

# WRPV302 2020 48-Hour Assignment

## Overview

You have from **8am on 21 October 2020, until 8am on 23 October** to complete and submit a solution to this assignment on the module's Learn site. The assignment covers topics from weeks 1-4.

Please read the *entire* assignment before starting to make sure that you fully understand what is required of you.

## Cheating & Plagiarism

This is *not* a group assignment, each individual must complete this assignment *on their own*. You may *not* submit others work (including Internet sources) as your own. At a maximum, you *may* copy 5 lines from an Internet source for the entire solution, but this must be referenced.

You will be required to complete a declaration prior to submitting your assignment. You *will not* be able to submit your assignment unless you make the declaration. **Anyone caught plagiarising another's work in this assignment will be submitted to the University's disciplinary procedure.**

## Submission

Prior to submitting, you will need to complete a declaration that you did not cheat or plagiarise anyone's work. Once the declaration has been made, you will be able to submit your solution.

When submitting your assignment, you must submit both the *project* and a compiled *APK* of the solution. These must be zipped into an archive and submitted on the module's Learn site.

Be sure to provide suitable comments in your code.

## MyPet Journal

You are required to write an Android app that will allow a pet owner to keep a journal on behalf of their pet<sup>1</sup>. The journal will be used to keep a record of *interesting* events in the pet's life. These include:

- going to the **vet**,
- **medication** that has been prescribed,
- **appointments** (play dates with other pets, hair appointment, ...) and
- **selfies** (pictures of the pet doing interesting stuff, such as skateboarding).



The minimum functionality of the app is described below, while Figure 1 shows some mock screen designs. Note that the app *only* needs to look correct on a mobile phone in portrait mode.

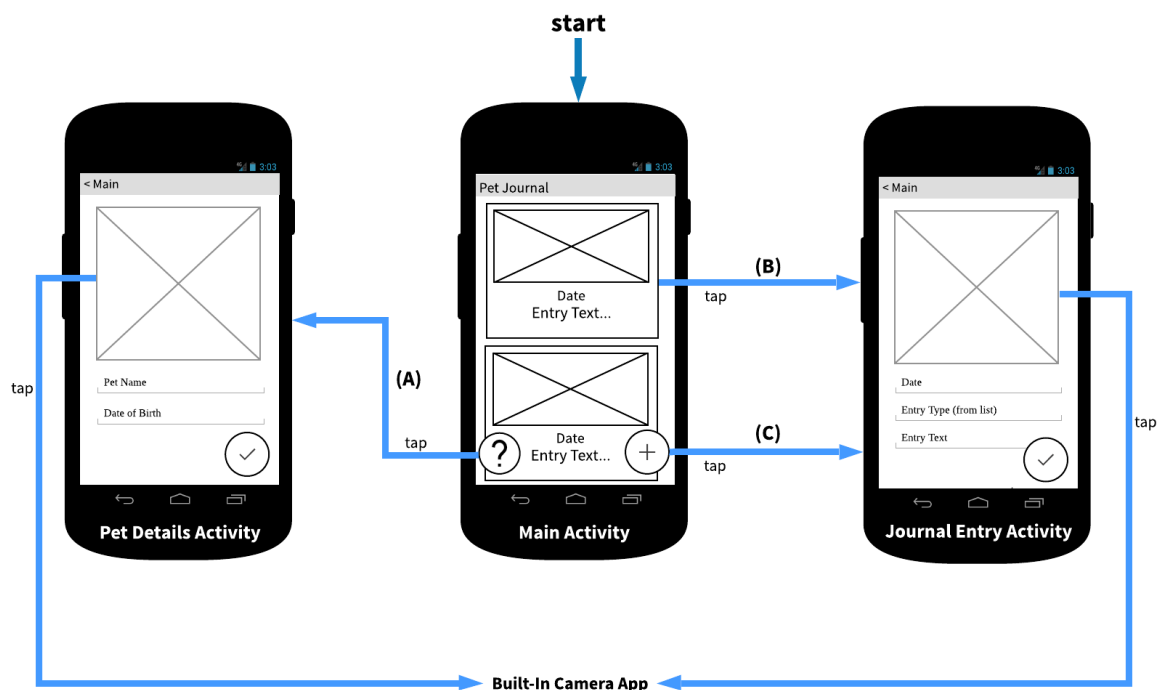


FIGURE 1: MOCK SCREEN DESIGNS<sup>2</sup>

### Main Activity

When the app starts, the user will be taken to the Main Activity. If this is the first time the app has been used and there are no pet details recorded yet, the user is automatically taken to the Pet Details Activity and requested to provide details of the pet (described below).

On the Main Activity, a recyclable list of cards displaying journal entries is shown. The entries are ordered from newest (at top) to oldest (at bottom).

<sup>1</sup> For this assignment, the assumption is that a person only has a single pet... A *real* app would have more pets... and yes, pet journals are a thing. Go look on the Play Store (search for "pet journal").

<sup>2</sup> Note that the screen designs are just suggestions. Your *actual* design may look different, but the UI design should be appropriate and usable (i.e. Lynette would be happy with your design).

Each entry displays the following:

- an image representing the entry;
- the date of the entry; and
- text describing the entry, but no more than 200 characters should be shown (the Journal Entry Activity will display *all* the text).

