br>- **Contact** will contain the contact details of the owner such as their name, phone number, email, shelter/store information (if applicable)<br>- The top of this page has the main photo of the animal, as well as the cover photo.<br>- Below the image, there will be the name, age, breed, and location of the animal.<br>- Clicking on the posts will **enlarge it**<br>- Videos will have a video icon on it<br>- Posts will have a timestamp on it<br>- At the bottom of the page will contain buttons, allowing the user to share the specific animal to social media, messaging, email, etc<br>- The other button will allow the user to follow/unfollow the animal |
| Individual About Page |  AD-About<br>12:38<br>Home<br>Search<br>Profile<br>Posts<br>About<br>Contact<br>Close AD<br>Save<br>Share<br>Purebred Algonie puppies<br>Date Listed: 14/08/2018<br>Last Edited: 14/08/2018<br>Documentation:<br>Microchip, Vaccination, Vethistory<br>Pet's Date of Birth: 01/07/2017<br>Pet Offered By: Breeder<br>Pet Age: Older Than Six Months<br>Rehome From: 26/08/2017 | - Same header as the "posts" tab.<br>- Has a description (usually written from the perspective from the pet e.g. "Hello my name is doggo")<br>- Separated into sections such as likes, dislikes, medical history, size, colour, etc<br>- Maybe also contain information about the shelter / previous owners if applicable<br>- This page will contain information that could not fit on the ad listing card<br>- The user would go to this tab if they would like to know more about the animal, based on its posts. |

| | | |
|---|---|---|
| Individual Contact Page |  AD-Contact<br>12:38<br>Home<br>Search<br>Profile<br>Purebred Algonie puppies<br>Posts<br>About<br>Contact<br>Close AD<br>Save<br>Share<br>Name of breeder: Fredrik Forsell<br>Phone number: xxx xxx x921<br>Email: email@email.com | - Has picture of owner (may not be necessary), and name.<br>- To combat spam, bots, the contact details must be clicked to reveal. It will also be unavailable without an account (maybe captcha too?)<br>- There will be email and phone numbers as methods of communication.<br>- In-app messaging is an optional feature<br>    - This would help managing all the contacts, as a user will be able to easily remember which animal belonged to this person.<br>- This page marks the end of the tabbed pane, and thus the location where the user takes action by contacting the owner(s). |
| Account Creation |  ACCOUNT CREATION<br>1. Upload photos<br>+<br>name<br>dob<br>medical<br>desexed<br>vaccinated<br>wormed<br>biography | - Once the user has signed into this app, they can choose to add a pet<br>- On the pet account creation page, the user can upload up to 6 images. There will be a recommendation to the user to upload a variety of different images, showing the animal in many situations/poses<br>- Simple text inputs for the names, dob, biography, likes, dislikes, etc<br>- Checkboxes for the medical history<br>- Dropdown for the breed<br>- Confirmation button at the bottom<br>- Once a user has confirmed their account, an email/SMS verification be also be required. |

| | | |
|---|---|---|
| Saved Posts Page |  | - Accessible through the tabbed pane, this page will show all the liked/starred/saved posts<br>- It gives the user an archive of the user’s favourite animal posts.<br>- This is useful as the user does not have to scroll down days if not weeks of posts to see the ones they really liked<br>- While not shown, this page may also be split into a tabbed pane of “saved posts”, and “followers”, the latter being a list of animals the user has followed. From here they can unfollow them quickly.<br>    - Due to this, the bottom tab may be renamed to “Me”<br>    - This tab will have account options, profile options, animal account options, saved posts, and a followers management page. |

