IMPERIAL COLLEGE LONDON

DEPARTMENT OF COMPUTING

A digital interface designed for sharing diagnostic medical imaging with patients

— FINAL REPORT —

by
Laura Hagege LH

Supervisor: Fernando Bello

Submitted in partial fulfillment of the requirements for the MSc degree in Msc. Computing Science of Imperial College London

September 2018

Abstract

In a world where access to data is becoming faster and easier everyday, how could we prevent the society from the desire to have a better and more systematic **access to medical data**? In particular, in the field of **Medical Imaging**, which propose a range of routinely used methods to produce precise representations of the inner humans body and diagnose serious diseases? And especially, in the **United Kingdom**, where medical imaging data sound like a well kept mysterious treasure? Apparently we can't, and some solutions are popping up on the way to deal with it.

Indeed, some researches have already been led in the United States concerning the creation of a **"patient portal"** [1]. The main idea beeing to provide the patient with a suitable graphical interface that would display their medical images and help them to understand what they see. **Dr Fernando Bello** and **Pr William Cox** took the bet of digging in that particular area to explore the benefits and risks that could come from the conception of such an interface. They ended up to propose the effective realization of this interface, as a final personnal project to the **departement of Computing Science at Imperial College of London** together with the **Chelsea and Westminster Hospital**.

Consequently, the following report offers an overview of the work undertaken during the last three month on the creation of **"A digital interface designed for sharing diagnostic medical imaging with patients"**. From specifications to conception, throught design and testing, the aim of the following document is to present the different stage of the interface realisation, including issues, skills earned, and the final output that will be left as a basis for further development.

Acknowledgements

”It is usual to thank those individuals who have provided particularly useful assistance, technical or otherwise, during your project. Your supervisor will obviously be pleased to be acknowledged as they will have invested quite a lot of time overseeing your progress.”

Person to thank:

- I would like to express my deep gratitude to Professor Dr Fernando Bello

- Will Cox

- Co-workers at Chelsea and westminster hospital

- Developper community over the internet and around me

Contents

1 Introduction

According to a study published in January 2018 from **National Health Service (NHS) England** [2], 41.4 million of **imaging tests** have been reported in England between October 2016 and December 2017. Indeed, **medical Imaging** exams are routinely used all over the world to explore internal body structures and/or diagnose diseases. This generic term gather together various clinical methods, such as, **Magnetic Resonance Imaging** (MRI), **Computerized Tomography** (CT-Scan) or **X-Ray**, which allow to produce a 2D or 3D representation of physical intern structures.

Methods used for those review are common, and since the introduction of the **DICOM standard** in 1985 [3], so is the professional storage and communication system. However, when it comes to sharing diagnoses with patient, each country get to developp its own methods. In the **United Kingdom**, it is less than likely that a patients get, or even ask, access to their medical data. On demand and providing payment, one can get his plain clinical images. But generally, patients have only access to their clinical report, by means of a general practitioner, and the medical images remain a mystery black box that no one get to see.

Simultaneously, with the evolution of technology making access to knowledge easier day after day, access to **medical informations** is also more coveted and especially concerning **imaging data**. However, the issue about sharing those sensitive data, is not only to promote access, but essentially to make those images affordable for the common mortal. Researches have already been made in the United States concerning the creation of a **"patient portal"** [4] - fully designed to facilitate **patient understanding** - exploring the related opportunities, and scaling different levels of benefit.

Taking all those facts into consideration, **Dr Fernando Bello** and **Pr William Cox** have decided to deepen the subject of designing a **patient portal**. First by exploring the **benefits and risks** from sharing medical informations with patient and thereafter by effectively building such an interface. Consequently, after a year of background researches, they offered to work on the creation of **"A digital interface designed for sharing diagnostic medical imaging with patients"**, as a final personnal project to the **departement of Computing Science at Imperial College of London**, together with the **Chelsea and Westminster Hospital**. Aiming to build an interface **suitably made for imaging patients** in order to let them access and understand their medical results in the most significant and comprehensible way.

The following report contains an overview of the work undertaken during almost three month beginning late May 2018 under the supervision of **Dr Fernando Bello** and **Pr William Cox**. The following parts will developp in a logical order - for better understanding - the several steps reached during the conception of the interface, including issues, skills earned and personnal review.

//Introduce the plan - will do once it is certain

2 Background and Related Work

2.1 Project context

As mentionned in the introduction, **medical imaging** methods are widely used everyday to diagnose more or less serious intern physical diseases. Using different imaging methods such as **Magnetic Resonance Imaging** (MRI), **Computerized Tomography** (CT-Scan) or **X-Ray**, this allow to produce a precise 2D or 3D representation of the internal structure of the human body. Since the introduction of the **DICOM standard** in 1985 - developped therafter - which provide a unified protocole to store and share imaging data, storage and communication has become easier between different healthcare services. However, when it comes to data sharing there is no uniformed protocole neither on the manner nor on the fact that those data should be sahred. Naturally, once a exam has been perfomed, patients will automatically be given their clinical results, in terms of wellness or sickness but medical images will not compulsory be shared with them. Depending on the issuing country, medical images can be shared, either automatically or on demand, more or less easily with a patient.

