Mediso

Innovation & Entrepreneurship Project Documentation

Table of contents

Executive Summaries

Architecture

The architecture part of this report is based on explaining the current architecture solution provided by the company, the future architecture solution proposed by the team doing this job and the potential risks of both implementations. Moreover, the background knowledge for both solutions is detached, as well as the application modules that are currently available with the implemented solution by Mediso.

The current application is sold as a license depending on the contract negotiated with the client. Also, the company prepares and configures the DICOM server in case the client requests it. In the case of an incidence, the client can contact Mediso and request support to fix the problem. The Operating System used is Windows.

The alternative application developed by Mediso is a new version that runs on Docker images and the Operating System will be based on Arch Linux. Apart from these innovations, the application works in the same way.

Among the potential risks analysed, they can be classified into several categories, like user management, change management, physical security, and performance and services. Inside each category, some criteria is defined and compared for both implementations. Finally, a recommendation is given for each part of the risks analysis.

Cybersecurity

As the following report on Cybersecurity at Mediso can be long to some, and a bit too detailed to others; especially to those that are coming from an executive role, we did our best effort to summarize the effective actions that we recommend to be taken: important topics in our analysis are Database Structure, Database Access Control and Remote Access.

We recommend the database installed locally to hold patient's data to be built so to minimize the attack surface towards both the local network and the internet. We suggest to divide it into two parts, one of them containing the anagraphics and the other holding the actual imaging and medical data; a patient can be identified through a unique string across the two databases. This way, each of the two parts can be kept in separate entities, be it physical servers or docker containers, and loaded into memory to minimize the chance for an attacker to obtain valid associations between sensitive clinical information and hints to identify who they belong to; a data leak severity can thus be significantly limited.

For Database Access Control, we recommend switching away from SQLite, towards either MySQL or Postgres. Another option that we cover is to consider Macaroons to solve the access control problem. Macaroons might offer better qualities, but possibly at the expense

of performance. In the decision which one of the three options to choose, we recommend that Mediso considers their requirements related to relative read/write performance in their applications; and to also consider their abilities in working with newer technologies (such as the Macaroons), or to decide to play it safe with technologies that have already been more broadly documented (such as MySQL and/or Postgres).

As for Remote Access, we will give advice on the need for a Virtual Private Network (VPN) and a Virtual Desktop Infrastructure (VDI). After this is made clear, we recommend a partner to set this up (Citrix). In addition, we recommend some general safe practices for password creation, and conclude with a discussion on the benefits of using two-factor authentication.

Business research

Medisos' product InterView Fusion has been evaluated from the business perspective. In the first part, all the steps and findings of the competitor research will be presented. In the second part, we proposed a new business model for the further development at Mediso which gives an overview of the key aspects to enlarge the scope of potential customers. To give an illustration, we chose some strategies to promote the brand image of Mediso which helps to build trust with existing customers and also among the new customers. In Conclusion, we describe a comparison between the ideal business model for Mediso and existing influential competitors to analyze and learn the opponent's business model which provides a deeper understanding to further develop the foundation and feasibility at Mediso.

Architecture

Introduction

This document is presented as the architecture analysis developed for the course I&E study related to the problematic presented by Mediso. One of the goals of Mediso is currently developing its application InterView™ FUSION and the goal of this report is to compare different architectures presented during our lessons. Thus, several aspects such as the physical security, cybersecurity risks and the alternative architecture proposed need to be evaluated.

The document is structured as follows. In Chapter 2 the activities, the partners and the technologies used by Mediso are analysed. In Chapter 3 the application under study is described, as well as the current application architecture, the proposed alternatives architectures and the actions to be taken in favour of that transformation. Afterwards, in Chapter 4, the risks of the migration are analysed, including the user management, change management, the physical security and the performance of the alternative. Finally, in Chapter 5 the conclusions of the are presented.

Background Knowledge

In this section we are going to provide a brief summary of the core technologies that are being used by Mediso in the hospitals to design and use the InterView™ FUSION software.

Network architecture

Due to the privacy requirements, the current network architecture applied by Mediso is local. Hence, for each hospital where Mediso has placed a software unit, the deployment is local. Therefore, the data generated by the hardware and processed by the software never leaves the local architecture of the hospital. In that way the medical standards for patient's data privacy are fulfilled. It is important to take this into account, so future deployments have to manage data to comply with the data privacy rules of the medical domain.

Hence, each hospital has a local server where all the data is collected. The software is also installed locally. All the data from the database can only be accessed locally.

DICOM Servers and Databases

DICOM is an abbreviation of Digital Imaging and Communications in Medicine. It is an international standard for the storage, the communication and the exchange of digital

medical images such as SPECTs or MRIs and other important data that are linked to the medical data, such as personal information.

The DICOM standard is encapsulated in the InterView™ FUSION software as well as in the hospital network and the local computer machines enabling the digital infrastructure of the hospital to communicate and store the data in an easy way.

The performance and safety of a DICOM network are critical, mainly due to the large volume of data which are transferred through it but also the importance of the data in terms of security and privacy. These data may include diagnostic images, patient and exam information, diagnostic reports, etc. For this reason, many medical institutions choose to build a dedicated, isolated and high-performance network as their DICOM network, separating it from the usual internal Local Area Network used for instance to exchange administrative data and files.

Containerization

Containers technology allows to allocate applications in an isolated way. They package code and all the needed libraries, system tools and system settings for the application to run. Hence, each application can be containerized having isolated resources. Containers also provide with security, meaning that data from a containerized application will not flow outside of the container. A container image become a container runtime when deployed on top of a container tool such as Docker or Kubernetes. Containers isolate the software from its environment, these containerized applications can run from different machines from the same container images. Several containerized applications can run in the same machine by sharing the OS kernel.

In contrast with virtual machines that virtualize the machine’s hardware, containers virtualize the OS. Therefore, containers provide enhanced portability, they are light weight and easy to boot. As well as containers present great opportunities in Mediso’s activities, there are also some issues that need to be analysed to avoid risks.

Cloud services

Cloud services provide to users with computational power, storage space etc via remote servers. Cloud technology has opened a new field that allow customers to perform its activities in the cloud without the need of deploying big local infrastructures. However, data privacy is one of the biggest concerns in cloud services. Therefore, deploying an application in the cloud means that all the data generated by the application will be in a remote server and that it will never be fully controlled by the application’s owner.

Application

The application approached in this document is a multi-modal visualization and evaluation software, whose name is InterView™ FUSION. It is built using technologies such as image processing algorithms and tools for measuring and evaluating the different medical imaging modalities. One of the core functionalities of this software is the study and analysis of the existing medical scanning techniques, such as SPECT, PET, CT or MRI. The evaluation is attached using various views and automated algorithms. For this purpose, there are several specialized tools that provide a detailed evaluation of the images, combining advanced visualizations and interactions.

The routine also contains modules that could be used in different scenarios, as well as a live reporting system, an option to work in the cloud that includes workstations and servers for the OS. These application characteristics are break down in this chapter.

A) Workspace architecture

Workspaces function as different screens organized in several tabs. The maximum number of screens open at the same time is 16. They have some features, like the option of adding or removing a workspace, the possibility of creating user-defined layouts, the duplication of a view to a new workspace, or the inter-workspace synchronization.