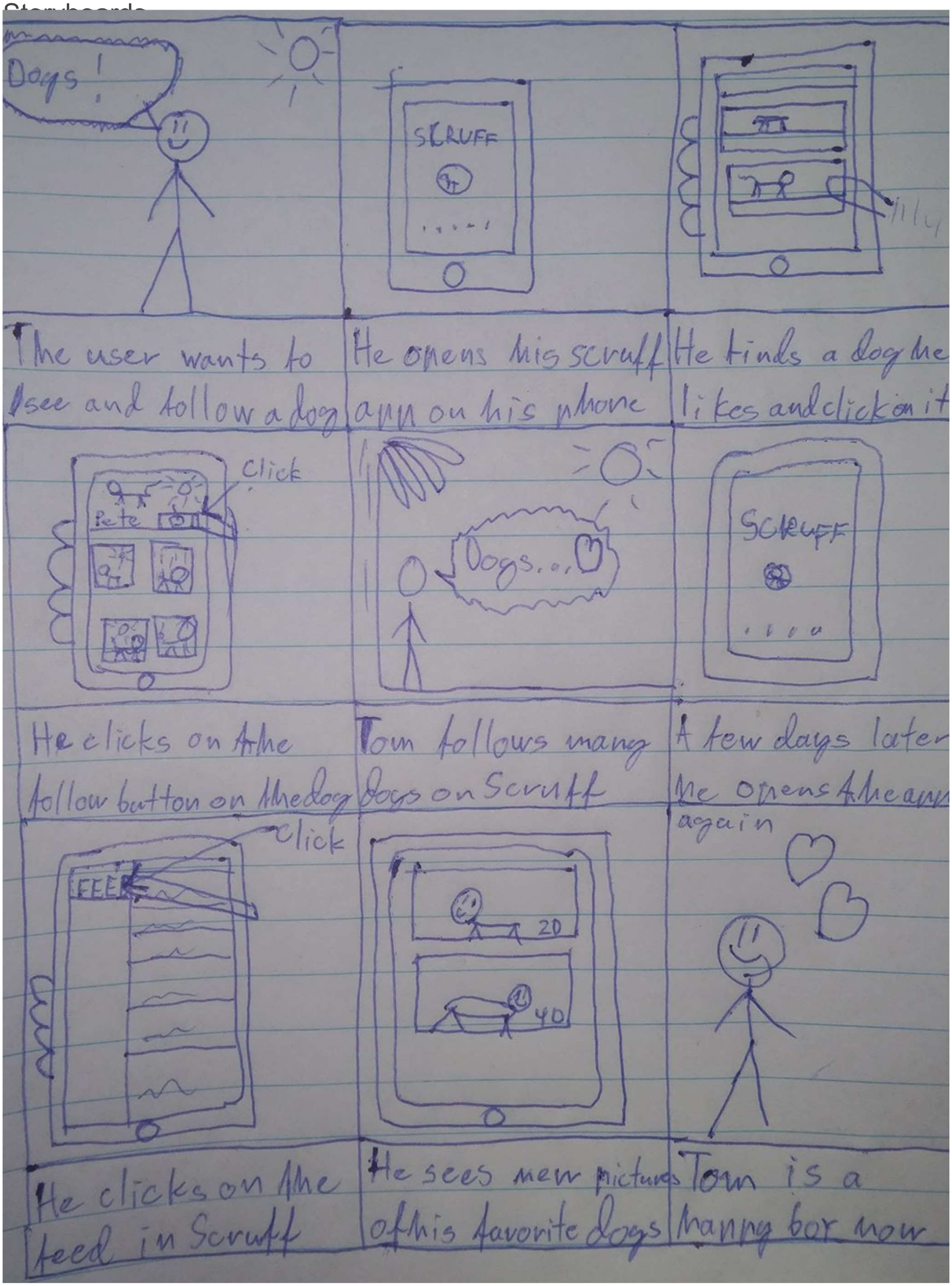
## User Stories

The following are a list of user stories pertaining to the application Scruff, and the indented points are each user stories' acceptance criteria:

1. As the manager of a dog shelter, I want to be able to show that the animals I sell need a home so that people who want to keep a stray dog will know to buy my animals.
    a. If the animal being presented belongs to an animal shelter or similar organization, it will be indicated in the profile of the animal and can be indicated to the user in the results page of a search.
2. As an owner, I want to be able to showcase my pet off properly so that people might be more interested in buying my pet off me.
    a. The profile for the animal will allow the owner to input data about the dog, such as type, breed, age, weight, etc.
    b. More general information can be elaborated on in a large "About" text box.
    c. The profile can be updated with images, gifs, and short videos, in a similar design to Instagram.
3. As a customer, I want a page to view more information of an animal, so I can make a better judgement as to whether this is the animal I want to have.
    a. There is a profile for each animal filled in by the owner that elaborates more about the pet, including type, breed, age, weight, other attributes, and an "About" section that goes into more detail.
    b. The profile will often be regularly updated with images, gifs, and short videos, like an Instagram profile.
4. As an owner, I want to be able to utilize images of my pet in my pet's profile so that people will know what to expect and maybe find the pet good-looking and purchase it off me.
    a. Multiple images of the animal are supported in the pet's profile.
5. As an owner, I want to be able to utilize gifs of my pet in my pet's profile so that I would be able to showcase off what my pet can do (tricks, etc.), without needing a full-length video.
    a. Multiple gifs of the animal are supported in the pet's profile.
6. As an owner, I want to be able to utilize videos of my pet in my pet's profile so that I would be able to showcase off what my pet can do (tricks, etc.) that cannot be properly shown in a short gif or single image.
    a. Multiple videos of the animal are supported in the pet's profile via embedding on YouTube or another video platform.
7. As a customer with a specific type of animal in mind, I want to be able to search for a specific animal with specific attributes (type, breed, age, weight, housebroken), so that I get the perfect animal.
    a. Users can search via a text bar, where they can search by exact name of the animal, or breed, or other attributes.
    b. Users can also utilise an advanced search tool where they can specify multiple attributes covering a wide range of aspects.
8. As a fan of pets, I want to be able to follow specific pets that I like and have a feed, so that I can sate my love of pets.
    a. There is an option to 'follow' a specific animal.
    b. There is a "feed" page which is regularly updated with any updated media posted on each followed animal's profile.
9. As a customer sympathetic to stray animals without owners, I want to be able to find animals based on whether their owners are part of an organization like animal shelter or a private person, so that I can own a stray animal and make a more positive impact to that animal's life.
    a. See 1.'s acceptance criteria.
    b. There is an option to limit a search to "shelters only" in the Search Page, meaning all the results of the search in the Search Page would come from shelters and other similar organizations.

10. As a user, I want the overall design of the app to be simple, incorporating only a few colours and fonts, so that I won't be distracted or disgusted by how the app looks whenever I use it.
    a. No more than three primary colours are used, and only one overarching font is used throughout the UI wherever applicable.
11. As a user, I want to login to my account, so that I can have access to it and use it.
    a. Users can login to an account using the login page, by entering their username and intended password.
12. As a user, I want to create an account with Scruff, so that I can do the actions that having an account allows you to do.
    a. Users can create a new account using the registration page, by entering their username, password, retyping their password, and entering their email address and phone number.
13. As a user, I want the option to logout, so that I can secure my account.
    a. Users can log out anytime they like in the user menu.
14. As a user, I want the app to be easily navigable so that I don't get lost or frustrated.
    a. The buttons to access the main pages will be located at the bottom of the screen in iOS, or on a left-side hamburger menu that can be opened in the top-left corner in Android.
15. As a customer, I want the results page of the application to be simple and well-laid out when I make a search, so that I can navigate through the results more easily.
    a. The results page will dedicate a large space in the vertical feed to each animal. All the results will contain a photo, name, breed, and location. Optionally the user will also be able to click on arrows located on the photos to browse through the animal's profile photos.
16. As a user, I want to be able to change my password, so that I can keep my account secure even when my password is made known.
    a. Users can change their password in the Settings page by clicking a link.
    b. The user must type their old password, new password, and confirm their new password by typing it again.

