

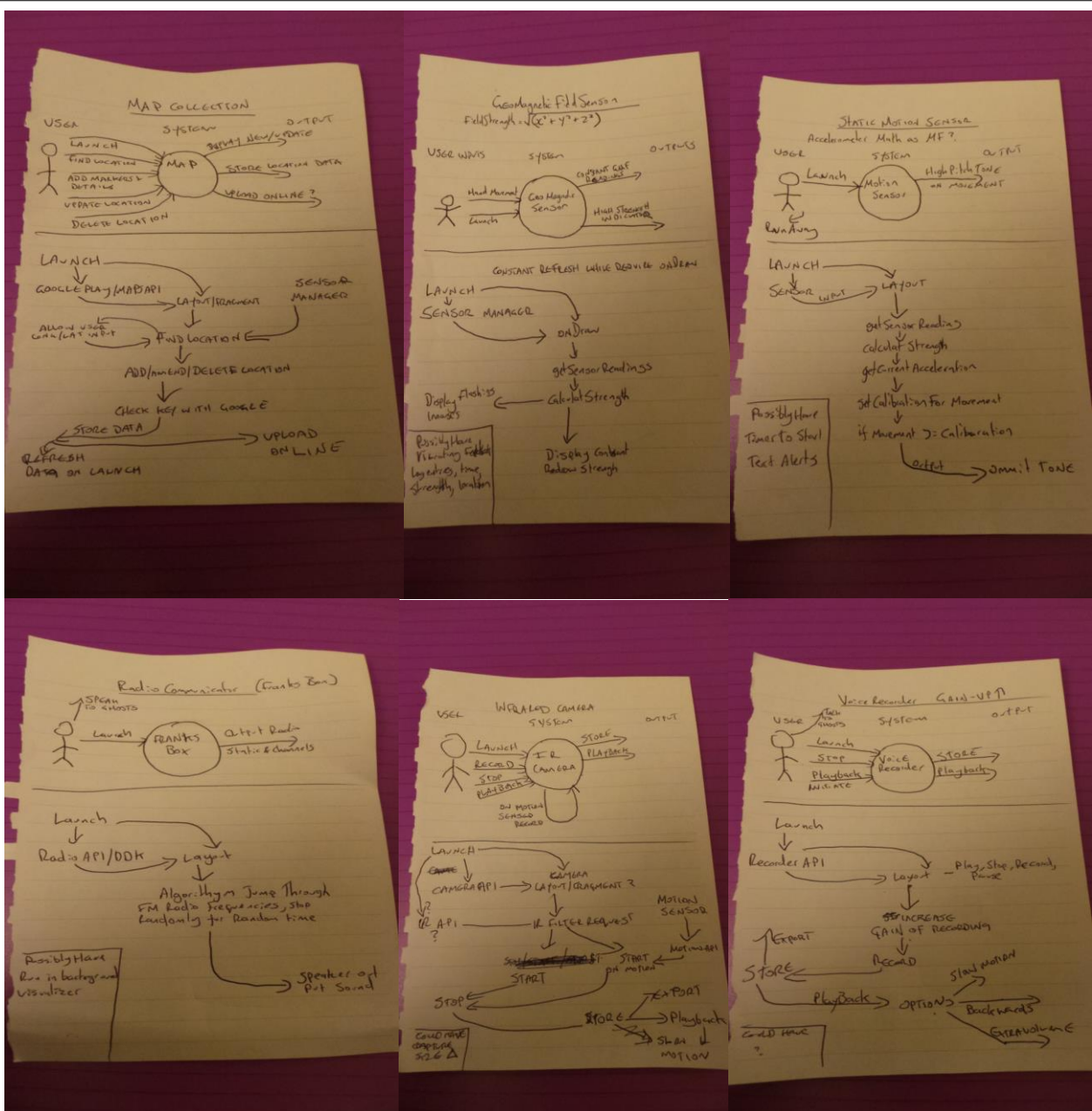
Introduction

I have been asked to research, design, implement and evaluate a prototype Android application. Below I will look at the inspirations I drew on to finalise my design and show the software modelling techniques I chose to begin my prototype. I will then give details of the final prototype I implemented and evaluate this against the initial idea, against similar Android applications and review my App to suggest possible improvements that I could have made.