B) Layout management

The layouts are other of the features included in the application. They provide helpful information for the purpose of InterView™ FUSION. The management of these layouts is also provided, with various features like the layout grid definition, the intelligent layout selector, the layout item exchange or the live layouts (palette, 3D views, orientation, view types...)

C) Views

The views are other of the core functionalities of the application, which range from basic to complex. Among them, we can find the “Volume View”, the “Tiled View”, the “Unified volume view”, the “Volume Rendering” view (VR), the “Maximum Intensity Projection” view (MIP), the “Time Activity Curve” view, the “Profile Curve” view or the “Histogram View”. They show information of the distinctive characteristics that the software stores.

D) Fusion engine

There is an extension fusion engine that is optimized to work with all kinds of modalities, up to four different images in parallel. Regarding the features of the fusion engine, it should be highlighted the hierarchical registration, the types of sampling modes (uniform and planes), the types of sampling spaces (union, intersection or user-defined bounding boxes) or the registration techniques, such as the manual, semi-automated and fully automated one.

E) Measurements

Regarding the measurements, there is a wide variety of ROIs and VOIs statistics taken to help the evaluation process. All of these measurements are stored in a ROI's/VOI's table, which is Excel compatible. Moreover, it is possible to export all these results. The main measurements types are ROI/VOI types, ROI statistics, the measurement types and the ROI evaluation tools.

F) Toolboxes

Toolboxes are provided to modify, control or change the content of the distinct views. There are several tools available, such as the "Quick functions", "Mouse modes", "Movie", "Bookmark", "Blending", "Palette", "Cut/Crop" and "Reorientation". Each of them has their own options, which turn out to be useful in concrete scenarios.

G) Operations and filters

Firstly, the arithmetic operations covered within the applications are useful to normalize both image size and image resolution. Once this is performed, images can be compared. Also, it is possible to set individual weights for the normalized images. Secondly, the application includes 2D and 3D filters for spatial and frequency domains. There are more than 15 filters between both domains, like Bilateral, Median, Gaussian, Metz or Wiener filters.

H) Live reporting

Other of the remarkable characteristics of InterView™ FUSION is the live reporting system that they have. It provides real-time interaction between several views. Once one of the views is captured, they could be altered in the report page. Other features of this system in the program are to represent measurements obtained and add comments, to set both header and footer attributes and to export these previously mentioned report pages into distinct formats (DICOM, PNG, JPG or PDF).

I) Dedicated modules

The application also includes three dedicated modules for different purposes. Each of them attaches one concrete task inside medical imaging domain. At first, there is a solution focused on planar image enhancement called Tera-Tomo™ 2D Planar Image Enhancement Module, which uses some algorithms to improve the image quality. The second dedicated module is called Tera-Tomo™ 3D SPECT Bone, and it focuses on boosting the results for SPECT bone structure scenario. For that purpose, it applies the anatomical information obtained from CT scan. Thirdly, the Tera-Tomo™ 3D CT is a dedicated solution that presents a reconstruction method that is used to generate optimal patterns for several body parts.

J) Scanning tools

Regarding the dedicated tools for each scan, they should be divided into the scan's types in several categories. Among them, they are found CT tools, SPECT-CT tools, PET tools and

PET-CT tools. There are almost 10 distinct tools offered to improve medical imaging analysis.

K) Cloud solution

In order to supply better performing of the solution, the app has both a client-server and a standalone workstation. Both server and workstations can be accessed remotely, and they are compatible with both Windows and Linux OS. Moreover, InterView™ FUSION provides extra-large dual monitors to improve the visualizations for hospitals.

L) Import/Export options

Finally, the last options included in the software is the import and export functions. As it works using a DICOM server, the most important importing option is to import from raw to DICOM format. Regarding the exporting options, it is possible to export everything at any time. Also, it includes disc (CD and DVD) burning capacities.

Application architecture

The main purpose of this section is to explain the various types of architecture solutions proposed. Firstly, the current status of the application architecture will be detached as well as the goal of the current developments. Afterwards, the alternative proposed status will be detailed. In this part, it is analysed how the current status can be improved into a better solution. In each of these first parts, it is also required to highlight the main problems and opportunities for improvement. Finally, the third part of this section is devoted to give a reasoning for the needed changes of both the architecture status, and to make a small comparison between them.

Current status

In order to address the current status of the application we have performed several interviews with Adam Istvan, responsible of the course IE Study at Eötvös Loránd University.

The current application is sold as a yearly/monthly license depending on the contract established with the client. Mediso also prepares and configures the DICOM server in case the client requests it in its Data Centre. Finally, in the case of an incidence, the client can contact Mediso and request support to fix the problem.

Regarding changes and versions in the application, they are performed upon client request and they are communicated to the rest of the clients just in case a software problem is detected, otherwise the change/version is only applied on the specific client application. Clients can not develop new changes on their own. Mediso can also develop new versions of the software and they can be notified to the clients.

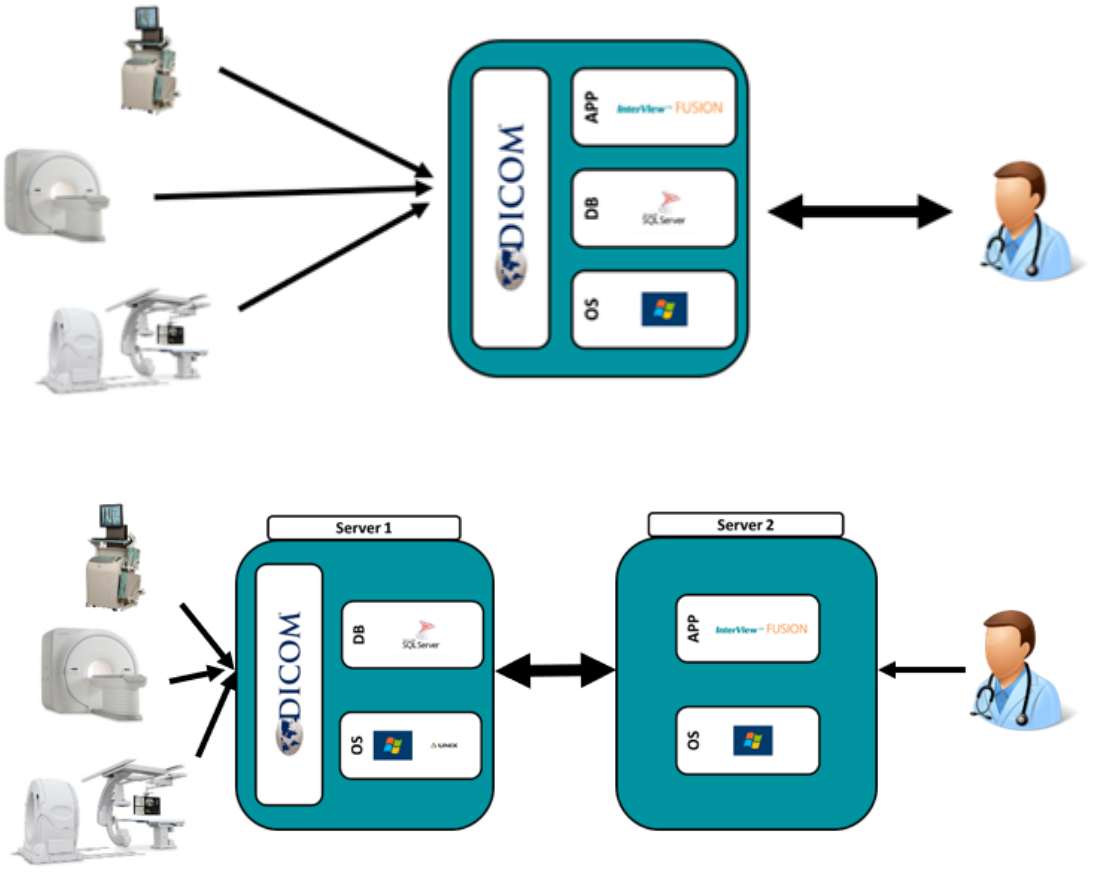
Mediso always maintain a user to perform the changes and/or install the new version in each client DICOM server with a VPN. The client cannot perform any changes on the application since it does not have any modules. There is no authentication to access to the application.