## Storyboards



The user wants to see and follow a dog

He opens his scruff app on his phone

He finds a dog he likes and click on it

He clicks on the follow button on the dog

Tom follows many dogs on Scruff

A few days later he opens the app again

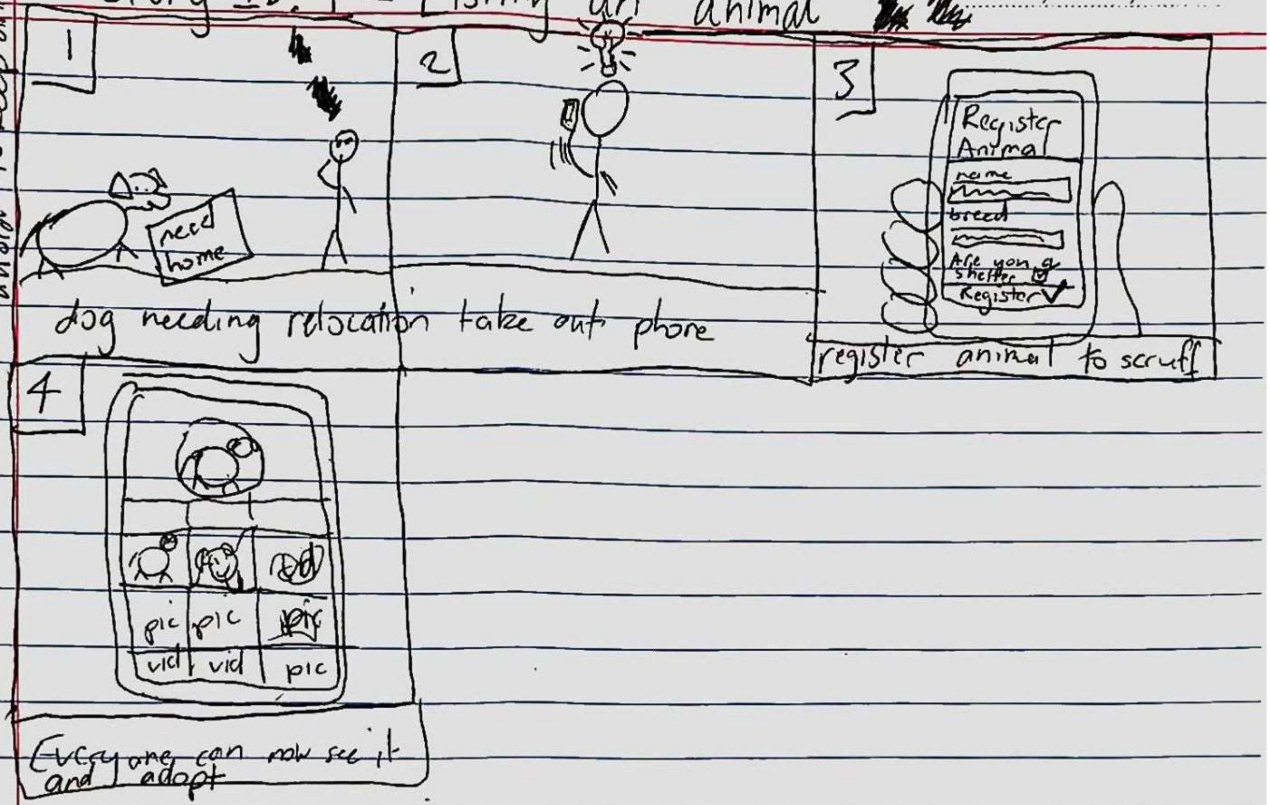
He clicks on the feed in Scruff

He sees new pictures of his favorite dogs

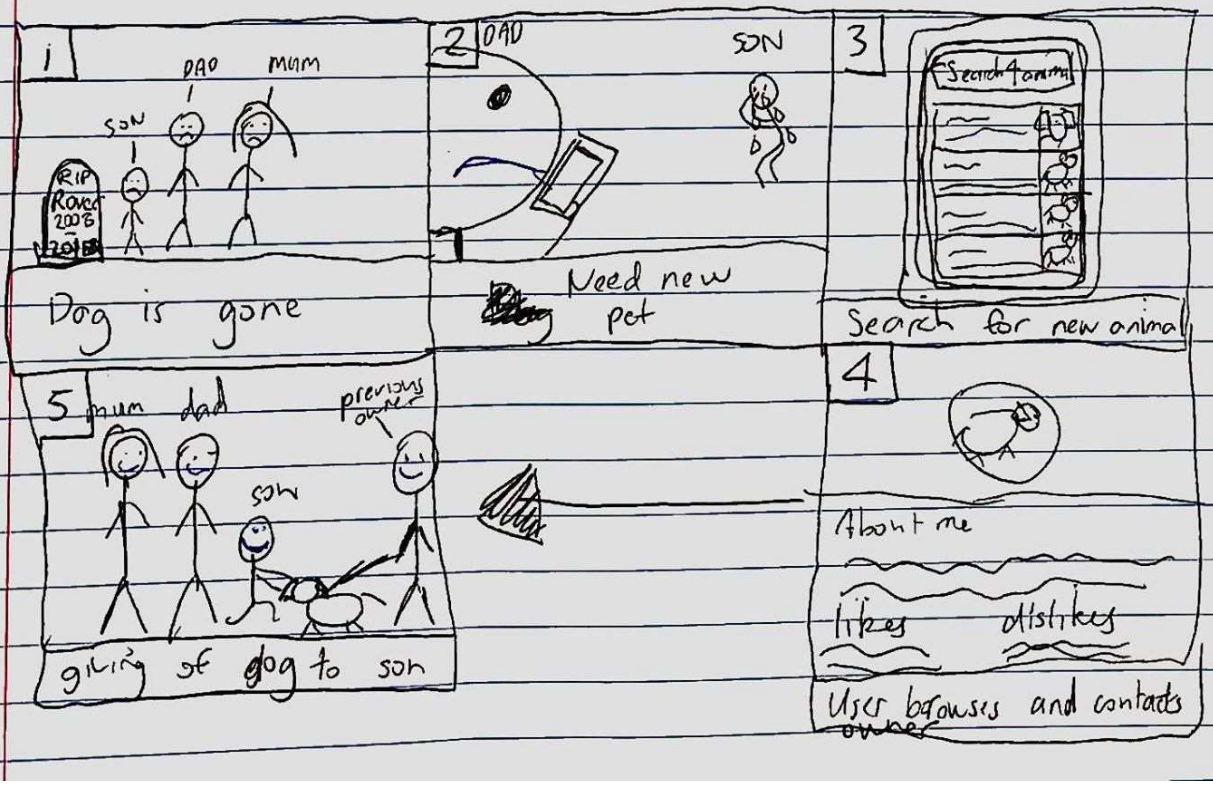
Tom is a happy boy now

owner can no longer afford to keep animal