Overview

The App I initially set out to build was a prototype of a multi-functioning ghost or spirit detector. I have used some similar Apps before for the Android system and always thought they could take more advantage of the platform. I am also interested in utilising available sensors for future projects. As there would be many different activity functions I could also apply Agile methods to its creation and build each section in sprints. From a purely layman perceptive of paranormal investigation I decided to design and implement; a Magnetic Field reader, a static motion detector, the ability to use the Android FM radio to jump between frequencies similar to a Franks Box, an Infra-red camera possibly with motion detection, a voice recorder with high gain and also a maps utility to record locations and information gathered. I read through the online Guides at Developer.android.com and [Using Device Sensor's in Android Wireless Application Development](#) to start my research on sensor management.

Software Design

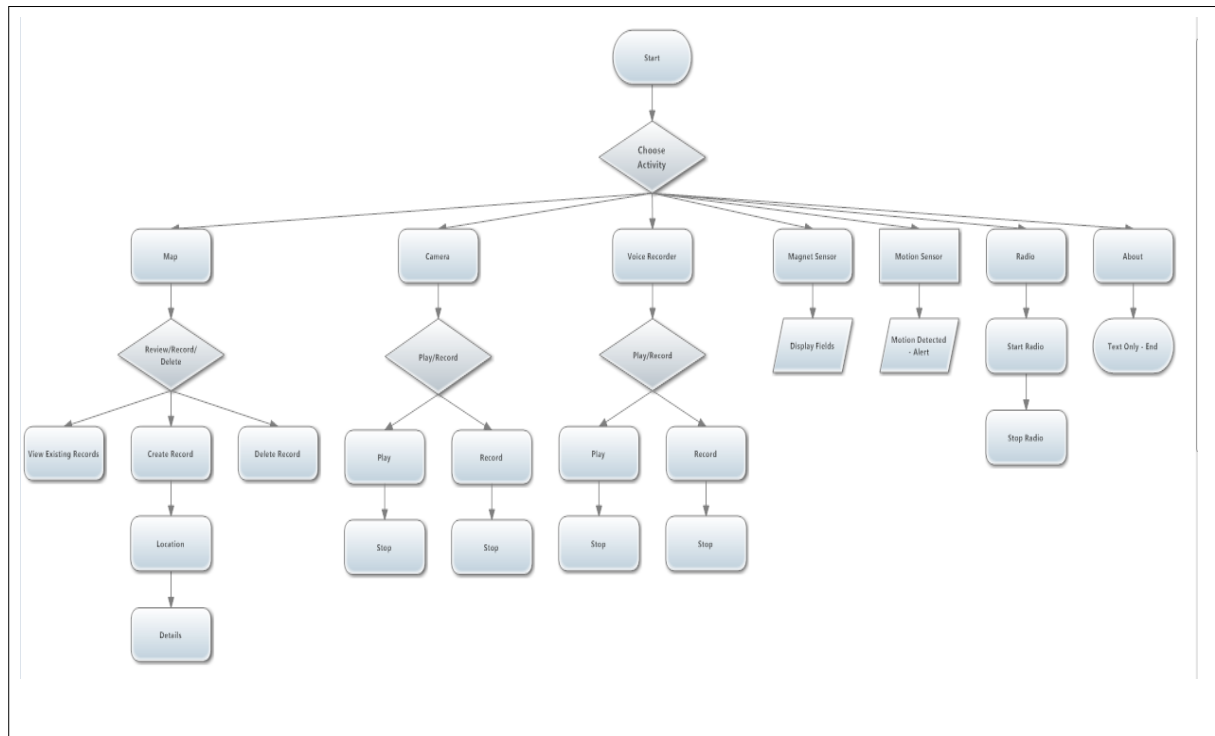


The image is a screenshot of the Kuler website, which is a platform for finding and creating color palettes. The interface includes a search bar, a list of color palettes, and a detailed view of a selected palette.

