

Christopher K. Hayes
B00692972
COM814: Project 2015-16

Dissertation

School of Computing & Information Engineering

Heart Sound Monitor Application

Supervisor: Janet Allison

Second Marker: Adrian Moore

1 September 2016

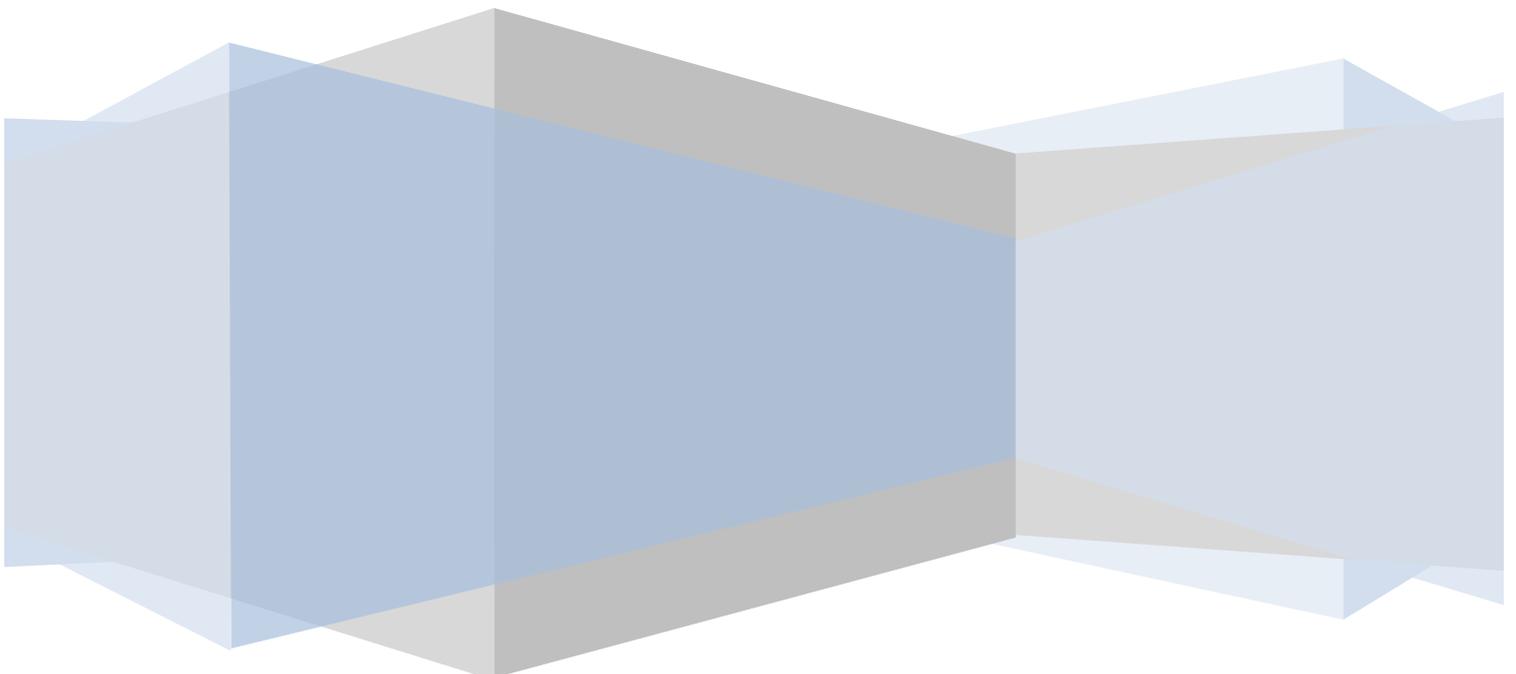


Table of Contents

Contents

| | |
|---|----|
| Abstract | 5 |
| 1. The Introduction | 6 |
| 1.1 <i>What is the problem?</i> | 6 |
| 1.2 <i>Rich Picture of Problem</i> | 7 |
| 1.3 <i>What are possible solutions?</i> | 8 |
| 1.4 <i>Potential Devices</i> | 8 |
| 1.5 <i>De-noising the Recordings</i> | 10 |
| 1.6 <i>Using a SQLite Database</i> | 10 |
| 1.7 <i>Rich Picture Diagram of the Solution</i> | 12 |
| 1.8 <i>Overall Expectation of the Project</i> | 13 |
| 2. Project Risks | 15 |
| 2.1 <i>Risks</i> | 15 |
| 2.2 <i>Ethical Considerations</i> | 16 |
| 3. The Analysis Chapter | 17 |
| 3.1 <i>Introduction</i> | 17 |
| 3.2 <i>Overall Description</i> | 19 |
| 3.3 <i>Functional Requirements</i> | 20 |
| 3.4 <i>User Characteristics</i> | 21 |
| 3.5 <i>Constraints</i> | 22 |
| 3.6 <i>Non-Functional Requirements</i> | 23 |
| 3.7 <i>Other Applications</i> | 25 |
| 4. The Design Chapter | 28 |
| 4.1 <i>User Interface and Interaction</i> | 29 |
| 4.2 <i>Software Architecture</i> | 35 |
| 4.2.1 <i>The Android Platform</i> | 35 |
| 4.2.2 <i>Application Architecture</i> | 38 |
| 4.3 <i>Database Design</i> | 39 |
| 4.4 <i>Overview</i> | 40 |
| 5. The Implementation Chapter | 41 |
| 5.1 <i>Various Designs</i> | 41 |
| 5.2 <i>Algorithms and Methods</i> | 42 |

| | | |
|---|---|----|
| 5.2.1 | <i>Database Implementation</i> | 42 |
| 5.2.2 | <i>Button Tapped (record, stop recording, playback)</i> | 44 |
| 5.2.3 | <i>Searchable Function</i> | 47 |
| 6. | Testing Chapter | 49 |
| 6.1 | <i>Introduction</i> | 49 |
| 6.2 | <i>Review</i> | 50 |
| 6.3 | <i>Overview</i> | 53 |
| 7. | Conclusion | 54 |
| 7.1 | <i>System Requirements</i> | 55 |
| 7.2 | <i>Future Developments</i> | 55 |
| 7.3 | <i>Summary</i> | 56 |
| Appendices | | 57 |
| <i>Definitions</i> | 57 | |
| <i>Entity Relationship Diagram (ER)</i> | 58 | |
| <i>List of Figures</i> | 59 | |
| <i>List of Tables</i> | 59 | |
| <i>Database Code</i> | 60 | |
| Bibliography | | 61 |

Dedication

I would first and foremost like to dedicate this work to Bronagh Kelly and my family, for their patience and support throughout this project.

I would also like to thank my supervisor, Janet Allison, for her guidance and priceless help throughout this project.

Plagiarism statement

“When submitting your work you are agreeing with the following statement: I declare that this is all my own work and that any material I have referred to has been accurately referenced. I have read the University’s policy on plagiarism and understand the definition of plagiarism. If it is shown that material has been plagiarised, or I have otherwise attempted to obtain an unfair advantage for myself or others, I understand that I may face sanctions in accordance with the policies and procedures of the University. A mark of zero may be awarded and the reason for that mark will be recorded on my file.”

Abstract

In resource-constrained environments, especially remote rural areas, diagnostic equipment leaves a lot to be desired. This has a particularly negative effect on those who are challenged with caring and diagnosing patients with certain medical issues, e.g. heart conditions. With the lack of equipment such as an ECG or a heart monitor, it could be extremely difficult to provide an accurate description of the medical issue, which would in turn create complications when diagnosing/caring for the patient. To obviate this issue, we can make use of widely available Android based smart phones to assist experts like practice nurses to get an initial diagnosis.

This application proposes a method in which medical staff can understand and facilitate the needs of the patients, but also assist doctors to help their patients in various ways. First and foremost, this project will feature a way in which the cardiologist can listen and monitor the sounds that the heart is producing. Using the microphone which every smartphone has, the application will utilise the sensitivity of this for recording such sounds, allowing the user to playback the audio recorded. In addition, this application will give the user the ability to compare heart sounds from previous dates, giving them a better comprehension of whether the patient's condition has improved or worsened. This is done by storing the patient information into a SQLite database. The benefits of this would be to enable the user to list the various types of data about the patient (ID, Name, Date of Birth, etc.), and also store the various recordings for future playback for that specific patient.

With the completion of the application, a new and improved version will assist the cardiologists/medical staff/users to provide more accurate diagnosis for their patients, including the ability to store all the heart records for their patients. With each recording, the user is able to save a comment stating the condition of the heart at that moment in time. The project itself will no doubt improve the resource-constrained environments in which cardiologists work in today.

1. The Introduction

In particular, we will focus on the use of the audio channel to determine heart rate (HR) and heart rate variability (HRV), to record the heart sound in real time in order to assist cardiologists for heart disease diagnoses. However, there is an issue with the actual recording of the sound as the heart beat is too faint for a normal sound recorder. As such, a solution would need to be achieved so that the recording is clear and accurate in order to facilitate the user when carrying out the task.

HR and HRV can be accurately assessed from acoustic recordings of the heart using only a smart phone or hands-free kit. This heart sound analysis program, which can run on a smartphone will be able to detect S1 heart sounds with a reasonable sensitivity and positive productivity. Evaluation of data recorded from smart phone demonstrates that the low-frequency response (<100 Hz) is key to the success of heart sound analysis on smart phones.

1.1 *What is the problem?*

As the 21st Century is well underway, the progression of technology is evolving constantly, it's on the rise and does not look like it will slow down anytime soon. Each year (or even month), there are more and more technological designs which continue to be succeeded by more advancements in the field. These include many different categories such as; applications, smart watches, computer hardware/software, etc. In the world of mobile technology, there are hundreds of new applications designed every day, which continue to push the frontier, with each giving more towards choice and innovation in the software industry. Companies such as Apple, Samsung and Microsoft all have their own software in which anyone can create an application. Samsung, in particular, uses the Android operating system and therefore the software Android Studio is an extremely useful tool in which you can design, build and add to the PlayStore (the market place where applications, games and tools that are uploaded for any user to download). Depending on what a user is looking for, there are thousands of options from which to choose.

This particular project will focus on the Health & Fitness category, as the application will be revised, and research will be designed to record the heartbeat of a patient/user. This application is designed to assist users (such as practice nurses), for those who visit more

rural areas in which equipment is needed to make a diagnoses when time is an invaluable factor in the medical sector. For a trained cardiologist/doctor, the vibration which resonates from the heart, can be vital in making a diagnosis. The heartbeat can provide a great deal of information about the health of a person and to the trained ear, it may be the difference between life and death. If caught early, heart disease can be treated, if a doctor is able to accurately hear irregular rhythms or vibrations they may be able to catch such illnesses early enough to save someone's life. On that note, the application must be accurate ... there is no room for error. Therefore, research is a key factor in the success of the application. While anyone from any background will be able to download the application, it will be directed at those in the medical profession. Additionally, anyone from a fitness based background could also use to application to record sounds of their heart while training or working out to aid in their performance.

The use of the audio channel to determine heart rate (HR) and heart rate variability (HRV), to analyse the recorded heart sound in real time could assist cardiologists for the diagnosing of heart disease.

One issue which could cause an inaccurate recording would be the background noise. The microphone or recording device would automatically pick out any white noise which would be heard over the important information which would be most valuable to the user. The Figure 1 illustrates the problem that this application will have to overcome in order to be successful.

1.2 Rich Picture of Problem

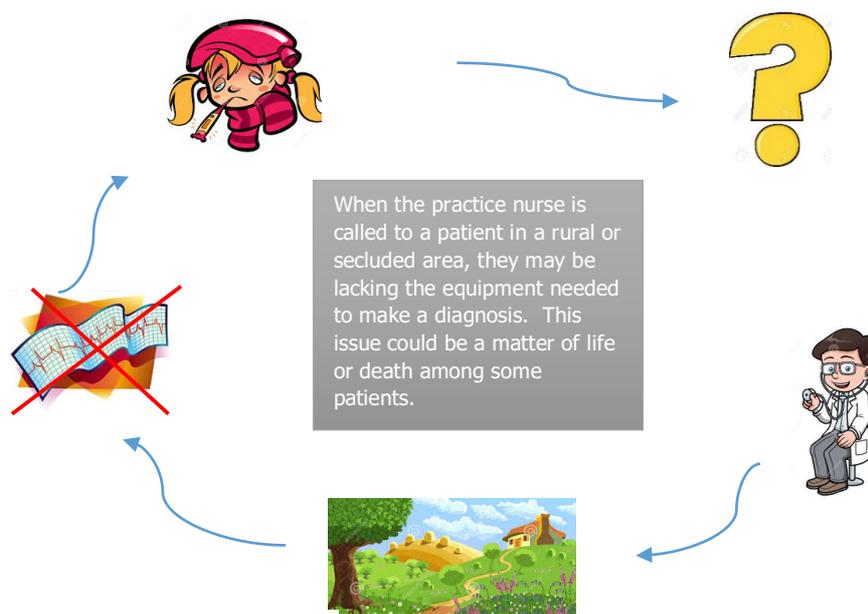


Figure 1: Rich Picture of the Problem

1.3 What are possible solutions?

The sound of the heart can give much information about a patient. When heard by the trained ear of a cardiologist or doctor, the heartbeat can tell whether someone is healthy or unhealthy (in terms of heart disease or other heart related issues such as high blood pressure). Heart rate variability is directly related to the body's interdependent regulatory systems and ultimately, their efficiency and health. "An optimal level of HRV within an organism reflects healthy function and an inherent self-regulatory capacity, adaptability, or resilience," McCraty and Shaffer write' (HeartMath, 2015). A recent article in the Global Advances in Health and Medicine provides much needed information about the heart and the Heart Rate Variability (HRV). This is extremely useful for cardiologist and those whose field involve the heart itself. It states, 'HRV began to be recognised for its importance in indicating or predicting various risk factors. Among these are fatal distress, autonomic nervous system dysfunction, heart disease, anxiety, depression and asthma among other health conditions' (HeartMath, 2015).