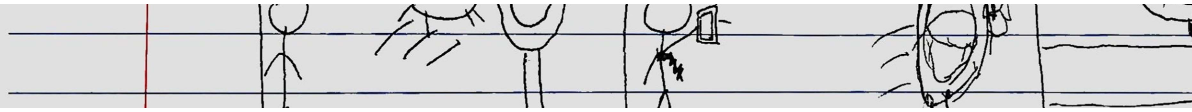
## Story ID: 1 – Listing an animal



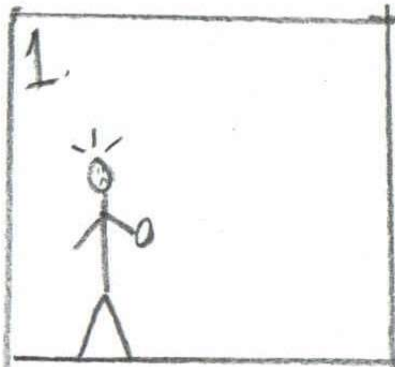
## Finding animals as end user story 3



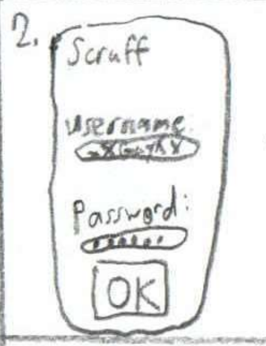
The top storyboard describes use cases 1 & 12. The bottom storyboard describes user case 3



This storyboard describes the user cases 2, 4, 5, 6

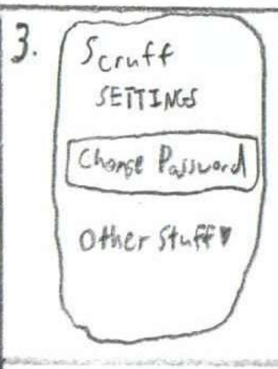


Man is worried about the security of his password after a recent hack of another account