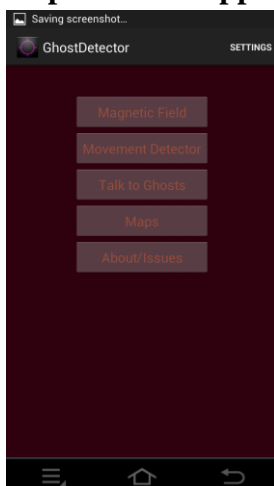
Color Palette Details:

- Base Color:** A color swatch with the hex code #3C3806.
- Color Selection Tools:** A color wheel and a color bar are visible, used for selecting colors.
- Color Data:** The palette includes several color swatches with their respective RGB, CMYK, and HEX values.

Color	RGB	CMYK	HEX
Base Color	60, 56, 6	0, 6, 90, 77	3C3806
Color 2	123, 12, 43	0, 90, 65, 52	7B0C2B
Color 3	343, 100, 18	0, 100, 72, 82	2F000D
Color 4	194, 90, 38	90, 22, 0, 62	0A4D62
Color 5	194, 95, 48	95, 23, 0, 52	065F7B

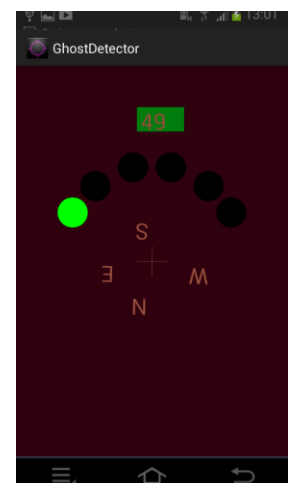


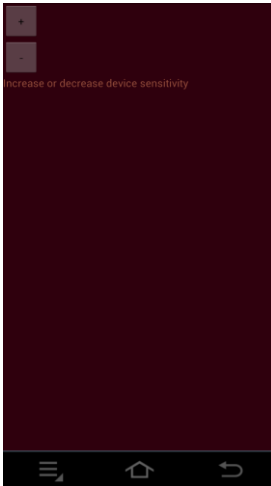
Implemented App Description



On implementation of my prototype App I completed 4 activities, not including the landing screen.

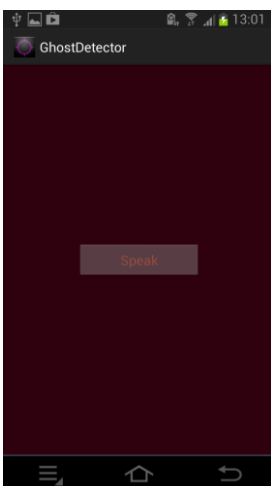
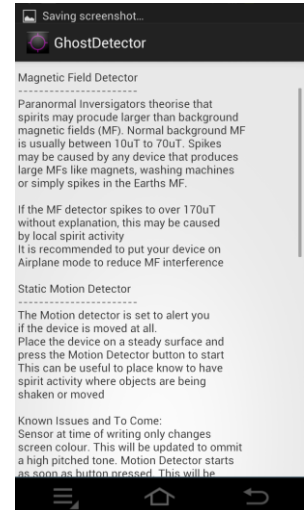
I first created a Magnetic Field reader. I used the resources at Samsung Developer of the SensorCompass to build my Magnetic Field sensor around. I then added the Magnetic Field sensor, calculations, readout and a Green, Yellow and Red light display to give easily seen feedback on the field strength.