In the limit of this project, the targeted area is the **United Kindgom** where effectiv access to medical imaging data is not one of a main concern. By default, once an exam is performed, only the medical report will be send to the referenced general practician. The patient will then need to take an apointment in order to get the information concerning his medical condition, more precicely to get the clinical report told by his doctor. No medical images would be shared or even shown at anytime of the process,neither to the medical expert nor to the patient. On demand, and provided payment of a notable amount, a patient can get his medical data on a CD or DVD, supplied with a default software reader built to deal with the special DICOM format. However, the kind of software that are provided on CDs or DVDs were initially made for clinician and may not be adapted for an average person and this is where the project emerges.

While patient desire to access their imaging data is growing, the idea to provide them with a suitable interface have emerged. By **suitable** this mean an interface that would only contains relevant and useful features, that would be instinctive and contains informations that are reliable for the user. Indeed, it is more than easy nowadays to find numerous of erroneous informations and explanations all over the Internet; the goal is to give the patient a safe environment where he could explore his data and understand his medical condition without any drift. Such interfaces are commonly called **"patient portal"** and the creation have already been some have already been created experimented locally on a sample of patient in Los Angeles [5]. However, this was, as my achieved work only part of a internal study and haven't reached the outside of it's location.

Nevertheless, for numerous reasons, building an interface for sharing medical diagnosis with imaging patient is not an innocuous task. Medical imaging are sentitive pieces informations and a misunderstanding can be significant, we could then ask what would be the risk and the benefit of creating a patient portal? These question have been subject to several litterature reviews [1] and especially to the realisation of a PhD by one of my supervisor **William Cox**.

2.2 Related Work

Before we started to actually discuss about the project, William provided me with several documents being either litterature reviews or articles (mentioned above) concerning works and researches directly related to the project. Among this he also provided me with the late stage report of his PhD containing the state of play of his work back then. Those papers have allow me to understand the background of the project and to identify the related issues and challenges.

The goal of Will's PhD is to identify the risks and benefits related to the creation of a patient-oriented interface. To be more precise I would cite directly from this report:

"Intuitively, there are benefits available from sharing images with patients. Indeed, there is a wealth of research available which assesses how visualisation aids increased understanding, or promotes communication. However, little research assesses the value of radiological images in this context. Moreover, no work assesses the risks associated with sharing patients images with them.This is the gap which this PhD will address. The research questions for this PhD are, therefore:

- *Is there additional benefit that can be extracted from diagnostic images?*
- *If so, what are the requirements to enable the realisation of this benefit?"*

Specifically, benefits are divied in two categories:

- *"Primary benefits benefits related to the rationale for acquiring the image, e.g. diagnosis, assessment, interventional guidance*
- *Secondary benefits benefits unrelated to the rationale for acquiring the image, e.g. education, communication, empowerment""*

At the moment, Will did a significant work on literatture review to set up the background and understand the issues related to the creation of a patient portal, I would cite "to ensure that this was a suitable subject for research and to identify any pertinent gaps in the existing knowledge.". He also carried out a survey, supplemented with 8 semi-structured interviews in order identify and scale the related risks and benefits. Results of those experience are available for consultation on Appendix 1. In the future the plan in to interview patient on the same model as clinician to evaluate the general opinion.

2.3 Goal definition

Once aware and familiar with the project background it was easier for me to understand the main issues and challenges and to have a precise idea of the goal to achieve. The objective of this project is not only to create a standard patient oriented interface but also to create an interface that can be **progressively adapted** in order to fit patient needs. On of the huge challenge is to create an interface that could be sufficiant and valuable, meaning it could provide enough information to guide the patient on the right track without any external contribution.

It has been agreed that we would together with Fernando and Will, define project specifications (in terms of design and content) and that all informations relative to benefits and content would be discussed mainly with Will. Fernando and William also advised me to install some already existing DICOM viewer made for clinicians in order the have an idea of the kind of interface that I could produce but while keeping in mind that the objectiv is not to simply do a copy of those readers.

The following part I will developp the project specifications that have been realized in the early stage of the project. Moreover, given that I have been given complete freedom in the choice of the tools and languages to use for building the interface I will explain the choices I have made according to this.

3 Define Specifications and Implementation Methods

3.1 Early specifications

Based on my previous experiences concerning the development tool for a third person, it was really important to me to define clearly the specification and limit of the project before starting to properly developp/ code it. This was on one hand to be sure to produce an interface that will suit my supervisors needs and on the other hand to avoid time waste in the future. Specifications concerning the graphical interface have been divided in five part: the basic requirements, the content, the design, the features and some further precisions.

During the early stage of the project we had several meetings with my two supervisor in order to define those specifications. These following specification are the specification we agreed at the time of writing the backgound and progress report.

- **Interface content:**
  - The interface should display patient images - images will be provided in DICOM format, and translated so that the patient can read them.
  - The interface must contain the clinical report and the simplifie version.
  - A link to NHS website will be given, so that patient could find general informations about their condition
  - Patient could get flag informations - to be filled by doctors - while exploring the images
  - Any other relevant informations related to what the DICOM files provides could be added
- **Interface functionalities:**
  The interface should provide:
  - One doctor oriented window: so, they can fill in datas (images, report) and add flag to images at their conveniance.
  - One patient oriented window: read only data (no modification allowed) and the possibility for patient to chat with doctors.

  My main concern - in the context of this project - is to focus on the patient oriented side and see how far I can lead this project. This part can be really time consuming as it might need to be oftently readapted following the needs of my tutors.

