Coursework Report

Mark Pereira
40286471@napier.ac.uk
Edinburgh Napier University - Module Title SET08114)

1 Introduction

QuickImageNote (the current Prototype name) is an app that allows the user to quickly and efficiently add comment windows of text to any image from the Gallery. There is much scope to this app; Quite often, artists, designers or even photographers may have an image or design that they may want to demonstrate to another party, such as editors, and would probably need to add well positioned notes to showcase focused parts of the image, especially on the go. The regular procedure is quite daunting, involving use of generalized or complex software with unwanted features and a difficult UI. ImageNote will be a quick and easy on-the go app, where the task can be done in a matter of seconds.The task is quite simple; simply choose a scanned/photographed image, tap the areas to add comments to, and save.

I chose this app since I enjoy doing artwork, and would wanted a way to show my drawing sketches and ideas to friends and family. However, the problem was that all my drawings were on paper. Since I don't own a personal scanner (as with many amateur artists), this makes it difficult to scan and share images. This was solved with the advent of Scanner apps. However, phone scanners don't pick up non-inked (pencilled) written comments well, while inking the writing or using any other means means affecting the physical drawing. I was quite impressed with Microsoft Office Lens' scan quality, so I thought I could simply eliminate the problem of the text scanning with this simple app.

2 Modelling

The app was designed in a pseudo-agile development manner (adapted for a single person). Due to the extremely short duration of my development cycle (mentioned in my Personal Evaluation), set goals had to be efficiently allocated or else omitted towards a later version of the prototype. A Time frame was then allocated according to dynamically identified areas of difficulty This program was primarily modelled from basic Hand-Sketches of the User Interfaces (Figure 1). The primary aim of the prototype was to construct the underlying app framework under which to develop further. This meant focusing on how to conveniently develop further features rather than the development of the features themselves. This approach increases convenience during future programming, and makes the program more maintainable. Primary goals contemplated at the modelling stage were:
How to structure a Layered Canvas; This meant that each layer would be controlled by the user independent of each other.
How to manage the canvas. The Canvas would be handled by user inputs, and each layer would need to controlled on touch. Includes Add, Update and Delete functions for the drawn Images.
How to render the canvas. It was decided this would be done in a render based fashion, where object position data was virtual and the physical position would be determined towards the end of a declared update (Figure 2).

3 Implementation

Though quite different from the initial prototype, many elements of the underlying framework have been successfully constructed. The main focus of difficulty was in developing the Canvas. As with any Software development method, unexpected hurdles were encountered at this stage. In most cases, this led to the importing of external functions and use of tutorial websites and forums to help in development. Examples of advanced features implemented or adapted through external sources are: Conversion of ImageView coordinates to Bitmap coordinates, Merging of Bitmaps and Use of Animated image transitions (See References). Other skills learned at development time were managing of Bitmap object coordinate spaces and Dynamic Canvas Class structures. Working on the Canvas eventually took up approximately 80Remaining features were implemented relatively easily, or had to be omitted due to time constraints, such as the file browser, help images/text, application artwork, and saving the image to storage.

4 Critical Evaluation

4.1 Canvas Design vs. Execution

Many intended canvas features were dropped, since the canvas development time was highly underestimated. Some easy to implement features I dropped due to time constraints are a Font picker to edit text Colour and Size, and a Scale scroll bar since Rectangle size is currently set constant in relation to image size. These issues could have been solved by adding controls on the Canvas to edit the construction variables, but were avoided due to lack in personal development time. More complex features I initially intended were touch animated motion and re-sizing, and Touch based selection. Features were also intended to export the image.

4.2 Market Comparison/Evaluation

The app was designed and developed according to my personal requirements, and as such, no other specific app inspired this choice. Of course, there are numerous apps which promise the ability to add comments and more to a given image, a typical example being the Android Photoshop app, or Textgram, which is close in relation. However, when I used these apps, I found myself doing tasks more complex in relation to what was required, such as having to accurately scale objects and navigate complex menus and rulers are non User Friendly. Other apps simply add captions to the images in a stored format, or add 'fancier' effects at the cost of efficiency. QuickImageNote has only a few options regarding font and shapes. Adding a note takes little more time than the User takes to type it. This makes the app very pragmatic and efficient.

4.3 Possible Improvements

Depending on User feedback, more primitive shapes and colours and fonts can simply be added at maintenance if desired. If Users desire custom Fonts or Word Art, a stock Artwork Library may be considered. The Canvas UI is relatively light-weight, so has room for improvement based on User feedback.