Consequently, there are no roles for the different types of users. The solution is installed in a server compliant with the DICOM standard (DICOM server from now on). The characteristics of the installation are:

Application: developed by Mediso.

Database: up to the client, but usually SQL Server. Does not need to be in the same server as the application. It contains DICOM files.

Operating System: Windows.



Regarding access from different hospitals, this can be done right now online if the hospitals are inside the client network and are able to access the DICOM server.

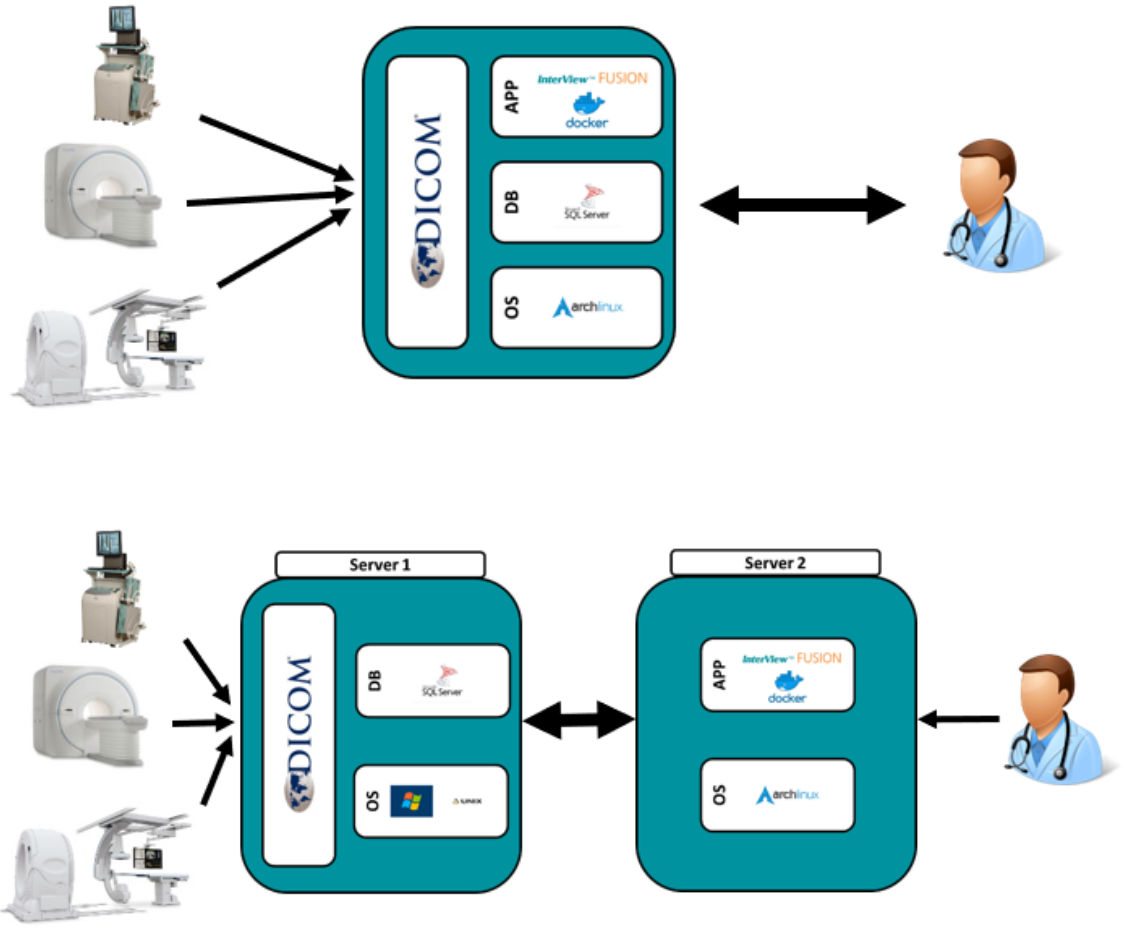
Change management does not exist in the application since it works by versions. Whenever a new version is released, it is sent to the client. This client is responsible for updating the software in its server (usually is installed by the Mediso user that is maintained in the server).

Right now, it is not possible to connect the application to other systems, the database needs to get fed by the hospital with its own data from other applications.

There are no profiles nor user management inside the application. Anyone with access to the DICOM server could be able to access to the application, so the security needs to be placed at an OS level.

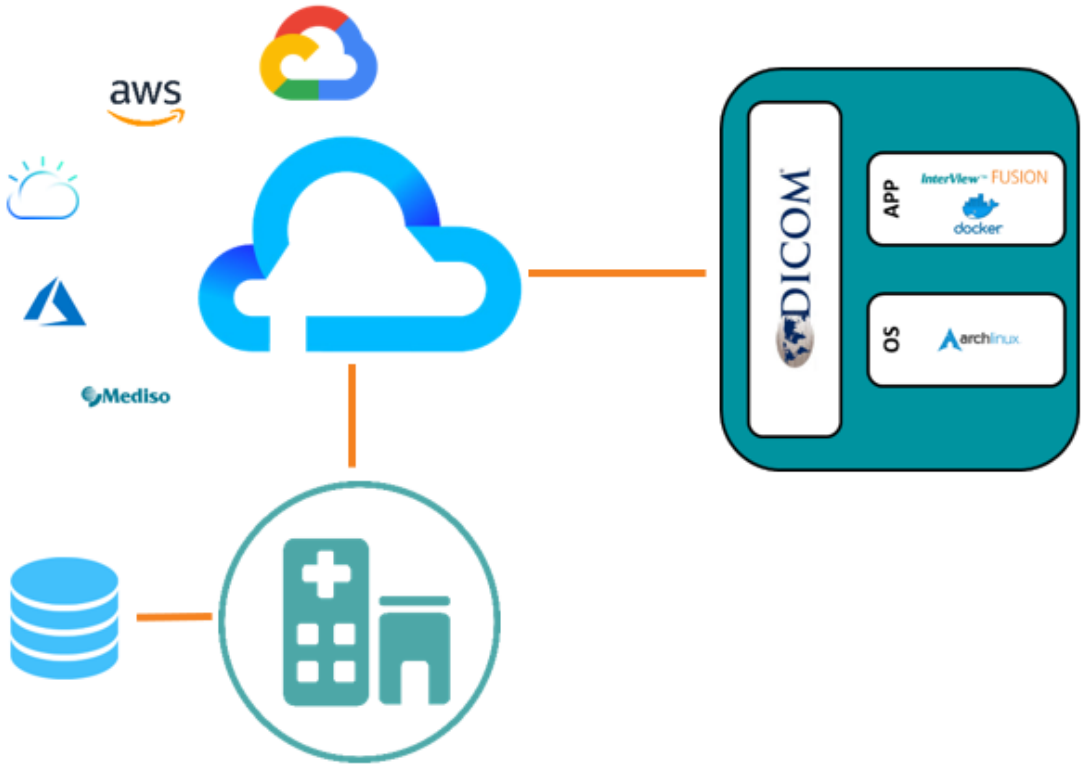
Future status

Right now, Mediso is developing a new version of the application that run on Docker images and the Operating System will be based on Arch Linux. The other parts of the flow diagram remain the same as in the current status of the application. This can be observed in the following figure:



Alternative status

The objective of this document is to compare the current status presented before with the alternative status already defined, where the DICOM server is placed in the cloud by MEDISO and client's access to their functionalities remotely.