With the purpose of the application to detect the S1 heart sounds, a complete understanding of what exactly these are is required. This S1 is the first sound that the heart gives off. It is caused by the abrupt closure of two valves within the heard – mitral and the tricuspid valves (these will be the M1 and T1 respectively). With the latter being much fainter than the first, the M1 produces the sound which is commonly heart with S1 heart sound. Although there is a **slight** delay in the T1 sound, these two valves produce the sounds which are almost simultaneous to each other. However, to the trained ear and with the appropriate equipment, the difference can sometimes be detected. Moreover, each of these sounds have different positions where they can be heard best – M1 is the loudest at the apex and T1 is most prominent at the left lower sternal border (Lome, 2016).

1.4 Potential Devices

The benefits for using this device are abundant. If it has the ability to record the slightest vibrations of the heart when held up to the chest, it will allow the data which is being recorded to be saved and analysed directly to/from the mobile itself. One problem which would have come from using other, less accurate microphone recorders would result in the data having to be converted from analogue to digital sound. However, as this microphone can be attached using the headphone jack to the mobile device, the application

would be able to analyse the material recorded. Moreover, the cardiologist would further benefit from the quick, more accurate results which follow. The adapter would provide a clearer sound for the microphone to record, resulting in more accurate findings. Aquarian Audio believes that this “outperforms most contact mics at very low frequencies. Use for sound effects, leak detection, surveillance, terrestrial studies, sporting events, or general phonography” (Aquarian, 2015)

After researching a number of blogging websites, various solutions were offered. These included using multiple contact mics, with some more promising than others. A number of the contact microphone recorders were designed for musical instrument use. This suggests that they would not have the capability to record such faint sounds (Parkinson, 2012). In addition, I contacted a musical microphone seller on Amazon. They simply said that in order for these to work, I would have to record the heartbeat in a sound proof (quiet room). For such requirements, it is unrealistic for those who would be using this application. The surroundings would obviously vary from user to user, and it would be impractical to require a soundproof room for the application to have a chance of success (and even then, it would not be guaranteed success).

Another option, which has the possibility of successful recordings, would be a digital stethoscope. After some research, they appear to provide accurate and sensitive readings. There are a number of models that provide clear readings, and have the capability to link with a set of headphones. Furthermore, they are able to change between the frequencies of individual readings. This particular feature would be extremely useful as each patient that the cardiologist/user would examine may require a different frequency.

However, this direction in terms of recording the sound of the heart for users would certainly have greater cost implications in terms of equipment. A model such as the ThinkLabs – Digital Stethoscope, is priced at \$499 (which is approximately £340). For this reason, it would be too expensive for those who wish to simply record the sound of their heart as precautionary/experimental reasons. In addition, for a user such as a practice nurse, who travels to more rural areas, this piece of equipment certainly has many beneficial features, however, it could prove to be too costly for them or their local hospital department (it would be doubtful that they would be given the expenses to cover such a substantial financial investment for the device), although this could be explored.

The simplest and most cost-effective option would be to use the actual smartphone as a device for recording the heart sounds. This would only be possible if the microphone

has the sensitivity to pick up the faint sounds of the heart. Testing will need to be carried out to see whether the android smartphone would be capable of this. If it is, the application would ultimately be much more accessible, portable and inexpensive for the users to utilise the app.

1.5 De-noising the Recordings

Due to the faint sound of the heart, the recording device would be required to pick up the sound with accuracy. The sound recorder on Android devices is able to pick up a light sound of the heart. However, the background noise is too heavy on-top of the recording. As a result, the white noise would need to be removed – giving a clear recording which will be essential to users.

The next step in processing the recording would be to cancel out any excess/background noise. This unwanted noise makes the actual recording 'fuzzy', leaving the user to hear an unclear recording. As a result of this, the user would have a much less reliable and accurate recording of the heartbeat, rendering it harder for the user to assist in any heart disease diagnose. After getting in touch with a contact-microphone seller online, I was informed that some of the microphones that are on the market would not have the quality needed for such a task. This would render these impractical, as the use of a proper microphone would need to be used in a room where the background noise levels would not be high – yet be able to record the faintest sound. If the data could be collected, then the recording would then need to be filtered for the user to listen to the heartbeat alone.

1.6 Using a SQLite Database

One of the key features, which would add an advantage over a number of similar applications, would be the utilisation of a SQLite database within the app. The purpose of such a database would be to ensure that the user is able to store multiple audio recordings for later playback.

As a consequence, the cardiologist would be able to compare previous recordings to monitor the condition of the patients – as they could be able to tell whether their condition is deteriorating or improving. Furthermore, if a link between a recording and a patient could

be made, then a smooth transition between adding a new patient and listening to previous recordings for an existing patient would most certainly increase the productivity of the application. On the other hand, there could be issues with converting the audio recordings into a file that could be saved into the database, and also being able to play it back after it has been converted. These records could also be uploaded to another computer.

Finally, the overall design of the database is vital for ensuring the functionality of this feature. With a number of relationships involved, it would be necessary to create tables which are linked with one key (e.g. Patient ID). The Patient ID would be the foreign key on one table and the primary key on another, providing the much needed connection, which would allow for full utilisation of the database within the application. This SQLite database will be discussed in detail at a later point of the report. The Entity-Relationship Diagram in the Appendix illustrates the Entity Relationship diagram that the database will be constructed around.

1.7 Rich Picture Diagram of the Solution

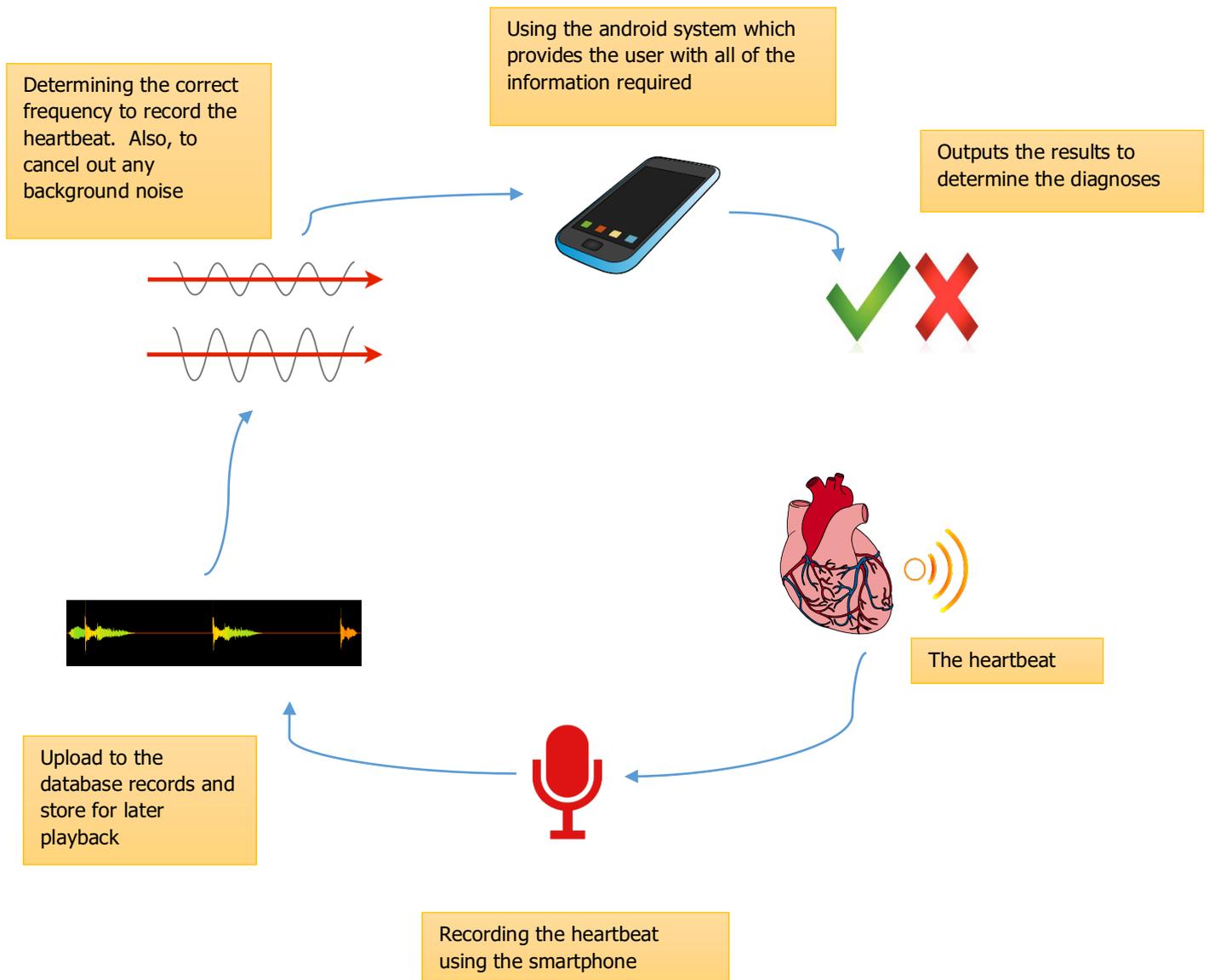


Figure 2: Rich Picture of the Solution

1.8 Overall Expectation of the Project

A thorough design is needed for such a project and with the issues that have been previously discussed, feasible solutions are required. With careful planning and experimentation, I believe that this project can bring new and exciting features to this particular area of study. If the issues of recording the vibrations of the chest, de-noising, and transferring the sound between analogue/digital, can all be resolved; then this new heart-monitoring application could provide a great deal of assistance to those cardiologists who require the equipment in more secluded areas of the country. That being said, it is fundamental that accuracy and reliability are vital to this project. Without these two core principles, it would be extremely difficult for any practitioner to make a proper diagnosis. With consideration to all of these factors, the likely overall achievement will depend on a number of elements:

- I) **Accuracy** – The application has the ability to record the sound of the heart accurately, and without any interference. This is extremely important because with even the slightest piece of interference, the authenticity of the data will be compromised and another reading will have to be taken. As such, a cardiologist will need to be able to hear the recording clearly.
- II) **User Friendly** – The ability for the user to utilise the application to its full potential and easily navigate through each page/screen is another vital facet on the overall achievement. Provided that the application is appropriately designed, the operator should have no issues using the application; which will result in greater accomplishment with relation to the overall aim. With this element, the equipment which is employed with the app would also be required to have ease of use. If the microphone or recording device was extremely complicated to use, then the overall system would not be as accurate or reliable as it is needed to be.
- III) **Multi-device compatibility** – With android, there are countless system which use this particular operating system such as; Samsung, Sony, HTC, among others. With this in mind, this application will need to not only be functional through different smartphones, but also different devices (i.e. tablets). This would ultimately allow a wider range of users to utilise the application, and in-turn, optimising opportunities for it to reach its full potential.

- IV) **Readily Available & Dependable** – Whenever the project is complete and ready to use, for it to continue to develop, there would need to be a level of dependability. In order for this to happen, it would have to be readily available for all those in the field to access it. These two factors are directly linked; as it stands available among the professionals, the dependability would need to be beyond question. As a result of this, more and more would download and operate the application to continue to drive the overall achievement of the success of the app and the project itself.
- V) **Data File Commonality** – files must be of the same format that is common and used within windows and other operating systems.
- VI) **Data Protection** – All patient information must be protected by the Data Protection Act of 1998. This means that all sensitive information must be secure as the information kept about patients is extremely sensitive.