Also, William recently sent me detailed general specifications concerning the patient oriented interface content/functions - Appendix 2

- **Interface design:**

  - Imaging display will depend on the provided images (MRI, CT) but not on the part of the body. Will also gave me on demand description concerning images to be display and the way to deal with it Appendix 2.
  - Provide a side by side or other relevant organization that would allow the patient to get the images and the report together in a relevant way

- **Further precisions:**

  - No access to any database will be provided for the current project (security issues)
  - Access to the interface will be local, patient would be given (upon request) a CD with their images loaded on the interface; this wont change patient access to datas but should make them want to access it
  - Interface should include user specification/precisions for patient
  - Benefits/specifications will have to be defined before starting implementation
  - Interface should be windows portable

- **Interface evaluation:**
  At some point the interface should be evaluted by a panel of patient that

  would be ask to use it and make feedbacks.

Eventually, thoses specifications have evolved during the realization of the project, adjusting the task the the work already performed and the encountered issues. Moreover concerning the interface Design, Will provided me with

3.2 Choosing tools- C++ and Qt

A lot of tools are available on the Internet to create GUI interface such as Qt, WxWidget, GTK+, FLTK, FOX and some others. I couldn't compare all of them so I have decided to focus on Qt, WxWidget GTK+.

- Tools Comparison:
  I looked accross the Internet to get some testimonies about the different tools and I tried to distinguish them following several criteria - see grid below. According to these criteria and considering that Qt is highly recommended for beginners, I have finally decided to use Qt for this project.

Criterias\Interface	Qt	WxWidget	GtK+
Chat doc	+++	+	++
Cross-Platform	Yes	Yes	Yes
Open Source	Yes	Yes	Yes
Langage	C++	C++	C++
Flexibility	High	Low	Medium
Performance	High	High	High
Documentation	+++	++	++

Figure 1: LQt, WxWidget, GTK+ comparison table

- Qt familiarization:
  In order to get used to this new tools I have decided to do Openclassroom tutorials [2]. Those tutorials have helped me to install QtCreator and to begin with some basic exercices to get familiar with Qt. I still have some tutorial to do at the current moment but I am feeling comfortable with it.

4 Understanding Imaging File Format: DICOM standard

4.1 Definition

1.Definition

DICOM file format is a special software integrated standard format dedicated to ease data communication between different facilities in the Medical Imaging Field. This standard has been defined by the American College of Radiology (ACR) and the National Electrical Manifactural Association (NEMA) in 1985. DICOM format defines among a lot of others data dictionnary, data structure, file format and comes with a TCP/IP protocole to facilitate data transfer. Before the creation of this standard, it was difficult for different facilities to exchange imaging and imformations, currently this format is widely use for all medical imaging areas such as CT (Computed Tomography), MRI (Magnetic Reasonance Imaging), X-Rays, Ultrasounds, etc.

DICOM format is a well structured piece of work and DICOM standard website [1] provides a large documentation, however, it remains a challenge to fully understand what this is all about.My objective in the following subsection is to give a basic overview of DICOM standard features. As complete the DICOM standard could be, I also used two well formed websites [2] [3], to build my understanding on the DICOM file format.

4.2 DICOM File Structure

2.DICOM Information Model structure

Above all DICOM has been created to facilitate information exchange in the real world of patient healthcare services according to Imaging Field.Information that are contained in DICOM objects are related to **Real World** object that could be Patient, Location, Sudies, etc. According to those objects the DICOM Standard has define a **Model of the Real World** - see Appendix X - to identify relationship and interaction between those objects.

Consequently, all information available in DICOM objects will be related to those instances. Based on the **Model of the Real World**, the DICOM standard has defined the **DICOM Data Model**.This Data Model is made of classes called **SOP Class(es)**; one SOP Class is made of one **DIMSE Service Group** together with an **Information Object Definition (IOD)** - see figure 1 .

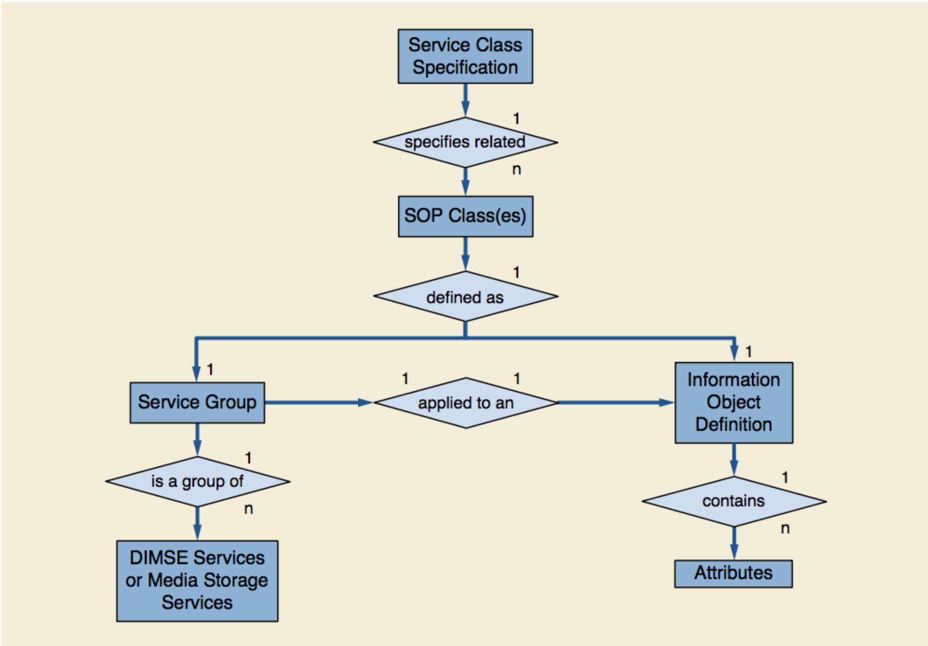


Figure 2: DICOM Informational Model Structure

Information Object Definition are abstract object which are intented to represent **instances of the Real World Model**. More precicely those objects aim at representing classes of element which have some common attributes. **Information Object Definition** are made of **Information Entities (IE)** that stand for the **Real World Object**. All **DICOM Object** must at least contains the SOP common module and the four main **Information Entities**: Patient, Study, Serie and Image.