This way the change management process would be done by Mediso itself without having to depend on the different hospitals, also different business models could be applied. This could be considered as an important advantage, as the company could work independently from their clients. Nevertheless, communication would play a bigger role in order to secure the information that will be travelling to the server.

As a summary, the changes made in this application update are the addition of the containerization solution Docker to the application, and the change to Arch Linux based Operating System.

Risk analysis

We present the risk analysis where we compare the current status (basically combining both the current status plus the future status presented in the previous section) and the alternative status presented (using Mediso owned cloud, not third party one).

User management

	Current status	Alternative status	Recommendation
Users gain unappropriated access to the application	**High impact** and **medium probability**. Since there is not authentication is possible that someone with access to the server could access also to the application if not managed properly by the hospital.	**High impact** and **low probability**. If developed in Mediso`s cloud, the user management could be handled by Mediso directly. So, it will be a shared risk since the database will be still in the hospital servers.	If Mediso can afford the risk of user management, it can provide some extra value for the product.
Users are created/deleted in an inappropriate way			
Permission of users are beyond their responsibilities			

Change management

	Current status	Alternative status	Recommendation
Changes are promoted into production by the developers	**High impact** and **medium probability.** New versions are sent to the clients and is their responsibility to install it, nevertheless usually the Mediso user is the one that install the new version.	**High impact** and **medium probability.** If handled properly, Mediso could manage the change management of the application in a centralized way.	By handling the change management in a centralized application, the risk will be addressed only once per new version and could finish with the problem of having different versions in different hospitals. Also, whenever a high impact or a security change is needed, it could be done directly without having to notify every client.
Unappropriated changes are made in the production environment	**High impact** and **low probability.** Since the clients cannot perform changes in the application, this risk can only happen if a Mediso user perform unauthorized changes.	**High impact** and **medium probability.** If they are handled properly by Mediso, these risks could be reduced to low probability.	
Test of changes are not appropriately performed	**Medium impact** and **low probability**. Changes are only sent to the clients when they are prepared and tested properly by Mediso.		

Physical security

	Current status	Future status	Recommendation
Data centre is not appropriately secured	**High impact** and **unknown probability**. Since each hospital has its own data centre and its responsible for its resilience.	**High impact** and **unknown probability.** If handled by Mediso, the risk probability will depend on how well secured its data centre is and how the resilience is managed.	By offering a secure solution to this problem (by addressing different certifications for resilience and secured communications for example) Mediso could also add value to its product.

Performance and service

	Current status	Future status	Recommendation
Load times are longer because of the network's saturation	**Medium impact** and **unknown probability**. Since this risk will be addressed by each hospital and depends on its network. Nevertheless, containerization provides more flexibility and isolation between modules leading to more effective architecture management.	**High impact** and **high probability**. Mediso will need to address this problem by measuring the performance of its network and acting based on the results. Probably it will need to report service indicators to its clients based on the contracts established with each one.	The current status provides less risk for Mediso since the transition to the cloud services offer will lead to a constant management of the service of the product.

Server memory overloads	**High impact** and **low probability**. Mediso needs to provide advice for the optimal machine needed and the best possible installation to avoid this problem. Also, containerization provides isolation and flexibility in dedicated resources for each application. Hence, avoiding memory overloads.	**High impact** and **low probability**. Mediso has a good knowledge of its tool and can deploy it in appropriate machines to avoid this problem.	Again, the current status is better for Mediso since it has no responsibility in maintaining the client server. Nevertheless, it can lead to an extra value to the product.
Different hospitals cannot connect	**Low impact** and **unknown probability**. It depends on how the product was sold and the client's network.	**Low impact** and **low probability**. In the case of using cloud services hospitals could share data. However, this approach is confronted with the data privacy in the medical domain. Not sensible data to privacy could be shared.	By using cloud services Mediso could even offer some networking service for hospitals in order to share information or diagnostics between them. Nevertheless, it carries a high risk based on the information that will be managed by the software
Scaling	**High impact** and **medium probability**. Containerization provides higher scalability. Anyway, the scalability depends also on each hospital needs.	**High impact** and **high probability.** Mediso will need to prepare its application and its data centre to face a potential high scalability. This can lead to extra costs and other risks.	The current status derives the scalability problem to the client. As we said before the cloud model will lead to extra costs, but it can offer high value to Mediso clients if handle properly

Conclusion

Based on the information we gathered during the lessons; we can say that the current development that is taking place will offer a better solution in terms of service to Mediso´s clients and it will probably help Mediso´s developers to perform future changes. So, it is a step forward.

After the analysis of the alternative status we propose, we think that a deeper analysis by Mediso is worth it. It will lead to high costs probably in order to arrive to the final solution, but it has the potential to offer a unique experience to Mediso´s clients if it can handle the regulation and privacy barriers in place.

Cybersecurity

Introduction

Cybersecurity has been gaining importance in the business world. In recent years, large companies and governments have been hacked, resulting in the loss of valuable data. In the case of Mediso, the risk would be that of losing sensitive patient data, and if this risk would not be properly addressed, this could lead to compliance issues (government/hospitals) and/or lose trust (patients). To help Mediso make these scenarios less likely, we have done the required research, and summarized everything into this report.

Our initial approaches

Our first steps were moved towards understanding the high-level structure of the target software in order to segment the attack surface and analyze possible weak points further. Initially, we came up with a simple description of the whole stack of the software and we brainstormed on a list of potential weaknesses while drafting ideas for their solution. The approach was to group them and to assign each group to a team member in order to perform a more detailed analysis of what the product could require in terms of security and privacy guarantees for customers while keeping an open eye on their commercial fungibility. In this first iteration, three main improvable areas were identified:

- **Database structure:** in order to reduce the attack surface over patients' personal data, we considered the idea of splitting the DB in a private section (with identifying data such as name and SSN) which is only loaded during a patient search, and a data section containing anonymized DICOM files along with a unique identifier. This way the container managing private data is only kept running when strictly necessary.

- **Database Access control:** at first a *Role-Based Access Control (RBAC)* is best suited, but also token-based access control has its advantages. As for requirements, imagery hardware should only be allowed to append files, doctors should only modify non-raw data and the administration should not have access to strictly medical information. This is somehow related to the structure but forms another section since it's a pretty vast field.

- **Remote access:** Doctors and other users should be allowed to run the software remotely in a seamless manner, thus opening a huge attack surface over the whole internet. Security concerns can be mitigated through the use of a VPN and a wise authentication structure; this shall consider password quality enforcement and recovery procedures, as long as two-factor authentication (2FA) methods.