2. Project Risks

2.1 Risks

The following table shows the risks that would threaten the success of the overall success of the project, in terms of management and research. The scale is from 0-5 (0 = no threat at all, 5 = the highest level of threat).

| Risk | Chance of risk | Threat | How to counter the threat posed |
|------------------------------|-----------------------|---------------|---|
| Time Management | 2 | 4 | Time management is key. Anticipating deadlines in advance will make the project run more smoothly than having a rushed plan of action. |
| Unforeseen absence | 1 | 3 | This risk is one that sometimes cannot be anticipated. Due to this, the threat is posed as quite high. |
| Lack of research | 2 | 4 | Keeping up to date with the new research that is published is vital. Additionally, making sure that the published research is properly revised. |
| Poor choice equipment | 2.5 | 5 | An issue that could cause a great deal of problems. Acquiring the proper equipment should counter this threat. |

Table 1: Risks and threat with the general management of the project

Table 2 shows the technical risks/threats that are posed to the over-all achievement of the project. The scale is from 0-5 (0 = no threat at all, 5 = the highest level of threat).

| Risk | Chance of risk | Threat | How to counter threat posed |
|---|-----------------------|---------------|--|
| Data loss | 2 | 5 | In order to counter this threat, multiple copies of the work will need to be saved. This means having numerous back-ups. |
| Technical issues | 2 | 3 | Constantly keep-to-date with software updates and repeatedly test the equipment. |
| Software-updates (version control) | 3 | 4 | Take extreme care with the coding and certify there are no issues with updates or coding. The reason for this is new software versions always need tested. |
| Issues with the Android Code | 3 | 3 | When coding, continuously try out the updated version of the project to make sure it is working correctly and as expected. Keeping a notepad with any software issues will also counter this risk. |

Table 2: Issues which could stem from technical difficulties

2.2 Ethical Considerations

This particular project is classed as a category Z, and therefore no evasive procedures are to be undertaken throughout the duration of the project. This included giving a diagnosis to a test heartbeat (which would cause unnecessary stress on that subject).

3. The Analysis Chapter

3.1 Introduction

Throughout the world, hundreds of thousands of patients a year suffer from heart related problems. In 2005, throughout the United States, there was a total of approximately \$400 billion spent in relation to cardiovascular issues. That was a colossal amount of money which has the potential to be reduced. Valarie Gay argues that the prospect of a smartphone application which has a hands-free kit would provide not only patients, but also those in the medical profession, a great deal of assistance in addition to the normal data as pulse and breath rate, when it comes to cardiovascular illnesses. In addition, for a traditional heart-monitoring device such as the ECG or Holter Machine, the patient would have the data recorded for a certain amount of time, while it would take 24-48 hours for a cardiologist to review the recordings of the heart sound (Valerie Gay, 2007). With this in mind, the purpose of this project is the use of the audio channel to monitor heart rate (HR) and heart rate variability (HRV) and analyse the recorded heart sound in real time in order to assist cardiologists for heart disease diagnoses. There is no question about the resources available to the national hospitals (and even private health care), however, sometimes these resources are very much stretched. This situation could mean that cardiologists/doctors may not have the equipment or time to review a patient's S1 heart sounds properly, especially if they are called to a rural environment (where resources are certainly not as readily available).

As a result of these findings, the demand for more equipment to assist those in making an initial diagnosis would be welcomed with open arms. The audience which this project is directed at would be those who are trained and experienced in the medical, and particularly, in the field of cardiology. If successful, medical staff such as practice nurses, who would travel or work in a more secluded area, or those who lack the sufficient resources would certainly benefit from this application – which provides a service that would aid in making any initial diagnoses. Moreover, the equipment itself would be hands-free and much easier to carry than a Holter Machine. The reading would be recorded using a microphone/recording device which would have the sensitivity to pick up the S1 sounds that the heart gives off. The device would be placed at the apex of the chest to maximise the readings and therefore give more accurate results.

Furthermore, the application should have the capability to cancel out any background noise that is picked up during each recording. This feature should be key as without a clear reading; the cardiologist would struggle to make a confident initial diagnosis. After much research, and many different options which deal with this specific part of the project, I believe that the **NoiseSuppressor** class in android studio would provide great assistance when reviewing the heart sound. As the cardiologist records the data in a calm and less-noisy environment (such as a house), this class should provide that extra bit of clarity when the user plays the recordings back.

However, even though this app could provide a great deal of assistance to those who need it (cardiologists who lack the resources and also patients who require the fast and accurate results to help resolve their slow diagnosis), it will not give a specific diagnosis. This means that without the trained ear of the doctor or nurse, this application will not state whether heart disease is a risk for a patient – it will need a practitioner in the cardiovascular field to make a prognosis. In addition, other patient data can be used in conjunction with currently accepted patient monitoring data. This will be made clear during the development and use of the app. On the other hand, an advice booklet will be made available to aid the user and supplement his knowledge base. For example, *Symptoms* will be described which will give further information on what to look for when reviewing the data from a patient's heart. The most common of which would be chest pain and discomfort in the chest area – dizziness, burning, aching and heaviness are all included factors (WebMD, 2015).

3.2 Overall Description

With reference to the market of heart monitoring in the smartphone network, there are countless applications which are similar in design and objectives. As previously mentioned earlier in the report, there are applications which are able to pick up the pulse of the heart and monitor it with great precision. This is done by placing the finger over the flash which is attached to the camera on the back of the smartphone. With each pulse, the small skin particles change ever so slightly and the quality of the cameras on phones in recent years have the ability to pick up on such small changes. However, this application will not be as straightforward in using the pulse. The basis for success would be to place a device on the apex of the chest and record the heart sounds which are given off from the S1 sounds. Therefore, using the pulse in the finger will not be practical on this occasion and although the overall goal is the similar, the application will attain results in a very different manner. In addition, there are some features which will be common for the development. One such aspect will be the ability to save recordings to review at a later stage. This will be discussed when the product's characteristics will be analysed. In relation to other systems, this application brings a new opportunity to the table. Although, there are a number of applications which use mobile stethoscopes such as the 3M Littman Electronic Stethoscope, these devices cost around £400 and with the restriction of budgets throughout the Health Service, such devices are of reach, financially speaking. Yet with the ability to record the heart sounds at such high standards, there is the market for another device which holds the same reliability, mobility and accuracy. This opening could be filled by this Android Heart-Monitoring application. With the combination of such features such as those which were discussed in *Existing Applications* and new/fresh aspects, prove this application will be proven to be a unique and manageable resource for the cardiological specialists.

After reviewing the potential option for a device which has the ability to pick up the sensitivity for the heart sounds, the reality is that a Samsung phone is capable to do such a task. After comparing the results found with an iPhone, the sensitivity is not quite as high a quality. However, the Samsung microphone was able to pick up the heart sound S1. Moreover, with the use of headphones, the sound quality was heightened, and this is extremely promising as the application will enable the sound to be de-noised (cancel out the background noise).

Finally, building a database which can store any audio recordings will further give the application the ability in providing users with a more unique and functional experience. With

other applications, some allow the user to store audio for later use, while others do not. However, there are a number of ways to achieve this. The first would be to have a database which stores the audio files, and the second would be to store them in an online account – both are feasible. Taking both of these options into consideration, it would be more practical for the user to store the recordings in a database. For example, if a cardiologist is in a rural area which has poor or no internet connection, then they format.

After analysing the functions and features that the existing applications have, and merging them with some distinctive new aspects, this application will prove advantageous when making an initial diagnosis for heart disease and other cardiovascular issues. The following information is a detailed list of the product functions which will be included within the application.

3.3 Functional Requirements

- I) **Heart Monitoring** – On the initial start-up of the application, the user will connect the device with the smartphone. Before the user begins the recording, they will have the option of selecting the time period in which they want to record. This variety of time periods will allow the user to record the data for the time required. The default time period will be set at 45 seconds to maximise the HR and the HRV of the patient. Once the recording is complete, it will ask the user whether they wish to listen back to the data or save it. This function will be useful for the user to listen back to a recording if they are not sure on the diagnosis or whether they want to save it for another time.
- II) **Cancelling out background noise** – As previously mentioned, the NoiseSuppressor class will be of great use for this application. This class 'removes the background noise from the captured signal'. Once attached to the audio file, the process of clearing up the recording will take place. The results of this function will give the cardiologist a much clearer recording of the heart sound, and in-turn, will provide them with more reliable and accurate data (the clearer the sound of the heart is, the easier it is for the user to make an initial diagnosis).
- III) **Pausing and listening back** – The specific function will allow the user to stop during a recording and listen back to a certain section of the data. If they

believe that they have heard something irregular, it will give them the opportunity to review the material which will further assist them in a diagnosis.

- IV) **GPS service** – A Global Positioning System may be included within the application. This would allow the user to save the data and also allow them to see when/where they made the recording for the patient. In a rural area, this would be a beneficial function as they would be able to recall the location of where they collected the heart sound data.

3.4 *User Characteristics*

- I) **Additional information** – This function will include some information about heart disease and symptoms to be aware of. For experienced cardiologists this material should be basic for their capabilities. However, for the new and inexperienced user, this function will help them to learn and grow within their field.
- II) **Saving the data for specific patients** - The reason why this function will be of use for any user, is because it will allow them to listen to a recording made at a previous date. It will also give them the information to compare a number of recordings that they may need when making an initial diagnosis. Moreover, this will give the application the organisation that it requires, in order to make it a success for the user. This will be done by saving the audio recording as an audio file on the device. As previously mentioned earlier, the file format is also vital. This would allow for transfer between multiple systems, which would further increase the user's ability to make an accurate diagnosis.
- III) **Option for User login** – A characteristic like this will give the application a more personal feel. With a nurse, they will need to be able to keep the information private (which is required for data protection legislation). Therefore, an individual Patient ID should be given to each patient, in order for the user to uniquely identify their recordings.
- IV) **SQLite Database** – A further feature that would be extremely necessary towards a smooth, functional application; would be the use of a database to store the audio recordings. With this database, the user would be able to listen to previous recordings and compare the sound of the heart in order to make a more up-to-date diagnosis, i.e. whether to patient is improving or not.

3.5 Constraints

Unfortunately, with every project there are some constraints which have the possibility of slowing down the development of the application. This may vary from hardware issues, software complications, safety, standards, etc. In this section of the report, these constraints will be explored and addressed in order to fully appreciate and prepare for the task ahead.

- I) **Hardware issues** – This constraint has a number of various factors which would need to be addressed. The functionality of the equipment and making sure that it works with the application would be the first step in delivering a high quality app. The recording device would need to fully record the S1 heart sounds to assist the user in making an initial diagnosis. Secondly, the noise-cancelling could be essential as if there is too much white noise in the recordings, the data would be rendered unusable and the overall aim of the project would be left in question. To summarise the hardware constraints, there would need to be measure put in place to guarantee the success of the app. For example, constant testing would be effective and provide the opportunity for any hardware problems to be resolved as soon as they occur.
- II) **Software Constraints** – Another problem which may have an overall effect on the project would be software constraints. A complete understanding of what each function does would be vital to ensure the project runs smoothly. Compatibility must be taken into consideration with this limitation. By this it is meant, that the application should be able to operate on a number of different devices such as phones or tablets. Each slice of code must be well-suited to the rest of the project, or there is a chance that not everything will work according to plan. To counter such constraints, a diary of all coding would be extremely useful so that any method or function would be recorded; and what exactly that function does. This organisation and keep records of all the software coding involved within the project (basic or advanced) is essential.
- III) **Safety** – The matter of safety is one that must be taken into consideration. This includes the safety of the test subjects from which the data is taken. The purpose of the application is to assist users/cardiologists, who are lacking in resources, when making an initial diagnosis. Therefore, while the application is an aid, it cannot make the final diagnosis itself. The category Z project label permits the cause of unnecessary stress on a subject, meaning that if it

'diagnosed a patient with heart disease' then it would create unnecessary stress on that subject (and it could be an incorrect diagnosis).

- IV) **Standards** – Finally, another constraint would be abiding by BCS and other standards. The British Computer Society has a set of written Standards, in addition to the different laws such as The Data Protection Act, which must be followed as a professional in the world of computers. These are guidelines on how a professional must act and work in the field. Keeping the information private for patients is a vital aspect for this application as private data may be at risk if it is not protected accordingly. Ensuring the safety of the patient and their information, all of these Standards must be taken into consideration.

3.6 *Non-Functional Requirements*

The non-functional requirement of the system is just as important as the *Functional Requirements*. This category includes what is expected of the system, such as reliability, security, etc. Within this section each of these requirements will be broken down and discussed. The non-function requirements could be described as what should automatically be a feature of the application, without the need for a focus group or questionnaire. The following list will analyse the non-functional requirements of the particular project.

- I) **Security** – Within every section of society, and specifically in the world of technology, security is an imperative feature which is always required. With every application, phone, computer; there are private details which need to be correctly safeguarded. As a result, this application should provide the same standard of protection to all the users, especially if they are working within the medical profession (which is a very sensitive sphere when it comes to private details).
- II) **Reliability** – The reliability of the project must meet the correct standard. For the application to assist with the cardiological diagnosis, it must have the ability to hear the sounds of the heart as required. If the initial diagnosis requires the cardiologist to first hear the heartbeat, then a clear and accurate recording must be made. On the other hand, if this is not possible for the application, then it is rendered ineffective. Therefore, a reliable system must be created to fully utilise the purpose of the project that works the same way every time.