The **DIMSE Service Group** specifies operations and notifications that can be applied on an IOD, **DIMSE** stands for DICOM Message Service Element, **SOP** are then used for message transfering between **IOD's**. Together those two element form a **Service Object Pair (SOP) class** that contains rules and semantics that rule the use of the services.

When DICOM was created the only instances where images, now other instances that are not images have been introduced but are not of our concern, therefore I will only use the term **DICOM Images** (DICOM Instance that are Image). As said **DICOM Images** are **Information Object Definition**, and must then contain the Image Module besides the four main entieies and the SOP common. This module will contain information about the image itself.

DICOM Images structure is defined on the Figure 2.

To more extend it is clear on the figure that each Information Entities has an attribute called UID, this stands for **Unique Identifier**. DICOM uses those identifiers to uniquely defines a wide variety of items to guaranty global uniqueness,

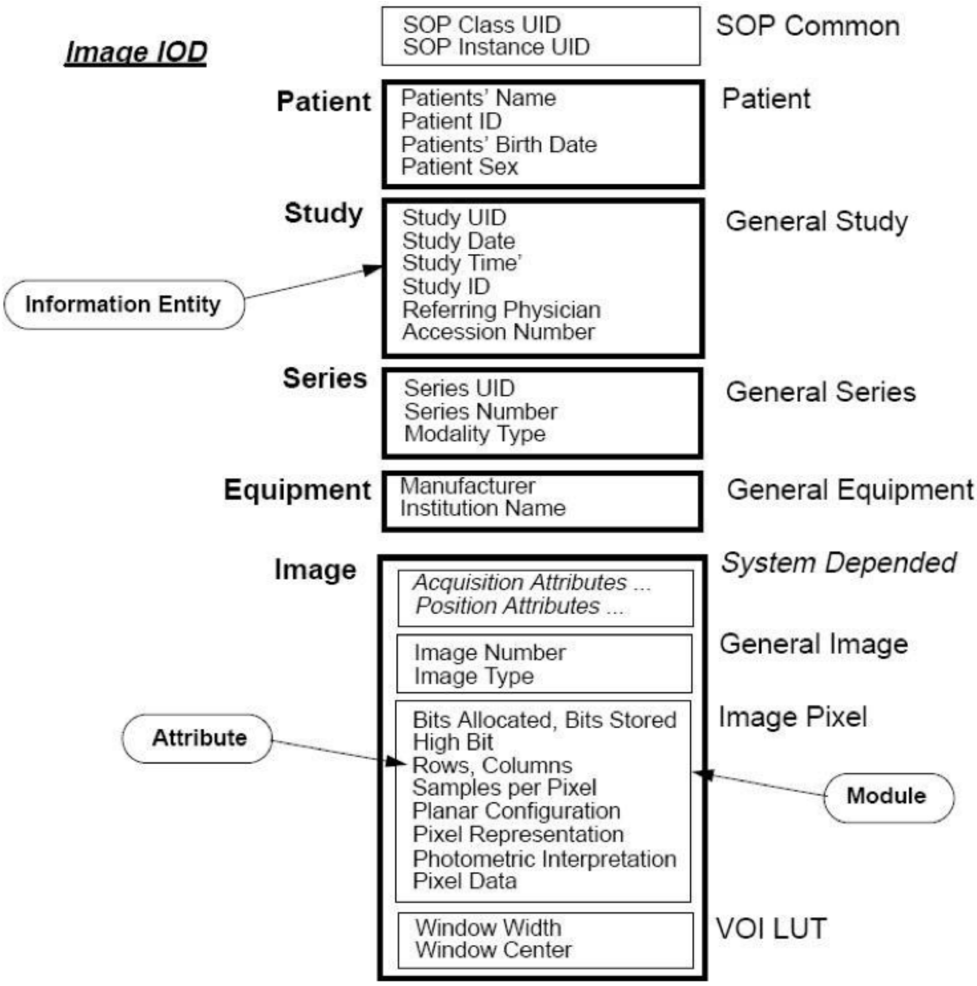


Figure 3: Image IOD structure

mainly among different countries, sites, equimment.

3.General File Structure

DICOM file objects contains informations about the **DICOM objects of the Real World**, which we know at this stage are held in **Information Object Definition**. Each DICOM file is composed of two instance: a **Header** followed by a **Data Set**.