Database structure

Mediso software currently implements a plain DICOM database in which all the data related to a patient is stored in a single file. This setting offers a large attack surface to database leaks because any information collected during the runtime of the server-side software potentially leads to the retrieval of sensitive personal data. Moreover, the anagraphics data regarding a patient is only used during the search in the database itself, and it's not necessary during the actual processing of the imaging information and diagnosis.

In order to reduce this attack surface to a minimum, one solution might be to split the database into two parts, containing:

- **Identifying data:** UID + name, SSN, address, doctor's name, …
- **Clinical data:** UID + imaging, blood composition, diagnosis, ...

These two parts would be managed by two separated docker containers and never loaded in memory at the same time: the usual flow would require the user to search in the database (by name, phone number, SSN, …), retrieving the UID and loading the clinical data through it. Leaking only one of the two DB sectors would not give the attacker really valuable personal data because no medical information could be linked to the actual person. This system would also allow the reverse search of a patient name given the clinical data. In case this feature is not needed in the regular flow of the software, which is in some sense expected, an easy improvement could be done to make such attacks more difficult. Instead of inserting the same unique identifier in both databases, the one together with clinical data could be hashed in order to make it difficult for large enough database to find the identifying information belonging to a given piece of medical data.

In case there was a need to publish or distribute personal data for commercial or research purposes, the database should be preprocessed in order to reduce to a minimum amount of information linkable in some way to patients. The *differential privacy* model allows to provide some confidence on the privacy level while ensuring the usability of the data for specific purposes. A deeper study of this matter would be strongly advisable in this case.

General considerations on architecture security

In addition to what we mentioned about the structural rework of the personal data storage, a few notes on some best practices on the matter.

A locally hosted database is generally preferable over one hosted in the cloud, in order to both achieve a smaller attack surface and a simpler adversarial model to comply with under data protection regulation. This server should be well protected within the hospital local area network through proper use of firewall and proxy and feature a separate append-only logging server in order to ease the discovery and diagnosis of breaches. A completely separated physical local network may as well be considered in particularly sensitive settings.

Depending on the business requirements, the maintenance, and thus administration access, can be given or not to the customer, but in the first case, a bit of training and proper documentation become necessary.

Docker, as a deployment system, allows to properly set burdens to the means of communication among different components of a service; by having these interfaces monitored and reduced to the strict necessary a lot is achieved in security guarantees. Apposite additional tools can be used to enhance the system strength, especially when third party provided containers are set up in production; they can either work *dynamically,* by analyzing the real-time behavior against a statistical model and reporting anomalies of different kinds or *statically*, introducing intermediate layers of containerization or modifying systems to ensure their expected behaviors. The composition of the two would provide a reliable enough set of rules and information to a security team (possibly outsourced) and ease their job a lot when monitoring and diagnosing.

Cryptographic secrets management

As one would expect, in order to reduce the possibility of significant data leaks and security breaches in general, plaintext storage of passwords and encryption keys should be strictly limited to runtime memory (RAM) and limited in time as much as possible. For example, a token can be generated through a PBKDF from each password and stored for the following checks, allowing to store the passwords into memory only for the limited amount of time required to recompute the token and compare it to the one stored in the database. PBKDFs are designed to be probably *slow,* in the sense that a single run will not influence performance too much (typically ~100ms), but enough to make brute-force attacks a lot less feasible. You can refer to the Password Hashing Competition for the state of the art in the topic.

Other secrets, such as encryption or signing keys (that should always be independent), can be stored encrypted in a separate database managed by an apposite container and only loaded when needed.

Database access control

From talks with Adam Istvan, we got to learn that Mediso uses SQLite for its database. So, our first step has been to look for ways to integrate a proper form of access control in the SQLite database, in particular, a Role-Based Access Control (RBAC) mechanism. However, our first searches did not return promising results. According to [1], a complete SQLite database is stored in a single cross-platform disk file and SQLite does not support multiple users, so anyone who has direct access to the file can read the whole database content. [1] proposes a security-enhanced version of SQLite, but it only supports Mandatory Access Control (MAC) and Discretionary Access Control (DAC), not RBAC, which we want. This meant that we had to continue our search, and look for alternatives to SQLite, that actually do support RBAC.

MySQL [2] is one of the database engines found that does support RBAC. Online we found a good tutorial on how to do this, consisting of two parts, which we propose as a possible guide for Mediso:

1. https://www.xaprb.com/blog/2006/08/16/how-to-build-role-based-access-control-in-sql/
2. https://www.xaprb.com/blog/2006/08/18/role-based-access-control-in-sql-part-2/

Postgres [3] is another alternative. [4] gives a good overview of the differences between Postgres and MySQL:

> *Postgres is an object-relational database, while MySQL is a purely relational database. This means that Postgres includes features like table inheritance and function overloading, which can be important to certain applications. Postgres also adheres more closely to SQL standards.*
>
> *Postgres handles concurrency better than MySQL for multiple reasons:*
>
> *Postgres implements Multiversion Concurrency Control (MVCC) without read locks Postgres supports parallel query plans that can use multiple CPUs/cores Postgres can create indexes in a non-blocking way (through the* `CREATE INDEX CONCURRENTLY` *syntax), and it can create partial indexes (for example, if you have a model with soft deletes, you can create an index that ignores records marked as deleted) Postgres is known for protecting data integrity at the transaction level. This makes it less vulnerable to data corruption.*
>
> *For simple, read-heavy workflows, Postgres might be a worse choice than MySQL*.

In the case of Mediso, we would think concurrency would be an important feature, and we would also expect that there would be a more equal amount of write/reads, meaning that Postgres might be the better choice, because it was built to be feature-rich, extendable and standards-compliant. However, Postgres is still less popular than MySQL (despite catching up in recent years), so there's a smaller number of 3rd party tools, or developers/database administrators available [4].

If deciding to go with Postgres, the documentation for implementing RBAC is provided in [5]. In our opinion, Postgres seems like an option that should be seriously considered, next to MySQL, as it seems to be the most future-ready option.

[7] might be interesting to check out as well, a Security-Enhanced Postgres.

[1]: https://cs.unibg.it/seclab-papers/2015/ACSAC/SeSQLite.pdf
[2]: https://www.mysql.com/
[3]: https://www.postgresql.org/
[4]: https://developer.okta.com/blog/2019/07/19/mysql-vs-postgres

[5]: https://www.postgresql.org/docs/9.0/user-manag.html
[6]: https://tapoueh.org/blog/2018/11/preventing-sql-injections/
[7]: https://wiki.postgresql.org/wiki/SEPostgreSQL_Introduction

Token-based Access Control

A more flexible alternative to RBAC is provided by token-based access control. To get an effective mechanism, this technology is often associated with more classical database level policies (like MAC or RBAC itself) in order to improve the user experience. By default, access in this setting is denied to everybody and granted only to who owns the necessary access token; his allows to tune in a really granular way which devices are allowed to read or write a certain resource enforcing without further effort secure connection (eg requiring IP coming from VPN), expiration times and allowing users to forward their own privileges to others, for example in the case of temporary collaborations with external doctors or technicians. However, using this technology likely makes it harder for developers to guarantee a smooth user experience without unnecessary access denials and if commercial requirements are simple enough to be fully deployable on RBAC this might be a bit overkill. Furthermore, enforcing access control at the server level instead of the database itself technically widens the attack surface; a risk that may be addressed by combining it with DB-level methods.