- III) **Portability** – With the nature of the project, and its drive to counter the lack of resources which are not readily available for cardiologists, portability is another non-functional requirement. By this, it is meant, that if the device and application were not easily transported, then it would further hinder the ability of the practice nurses, to make the initial diagnosis. However, if the device (smartphone) is portable, then this would effectively give the users an additional resource in which they can make a quick and decisive decision on the prognosis.
- IV) **Performance** – The topic of performance is another non-functional requirement. This application needs to be fast and accurate to fulfil its purpose of assisting the user. This can be done by perhaps restricting a feature which would slow down the process speed and ultimately hinder the performance of the app. One example of this would be a database within the application. Although a worthy intention, it would slow down the ability for the cardiologist to listen to the heart sounds. As the overall design is to allow the user to hear the HR and HRV, then this must be the top priority when it comes to features. Therefore, if another feature impedes this, an evaluation and decision must be made as to whether it is a necessary function.
- V) **Availability** – With the intended audience of the application to be the medical profession, it should be readily available to all who would require such a resource. In order to achieve this, promotional letters could be sent out to give the potential users the chance to be informed and then utilise the app within the field. In addition, allowing the app to be accessible via the PlayStore, would not only give the users the chance to download it, but also give the developers the opportunity to see how effective it is through the use of the application.
- VI) **Maintainability** – Another factor which impacts the overall success of the project is the ability of the application to continue to grow and develop. With this in mind, the app needs to have the opportunity to update and stay 'in the loop' with new features which could be available for it. If, for example, there is a new class or a new piece of information about heart-disease, then the app could implement it within the original design and be able to assist the user in a resource constrained environment.

3.7 Other Applications

To begin with, the research that is required will be essential to discovering the key to success in this project. Therefore, it would be vital to look over any previous applications which have been created for such tasks. After much research, I have found that numerous applications record the heartbeat and the pulse that runs through the body. However, the majority of these apps record the data using the flash on the back of a smartphone. The website, *LiveScience*, rated the best heart-monitoring applications. The following information shows the top applications which were found during research on other apps with similar purposes.

Cardiograph – Heart Rate Monitor

(<http://www.imedicalapps.com/2012/09/cardiograph-app-heart-rate-monitor/>)

- User is invited to watch an introduction tutorial on how the app works
- Comparison to the ECG device used in the hospital
- It contained a disclaimer message
- Help, history and profile buttons
- Allowed users to enter name, age, and time they were last logged in
- Gave previous measurements taken for these users which the application was able to save
- Using the pulse, it is able to record the heartrate.
- GPS was also involved when recording the heartrate (when and where)
- Ability for the user to email or print the recordings (they were presented in the ECG form)
- Previous research showed that applications from the mobile was generally accurate in recording the heartrate
- \$1.99

Pros

- ✓ Inexpensive
- ✓ Ease of use
- ✓ Reasonably accurate
- ✓ Updated regularly

Cons

- ✓ Older patients may have trouble finding the right spot to find the heart rate
- ✓ Pricey if you aren't able to use it

Instant Heart Rate app

(<http://www.livescience.com/49653-best-heart-rate-monitor-apps.html>)

- Reasonably accurate to within 1/2 beats per minute
- Gave different reading for multiple states of living – just woke up, after exercise, resting
- Easy to use and has clear instructions
- Gives push notifications to remind the user to record their heart beat
- Allows the user to add notes for each piece of data recorded
- When a smartphone's flash illuminates the skin, its camera can capture the miniscule colour changes that happen each time the heart beats. (Because you have to hold fairly still for this process, these apps don't provide the sort of continuous monitoring that external heart rate monitors can provide)
- Your maximum heart rate should be 220 minus your age
- \$1.99

Pros

- ✓ Ease of use
- ✓ Very Accurate
- ✓ Various heart reading states; e.g. resting or exercising
- ✓ Updated regularly
- ✓ Option for notes on each reading

Cons

- ✓ Older patients may have trouble finding the right spot to find the heart rate
- ✓ Sensitive when doing a reading as the user would have to hold still for an accurate reading
- ✓ Useful for certain functions, but does not seem suited for a medical practitioner
- ✓ Costly for a mobile application

Runtastic Heart Rate Pro

(<http://www.livescience.com/49653-best-heart-rate-monitor-apps.html>)

- Runtastic stores your last 10 measurements in handy graph and list format. It provides less detail for tagging measurements, but you can enter whether you measured your heart rate while resting, before or after exercise, or at your max heart rate

- You can then filter the data using these tags
- The app includes a reminder feature and the ability to share results via Facebook, Twitter or email
- Stored in an online account where it can be accessed from any device
- Annoying for users when it provides useless toast pop-ups and information while trying to focus on other aspects of the app
- \$0.99

Pros

- ✓ Allows the user to see each recording in graph form, utilising the ability to read wave graphs
- ✓ Ease of use
- ✓ Gives the user to option to store their recordings to an online account
- ✓ Averagely priced for a mobile app

Cons

- ✓ Online account storing may be useless if the user has no connection to Wi-Fi or mobile internet
- ✓ Distractions for the user when some toast pop-ups off-putting

4. The Design Chapter

For the designing and development stage, it is very important that the whole application runs smoothly and is easy for the user to navigate through. Everything, from the colour scheme to the backgrounds and buttons, must be carefully thought through to ensure maximum satisfaction from each user. If the system was complicated and not consistent, any user would more than likely prefer not to use this app, rendering it ineffective in the overall purpose of the project. A plan must be put forth in order to create a user friendly and productive app.

Within this chapter, the software methodology will be analysed, in addition to the prototype and any other interface designs that will be implemented throughout the application. I believe that the Agile Methodology would be the most effective for this specific project. There were a number to consider which all had a number of attractive ideologies, these included the like of;

1. Crystal Methods Methodology – This method is based around the different talents that each person on the project has. Each can bring a different skill to the overall success and therefore, their tasks should be suited to their abilities.
2. Dynamic Systems Development Model – There are a number of principles which are included within this method. Points such as integrated tested throughout the development stages, active user involvement and collaboration and cooperation between stakeholders can be found in the definition of this idea.

However, after reviewing these different methodologies, the Agile Method appears to best fit this application development. For this method, there are short time frames in which requirements for the progress must be made. These are called iterations. "Each iteration is like a miniature software project of its own, and includes all the tasks necessary to release the mini-increment of new functionality: planning, requirements analysis, design, coding, testing, and documentation" (Professionals, 2016). Therefore, each section of the project must be broken down into individual tasks, given a time limit for each, and successful complete each task within the designated time period.

For a productive user interface, we must understand a number of points.

- Who exactly will the users be? E.g. Professionals, teachers, children...
- What are the tasks that the user must perform in order for the application to work successfully?
- Is there any specific content which must be included for the subject?
- What will the user environment be like? e.g. under what conditions will the application be used?

As these questions are being answered, it allows the developer the opportunity to prioritise certain tasks over others. For example, the capability of the system to perform the task of recording the heartbeat is essential and must be a priority. Additionally, an easy and non-complicated interface must be set up, ensuring it is user friendly. If this is not the case, the user would find it tedious and difficult to perform any task that it is required for. These issues would be a priority for the developer to resolve. A smooth and reliable application would give the project a much greater chance of success.

4.1 *User Interface and Interaction*

As with every application, the User Interface is one of the most important aspects of any app. The user interface is an important element of mobile application because the display is smaller and this has a great impact on the design aspect of each project. Widgets, touch and keyboards play a role in this interface, allowing for an ease of accessibility in using the device. These interfaces are included in most mobile platforms and may share a "common look or feel". According to Wasserman, it is in the interest of the developer to take into consideration the platform standards, in particular the touch screen facet of devices, and this may differ from platform to platform. The user interface will strongly be influenced by the design issues related to each device; e.g. weight, size, etc (Wasserman, 2010). The design of the user interface will allow for a level of intuitiveness from the user, when it is easy to use and understand without any complications or bugs.

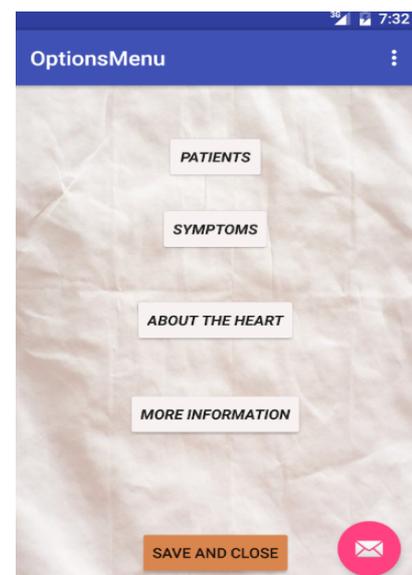


Figure 3: Opening Menu of the Application

Figure 3 shows the initial menu which the user is presented with after the welcome page. On it, each button provides the user with a tool to utilise each function which was discussed in the Analysis and Design Report from Jan. 2016. An 'easy on the eye' interface, with a simple background, would certainly give the user a simplistic, yet sophisticated feel with good contrast. The benefit of using the layout with light colours and refined buttons, seen in figure 4, provides a feeling of simplicity from the application which the user will, no doubt, appreciate. Another facet, which the user would take into consideration, would be easy transition from the 'rustic paper' background into 'cream-like' coloured buttons. The use of heavy and bursting colours such as pink/yellow or a bright red, would only give the user an aggressive, 'in-your-face' feel about the application.

Upon opening the Patients Activity, the user will have the option to search for an existing patient via their Patient ID, or to add a new patient to the database. This is shown in Figure 4. Initially, the user will enter an existing patient, which will then result in the top blue text field. This will provide the user with the Patient ID, Last name and then the First name. The next option would then be to search for previous recordings for that patient. After entering the Patient ID, the SearchView would return any recordings that match the ID that is entered. Shown in Figure 4.

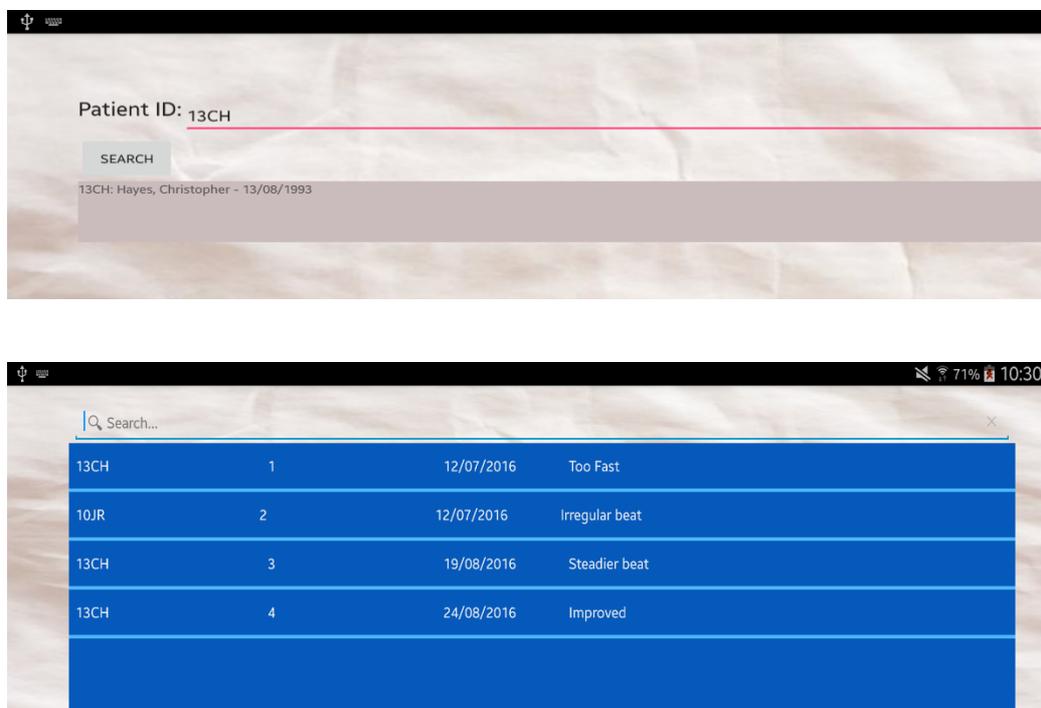


Figure 4: Activity which the user has the option to search for an existing patient, or add a new one

Moving from the Options Menu, one of the most important activities which the app provides would be the Recording function. The user would utilise the 'Record' button to continue through to recording the audio for the heart. Figure 4 shows how the layout and interface provides an extremely simple, yet effective way of allowing for such functionality. As seen from the image below, there are a number of buttons which allow for the functionality to record the audio clip, stop and playback a recording. The ImageView in the middle of the screen simply lets the user know that this activity is used for recording the heartbeat. In the top right-hand corner, there is a button which allows for continued use to visual the audio file. Once the user has successfully recorded an audio recording of the patient, the user will playback to validate that the audio is of accurate quality. Next, the user detail will need to be entered. As seen from Figure 5, there are background hints of text with each Edit Text field. This gives the user to format from which to save the patient details, e.g.

- I. Recording ID – 1
- II. Patient ID – 13CH
- III. 13/08/2016
- IV. Comment

The final aspect of this page would be to hit *Add Audio*. This button simply takes the audio file and places in within the Recording Table in the database. In the Implementation chapter, the methods for doing this will be discussed, along with the importance of an



Figure 5: Screenshot of the Recording activity

organised database. Finally, when the user had added the audio file for the patient to the database, they are able to all the recordings for each of their patients.