My next Activity is a motion detector. I used a tutorial code from vogella.com for the main accelerometer and added buttons to increase and decrease the sensitivity. I have started but have not yet completed implementing the Tone Generators.

Another activity is a random word generator with Text-to-Speech capabilities. I used the Text-To-Speech code from jamesellsey.co.uk. I then added an array of 830 common English words, a couple of random statements and a button to press for user interaction to talk to spirits and then get a speech output. The array is currently in the class.



My final activity is using the Google Maps API which is signed and working in development mode. A side bar is available so that the user can input their latitude, longitude, location and details of their find. At this time this data is not able to be saved.

The overall design of the App is in dark colours. I selected this theme because if the App is to be used in the dark, a bright display could depreciate the user's vision for objects around them. After choosing my primary dark red colour I used Adobe Kuler to create a colour palette to complete the rest of the Apps theme.



Comparison against Original Concept

The final prototype of my App contains some features of my original concept, but also the design changed over development and other features were either not complete or not included.

Neither the infra-red camera nor the high gain voice recorder was included in my prototype.

The magnetic field sensor was completed with success and was my first sprint. It attempts to replicate the look of actual devices used with its illuminating light system. The movement sensor was also added successfully but lacked the feature of the Tone Generator.

The map application was completed to a basic level. Users are able to navigate the map and add location and details. The user has to manually add their latitude and longitude locations, and also the lack of ability to save the details.

Building a radio scanner was not possible for my test device. I decided to change direction and build a random word generator with text-to-speech output.

Comparison against Existing Apps

I feel that this App, even in its prototyping stage, stands up well in functionality against existing Apps. There are about 70 Apps available in the Play Store. Most of these seem to be based around a ghost radar, and also have a very fun cartoon style theme. Some examples of these are “Ghost Detector”, “Ghost Radar” and “Ghost Meter PKE Detector Free”.

Another common App is for pranks, where the user can set up ghost readings to fool other people. “Ghost Finder” and “Spirit Scanner” are of this type.

There are several Apps very similar to my prototype, and most of these are based on displaying the magnetic field strength. The most comprehensive and perhaps professional feeling App available is “EMF Detector”. This App concentrates on the Magnetic Field sensor and provides excellent user feedback. This App has less functionality than my prototype. However, the Magnetic Field display provides a better user experience. Due to its appearance it feels like a less like you are holding a phone, and more like an actual EM sensor. It also has more detail on the field strength user output.

User Feedback Evaluation

Immediately upon starting a user evaluation I realised that I could have problems with an evaluation after looking at potential user groups of the App. I started with building user groups of people who would install and use this App in most of its entirety, and also retain this App on their device period of at least six months. My user group was mostly limited to people 18-65, have an interest in the paranormal, are technologically knowledgeable and are active in paranormal investigations. I could also repeat this user group without the “active in paranormal investigation” but instead have a casual interest in the field. I then created three personas to understand if they would have any interest in the App. The results showed some passing interest but no real long term usage.

I decided to get user feedback from three people with different experiences. The first person, user 1, was very familiar with the topic and knew in theory how the system worked. The second and third users, user 2 and user 3, had absolutely no knowledge of the subject. I gave the users two tasks to complete and then observed their actions. For user 2, I provided a more detailed brief of the technologies being used. I then followed the observation up with interviews asking the same questions on usage, design, feedback and familiarity.

User 1 started methodically progressing through the App and using most of the functionality with ease. They struggled using the map as they did not know their latitude and longitude and quickly skipped this section. They used the Ghost Talk section at length.

I briefly described the functions to User 2, and asked them to read the about section in its entirety. After finishing the About section, User 2 successfully went through the App using all the functions including the map, but also spent more time playing with the Ghost Talk section. They had difficulties in navigating back to the main screen.

User 3 had issues understanding any of the functions. They quickly looked at the about section but did not read it, and went through all the functions without using them until they reached the Ghost Talk section. Again they used this on length for the fun factor.