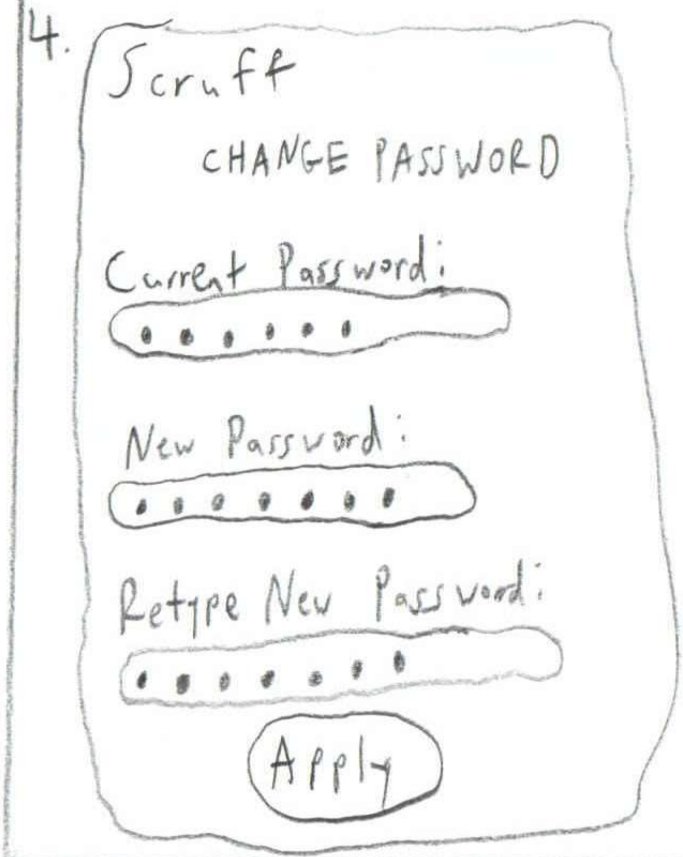


**Covers Use Case 11**

He logs in to his Scruff account, typing his username and current password

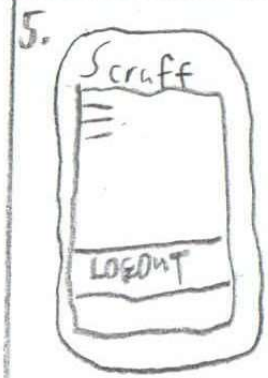


He goes to settings and chooses the "Change Password" option



**Covers Use Case 16**

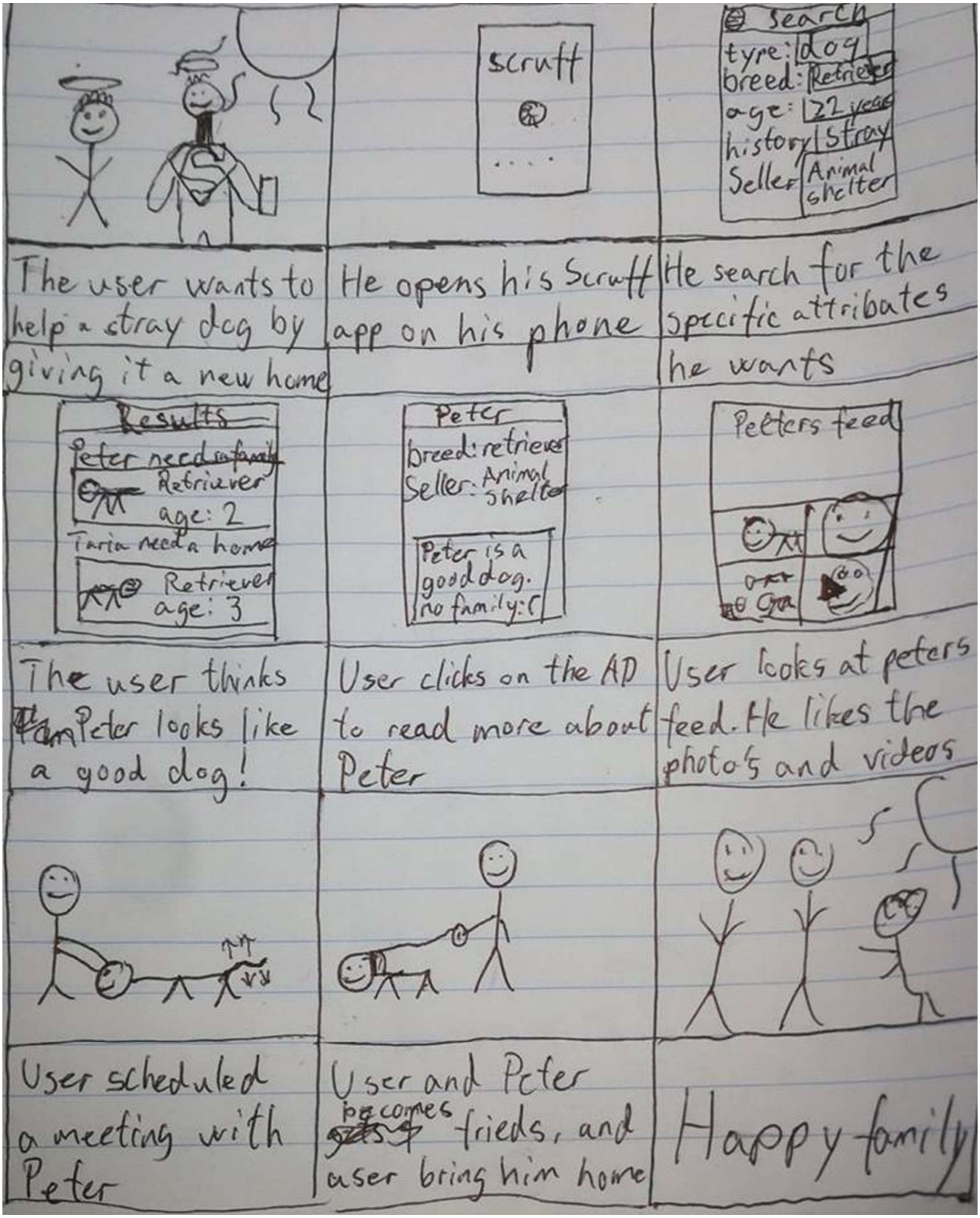
He enters his current password, then enters the new password, then retypes the new password