If the user does wish to see all the recordings, they will be able to do this in the *List of Recordings Activity*. As each audio file is saved into the database, they will then be inserted into a Listview which stores each file. The purpose of this would be so that the user is able to see the collection of recordings; with each ID, Patient ID and the date that the recording took place. While there will be a search function implemented into the application, this Listview will display the recordings in an orderly fashion e.g. 1,2,3,4... while in the search function they will simple display the recordings that apply direction for one patient. The next chapter will discuss in detail how these audio files are stored, and why it is extremely useful for the user to simply click the row, with the details for the patient, and the recording will begin to playback. Figure 6 shows the display method choose and how each audio file will be organised in the Listview.



Figure 6: The display of the Listview which stores the audio files

The development of the User Interface and how this interface provides the smooth transition between an elegant application and its functionality. The next section to be discussed in this report will be the information pages which will be provided for the user.

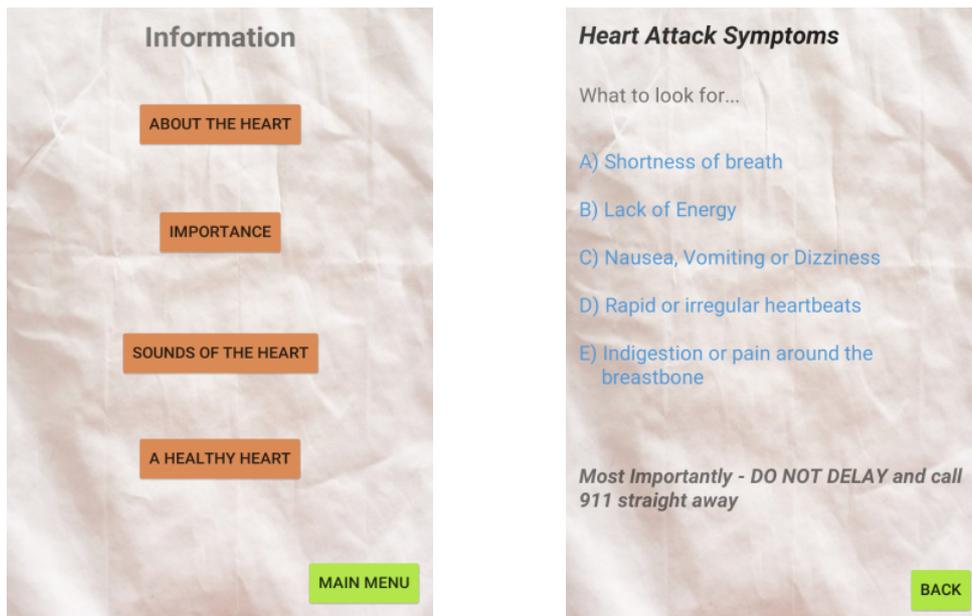


Figure 7: How the information will be presented to the user.

Figure 7, shows the outlines for the informational activities. These give the user much needed information about the heart, what it does, how the sounds are created and a number of further factors to consider. In addition, there is a section of the application which provides information about various issues that can occur with the heart. Figure 7 shows the page that gives details on heart attack symptoms. These pages are certainly not major features that the application provides, but they do give a more complete feel when using this app, they are not exhaustive but should give some insight into the heart. Furthermore, if a cardiologist or doctor has the symptoms provided for them during a screening or recording, then they are able to provide a more accurate diagnosis, if it is required. The 'More Information' section of the application informs the user of various facts, figures and links for any user, it provides them with a clearer picture of how heart issues affect the United Kingdom as a whole. For example, the number of deaths which are related to heart problems, the percentage of those which are diagnosed with any heart related illnesses. These three pages (Symptoms, About the Heart, and More Information) will all follow a similar theme and look. This will result in a consistent and smooth interface for the user to enjoy while successfully achieving the overall aim of recording each piece of data for each individual patient that the user is monitoring.

From Figure 5, the user has the option to add a new patient to the database. Figure 9 is a screenshot of how this page will be displayed and organised. Each button which is available provides a function that the user will be able to utilise such as adding new patient

data, or updating a record from the database. Using the SQLite database which is supported on the Android system, the application will deliver a cleaner and more useful app to achieve the overall aim of the project.

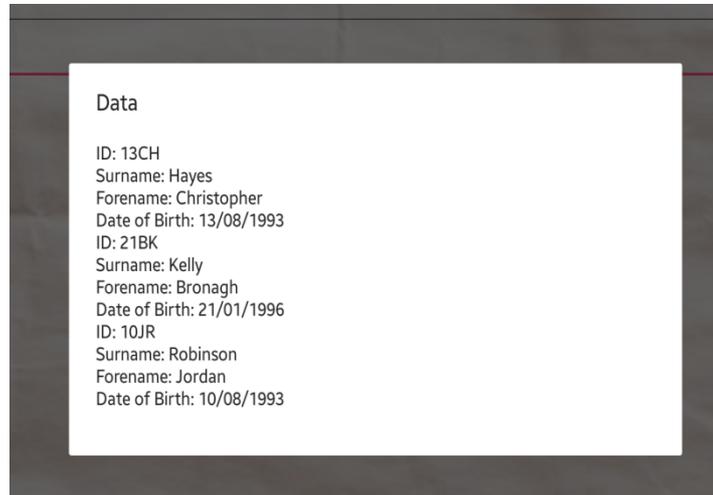


Figure 8: How the patient information will be displayed.



Figure 9: Allows the user to add new patient details.

With the use of Figure 8 and 9, the outline of how the database and activity to store the patient information can be seen. This is a starting point for further development as the audio recording will need to be attached to each patient accordingly, this will be saved in the Recording Table – with a foreign key link (Patient ID) between the Patient and Recording Tables. The Entity Relationship Diagram (found in the Appendix) shows the relationships for which the patient’s information will be stored, in addition to how the tables will be linked.

4.2 Software Architecture

With the user-interface (how the application will look), the software architecture is the next step in development. This section of the project is extremely important as it allows the developer to break down each piece of software that they would need and describe in detail what it brings to the project. With Android being the operating system which will be used to create the application, an analysis will be made on the different aspects of this system. This platform will be the building block from which the app will be built. In addition to the Android operating system, other software which will be involved, and also any other libraries, will be included in this chapter.

4.2.1 The Android Platform

The first and most important piece of software, is the one that the project will count on to run, Android. This system is an extremely common and reliable platform. Smartphone developers such as Samsung, Sony, and HTC all provide the Android operating system. Figure 10 shows the systematic architecture of the android based platform.

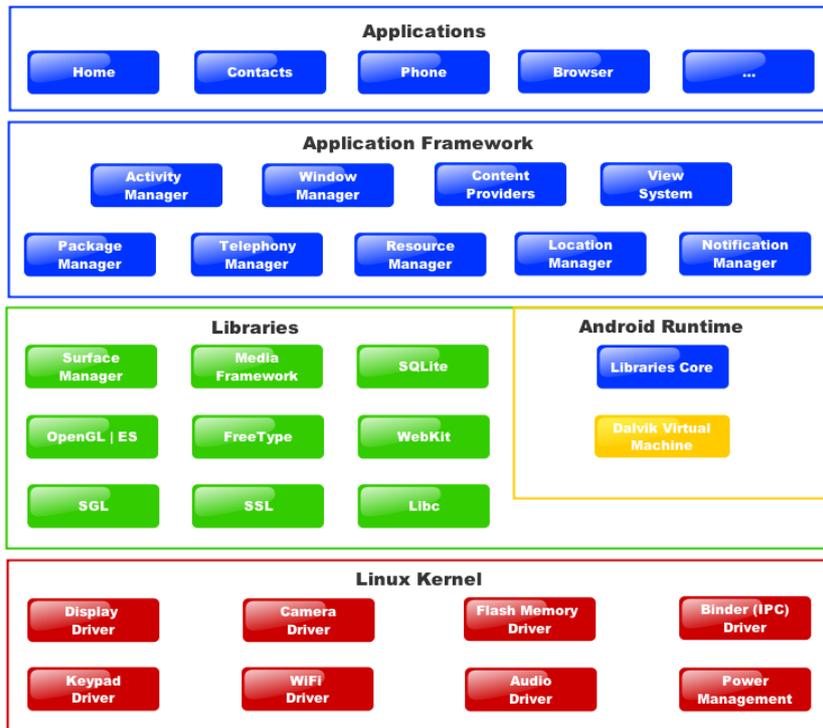


Figure 10: The various layers within the Android Platform.

As seen from the image above, it is broken down into a number of different categories. At the very bottom, the Linux Kernel is implemented throughout the system. The purpose of this is to “provide a level of abstraction between the device hardware and the upper layers of the Android software stack” (Techotopia, 2014). This Linux Kernel gives the platform a number of properties which are very desirable, they include:

- Pre-emptive Multi-tasking
- Memory
- Process and Power Management
- Provides a network stack
- Device drivers – audio, Wi-Fi, displaying the screen (Techotopia, 2014).

Moreover, with Android, there are many benefits in comparison to other software application programs. Android allows the users the ability to utilise code that is already written within the API of the platform, providing a high level of consistency throughout the project. The Media Framework Library is a feature of Android which will be used to great effect for this app. While the heart sounds are being recorded, this library will allow the device which is using the app, to listen and assist the cardiologist with the ability to hear the

recording. It will support the purpose of the project and give the information required to make the initial diagnosis.

The second feature of Android which will allow further development is the *Resource* and *Activity* Managers. The Activity Manager takes care and looks after all of the features during the lifecycle and activity stack. The Resource Manager gives the developer the ability to “access to non-code embedded resources such as strings, colour settings and user interface layouts” (TutorialsPoint, 2016). These particular aspects of Android will give the developer the control over the application that is required in order to achieve the overall aim of the project.

Finally, the Hardware Application Layer (HAL) within this platform is vitally important. This will allow different pieces of hardware, such as microphones, recorders and headphones to be used to assist the user. For this specific project, all of the are a necessity as without these hardware devices, that application would not be able to record the sound of the heart, nor would the user be able to play the data back (either at the time or at a later date). The HAL allows the developer to “implement functionality without affecting or modifying the higher level system” (Android, 2016).

4.2.2 Application Architecture

As with the User Interface, the architectural design must also be carefully developed. A clear outline will give the developer the understanding which is required for a successful application. If the architecture is designed correctly, and efficiently, then the result will be a more straightforward path that the user/developer can follow. Figure 11 is a diagram of the initial architectural strategy which will be put forward and will also allow for ongoing development.

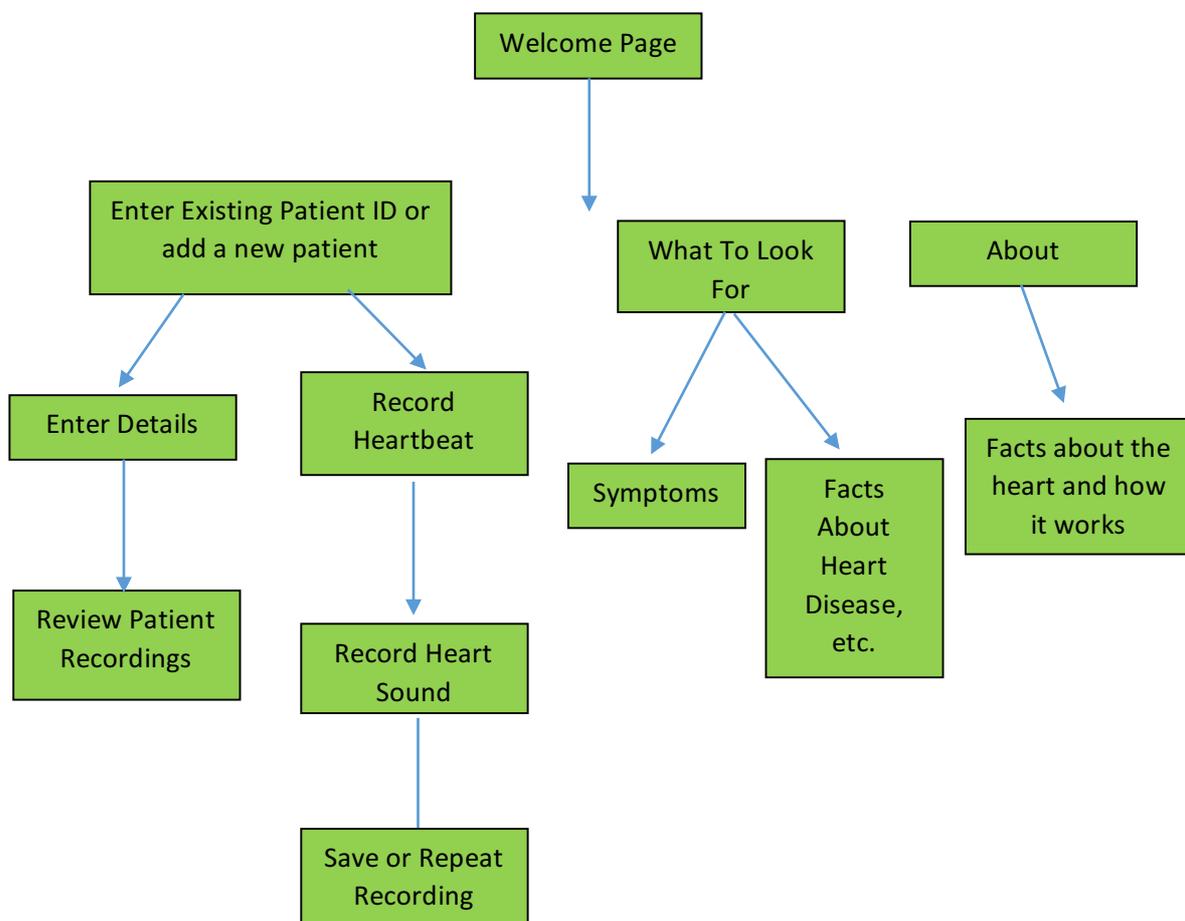


Figure 11: Basic design of the application and the links between each page.