A pretty fresh technology that fulfills these promises, called *Macaroons,* was developed by the Gmail team and has interesting properties for our purposes. A macaroon can be seen as an enhanced version of a cookie, like the ones that bother our browsers all the time. It consists of a small file with a digital signature attached (actually a Message Authentication Code, more on this in [8] and [9]) that a user stores on his machine and can be sent to the server together with a database request as a proof of ownership of the access right. The use of symmetric cryptography makes macaroon creation and validity check really efficient and it allows us to simplify the server-side access structure by moving it on the client. Macaroons' efficiency allows them to create on a per-patient basis, having them created by the server when a new patient is accepted and distributed to the necessary hospital staff to grant them the ability to read, modify or append on all or part of the patient's data. For example, one *append-only* macaroon can be given to the data imaging client software, a *read-only* to nurses and the administrative staff can be limited to access non-medical information only.

In case a user needs a consult from someone that doesn't have access to some data, he can "share" this data by sending the related macaroon, possibly appending additional caveats such that a shorter expiration date or forbidding modifications. This complexity can be completely hidden from the end-user and managed by the client software, for example through a *share* button with associated dialog window to set the necessary conditions.

Potentially the database access control could be enforced through macaroons only, requiring password insertion only while taking in charge of a new patient, in order to receive the related macaroon, and data access could be granted without further authentication from the

same device until expiration. Macaroon update and synchronization could be done in the background during software used when the user is logged in. This would allow to easily share access rights to external players that use the same software without additional setup and account setting, for example for a temporary consultancy with a doctor from another hospital.

As compared to RBAC, this solution might allow better compliance with the “least privilege principle” while offering an integrated way to share content among collaborators and easing the load on the server-side. This level of flexibility has to be paid through a higher level of specialization of the IT department that needs to be able to translate internal rules into macaroons language and to properly implement the rules on how they’re distributed to client machines. Moreover, the hack of a doctor’s computer leaves the attacker with the same privileges of the doctor himself, which doesn’t necessarily happen with a classical password based system.
In conclusion, we suggest partial or complete use of macaroons in case some resources can be spent to produce and maintain a more flexible system.

[8]: http://hackingdistributed.com/2014/05/16/macaroons-are-better-than-cookies/
[9]: https://en.wikipedia.org/wiki/Message_authentication_code

Remote access

Remote access management for a medical system includes the monitoring and controlling access for the employees to the sensitive patient data inside the organisation's network. Thereby minimizing the data theft risk and losing the data due to malicious activity, if properly implemented.

There are multiple ways to provide remote access control for a network: by Direct Line, by Virtual Private Network, by Deploying Microsoft Remote Application server, etc. On investigating every method listed above, the Direct Line is expensive and is prone to hardware damage resulting in an insecure state of communication (unreliable). Microsoft Remote Application server is an external service, deployed with limited load balance, and has difficulty in installing and configuration.

This is why we will continue with the Virtual Private Network option, which uses the Internet to connect to the network remotely. It supports encryption and tunneling techniques for secure data transmission.

Virtual Private Network (VPN)

We could have decided to write this section ourselves, but while searching for information, we found an excellent post online [10], and honestly, couldn't have said it better ourselves:

> *Virtual Private Networking is a method of creating a smaller private network that is running on top of a bigger network. Computers connected to a VPN act as if they are*

connected to the same network switch even if the other computer is halfway around the world. Citrix is a company that provides services and applications that operate on a VPN and allows users to access files and applications on a server remotely.

Creating a VPN can be accomplished with the use of a wide variety of software that can be acquired from different sources. Setting-up a VPN from scratch can be a complex task as there are a wide variety of concerns that needs to be addressed, not the least of which is security. Citrix provides an all-in-one service as it is capable of handling the huge majority of things needed to create a fully working VPN.

[10]: http://www.differencebetween.net/technology/internet/difference-between-citrix-and-vpn/

Virtual Desktop Infrastructure (VDI)

After having built a Virtual Private Network, it would be time to think about how to virtualize the connection. One good option for this would be Citrix, as also mentioned in [10]. On their own website [11], Citrix describes VDI in the following way:

[Citrix] offers critical benefits to IT, including improved security and centralized desktop management, but the ability to securely access a remote Windows or Linux desktop isn't enough in today's organizations. The demands placed on IT require a more evolved approach. Organizations need to optimize user experiences to increase adoption, maximize capabilities by minimizing the number of consoles to manage, and ensure sensitive data stays safe.

However, recently Citrix got in the news in a way that could be called 'bad', see [12]. A part of the story is quoted below to get an idea:

On Dec. 17, 2019, Citrix released security bulletin CTX267027, which identified a vulnerability in Citrix Application Delivery Controller (ADC) and Citrix Gateway. This vulnerability, assigned CVE-2019-19781, could allow an unauthenticated attacker to perform arbitrary remote code execution via directory traversal. This vulnerability received a score of 9.8 and was deemed Critical. On Jan. 8, 2020, Tripwire provided a very detailed explanation of the CVE that we recommend reading.

Based on this background, many offensive security professionals described their ability to weaponize CVE-2019-19781 but signaled their plans to keep their exploitation code private, in favor of providing defenders with scanners. However, on Jan. 10, 2020, FireEye observed the public release of a code repository containing proof-of-concept (PoC) code that assisted and automated the exploitation of this vulnerability—at which point multiple reputable offensive security professionals released their previously-private tooling. As of that night, FireEye network appliance telemetry showed reconnaissance for vulnerable Citrix instances conducted from anonymizers, such as Tor, and known-blacklisted IP addresses. Other organizations publicly shared honeypot data indicating a similar trend. Shortly afterwards, we observed weaponized versions of this exploit used to gain a foothold in victim organizations.

There might be some technical terms in this story but the overall idea should be clear: a lot of organizations, including the Dutch government, had to temporarily shut down their Citrix solutions because of this. While this might make you want to avoid Citrix, we think that this actually shows that its services are being tested by professionals, and are being patched. Not knowing about vulnerabilities might be nicer, but definitely not more secure. Citrix claims that *today, more than 100 million users across 400,000 organizations – including 99% of the Fortune 500 – trust Citrix to power a better way to work* [13].

[11]: https://www.citrix.com/digital-workspace/virtualization-vdi.html
[12]: https://www.fireeye.com/blog/products-and-services/2020/01/rough-patch-promise-it-will-be-200-ok.html
[13]: https://www.citrix.com/about/

Two Factor Authentication

Apart from everything said above, there are also other important risk elements, of which one is not having two-factor authentication.

Compromising a weak single factor entry to an application (like a password) is easy nowadays, with the tools available in the market. Hence, whenever we sign-up an account with username and password, a strong password is required. A strong password is defined based on its effectiveness against brute-force attacks. According to NIST, a password should be minimum 8 to maximum 64 characters of uppercase, lowercase alphabet, numbers, and symbols, and shouldn't be reused in any other accounts which eventually compromise multiple accounts.

For more secure account management in Mediso, we suggest using two-factor authentication. All the actors in the system will have an authenticator application on their phone, which generates a new 6-digit code for every 30 seconds, called *time based one-time password*. This acts as a shield between the password clearance (first factor) and the application gateway. Two-phase authentication is safe from dictionary attacks, something one-phase isn't necessarily, as any Password Stealing Ware (PSW) could guess the username and password combinations if the entropy for a user to provide a strong password is low. Following from this, we suggest the use of two-factor authentication, along with a random password generator for the first phase (like RSA SecurID software tokens) to make sure all passwords are strong.