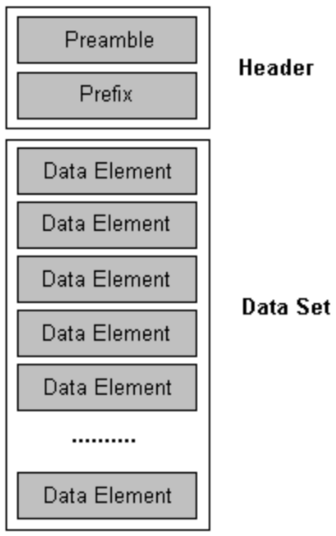


Figure 4: Basic DICOM File Structure

The Header contains 128 bytes preamble (which are all set to zero if it is not used) followed by 4 byte DICOM prefix (DICM). The header is not necessary included in the file but is usefull to make access to data easier, indeed the prefix allows to quickly acknowledge DICOM format. Besides, no structure is required for the preamble.

The Data Set is organised as consecutive **DICOM Data Element** (or Data Attribute) referenced in the DICOM standard. Those Data Element can represent various informations, from the patient name and birth to the pixel image information. More precisely one **DICOM Data Element** is one unit of information corresponding to an encoded **Information Object Definition Attribute** defined above.

DICOM Data Element are Tag Element therefore DICOM can be said to be a tag file format meaning each element is referenced by a unique **Tag Number** which define the element and its properties. The Data Set order Data Elements by increasing Tag Number. Each Data Element is made of the consecutive fields:

– **Tag Number**: ordered pair of 16bits unsigned integer of the form (gggg,eeee) representing the Group Number Followed by yhe Element Number. Eg:

(0028,0010): Group Number 0028 correspond to the Image group, Element Number 0010 correspond to the row and especially to the length of the image in pixels

– **Valure Representation**: defines the data type of the element, can be omitted because the Tag Number already implies the data type
– **Value Length**: either 16 or 32 bits, defining the length of the following value
– **Value Field**: even number of bytes containing the value of the element; the value field can contain the Value Multiplicity, which specified the number of value that can be encoded in the value field.

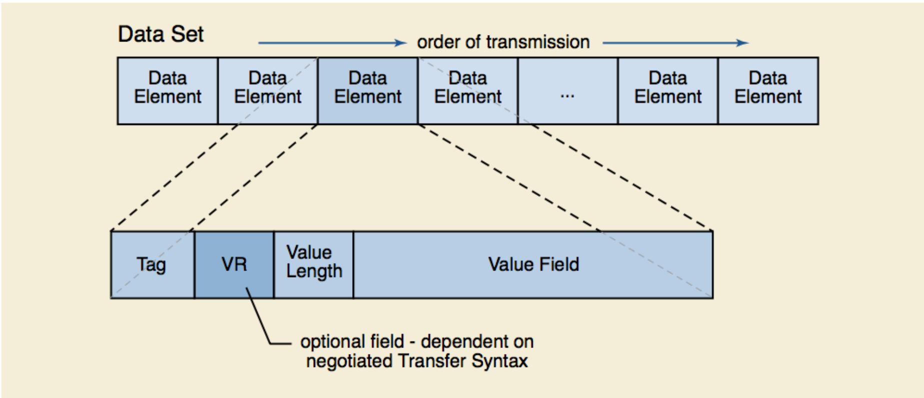


Figure 5: Data Set and Data Element structure

4.The DICOMDIR

In order the get their DICOM imaging data, patient will be given their DICOM folder. Every basic DICOM folder should contain one element called **DICOMDIR**. This **DICOMDIR** is a **DICOM File Object** containing paths to every DICOM Element that is in the folder; paths are organized within patient, study, serie, image. It also contains the genral informations according the the situation (patient information, study ID, date). The folder might contain other non DICOM element (e.g the report) that won't be recorded by the DICOMDIR. This DICOMDIR is the entry point for my interface and allows me to browse through the DICOM provided file and find path to images.

5 Implementation

5.1 Code Structure

1.What is DCMTK
DCMTK is an open-source collection of libraries and applications that has been made to deals with the DICOM format while developping an independant application. This software is widely used by hospitals, companies or private individuals that aim at creating DICOM related desktop software. DCMTK provide several classes and script in order to treat, construct, and convert DICOM files and an internal worklist server in order store, send or receive images.

DCMTK source code repository is available on Github and free for downloading. It is available for both Windows and Unix operating systems. For the purpus However dcmtk toolkit doesn't came as a prebuilt library

2.DCMTK classes and DICOM Element access

5.2 DCMTK library and Qt

5.3 Render DICOM Images with Qt and DCMTK

The first challenge of this project is to render a DICOM Image. As already explained the DICOM format is not trivial

I don't know what would be written about dcmtk and DICOM

Among other information I think this is relevant to explain how I managed to display the dicom images because this is not a trivial format

QT provide a class QIMage qnd blablabla

and had to get through dcmtk documentation to find the relevant function I will use !

Given a DICOM folder, this should contain at least the DICOMDIR file and another folder call DICOMDAT containing paths to series folder and images.

Usually DICOM folder contains only on study composed of multiple series, each serie being stored in separate folders. Then one serie file can either contain:

- One single frame DICOM Image
- Several single frame DICOM Images
- One multiframe DICOM Image

Objectively, single and multi frame images only differ by the size of the file.

1. Display one image:

The DCMTK library that I installed contains several classes that should make DICOM Images treatment easier. The class I used to deal with DICOM Images files is called DicomImage, the class structure and related functions are available on DCMTK website [8888]. The class is provided with four different constructors and depending on the given parameters this class allows to deal with single frame and multiframe images. Information about the constructor I choosed are available on Appendix XXX figure YYY

- Display single frame DICOM Image:

  *DicomImage *DcmImg = new DicomImage(path)*

- Display multiframe DICOM Image:

  *DicomImage *DcmImg = new DicomImage(path,0,index,1)*

  index being the index of the frame to display

The variable *path* is a string and contains the absolut path to the DICOM Image file.