4.3 Database Design

One factor which is vital for ensuring the highest level of user interaction, is the correct design of the SQLite database. If this is not addressed in the correct manner, then the application runs the risk of being completely inept for the user's needs. Therefore, time, effort and research was needed to safeguard against such a risk.

First and foremost, the database must be designed around the user's requirements. In the case of this application, the user will want to record certain information about the patient:

- Name
- Patient ID
- Date of Birth
- Heartbeat audio (3GPP)
- Date of recording
- Recording ID
- Comment about the recording

As a result of all these requirements, a database must be set up to connect all of this information for the user to utilise whenever they need it.

Secondly, how will these different types of information be organised? For example, will they all be stored in one table? Or will it require more than one to properly address the user's needs for storing information about the patient? This application will contain two tables for storing the information. The first table will be labelled as the **Patient_Table** and the second will be the **Recording_Table**. This will split the information into two tables but will have a link between them so that the user is able to fully access all the patient information when required.

As seen from the ER Diagram in the Appendix, the relationships are defined with every factor taken into consideration. Patient ID in the Patient Table is the primary key from the database. With reference to the Recording Table, the Patient ID** gives the foreign key that references the Patient ID in the Patient Table. This gives the user to ability to save individual recordings to individual patients. The relationship between each table can further be described as:

- **One** patient can have **many** recordings (one-to-many)
- **One** recording can only be saved to **one** patient (one-to-one)

A further benefit of the organisation of the database, is that it also allows the user to search for a certain patient via their Patient ID, and this would result in all of the searched patient's audio recordings to be available for the user to listen to¹ and transferred.

4.4 Overview

With an understanding of the User Interface and the software architecture, the progression of the project has taken a new step. The feature and design of the Android Platform, make it an easy choice when designing an application. The Multiple Libraries, Hardware Application Layers, and the Device Managers give a great deal of flexibility and consistency which provides the application with a solid base on which to start building. Furthermore, the necessity for external hardware equipment for this project, and by using Android it allows the room for modification without affecting the overall operating system. As the project continues to develop, new ideas will be put forth. These will all have to be examined to decide whether they are necessary, or if these would bring a new, exciting feature which the application would benefit from. To conclude, the understanding and knowledge of the architecture of a system/application is without a doubt an invaluable asset, and therefore it always needs to be carefully considered before any development begins. This will, hopefully, provide a new application which will be easy, useful, reliable and exciting for users in the field of cardiology.

¹ See Appendix: Database Code.

5. The Implementation Chapter

5.1 *Various Designs*

When it comes to any application development, no matter the scale, there are always going to be various designs which were initially considered at the beginning and which could have been 'potentially' implemented. This section comes with the successful ideas, and those that were not practically feasible. The development of this application is no different. This section of the Design Chapter will discuss any ideas which were brought in to improve the quality of the app for the user, and also those concepts which were not included in the final design.

After the initial analysis and design, noise suppression algorithms became a major factor for implementing when an audio file was created. This was thought to be a necessity for such an application. However, after reviewing a number of devices, such as the HTC One or the Samsung Note II, it was decided that the quality of microphone was a great deal more reliable and accurate than initially thought. The idea behind this came from an older generation of Samsung devices, which left the audio recording sounding very muzzled or fuzzy. Nevertheless, after testing the application on a number of newer devices, this idea of noise suppression was dropped in order to increase user function within the SQLite database. Moreover, the most effective and reliable method of cutting out background noise would be to implement *Fast Fourier Transformation* to the audio recorded with the application. This required a great understanding of such a technique, especially with translating it into the Java language.

What is Fast Fourier Transformation (FFT)? To simplify what FFT does, it is "a mathematical function that can be used to show the different parts of a continuous signal. A Fourier transform shows what frequencies are in a signal. For example, consider a sound wave which contains three different musical notes: A, B, and C. Making a graph of the Fourier transform of this sound wave (with the frequency on the x-axis and the intensity on the y-axis) will show a peak at each frequency which corresponds with one of the musical notes." (Wikipedia, 2014). The reason why this computing function would be extremely useful for such an application, would be to ensure the recording would take both frequencies (high and low) and ultimately filter out the higher frequency noises (e.g. background noise). This would result in the lower frequency sounds being displayed and given back to the user. As discussed previously in the report, the heart beat is an extremely

low frequency so this function would return an audio with no background noise, with a clear heartbeat being audible. Unfortunately, to fully utilise this method, a complete understanding of how this process works is necessary, and then converting the methodology into Java for Android Studio. Alternatively, with the power of newer phones, and the ability to record such sensitive sounds, this application can be implemented and utilised by the more powerful microphones which are found on newer generation devices.

One element which was going to be featured in the application was a visualiser for the recording. After recording the audio, the user may have had the option to visualise the heartbeat wave with the use of the Visualiser Class which android provides. The Visualiser activity was under development but when the application is a finished product, it will, unfortunately, not have a wave like graph which would represent the sound waves of the heart. The reasoning behind this decision, not to include the Visualiser, was simply the desire to focus more on the efficiency of the app, rather than having features which enhance the user interaction, but this was at a cost of a less developed product. After discussing this with a number of individuals, the decision was taken to focus more on the necessary components of the application, e.g. recording a heartbeat, storing it in a well organised database, and then having that audio file for playback at any later date. If any of these features suffered as a result of attempting to implement the visualiser, then the overall productivity and user interaction would also suffer in future developments.

5.2 Algorithms and Methods

5.2.1 Database Implementation

One of the key features of the Heart Sound Application is the ability for the user to listen to previous recordings, play them back, and compare them to see if the patient's condition has deteriorated/improved. This facet is vital in the success of the application, without being able to compare older recordings to more recent audio files, the user (e.g. a nurse out in the field) would not be able to make an accurate and confident diagnosis. This would give doubt in the diagnosis and render the application incapable of its purpose – allowing the user to make a clearer and faster diagnosis on current patients based on the condition of their heart. The capability to store these audio files in an SQLite database gives the application more appeal to users as they would most certainly be able to compare older

recordings to more recent ones; leading to a more accurate diagnosis. The following segments of code show how this is accomplished within the application.

```
//Adds the audio which was recorded into the Second table - foreign key (Patient ID) links to the Patient table
public void addAudioData (String ID, String patientID, byte [] recording, String date) {

    SQLiteDatabase db = getWritableDatabase();
    String sql = "INSERT INTO " + RECORDING_TABLE + " VALUES(?, ?, ?, ?)";
    SQLiteStatement sqlStatement = db.compileStatement(sql);
    sqlStatement.clearBindings();

    //Binds both the String and Blob, and finally executes the statement by inserting it into the table
    sqlStatement.bindString(1, ID);
    sqlStatement.bindString(2, patientID);
    sqlStatement.bindBlob(3, recording);
    sqlStatement.bindString(4, date);

    sqlStatement.executeInsert();
}
}
```

Figure 12: Inserts the information given into the RecordingsTable

```
//Query to access the Recording table within the database
db = new DatabaseHelper(this, "recordingsTable", null, 1);
db.QueryData("CREATE TABLE IF NOT EXISTS recordingsTable (id INTEGER PRIMARY KEY, PatientID TEXT, Recording BLOB, Date TEXT, FOREIGN KEY (PatientID) REFERENCES patientTable (PatientID));");

//OnClick method to add the recorded audio to the database
addOnButton.setOnClickListener((v) -> {

    String recordID = recID.getText().toString();
    String patientID = patID.getText().toString();
    String date = addDate.getText().toString();

    db.addAudioData(
        recID.getText().toString(), patID.getText().toString(),
        FileLocal_To_Byte(outputFile), addDate.getText().toString()
    );

    //addDate.setText("");
    Toast.makeText(Recording.this, "Adding...", Toast.LENGTH_SHORT).show();
});

addOnClickListenerRecordings();
```

Figure 13: This function adds the audio into the database via OnClick Listener

Figure 12 and 13 shed light on how the information is stored when a heartbeat is recorded. Figure 12 is the basis for adding audio into the SQLite database. This function creates an SQL query for inserting data into a table, in this case the Recording Table. It then takes the parameters which are given, and inserts them accordingly. Each String is added to the appropriate column within the table, leaving the actual audio file being stored as a byte file (rendering it possible to store it in BLOB format). The reason why this is

necessary is that this is the only format enabling the storage of audio files within a database of an app. After each parameter is binded, the method finally executes the insert function. In Figure 13, the data is taken from EditText fields which are entered by the user. The *addAudioData* method then casts the text within these fields, and applies them to a String variable which is stored in the Database. The *fileToLocalByte* method allows for the local file which was created as an instance for each recording to be converted into byte [] format. This is key, as it allows the file to be stored as a byte array within the database; only to be converted back at a later date to allow for playback. In order for the user to know if the 'add' is successful, a toast will pop up to inform the user that the audio has been stored into the database.

5.2.2 *Button Tapped (record, stop recording, playback)*

After a great deal of research on various methods for implementing the microphone into the application, one particular method was more appealing than the others. This method was known as a *Button Tapped* function. This method calls three functions within the activity which then link to buttons within the page. These methods are:

- *beginRecording()*
- *stopRecording()*
- *playRecording()*

These three function give the application the simple yet required features for the user. They allow the device microphone to be utilised by the app, along with stopping, and playing-back of any audio. Figures 14-17 show the code behind each function, along with how they are implemented within the *Button Tapped* method.

```
//Method which links the buttons in the activity view to the functions
public void buttonTapped (View view){

    switch (view.getId()){

        case R.id.btnStart:
            try{
                beginRecording();
            }catch(Exception e){
                e.printStackTrace();
            }
            break;
        case R.id.buttonStop:
            try{
                stopRecording();
            }catch(Exception e){
                e.printStackTrace();
            }
            break;
        case R.id.btnPlayback:
            try{
                playRecording();
            }catch(Exception e){
                e.printStackTrace();
            }
            break;
    }
}
}
```

Figure 14: ButtonTapped()

```
//Method which begins the recording once the button is tapped
private void beginRecording() throws Exception{

    ditchMediaRecorder();

    //sets the file path for the recording to be saved
    File outFile = new File(outputFile);

    if (outFile.exists()){
        outFile.delete();
    }

    recorder = new MediaRecorder();
    recorder.setAudioSource(MediaRecorder.AudioSource.MIC);
    recorder.setOutputFormat(MediaRecorder.OutputFormat.THREE_GPP);
    recorder.setAudioEncoder(MediaRecorder.AudioEncoder.AMR_NB);
    recorder.setOutputFile(outputFile);

    recorder.prepare();
    recorder.start();

    Toast.makeText(getApplicationContext(), "Recording started", Toast.LENGTH_LONG).show();
}
}
```

Figure 15: beginRecording()

```
//method to playback the recording
private void playRecording () throws Exception {

    ditchMediaPlayer();

    mediaPlayer = new MediaPlayer();
    mediaPlayer.setDataSource(outputFile);
    mediaPlayer.prepare();
    mediaPlayer.start();

    Toast.makeText(getApplicationContext(), "Playing the Recording", Toast.LENGTH_LONG).show();
}
}
```

Figure 16: playRecording()

```

//method which stops the recording once finished
private void stopRecording(){
    if (recorder != null){
        recorder.stop();
    }

    Toast.makeText(getApplicationContext(), "Audio recorded successfully", Toast.LENGTH_LONG).show();
}

```

Figure 17: stopRecording()

As seen from Figure 14, the methods found in Figures 15-17 are called within the *button tapped* function. This allows the user to call one of these methods when one of the following buttons is hit:

- btnStart
- buttonStop
- btnPlayback

In the *beginRecording()* method, a new instance of the *MediaRecorder* is created for each time the button is hit. Moreover, the file from where it is saved is also created as a new file for each audio. The *AudioSource* is further linked to the microphone, allowing the user to utilise the device to its full potential. However, within the *Android Manifest*, permission must be given to allow the microphone to be used by the current application. Finally, if the audio recording is successful, a toast will 'pop-up', informing the user that the device is currently recording the sounds of the heart.

The method *playRecording()*, takes full advantage of the *MediaPlayer* function within *Android Studio*. Using the same **output file** that the audio was recorded to, it allows the *DataSource* to call this file, prepares it for playing, and allows the user to hear the audio that was recorded during that session. Similarly, the *playRecording()* function also has a toast will allows the user to know that the audio is now being played via the speakers.