Business research report

Research

Process of finding competitors

One of the first and basic steps to getting acquainted with the market is to do the initial competitor research. This can be done in many different ways, but the outcome is a way of visualising data from which the reader can easily see who the competition is and what it is doing. Here a simple Excel table was chosen, which was split into multiple categories:

Company	Product (with a link)	Features	Privacy	Internet	Market	For who is it meant	General comments

First, the name and the link of the competitor company was given, second, the link and the name is the competitor's product was given. Here we were careful to list only products that are directly relevant and fairly similar in function to Mediso's product InterView Fusion. If we had not implied this filter, the list would grow too large and more irrelevant. The third column lists the basic features of the competitor's product. This is important to give the reader some basic data about the product to be able to have some context. The fourth column gives information on whether the product needs to be connected to broad internet to operate. The market column gives the reader some idea to which market the product is generally aimed at. The sixth column shows for which department is the product meant for. And the final column shows any additional comments and observations that could not fit into other categories. This is often considered as the most important column as its usually the source of the greatest insights and biggest help when distinguishing yourself from the competition.

The process of gathering information was fairly straight forward. We searched on the internet for firms that were giants in the medical devices area and researched their website pages for any details on their products. Here the mentor was of great help, as he was able to point out the relevant firms which made a search that much easier. It is also noticeable from the table that some data was unable to be found due to lack of them actually being on the companies website. In the end, we also searched for what we could find on the InterView Fusion from Mediso itself, just to be able to determine how searchable it is and what data can actually be found online about it. We also used it to compare it to the other entries. The full competitor table can be found at the end of the document.

Competitor analysis

In the end, nine products were analysed (some of the companies, like Siemens and Fuji had even multiple products that we deemed interesting). All the listed products were a kind of solution to connect multiple devices either within the section, within the hospital or even between multiple hospitals.

Interestingly, even though all of the products offered connection solution, none of them mentioned or addressed any privacy or security issues. This strikes us as really surprising since these solutions are all dealing with very sensitive medical data, leakage or misplacement of which would likely to imply serious issues. One of the reasons might be that the websites actually give information about the medical-relevant features of the products and that the security measures are disclosed only after the client has shown sufficient interest.

The second observation is also that none of the products mentions any ability for the system to be systematically updated software-wise. After some consultation with the mentor, it was concluded that modern medical devices either don't get updated, or this gets done by a product engineer coming to the hospital physically and installs all necessary updates via CD or USB key. Not only does the lack of on-line connection not enable the providing company to update their machines in an efficient manner, it also impairs them to do any number of useful actions, such as preemptive maintenance, real-time usage analysis, online diagnostics, … This strikes us as a huge missed opportunity for providing companies.

The third major observation is also that the Medisos' product and Mediso, the company, in general, was very hard to find if not sought by specific (pre-known) names. When searching for phrases like "medical IT systems", "medical platforms", "IoMT",... with which other competitor companies were easily found, turned out not to be fruitful search terms for company Mediso. This could be for multiple reasons, one of which might be that other companies are relatively bigger and are thus more likely to turn up on the first page of the search result. Nevertheless, Mediso could invest some resources into search engine optimisation and relevant online marketing as this would make it more visible to future prospective customers.

To categorize all the competitors, the following comparison chart was used:

1 - Same tech, Same Vertice
2 - Different tech, Same Vertice
3 - Same tech, Different Vertice
4 - Different tech, Different Vertice

This assigns certain traits to each class - from 1 to 4, to see how relevant the competitor's product is (1 being the most relevant and 4 the least). If the product is using similar technology to work, it belonged either in classes 1 or 3, and if it solved the same kind of classes it belongs to either class 2 or 4. With this classification is now easy to see which product is the most relevant by a reader just sorting by category. Thus in the first category, there is only one product: QLAB Cardiac Analysis from Philips. This is because we see it as the most similar to the Medisos' InterView Fusion and should be in our opinion the most closely followed competitor product.

The full competitor analysis table can be found on this link:
https://docs.google.com/spreadsheets/d/1qymNs97En_2340-btqwTlo8ovlNzTRD8W8bv8LaOlUM/edit?usp=sharing

Possible Business Model Canvas

After having valuable insights from the research perspective we constructed a business model that will provide a significant growth and development for Mediso in the future. Some of the major key partners include the best server providers in the market such as Bluehost or HostGator because they have a reliable uptime of 99.8% and a speed of 369ms. There are also other providers but they are compromised on reliability and speed. On Further discussions with our mentor we have also understood that the co-operations formed with hospitals and clinics also play a vital role as key partners. However, Consulting services and Insurance companies also fit into the frame as partners.

Our key activities will include development of new software, upgrades to the existing software,Marketing and potential customer consulting by having analysis on feedback with the general opinion on the services from the customers. There is nothing that excites potential customers more than feeling different from the rest. In order to achieve this effect, we can send special offers with a special price for the customers who are already using our services and also willing to use further products by the company in the future. A small percentage will be more than enough for them to continue to trust in us. The customer looks for innovation and something that really grabs their attention.On the other hand by being updated with the current and best technology we can break the ground and always win over competitors.

Moreover, The value to the customers can be provided with quick and simple way to diagnose patients with our products. The data needs to be seamlessly synced with all devices and maintain the maximum security since it would contain sensitive information and a minor security breach could lead to the loss of potential customers. Having said that, we would need key resources like developers to build our software, human resources in terms of staff and computer programmers, and other basic requirements such as electricity,computers and internet to further maximize the growth of the company.

Significantly, The customer segments include the hospitals that are already using the services and most importantly third party hospitals who are not yet partners with Mediso. Despite this, The customer relationship will be primarily online and in some cases third party contractors. The customers can be provided with support and maintenance of the software.However, The customers will be reached with various communication channels such as Webpage, Facebook, Advertisements, Instagram and E-mails. The purchase and delivery services would be both Online and Offline, for reaching the broader section of the society. Medical conferences could also be conducted to promote the services we offer that could help the customers get deeper insights with respect to the product. For after sales, we propose to have a call center/helpline and standard requests can be responded by a chatbot 24/7.

The revenue streams can primarily be based on pay per usage of the software and hardware could be one time payment. Depending on the performance, over the period we can intend to generate revenue through advertisements and promotions. All things considered, The cost structure includes the fixed costs such as development of software and hardware, cost of marketing, cost of servers and external costs for the lawyers and costs related to testing the products. The variable cost includes the salary of the human resources.

Comparing Mediso business plan to other companies

While doing research on Mediso's competitors, we have listed several products and companies that are directly related to our product InterView Fusion or who are functionally similar. Each product is a solution proposed by their company for different medical scenarios, but we still find their common shortcomings such as privacy,security issues and timely software updates. Mediso's product which cover all this function defects will have unique innovations that attract users.On this basis, we have given an ideal business model as above. In order to convince our customers and strengthen their trust in our products and services, we select three of the most influential competitors: Siemens, GE and Phillips to compare with proposed Mediso business model.