After interviewing the Users I ascertained that the navigation system should be updated, as users not used to Android phones had issues going back to the landing screen, and they would also like to jump navigate directly to and from each section. They liked the colour scheme and thought it was suitable for purpose. They agreed that the map was not something they would use out of all the functions as it required too much user input and would require updating if they were to install this App. They all liked the Ghost Talk section and I got request to make this a separate App but have it make up whole sentences. User 3's feedback was almost, but not completely, limited to "It was weird".

Possible Improvements

My App will certainly require updates if I was to move this out of prototyping stage. The main update has to be the map section. This would require a long press on a location of the map where the user is, to be able to input details. It also requires at a minimum that this data can be saved onto a database either locally or online, and then load up when the map is launched. The map could also have a normal search for location function, and share online option.

The Ghost Talk activity requires pulling its information from a database or local dictionary and not having the information in an array in the class. It could also have more words, names and places.

Outside most of the minor issues in this prototype, if this app was to be used as a comprehensive paranormal device, I feel that completing my original actives of an IR camera with an external motion detector and voice recorder would be a major bonus to its usefulness in the field. Possibly adding more connectivity so more than one device could share information would also be compelling. If a small team of investigators had about ten android devices with a complete app installed, they would be able to record a detailed investigation from setting up IR camera zones, taking hand-held readings in specific locations to having a team member monitor all the activity remotely.

References and Resources

Conder, Shane, and Lauren Darcey. Android Wireless Application Development. Upper Saddle River, NJ: Addison-Wesley, 2010. Print.

Jones, Matt, and Gary Marsden. Mobile Interaction Design. Chichester, England: John Wiley & Sons, 2006. Print.

Lee, Wei-Meng. Beginning Android 4 Application Development. Indianapolis, IN: John Wiley & Sons, 2012. PDF

Elsey, James. "Android; A Really Easy Tutorial on How to Use Text To Speech (TTS) and How You Can Enter Text and Have It Spoken." James Elsey. Jameselsey.co.uk, 26 Feb. 2011. Web. 18 Mar. 2013. <<http://www.jameselsey.co.uk/blogs/techblog/android-a-really-easy-tutorial-on-how-to-use-text-to-speech-tts-and-how-you-can-enter-text-and-have-it-spoken/>>. - Resource - Code used in it's entirety. I left in the comments as they were also very helpful

Francisco. "OGDEN's BASIC ENGLISH (1000 Words) 830Â words." OGDEN's BASIC ENGLISH (1000 Words). Vocabulary.com, 25 Sept. 2012. Web. 18 Mar. 2013. <<http://www.vocabulary.com/lists/190098>>. - Resource – 830 Common English Words

"Kuler." Kuler. Adobe, n.d. Web. 18 Mar. 2013. <<https://kuler.adobe.com/>>.

"Mobisys Technology Blog." Â» Android. Mobisys Tech Blog, 12 Dec. 2012. Web. 18 Mar. 2013. <<http://mobisys.in/blog/tag/android-2/>>. – Resource - Used to setup Google Maps API only

"Sensors in Android | SAMSUNG Developers." Sensors in Android | SAMSUNG Developers. Samsung, 17 Sept. 2012. Web. 18 Mar. 2013. <<http://developer.samsung.com/android/technical-docs/Sensors-in-Android>>. – Resource - Sample Code used in it's entirety

"Storing and Searching for Data." Android Developers. Android, n.d. Web. 18 Mar. 2013. <<http://developer.android.com/training/search/search.html>>. – Resource - Not available to users at this time

Vogel, Lars. "Android Sensor - Tutorial." Android Sensor - Tutorial. Vogella.com, 07 Mar. 2012. Web. 18 Mar. 2013. <<http://www.vogella.com/articles/AndroidSensor/article.html>>. – Resource - Code used in it's entirety. Toast listener removed