Finally, the *stopRecording()* function stops all use of the microphone. This gives the user full control of the time for which a recording is taking place. After the user has finished recording the patient's heart sounds, then they are able to stop the microphone from taking in anymore audio sounds.

5.2.3 Searchable Function

The final feature which will need to be discussed is the **Searchable Function** that allows a user to search the patient information. This is achieved by searching for the Patient's ID, which will in-turn search the SQLite database for a match. Not only will the application have the ability to search for the basic information such as the patient's surname and forename, they will also be able to search for recordings that match that patient. For this to be possible, SearchView is needed to be implemented. This searches the existing recordings for a match of Patient ID and results with any that is a match. For example, if a user searches for that Patient ID: 13CH; then any recordings which were saved under that patient's ID, will be available for the user to listen to – and compare to recordings done more recently. This benefits the application and also the user's interaction with the application as they would be able to give a more accurate diagnosis (which is the overall aim of the project).

```
//Method to search within the database for a certain patient based of their Patient number, and returns the results
public String searchData (String patientID) {
    SQLiteDatabase db = this.getWritableDatabase();

    String [] columns = {COL_1, COL_2, COL_3};

    Cursor cursor = db.query(PATIENT_TABLE, columns, COL_1 + " = '" + patientID + "'", null, null, null, null);

    StringBuffer stringBuffer = new StringBuffer();
    while(cursor.moveToNext()){

        int index1 = cursor.getColumnIndex(COL_1);
        int index2 = cursor.getColumnIndex(COL_2);
        int index3 = cursor.getColumnIndex(COL_3);

        String patient = cursor.getString(index1);
        String surname = cursor.getString(index2);
        String forename = cursor.getString(index3);

        stringBuffer.append(patient + ": " + surname + ", " + forename + "\n");
    }

    return stringBuffer.toString();
}
```

Figure 19: searchData()

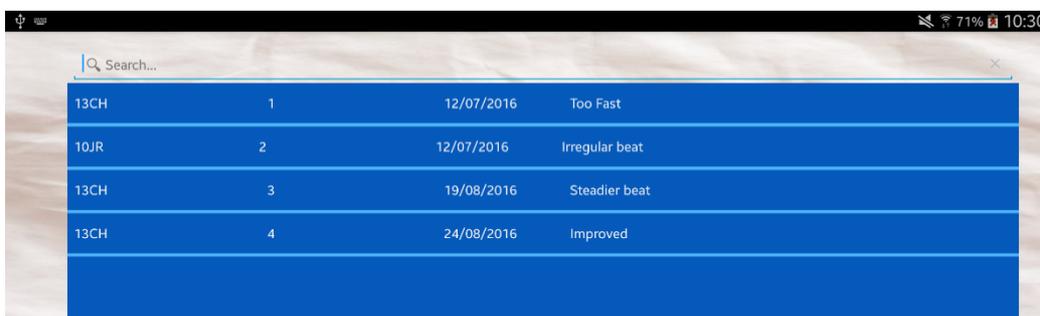


Figure 18: SearchView

As seen from Figure 18, within the `searchData()` method, the query calls the Patient Table, and uses the parameter of `patientID` to match any ID's that are stored within the table. The cursor moves through the list, picking up any that are a match, and displays them in the following format:

"13CH: Hayes, Christopher"

Figure 20 shows how these methods are called within the *Patient Options* Activity.

```
//calls the search function to display a searched patient
public void search (View view){

    String string1 = searchID.getText().toString();
    String string2 = db.searchData(string1);

    textView.setText(string2);

}
```

Figure 20: search() function

As seen from the figure above, the editText field '`searchID`' is used to pass as the parameter of the `searchData()` method. When the user enters a Patient ID into the editText, it will result in being used for these methods. After the search results are found, they will be displayed into a text box for the user to utilise as they wish. '`String2`' is the results which are matched when the methods search each table for a matching patient ID.

6. Testing Chapter

6.1 *Introduction*

After the software is completed, the application is required to be put through testing, in order to assure it is of a correct standard. The HeartSound App was no different, and this meant that there were a number of features which were tested. They include:

- Unit – testing the application on various devices
- Functional – testing all the functional features within the app
- System – testing the app within certain processes
- Performance – testing characteristics, e.g. concurrency or file sizes
- Security – ensuring the application protects the user’s information

Each test provides various answers for the developers, and only after all these have been completed, is the application ready for release. Within this chapter, an analysis of various methods of testing will be addressed. The testing also requires that an external user processes each function and gives feedback (Overbaugh, 2015).

The hardware that this project will be tested on includes a Samsung Galaxy Tab Pro tablet, HTC One smartphone and a Samsung A3 smartphone. During each test, there will be a number of factors that will be considered, such as:

- Speed – the response of the system when the user is utilising the app
- Accuracy – how accurate the audio is recorded and if the information of each patient is correct
- Integrity – certain functions of the app work independently to each other, without disrupting the overall user experience
- Errors – All errors are successfully resolved and once again, do not disrupt the user experience.

The results will be displayed in table, which will cover the areas above, and how well the application handled each device. Moreover, if there were disruptions or errors that occur, they will be discussed and resolved accordingly.

6.2 Review

Table 3: Samsung Galaxy Tab Pro Tablet

| | Speed | Accuracy | Integrity | Errors |
|-------------------------------|--------------|-----------------|------------------|---------------|
| Functionality | | | | |
| Heartbeat recording | PASS | FAIL | PASS | PASS |
| Database implementation | PASS | PASS | PASS | PASS |
| Playback recording | PASS | FAIL | PASS | PASS |
| Saving patient information | PASS | PASS | PASS | PASS |
| Searching patient information | PASS | PASS | PASS | PASS |
| Informational pages | PASS | PASS | PASS | PASS |

Issues

The testing with the Samsung Tablet was mostly a success. The majority of the categories were able to be cleared with very few errors. However, the accuracy of the recording was very limited as the ability of the microphone was not of the highest quality. As a result, the Tablet failed this part of the testing because there was a great deal of background noise which interfered significantly with any recording. This would cause any user to question the ability of the application as there would not be accurate recordings available. Furthermore, this would hinder any diagnosis because there would not be the quality of recording to confidently give a diagnosis to any patient.

On the other hand, with reference to the implementation of the database, the Tablet was able to handle and display any SQLite queries which were included in the code. It was extremely quick when adding information to the database with the time being almost instantaneous to when the button was clicked. In addition, the display of the informational pages was very quick moving from page to page, and the overall layout was appropriately clear.

Table 4: HTC One Smartphone

| | | Accuracy | Integrity | Errors |
|-------------------------------|------|-----------------|------------------|---------------|
| Functionality | | | | |
| Heartbeat recording | PASS | PASS | PASS | PASS |
| Database implementation | PASS | PASS | PASS | PASS |
| Playback recording | PASS | PASS | PASS | PASS |
| Saving patient information | PASS | PASS | PASS | PASS |
| Searching patient information | PASS | PASS | PASS | PASS |
| Informational pages | PASS | FAIL | PASS | FAIL |

Issues

Comparing the HTC One smartphone to the Tablet which was previously tested, the microphone is of a very high quality. This is extremely beneficial for the user as the audio recordings would be more clear, which in turn allows for a more accurate diagnosis. The background noise is almost non-existent with the heartbeat heard very clearly. The database is also very fast. The HTC One testing was no quicker or slower than the tablet when adding information to the database. The reason why the accuracy was 'PASSED' in the table above is because the audio was heard very clearly, which is why the tablet failed this section.

In contrast, the informational pages failed this test. The reasoning behind this was because the text fields were jumbled up, with nothing be read clearly. To combat this error, each textbox was made smaller, making sure that they did not interfere with each other. This resulted in a clearer activity, with the user able to read and take advantage of any information which was presented in these pages.

Table 5: Samsung A3 Smartphone

| | | Accuracy | Integrity | Errors |
|-------------------------------|------|-----------------|------------------|---------------|
| Functionality | | | | |
| Heartbeat recording | PASS | PASS | PASS | PASS |
| Database implementation | PASS | PASS | PASS | PASS |
| Playback recording | PASS | PASS | PASS | PASS |
| Saving patient information | PASS | PASS | PASS | PASS |
| Searching patient information | PASS | PASS | PASS | PASS |
| Informational pages | PASS | FAIL | PASS | FAIL |

Issues

The final device on which the application was tested was a Samsung A3 smartphone. The results were very similar to the HTC One. The microphone was very powerful, picking up a very accurate recording of any heartbeat that it was tested on. Furthermore, the speed and integrity of the app was very high. It was successful in implementing the database and had no issues with utilising all the features of the database (adding a new patient, saving a recording, searching). This was encouraging as the features were more accurately employed with the smartphones when compared to the tablet, which lacked any real accuracy with recording a patient's heartbeat.

However, the HTC One was able to record a heartbeat with little background noise interfering with the recording. The Samsung A3 did pick up limited white noise when carrying out this task, although, it was not enough to interfere with the accuracy of making a diagnosis. Moreover, the Samsung A3 encountered the same issues with the display of the

information pages. The texts were jumbled up and unreadable. This was resolved at the same time as the HTC One bug fixing as they were similar issues.

6.3 Overview

After much review and testing, the application was mostly a success. There were a few issues such as the quality of recording (Tablet only) or the format of the display of information (Smartphones only). These errors were exclusive depending on device. However, as an overall verdict, the application ran very well on both platforms. All features were generally of high quality. The limitations were clear, allowing for further development in the future, which is key for a successful app.

The user feedback was also very beneficial. Suggestions came in the form of the possibility of using a headset to listen more carefully to the recordings. With reference to the Samsung A3 or tablet, the background noise sometimes overpowered the actual recording. This resulted in failed attempt for a patient's heartbeat recording, requiring it to be redone. Moreover, a number of users identified an inconsistency with the colour of buttons which was resolved by a matter of carefully going through each page to make sure the button colours were consistent.

On the other side of the argument, users did provide positive feedback. They were particularly pleased with the ability to save a recording to a particular patient. This allowed for ease of use and a very smooth user interaction which was appreciated. In addition, each page smoothly transferred to one another with any delay or lag. In summary, the testing phase of the lifecycle of this application provided much needed feedback and allowed for the developer to see any errors which arose as a result of such thorough testing, and a very insightful experience resulted.

7. Conclusion

The aim of this chapter is to discuss the project as a whole, summarising the overall objectives and achievements of the application. In addition, any lessons and recommendations will also be discussed, which will further develop the future for this project and app. Finally, features which were not achieved, along with any improvements that would increase the user interaction will feature within this chapter.

The HeartSound Application is designed to increase the interaction between patients and medical users to increase the accessibility for devices which would be capable of recording a patient's heart, without having any hospital equipment (e.g. ECG machine). Moreover, while there are a number of applications which do provide features such as listening to the sounds of the heart, none are designed specifically for this reason, being more directed to a health conscious audience.

After establishing the key features which the application required, they were successfully implemented throughout. At the beginning of the project, there were ideas which were initially evaluated and thought to be key features. However, after much research and discussion, the main focus was towards having an application which is able to successfully record the sounds of the heart, save these audio recordings to an SQLite database, and ultimately search for patient information. These primary features provide users with a simple, smooth and very efficient product which can be utilised in the medical profession.

Moreover, the time and research which was given to the basic understanding of building an app and the benefits of Android Studio were crucial to successfully identifying the essentials for such a project. After this initial research, comprehending the programming language of Java was required. This combination of various investigations provided the building blocks to move forward in the project.

Finally, the desire to implement an SQLite Database pushes the boundaries of a simple app which is only capable of listening to one recording at a time. This fundamental structure, which is designed to store patient's information in an organised and logical way is a key feature which allows for more creative and effective user interaction, which emphasises the need for such an application within the medical profession. Not only will help provide a more accurate diagnosis, it will provide greater confidence to those who utilise the app. The detail and decisive work which was put into making a user friendly

interface, that was smooth and elegant was extremely challenging. This was done by working with neutral colours and a desire not to complicate the activity pages with a mass of various buttons. Instead, by good contrast, using buttons only when necessary and not including any colours which were 'in your face', I believe this was achieved.

7.1 System Requirements

At the beginning of the project, when the functional and non-functional requirements were initially specified, a methodical development was vital. Throughout the lifecycle of the application, these requirements were constantly being assessed, evaluated and revised. However, while some of these requirements were not always completed, they were addressed appropriately. In the next section, these functional and non-functional requirements will be discussed, with reference to future developments. In contrast, those which were developed and implemented in the project, were always carefully analysed before making any changes. This has resulted in the application addressing, not only the needs of the medical market, but also the characteristics which such a profession would very much appreciate.