Once I got the DicomImage object, I need to get the pixels in order to have the opportunity to use QImage class thereafter. Here again DCMTK provide me with the function *getOutputData()* - see Appendix XX figure ZZ -. The corresponding line of code is:

$uint8_t * pixelData = DcmImg- > getOutputData(8)$

Explain what is $uint8_t$

Finally I only need to use two classes provided by Qt to render the DICOM Image on my application:

$uint8_t * pixelData = DcmImg- > getOutputData(8)$

QIMage only takes pixel and a scene can only display QPixmap element see appendix

2. Store and display successive images of the serie

6 Design and Features

6.1 Design with QtDesigner

6.2 Features implementation

7 Results

7.1 Final output

7.2 Third party feedback

8 Evaluation

8.1 Personal evaluation

–¿ explain some choice
–¿ show calcul for different plan output

8.2 Ethic and LSEPI checklist

9 Conclusions and Future work

10 Appendix

10.1 Appendix - Expert Questionnaire Results from William PhD

Expert questionnaire

The questionnaire (APPENDIX II) was distributed to clinical imaging experts and 131 responses were received. Of these, 121 were valid responses. The questionnaire assessed:

- Participants' professional background and length of experience (Questions 1 & 2)
- Attitudes to potential benefits and risks of images sharing (Question 3)
- Support for image sharing (Question 4)

The findings from the questionnaire were as follows:

- Participants confirmed agreement by majority with the benefits/risks identified through the literature review
- Participants demonstrated support for sharing images with patients (79% (n=95) agree/strongly agree that image sharing is a 'good idea')
- Acknowledgement of the importance of context in image sharing to enable benefit realisation/mitigate risk

The questionnaire did not provide information on the following:

- Specific contextual factors necessary for safe and successful image sharing
- Information on how to manage safe and successful image sharing

Expert interviews

8 semi-structured interviews were subsequently undertaken with expert clinicians. The interviews were designed to confirm and expand on the findings from the questionnaire as well as to consider the requirements for enabling patients to extract benefits from access to images whilst mitigating the associated risks. The interviews were transcribed and subjected to thematic analysis utilising a constructive grounded theory approach (Charmaz 2006).

Results from the interviews included:

- Confirmation of the benefits and risks of image sharing identified through the preceding study
- Identification of key considerations for supervised image sharing:
    - Why share?
        - The importance of rationale
    - Who shares?
        - Clinician type
        - Patient type
    - What is shared?
        - +ive/-ive findings
        - Supporting information
    - When should sharing occur?
        - Supervised/independent?
        - On demand?
- Key requirements for benefit realisation via independent access to images
    - An explanatory layer
        - Clarity regarding user responsibilities
        - Understandable/accessible/trusted supporting information
    - Interactivity
    - Support mechanism availability
    - Data security

The data from these interviews will be further analysed in the context of data gathered from the planned patient interviews. This will involve checking that codes remain 'fit' and 'relevant' and will enable the identification of agreement and difference between the two groups.

Figure 6: Expert Questionnaire Results

10.2 Appendix - General Specification

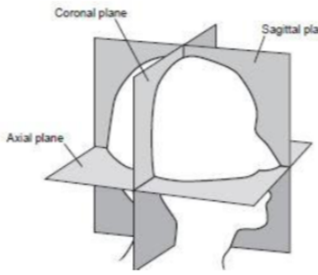
General requirements:

Initial interface (before opening imaging)

- Password control
- Disclaimer
    - For information only
    - Do not try to interpret
    - Any queries contact your healthcare professional
    - Some people may find seeing images of themselves upsetting, please consider whether you want to view them
    - Be aware that these images constitute your personal medical data – please be mindful of who you share them with
- ... any further which may be identified/emerge over course of the project
- Will need to be some thought given to how to make the functionality designed for clinicians accessible/useful to end user

Accompanying information

- Relevant abnormality information (from patient.info)
- Report – could terms be linked to definitions?
- Simplified report
- Normal comparator/diagram
- FAQs
- ... any further which may be identified
- Ability to display:
    - Multiple studies. E.g. interval studies.
        - This is when an X-ray of a broken wrist is taken at the time of the accident and then another X-ray is taken 6 weeks later to see if it has healed
            - Helpful if these could be viewed side by side
    - Multiple images side-by-side (e.g. axial view (top to bottom view) & sagittal view (side to side view)
        - See diagram below for the imaging planes:



  - Flagged image from a series opens first
    - It would be helpful to know how many times the user accesses the more detailed imaging vs how many times they only review the flagged image
  - Highlight abnormality
    - Can be toggled on and off
  - Hover over labels (anatomy)
  - Annotations made by clinician
    - Can be toggled on and off
  - Ability to remove any personal information/annotations (for users who want to share their image without revealing information)
  - Ability to share image onwards (e.g. via email)
  - ... any further which may be identified

Basic manipulations:

- Zoom
- Adjust contrast/brightness
- Invert greyscale
- Measure
- Ability to reset to original image appearance/undo manipulations
- … any further which may be identified
  - Useful to have an explanation for each of these
  - Pop up box?
    - What they do/how they may affect the image appearance
    - Disclaimer/warning that changing the image appearance may make it look artificially abnormal