**Covers Use Case 13**

After Applying, he locates the logout button, and logs out of his account

Man is now reassured that his account is safe with a different password

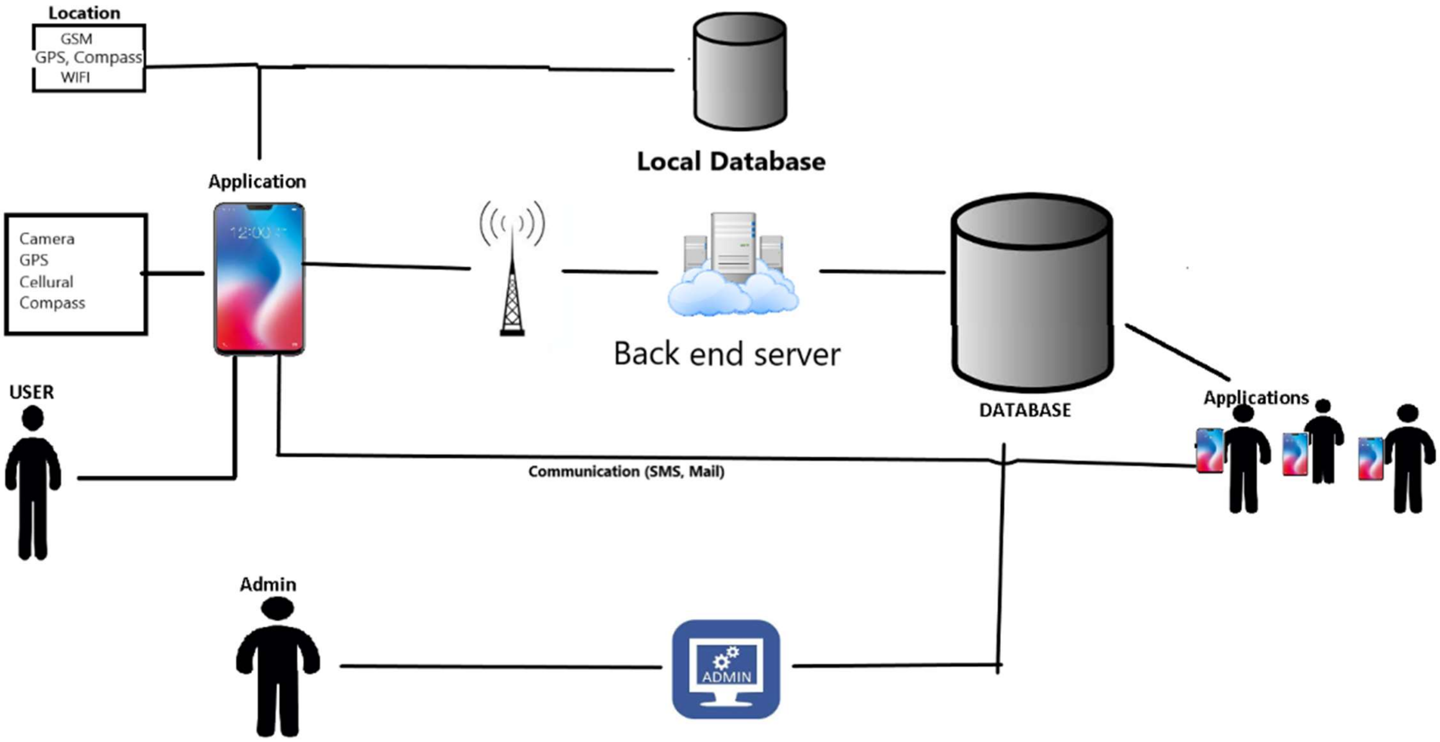

scruff
search
tyre: dog
breed: Retriever
age: 2 year
history Stray
Seller Animal shelter
The user wants to help a stray dog by giving it a new home
He opens his Scruff app on his phone
He search for the specific attribates he wants
Results
Peter need a family
Retriever age: 2
Taria need a home
Retriever age: 3
Peter
breed: retriever
Seller: Animal shelter
Peter is a good dog. no family :(
Petters feed
The user thinks ~~Tam~~ Peter looks like a good dog!
User clicks on the AD to read more about Peter
User looks at peters feed. He likes the photo's and videos
User scheduled a meeting with Peter
User and Peter ~~gets~~ becomes frieds, and user bring him home
Happy family

This storyboard describes user cases: 3, 7, 9 and 15

# Architecture

## Implementation of System Architecture

**System Architecture diagram:**



We want to let our app have the ability to send and receive pictures and videos to a database. Users will be able see other people's content, and also be able to upload their own. Users will either upload a file or take a picture/video directly through the application. To make sure that the advertisements are following the rules, we will have to implement an administrative service so that we can easily remove unwanted content from the database.

In some cases, the application will ask the user for access to their GPS device. These coordinates will be used to calculate the approximate distance between the seller and user.

The application will also let the user communicate directly through their email and phone number shared on their advertisement page. We are looking into the possibility of creating a direct messaging method within the application.

To make sure that users don't break the rules of our application we will implement an admin application that will have access to the main database. This way we can have a reporting system, giving us the ability to edit or delete advertisement and manage user accounts.

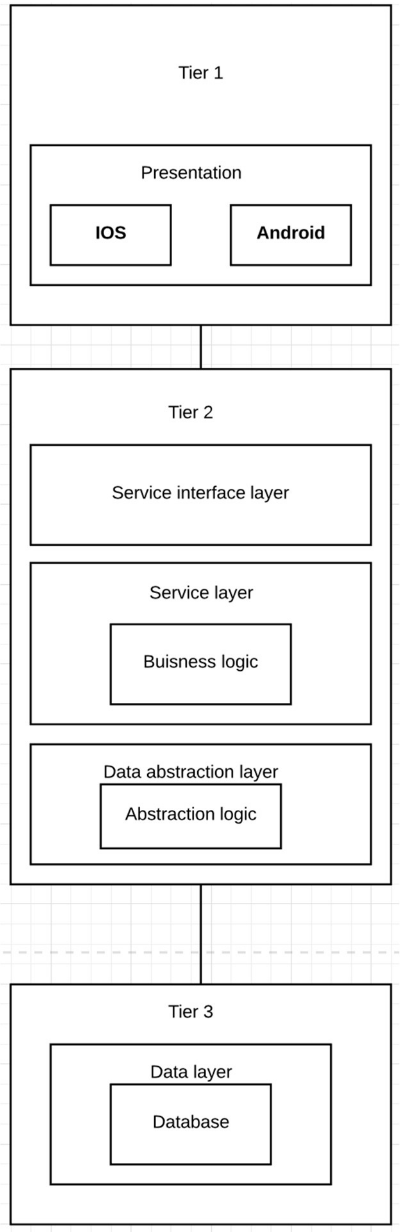
IVA:

We have decided to follow the best practices for our system architecture. We reviewed the technical requirements to enable our application to work as intended:

Our requirements for the system was the following:

- Security: we needed to secure our data and authorization process from unwanted access.
- Data and business logic: Our application requires communication with a external database and external logic (unknown to the end user)
- API: We wanted to seperate the presentation (the UI/application) from most of our logic. This means that our application will need to fetch data from our database. This functionality will be enabled through API's. The API's will be the only interface in which our application communicates with our database and logic.
- We also wanted to separate our business logic from our database. This requires a backend server with a interface to clients (the API) and a "external connection" to our database.
-

After these considerations we decided to go with the multi tiered system architecture, with 3 tiers.



The reason behind this is to meet the requirements state above.

**Tier 1:** Is what a normal user sees, it contains the presentation layer. That's the app that the users use to interact with the system.

**Tier 2:** This tier contains three layers:

- The service interface layer. This is will be the API that the presentation layer interacts with by making requests to this layer. The requests accepted will be defined here in order to limit access to our data and business logic. It will also interact with the service layer.
- The service layer contains is were our logic happens, like calculations done on the backend. And prepare data for delivery to the client. The service layer communicates with both the Service interface layer and the data abstraction layer.
- The data abstraction layers handles communication with our database. And delivers data to the business logic layer. It also takes requests/orders from the service layer and returns results.

**Tier 3:** This tier contains our database system. Its seperated from the other layers. Separation of the database from our tier 2 layers may help with security because we can define a set of accepted requests from the Abstraction logic. This will make a malicious request from the client to go through two different verifications before any data is changed or delivered.