7.2 Future Developments

While this application was organised, robust, and advantageous for the medical profession, there are always areas for improvements. Moreover, as the app is utilised in the appropriate field, certain aspects of the design could provide for future initiatives to achieve the overall aim of providing a solid and reliable application for recording patient's heartbeats, without the need for heavy hospital equipment. This section will suggest the areas which could possibly be improved for an enhanced user experience. These include:

- i. Login system for the user to provide more security
- ii. Visualiser to provide the heartbeat in a graphical form
- iii. Fast Fourier Transformation to cancel out any background noise, providing a clearer reading
- iv. Links to email accounts in order to send certain recordings for a closer review

- v. Linking the SQLite database to an online cloud, giving the user the opportunity to have access to their patient files from any device

I believe that the project was a success and with the opportunity for these future developments, the app will go far beyond any other application on the market. This will also give medical users a very powerful tool in their ability to give an early diagnosis for their patients, and could significantly add value to the success of this project.

7.3 Summary

The inclusive objectives of this project were to deliver a reliable, accurate and efficient application from which users in the medical profession would be able to hear and record a patient's heartbeat. In addition, the application should offer these requirements without having to transport hospital equipment. I believe this application would not only open new doors for such advancements in the smartphone world, but also offer users such as practice nurses, the chance to really tackle diseases of the heart, knowing they can rely on the results from this application. The overall implementation and structure supports a strong basis from which to continue, but it provides the opportunity for further development in order to take it to the next level of functionality. As with any application, as it gets more and more utilised, further evolutions and refinements will become evident and it should be able to provide continuous support to the user.

Appendices

Definitions

Heart Rate (HR) – The number of times that the heart beats per minute (MedicineNet, 2016).

BPM – Beats per minute (Farlex-Dictionary, 2016).

Heart Rate Variability (HRV) - the beat-to-beat alterations in heart rate. For a healthy individual, the ECG recording would show that the patient has periodic variation in R-R intervals (Kawachi, 1997).

Electrocardiograph (ECG) - a tracing that represents the patient's heart's electrical action which is 'derived by amplification of the minutely small electrical impulses normally generated by the heart.' (Miller-Keane, 2007).

Heart Disease – 'Conditions that involve narrowed or blocked blood vessels that can lead to a heart attack, chest pain (angina) or stroke. Other heart conditions, such as those that affect your heart's muscle, valves or rhythm, also are considered forms of heart disease.' (Staff, 2014)

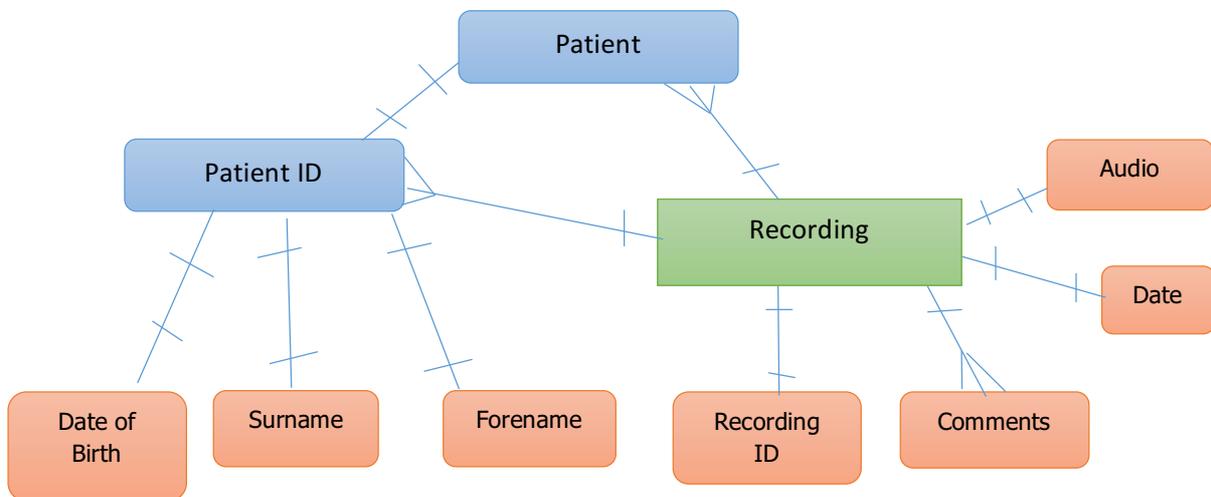
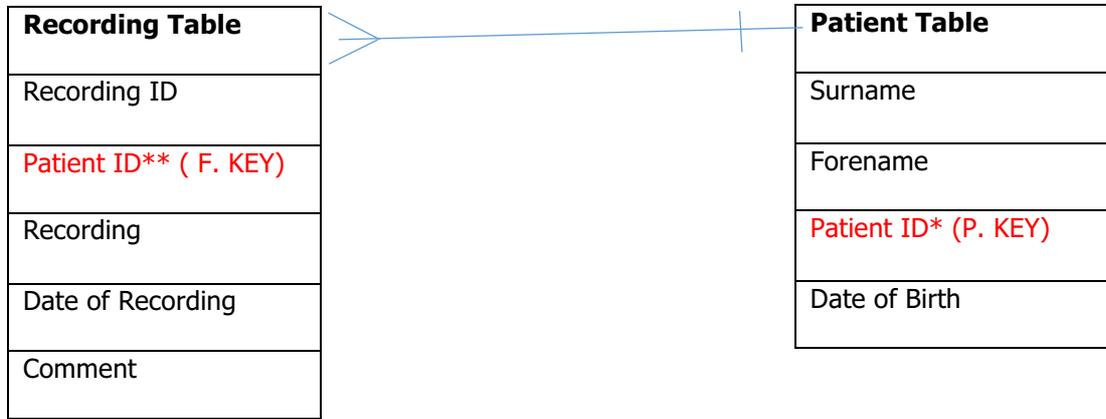
S1 Heart Sound – The first sound which is given off by the heart as the systole begins. It is created by the mitral and tricuspid valves closing abruptly. (MedEdu, 2015)

De-noising/Noise Cancelation - The process which cancels out the white noise/background noise that is found during each recording. This will result in the vital information of the heart beat to be heard more clearly (AllWords, 2016).

Background Noise – 'Any type of noise that is not the sound that you are specifically listening to or monitoring'. (Collins-Dictionary, 2016)

Frequency - The rate at which something happens during a certain period of time (i.e. 72 Beats per Minute) (Cambridge-Dictionary, 2016).

Entity Relationship Diagram (ER)



List of Figures

FIGURE 1: RICH PICTURE OF THE PROBLEM

FIGURE 2: RICH PICTURE OF THE SOLUTION

FIGURE 3: OPENING MENU OF THE APPLICATION

FIGURE 4: ACTIVITY WHICH THE USER HAS THE OPTION TO SEARCH FOR AN EXISTING PATIENT, OR ADD A NEW ONE

FIGURE 5: SCREENSHOT OF THE RECORDING ACTIVITY

FIGURE 6: THE DISPLAY OF THE LISTVIEW WHICH STORES THE AUDIO FILES

FIGURE 7: HOW THE INFORMATION WILL BE PRESENTED TO THE USER.

FIGURE 8: HOW THE PATIENT INFORMATION WILL BE DISPLAYED.

FIGURE 9: ALLOWS THE USER TO ADD NEW PATIENT DETAILS.

FIGURE 10: THE VARIOUS LAYERS WITHIN THE ANDROID PLATFORM.

FIGURE 11: BASIC DESIGN OF THE APPLICATION AND THE LINKS BETWEEN EACH PAGE.

FIGURE 12: INSERTS THE INFORMATION GIVEN INTO THE RECORDINGSTABLE

FIGURE 13: THIS FUNCTION ADDS THE AUDIO INTO THE DATABASE VIA ONCLICK LISTENER

FIGURE 14: BUTTONTAPPED()

FIGURE 15: BEGINRECORDING()

FIGURE 16: PLAYRECORDING()

FIGURE 17: STOPRECORDING()

FIGURE 18: SEARCHDATA()

FIGURE 19: SEARCHVIEW

FIGURE 20: SEARCH() FUNCTION

List of Tables

TABLE 1: RISKS AND THREAT WITH THE GENERAL MANAGEMENT OF THE PROJECT

TABLE 2: ISSUES WHICH COULD STEM FROM TECHNICAL DIFFICULTIES

TABLE 3: SAMSUNG GALAXY TAB PRO TABLET

TABLE 4: HTC ONE SMARTPHONE

TABLE 5: SAMSUNG A3 SMARTPHONE

Database Code

OnCreate () method to create tables

```

@Override
public void onCreate(SQLiteDatabase db) {

    //Create the tables
    String data = "CREATE TABLE " + PATIENT_TABLE
        + " ( PatientID TEXT PRIMARY KEY ,"
        + " Surname TEXT , "
        + "Forename TEXT ) ";

    String recordings = "CREATE TABLE " + RECORDING_TABLE
        + " (id INTEGER PRIMARY KEY,"
        + " PatientID TEXT, "
        + " Recording BLOB,"
        + " Date TEXT,"
        + "FOREIGN KEY (PatientID) REFERENCES " + PATIENT_TABLE + "(PatientID));";

    //Execute the SQL query - creating the tables
    db.execSQL(data);
    db.execSQL(recordings);
}

```

Initialise Tables and Columns of Database

```

//Declare Local Variables for Patient table
public static final String DATABASE_NAME = "patient.db";
public static final String PATIENT_TABLE = "patientTable";
public static final String COL_1 = "PatientID";
public static final String COL_2 = "Surname";
public static final String COL_3 = "Forename";

//Declare Local Variables for Recording table
public static final String RECORDING_TABLE = "recordingsTable";
public static final String COL_A = "ID";
public static final String COL_B = "PatientID";
public static final Blob COL_C = null;
public static final String COL_D = "Date";

```

Bibliography

AllWords, 2016. *Denoising*. [Online]

Available at: <http://www.allwords.com/word-denoising.html>

[Accessed 15 08 2016].

Android, 2016. *Android Interfaces and Architecture*. [Online]

Available at: <https://source.android.com/devices/>

[Accessed 15 01 2016].

Cambridge-Dictionary, 2016. *Frequency*. [Online]

Available at: <http://dictionary.cambridge.org/dictionary/english/frequency>

[Accessed 15 08 2016].

Collins-Dictionary, 2016. *Background Noise*. 10th ed. Boston: HarperCollins Publishers.

Farlex-Dictionary, 2016. *BPM*. [Online]

Available at: <http://medical-dictionary.thefreedictionary.com/bpm>

[Accessed 14 08 2016].

Jackson, W., 2015. Digital Audio: Technology and Terminology. In: *Android Studio New Media Fundamentals: Content Production of Digital Audio/Video, Illustration and 3D Animation*. Lompoc: Apress, pp. 33-41.

Kawachi, I., 1997. *Heart Rate Variability*. [Online]

Available at: <http://www.macses.ucsf.edu/research/allostatic/heartrate.php#definition>

[Accessed 11 01 2016].

Lome, S., 2016. *Heart Sounds*. [Online]

Available at: <http://www.healio.com/cardiology/learn-the-heart/cardiology-review/heart-sounds#S1>

[Accessed 10 01 2016].

MedEdu, 2015. *S1 Heart Sound Page*. [Online]

Available at: <http://www.easyauscultation.com/s1-heart-sound>

[Accessed 11 01 2016].

MedicineNet, 2016. *Definition of Heart Rate*. [Online]

Available at: <http://www.medicinenet.com/script/main/art.asp?articlekey=3674>

[Accessed 10 01 2016].

Miller-Keane, 2007. *Encyclopedia and Dictionary of Medicine, Nursing, and Allied Health*. 7th ed. Philadelphia: Saunders.

Overbaugh, J., 2015. *What are the top software testing methodologies?*. [Online]

Available at: <http://searchsoftwarequality.techtarget.com/answer/What-are-the-different-software-testing-methodologies>

[Accessed 14 08 2016].

Professionals, A. o. M. T., 2016. *Software Development Methodologies*. [Online]

Available at: <http://www.itinfo.am/eng/software-development-methodologies/>

[Accessed 14 01 2016].

Staff, M. C., 2014. *Heart Disease Definition*. [Online]

Available at: <http://www.mayoclinic.org/diseases-conditions/heart-disease/basics/definition/con>

20034056

[Accessed 11 01 2016].

Techotopia, 2014. *An Overview of the Android Architecture*. [Online]

Available at: [http://www.techotopia.com/index.php/An Overview of the Android Architecture](http://www.techotopia.com/index.php/An_Overview_of_the_Android_Architecture)

[Accessed 15 01 2016].

TutorialsPoint, 2016. *Android - Architecture*. [Online]

Available at: http://www.tutorialspoint.com/android/android_architecture.htm

[Accessed 15 01 2016].

Valerie Gay, P. L., 2007. A Health Monitoring System using Smartphones and Wearable Sensors.

International Journal of ARM, 8(2), pp. 29-36.

W3-Schools, 2016. *SQL Joins*. [Online]

Available at: http://www.w3schools.com/sql/sql_join.asp

[Accessed 13 80 2016].

WebMD, 2015. *Symptoms of Heart Disease*. [Online]

Available at: <http://www.webmd.com/heart-disease/guide/heart-disease-symptoms>

[Accessed 11 01 2016].

Wikipedia, 2014. *Fast Fourier Transformation*. [Online]

Available at: https://simple.wikipedia.org/wiki/Fourier_transform

[Accessed 13 08 2016].