Interactivity

Contact information for queries

Chat functionality

10.3 Appendix - Imaging Specification

Specific requirements by imaging modality:

Conventional X-ray

- The most common form of imaging
- Usually 2 images per body part imaged
  - e.g. Dorsi-palmar (DP) wrist (axial plane - top to bottom view) & lateral wrist (sagittal lane - side to side view)
    - DP wrist :

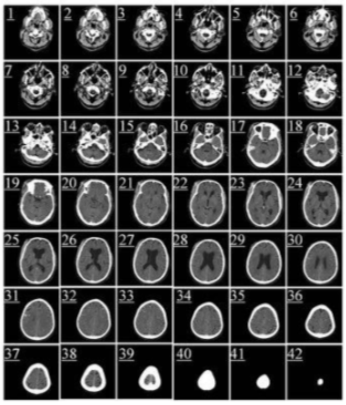
    - Lateral wrist:

- May be more than one body part imaged
  - e.g. hand (2 images) & elbow (2 images)
- May occasionally contain more than 2 images, if initial 2 don't display all the information –
  - e.g. if a leg is longer than the image size, need 1 image for top of leg, 1 image for bottom of leg

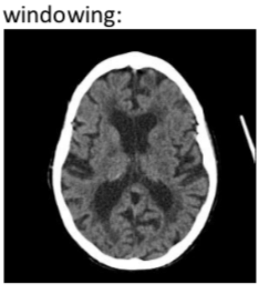
Computed tomography (CT)

Format information

- Multiple images in series with sequential slices to cover all relevant anatomy
  - e.g. brain scan:



- May contain imaging in more than one plane/utilising windowing to highlight anatomical structures of different density
  - e.g. brain images in axial/sagittal/coronal plane
  - E.g. brain images which have been 'windowed' to show only the bone of the skull in order to make it easier to identify fractures
    - Normal windowing:



    - Bone windowing:

Requirements for display

- Should be able to open the study with the 'flagged' image first. User can then view further imaging if necessary
- It would be helpful to know how many times the user accesses the more detailed imaging vs how many times they only review the flagged image

Magnetic resonance imaging (MRI)

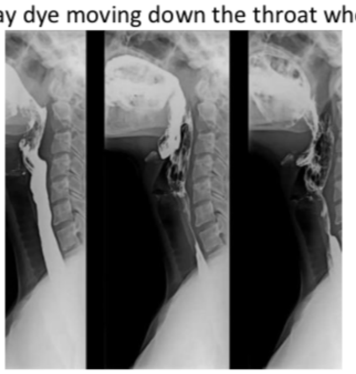
- Similar to CT
  - Can contain a lot of images which are arranged in sequences
  - Can have images displayed in different imaging planes
  - Different sequences show different parts of the anatomy by demonstrating different types of tissue (similar to windowing in CT)

Ultrasound (U/S)

- Similar to CT
  - Can contain a lot of images, although sometimes only 1 or 2
  - All images will be acquired in the same imaging plane

Fluoroscopy

- Similar to CT
- Can contain a lot of images
  - Images may be acquired in multiple anatomical planes
  - Images sequences may demonstrate time-lapse movement (similar to a video)
    - E.g. X-ray dye moving down the throat when swallowed



    - Helpful to be able to present these images like a slideshow so that they look like a video

Other points:

I have access to limited DICOM datasets but I do have some CT studies and fluoroscopy

These are quite large files and come on a CD bundled with an existing DICOM viewer – do you have a preferred secure method for me to transfer this data to you?

I also have a large bank of conventional X-ray images which are in fw.png format. These should be easier to transfer. It would be good to use these, too.

10.4 Appendix - DCMTK: Relevant classes and functions declarations

◆ DicomImage() [1/5]

```
DicomImage::DicomImage ( const char *          filename,
                         const unsigned long   flags = 0,
                         const unsigned long   fstart = 0,
                         const unsigned long   fcount = 0
                       )
```

constructor, open a DICOM file.

Opens specified file and reads image related data, creates internal representation of image data. Use getStatus() to obtain detailed information about any errors.

Parameters

- **filename** the DICOM file
- **flags** configuration flags (see diutils.h, CIF_MayDetachPixelData is set automatically)
- **fstart** first frame to be processed (optional, 0 = 1st frame), all subsequent use of parameters labeled 'frame' in this class refers to this start frame.
- **fcount** number of frames (optional, 0 = all frames)

Figure 7: DicomImage Class Constructor

◆ getOutputData() [1/2]

```
const void* DicomImage::getOutputData ( const int             bits = 0,
                                        const unsigned long   frame = 0,
                                        const int             planar = 0
                                      )
```

inline

render pixel data and return pointer to internal memory buffer.

apply VOI/PLUT transformation and (visible) overlay planes. internal memory buffer will be delete for the next getBitmap/Output operation. output data is always padded to 8, 16, 32, ... bits (bits allocated). Supported output color models: Monochrome 2 for monochrome images and RGB (or YCbCr_Full if flag CIF_KeepYCbCrColorModel is set) for color images. The rendered pixel data is always unsigned.