The cards are colour coded based on the type of entry. You may decide on the most appropriate colour to use for each of the entry types (vet visit, medication, appointment and selfie). The image that is displayed is an icon based on the type of entry. *If* a photo has been taken for the journal entry, then the image used will be a scaled down version of the photo instead. If the icon is used, then the image should be smaller, whereas if a photo was taken, then the image should be larger. You may download and use icons and images from Flat Icons<sup>3</sup> for this assignment.

Tapping on an entry's card (B) will take the user to the Journal Entry Activity with the selected entry's details filled in.

Tapping on the floating + button (C) will create a new entry and take the user to the Journal Entry Activity to fill in the details. The current date will be automatically filled in for the new entry.

Tapping on the floating ? button (A) will take the user to the Pet Details Activity and allow the user to view and edit the details.

The entries list may be themed as desired to be visually appealing. The Google Keep styling, shown in Figure 2 may be used as inspiration (because staggered lists are purdy).



FIGURE 2: GOOGLE KEEP

### Pet Details Activity

In the Pet Details Activity, the user can view and enter details of the pet. The user may take a photo of the pet by tapping on the pet's image (the box with the cross through it). This will take the user to the built-in camera app and allow the user to take a picture<sup>4</sup>. Once a photo has been taken, control will be returned to this app and the photo will be used.

You may use code provided on the Android Developer site<sup>5</sup> to assist you will accessing the built-in camera functionality – **note that this is the *only* code you are allowed to “plagiarise” for this assignment!** This code *does not* count towards the 5 lines of code allowed to be used from the Internet.

Tapping the floating ✓ button will save the details.

This Activity should be parented to the Main Activity to display the return option at the top of the Layout.

### Journal Entry Activity

In the Journal Entry Activity, the user can view and edit details of an entry. The user can select the type of entry from a *pre-defined* list (vet visit, medication, appointment and selfie)<sup>6</sup>. Changing the type

<sup>3</sup> <https://www.flaticon.com/>

<sup>4</sup> If you are using the emulator, you may connect a webcam and have the emulator use *this* as the camera.

<sup>5</sup> <https://developer.android.com/training/camera/photobasics>

<sup>6</sup> i.e. the user *does not* type the type of entry.

of entry will change the image displayed, *if* a photo has not been taken for this entry already. Icons representing specific entry types should be used by default (from Flat Icons) unless a photo has been taken.

Tapping on the image (the crossed-out box) will allow the user to take a photo using the built-in camera app and used as the image associated with this entry.

When a new entry is created, the current date is automatically assigned. The user can change this by entering another date. Only *valid* dates should be accepted.

Tapping the floating ✓ button will save the details.

This Activity should be parented to the Main Activity to display the return option at the top of the Layout.

### General

Appropriate confirmation and error messages should be used as needed.

The pet's details and all the entry details must be saved and loaded at appropriate times.

The app should work for *two* languages (English and one other).

Appropriate, consistent styling should be applied to the app.

Use comments, naming conventions, SOLID, etc. appropriately in your code.

## Rubric

The following rubric will be used when assessing submissions. This may change slightly once assessment begins, but will not differ majorly from what is specified below:

Criteria	Description	Weighting
<b>UI</b>	Good, consistent UI design of layouts throughout the app. This includes consistent theme and styles, well laid out, appropriate for mobile devices (specifically a phone).	10
<b>Resources</b>	Correct usage of string resources to allow multiple languages.	10
<b>Data &amp; Persistence</b>	Appropriate data structures used to represent the pet and journal entry data, etc. This data should be saved and loaded at appropriate times to ensure that data is not lost.	10
<b>Recyclable List</b>	Appropriate implementation of a recyclable list (and associated classes).	10
<b>Card Layout(s) and Functionality</b>	Card layout for a journal entry. Displays image (or thumbnail of photo), text limited to 200 characters. Tap functionality enabled. Smaller/larger image based on whether a photo was taken for the journal entry or not.	10
<b>Built-in Camera app used</b>	Used the built-in camera app to take photos. Use of code from the Android Developer website is allowed. The photos should be “remembered” when loading/saving the details of the journal and pet. Correctly sized images should be used.	10
<b>Navigation</b>	Multiple Activities and the ability to navigate between them without loss of context. Parameters passed in Intents as needed. (A), (B), (C) implemented, along with back navigation.	5
<b>Main Activity</b>	Functionality expected: <ul style="list-style-type: none"> <li>• Loading;</li> <li>• Initial behaviour (no pet set) implemented;</li> <li>• View/Edit Pet Details; and</li> <li>• View/Edit/Add Journal Entry.</li> </ul>	10
<b>Pet Details Activity</b>	Functionality expected: <ul style="list-style-type: none"> <li>• Loading;</li> <li>• Photo taking; and</li> <li>• Saving.</li> </ul>	10
<b>Journal Entry Activity</b>	Functionality expected: <ul style="list-style-type: none"> <li>• Loading;</li> <li>• Date setting and validation;</li> <li>• Photo taking; and</li> <li>• Saving.</li> </ul>	10
<b>Code</b>	Was appropriate internal documentation (comments, naming, etc.) used? Was the code SOLID?	5
<b>Total</b>		<b>100</b>