## Flexibility and Maintainability

Thanks to Xamarin, Scruff will be able to perform well on the main phone software technologies: Android, iOS, and Windows Phone. It will be functional up to fairly late versions of each OS (Android Oreo 8.1, iOS 11.4.1, Windows Phone 10), and will be functional with earlier versions as it isn't a complex application. It will be functional on the vast majority of devices running any of the above OS's, although the application makes use of camera and location sensors, none of these sensors are essential to the application's basic operations.

Scruff will be able to support major changes in features. The coding principle of encapsulation will be enforced, so all functions will be seperated in their different classes, and there is less chance of the change of one method negatively clashing with another method somehow. The UI will be contained within one class, and based on a single point-of-entry, so that any changes to the UI will not reflect badly on the rest of the application.

Scruff is somewhat flexible logically. The search page and settings page can be accessed anywhere, and its access can change without affecting navigation significantly. However, the results page is only generated as a result of search page, and the ad/individual animal page can only be accessed via the results page. The fonts and colours are minimal and can be changed with ease.

Although the app is aimed at selling dogs, it can be extended to cats, birds, rodents, fish, and other animals. The only attributes that would need to be changed is different "breeds", most other attributes are applicable to all animals. Some attributes, however, will have to be skipped depending on the animal's species, as they may not be applicable to all animals (e.g 'housebroken' or the concept of a stray animal. Can a fish be 'housebroken' or 'stray'?)

Probably the most inflexible portion of Scruff is having to load gifs and videos in the ad pages/individual animal pages; not every phone necessarily has the processing power to play gifs and videos, and this can be taxing for people with weak phones that desire to view the ads for each animal.

## Integration

Scruff supports being integrated with new technologies, and would probably end up relying on these technologies if the app were ever to go for a public, wider release.

Scruff can be easily integrated into a web service. The app shares similar qualities and has similar objectives to certain websites that are also focused on giving animals a good home (refer to "Mobile Application Reviews", especially "RSPCA Adopt A Pet"), and all of the aspects of the app could be easily converted to a website format that devices other than mobile phones can access. A website could be used to branch out the application's possibilities, allowing PCs and laptops to partake in the application, and allowing for more complex functionalities that websites can support and phone applications cannot.

An application like Scruff can easily rely on a cloud service to host the mass of media, profiles, and other information, since the application wouldn't rely on a specific, local server, and can be easily pushed to several remote servers hosting a cloud. Having to physically host, wire, and program a multitude of servers to host the architecture of the application as well as the data collected can be troubling, difficult, and expensive, so a cloud solution would be preferable.

## Testing

Scruff is designed to support comprehensive testing, including unit tests, integration tests, system tests, and acceptance tests.

Since encapsulation will be implemented throughout the system, and each method and function are more-or-less isolated from one another, each function will be independently tested on its own.

Using Nunit, Scruff will be capable of UI, or “whitebox” testing, as well as general “blackbox” code testing and user experience testing.

There are many challenges when it comes to mobile app testing, most of which can be summarized in the four following sets:

1. Connectivity
2. Device Constraints
3. Device Diversity
4. Installation and Maintenance

The design of the application will address any problem in all of these sets:

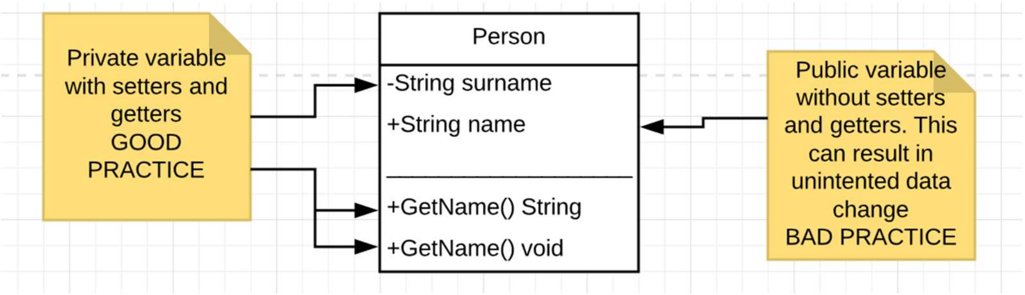
- None of the unit tests will rely on connectivity as it will not be launched online at the moment.
- Xamarin allows testing over a wide, diverse range of OS’s.
- As specified in “Flexibility and Maintainability”, the app is simple enough to be run on most phones, even a phone without good sensors.
- With the way the application architecture is structured, there will rarely be a unit test that will become redundant thanks to an update.

## Implementation of Software Architecture

The first thing we considered was how the application would operate. We would like to have the opportunity to add more features in the future without adding too much complexity to our codebase and avoid common bugs. Avoiding bugs and keep information accurate

### **Encapsulation**

Since our software will have many classes and methods we will have to encapsulate classes and methods in the classes. This means that we will restrict access to objects methods and variables. For example where we can we will avoid having public variables or static methods that modifies objects. We will have to discuss these considerations during development. When our classes and methods are finished with strong encapsulation, it will pay off since it limits potential mistakes when connecting everything together.


Private variable with setters and getters GOOD PRACTICE
Person
-String surname
+String name
+GetName() String
+GetName() void
Public variable without setters and getters. This can result in unintented data change BAD PRACTICE

We will enforce this setup of classes and methods in order to avoid problems with unintended changes of objects or method runs. We will also encapsulate classes whenever possible.

### **Façade**

We will also use façade to an extent when we have multiple classes which tries use the same features. For example when communicating with our backend, file system, sensors. We are considering having for example one class for communications with our backend. This class will act as a single point of contact for the classes/objects/methods that want to communicate with the backend server. So when we are developing or troubleshooting we won't have to look in multiple classes for methods relating to backend communication.