5 Personal Evaluation

Before this coursework, I was not the most convinced that Android development would be as interesting as developing PC applications. However, the coursework has shown and convinced me that Android has undergone quality development with the advances of the Smart-phone market, and isn't as basic as I once assumed. Because I underestimated the scope of this Coursework, and conflicts with other Coursework, I brought upon myself a tight time constraint of 4 Days to build the app. Despite the long string of hours working on it, the experience certainly was interesting and memorable. This was a good opportunity to build upon my java skills, and certainly helped develop my skills working with 2D graphics, which is my personal area of interest. This really helped me, both as an artist and a programmer. I acknowledge that the final product is quite a ways away from what I expected, and though it seems incomplete, I feel that a couple weeks more of development time would have been enough to fit most intentions. In any case, I can at least acknowledge that the developed app is a good framework for future development.

6 References

6.1 Functions Adapted in Prototype:

Cross Fade Animation:
https://developer.android.com/training/animation/reveal-or-hide-view.htmlCrossfade

Overlay Bitmaps:
answered May 16 '12 at 10:41 RajaReddy PolamReddy
https://stackoverflow.com/questions/10616777/how-to-merge-to-two-bitmap-one-over-another

Convert ImageView Coordinates to Bitmap Coordinates:
answered Mar 30 '12 at 15:22 akonsu
https://stackoverflow.com/questions/4933612/how-to-convert-coordinates-of-the-image-view-to-the-coordinates-of-the-bitmap

6.2 Images Used:

Jellyfish:
https://androidhdwallpapers.com/apple-watch-wallpaper-jellyfish-art-nature-white/

Prescott Park:
https://www.portsmouthnh.com/listing/prescott-park-boat-dock/

Colosseum:
https://www.dreamstime.com/royalty-free-stock-photography-colloseum-rome-circa-tourists-most-visited-landmark-rome-december-rome-italy-image32943857