Parameters

- **bits** number of bits per sample used to render the pixel data (image depth, 1..MAX_BITS, 0 means 'bits stored' in the image) (MI_PastelColor = -1 for true color pastel mode, EXPERIMENTAL)
- **frame** number of frame to be rendered (0..n-1)
- **planar** 0 = color-by-pixel (R1G1B1...R2G2B2...R3G3B3...), 1 = color-by-plane (R1R2R3...G1G2G3...B1B2B3...) (only applicable to multi-planar/color images, otherwise ignored)

Returns

pointer to internal memory buffer containing rendered pixel data (if successful, NULL otherwise)

Figure 8: getOutputData function definition

10.5 Appendix - Qt: Relevant classes and functions declarations

QImage::QImage(int *width*, int *height*, QImage::Format *format*)

Constructs an image with the given *width*, *height* and *format*.

A null image will be returned if memory cannot be allocated.

Warning: This will create a QImage with uninitialized data. Call fill() to fill the image with an appropriate pixel value before drawing onto it with QPainter.

Figure 9: QImage contructor

QPixmap QPixmap::fromImage(const QImage &*image*, Qt::ImageConversionFlags *flags* = Qt::AutoColor) `[static]`

Converts the given *image* to a pixmap using the specified *flags* to control the conversion. The *flags* argument is a bitwise-OR of the Qt::ImageConversionFlags. Passing 0 for *flags* sets all the default options.

In case of monochrome and 8-bit images, the image is first converted to a 32-bit pixmap and then filled with the colors in the color table. If this is too expensive an operation, you can use QBitmap::fromImage() instead.

Figure 10: QPixmap static function

10.6 Appendix - Interface Design Specification

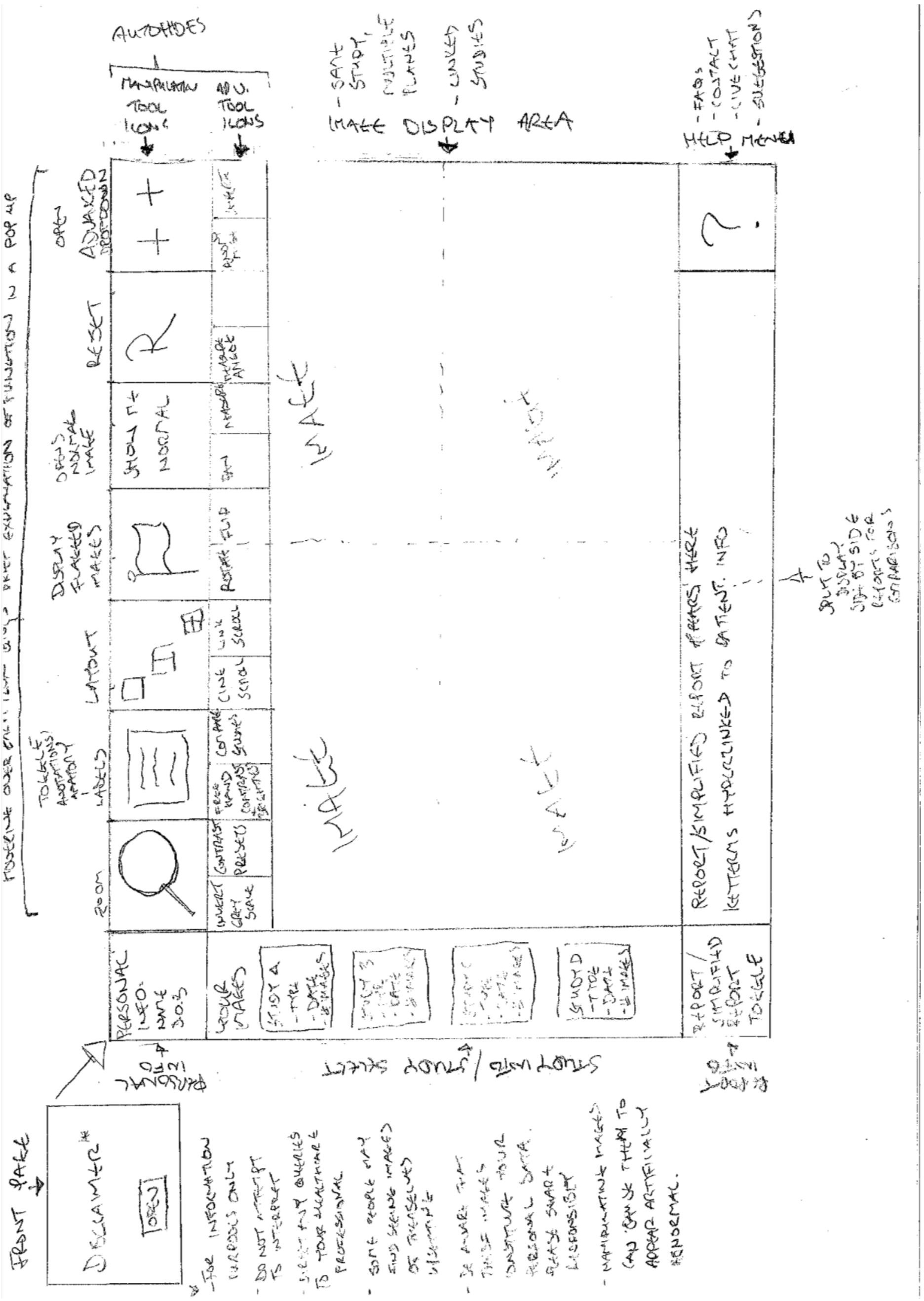


Figure 11: Interface Design Drawing Specification