### **Model-View-Controller implementation**

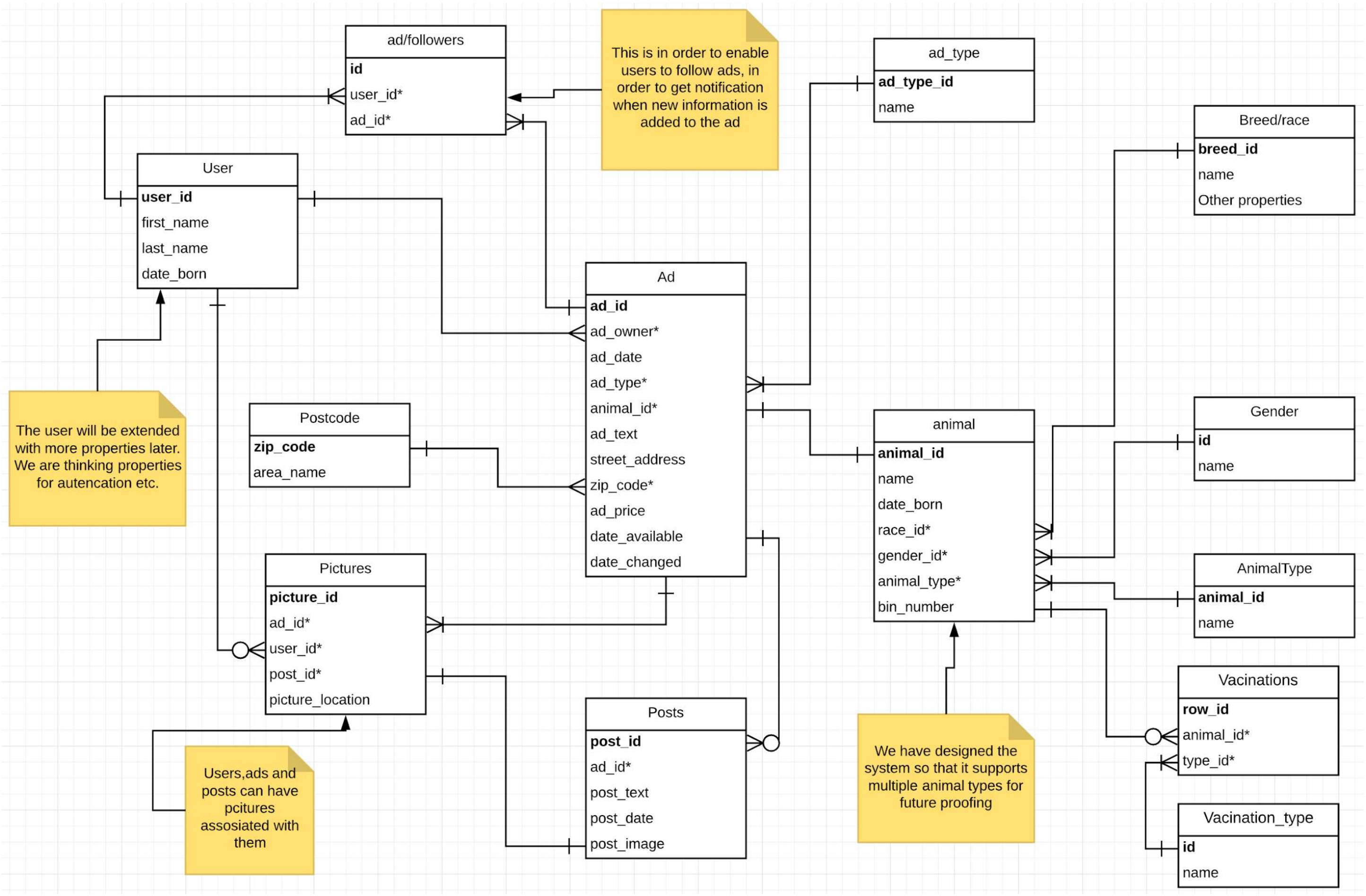
We have decided use a model-view-controller implementation in our structure. We will have three main types of classes.

- View classes (for example one per view in our app), this will be our user interface related classes. These will have no direct connection to our actual objects and object properties that we will display on the screen. They will be connected a controller which sends the data from our model (objects) to the view class.
- Controller classes will be responsible for transferring data from our objects/classes to the UI (views) and the other way around when submitting data from the view to Models.
- Our Model classes will be classes and instances of these will, contain the actual data in the application will use. For example when we have loaded content from our backend we will create objects of these classes which then in turn can be read by the Controller and transferred to the View.

### **Data model**

This is our data model for the application. It has been designed with upgradeability in mind. Here are some of the important design decisions we have made:

- In the future we want to have the ability to add new animal types. We will start with only dogs in mind. If you look at the animal class/table, you can see that it will support new animals just by adding data in AnimalType class/table.
- We have also enabled “follow an ad” functionality where users can follow ads in order to keep updated.
- We also support many “ad types”, this can range from adoption, sale etc.
- We will also be able to store relevant facts about the animal such as Vaccinations, gender, breed/race etc.
- We support posts on each ad in form of pictures. So if you make a ad, you can post images of the animal after the ad has been created.


ad/followers
id
user_id*
ad_id*
This is in order to enable users to follow ads, in order to get notification when new information is added to the ad
ad_type
ad_type_id
name
Breed/race
breed_id
name
Other properties
User
user_id
first_name
last_name
date_born
Ad
ad_id
ad_owner*
ad_date
ad_type*
animal_id*
ad_text
street_address
zip_code*
ad_price
date_available
date_changed
The user will be extended with more properties later. We are thinking properties for autencation etc.
Postcode
zip_code
area_name
animal
animal_id
name
date_born
race_id*
gender_id*
animal_type*
bin_number
Gender
id
name
AnimalType
animal_id
name
Pictures
picture_id
ad_id*
user_id*
post_id*
picture_location
Vacinations
row_id
animal_id*
type_id*
Posts
post_id
ad_id*
post_text
post_date
post_image
We have designed the system so that it supports multiple animal types for future proofing
Users,ads and posts can have pcitures assosiated with them
Vacination_type
id
name