7 Design Sketches

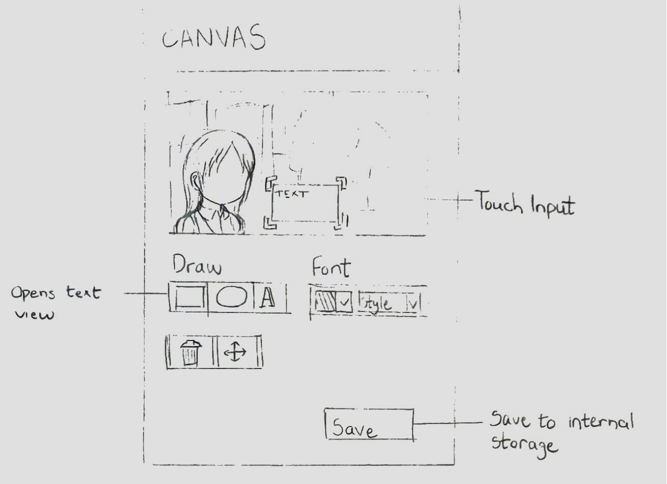


Figure 1: **Canvas Design** - Inital Canvas Design Concept

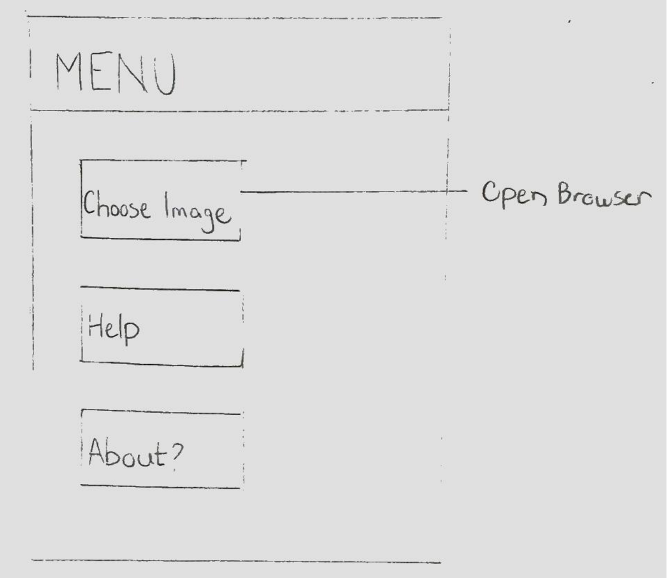


Figure 2: **Menu Design** - Initial Menu Design Concept

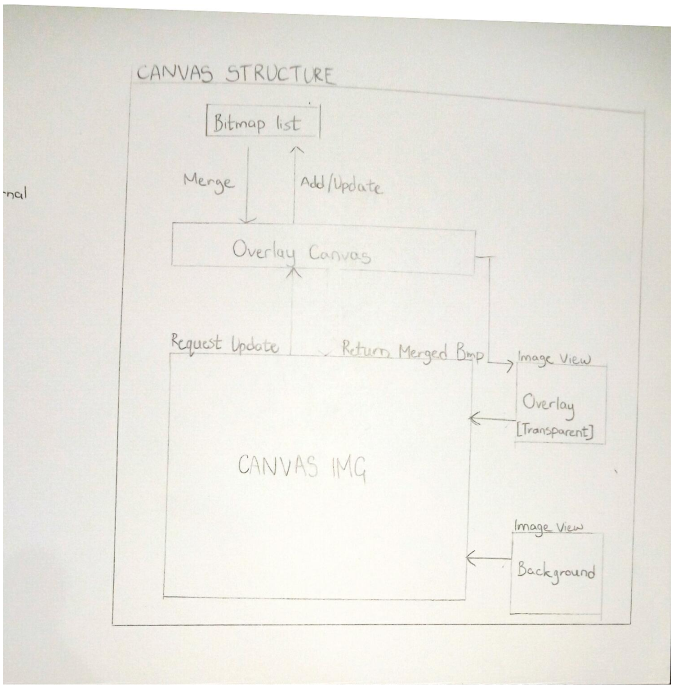
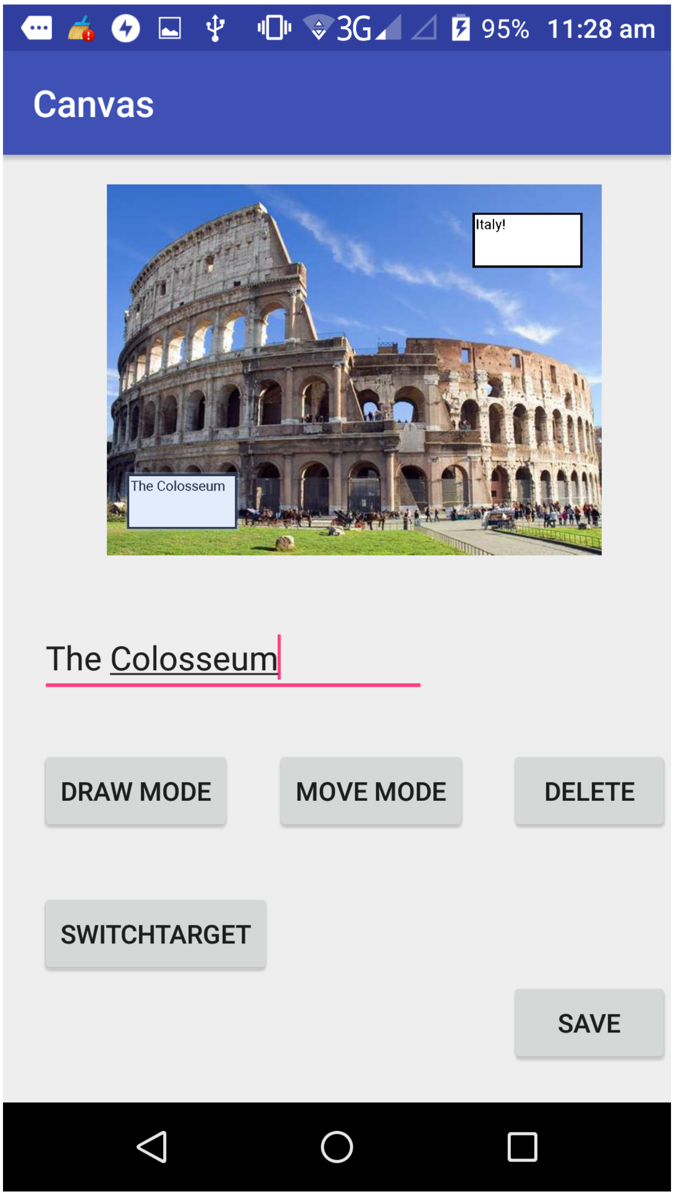


Figure 3: **Canvas Structure Design** - Initial Canvas Design Structure. This allows independent layering.



The Colosseum

Figure 4: **Canvas Structure Design** -App Canvas Screenshot