GE Healthcare

GE Healthcare is also very comprehensive. The entire company is in a leading position in the fields of medical imaging, information technology, medical diagnosis, patient detection and monitoring, disease research, drug development, and biopharmaceuticals. In terms of imaging products alone, each of the major product lines has its own fist exhibits. The market share of CT and MR is one of the highest in the world. The market share of nuclear medicine products once reached an alarming 70% in China. Ultrasound's products are well-known for their complete product lines covering high, middle and low end. Even for those general-purpose, monitoring products that are very sensitive to costs, GE Healthcare can also come up with very competitive products.GE focuses on overall strategy and market positioning, especially on precision medicine. In the emerging markets, GE Healthcare's investment has adopted an in-region, for-region strategy. However, due to the interoperability

between traditional solutions and GE's newly released products, GE has not made much progress in affecting informatics.

Siemens

Siemens' medical imaging is their core driving force. And they recently changed their business model to operational purchasing model. This will become increasingly important as suppliers seek partners for long-term risk-sharing contracts to meet value-for-medical plans, especially in mature markets where Siemens has long occupied an important position. In addition, Siemens maintains close ties with Cerner. This company helps Siemens Medical to resume and develop related products and solutions outside of the diagnostic imaging business to meet customer needs.

Phillips

As GE Healthcare continues to streamline itself, concentrating its core focus on medical imaging, clinical care and life sciences, Philips is continuing to expand their medical business with the intention of reaching long-term large-scale cooperation agreements with the largest medical systems and suppliers providing an all-you-can-eat model. In addition, Philips is also actively promoting subscription-based and risk-sharing contracting. However, the implementation of this strategy does face challenges, large-scale, complex long-term transactions usually require long lead times for procurement. Therefore, in order to maintain the share in the medical field and ensure that the risk sharing results can be profitable, Philips has taken great pressure and actively selected regional markets for products so that they can adapt to existing, localized market conditions. This is why Philips has many benefits in the product area, so they don't need to adjust individual product structures to adapt them to specific markets. Except for this, Philips is the only company that does both B2B and B2C in these companies.

Mediso

Mediso is more similar to GE Healthcare's target market than Siemens and Philips: the core goal is to provide nuclear medicine and modern hybrid imaging technology. Selectable software and hardware pay-per-use or one-time flexible purchase models plus additional discounts from previous promotions can attract a large number of users, while InterView Fusion for accurate comparison and joint analysis of multimodal image data and privacy protection Medical servers that can ensure the security of medical image data and high-quality after-sales service will retain users. Of course, through a preliminary survey Mediso needs to invest more money in advertising to increase user channels.

Data Science

Introduction

Medical Image Analysis provides a forum for the dissemination of new research results in the field of medical and biological image analysis, with special emphasis on efforts related to the applications of computer vision, virtual reality and robotics to biomedical imaging problems. With the development of computation and popularity of Artificial intelligence(AI). AI in medicine is also a fast-growing field. The rise of deep learning algorithms, such as convolutional neural networks (CNNs), offers fascinating perspectives for the automation of medical image analysis. In this project, we carried out a research on doing the semantic segmentation of the heart on the CT images using a state-of-art deep learning model - U-Net.

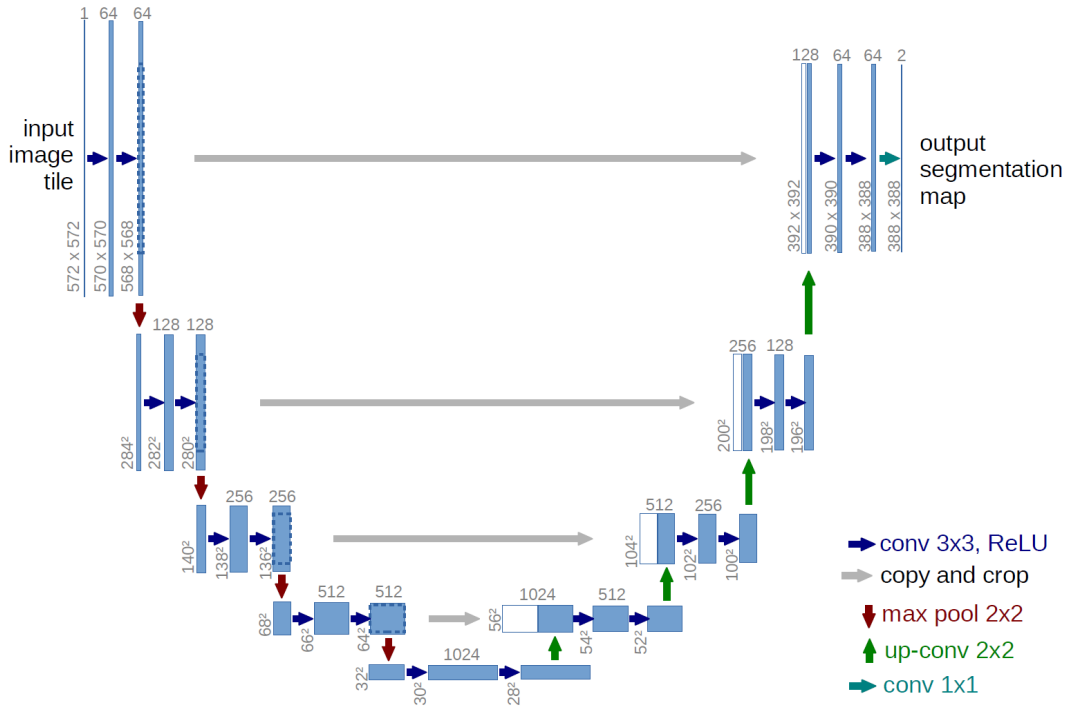
Semantic Image Segmentation

In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as image objects). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

As to semantic segmentation, traditionally, thresholding-based algorithm such as Hough transformation can detect the object of the image and distinguishes it from its background, edge detection based algorithm such as Laplacian algorithm. In recent years, with the development of deep learning, semantic segmentation gained a lot of attention. Feature-encoder based model VGGNet and ResNet, regional proposal based model like RCNN and Fast-RCNN can achieve semantic segmentation in more complex images.

U-Net Model

The main idea behind CNN is to learn the feature mapping of an image and exploit it to make more nuanced feature mapping. This works well in classification problems as the image is converted into a vector which used further for classification. But in image segmentation, we not only need to convert feature map into a vector but also reconstruct an image from this vector. This is a mammoth task because it’s a lot tougher to convert a vector into an image than vice versa. The whole idea of U-Net is revolved around this problem.



The architecture looks like a ‘U’ which justifies its name. This architecture consists of three sections: The contraction, The bottleneck, and the expansion section. The contraction section is made of many contraction blocks. Each block takes an input applies two 3X3 convolution layers followed by a 2X2 max pooling. The number of kernels or feature maps after each block doubles so that architecture can learn the complex structures effectively. The bottommost layer mediates between the contraction layer and the expansion layer. It uses two 3X3 CNN layers followed by 2X2 up convolution layer.
But the heart of this architecture lies in the expansion section. Similar to contraction layer, it also consists of several expansion blocks. Each block passes the input to two 3X3 CNN layers followed by a 2X2 upsampling layer. Also after each block number of feature maps used by convolutional layer get half to maintain symmetry. However, every time the input is also get appended by feature maps of the corresponding contraction layer. This action would ensure that the features that are learned while contracting the image will be used to reconstruct it. The number of expansion blocks is the same as the number of contraction block. After that, the resultant mapping passes through another 3X3 CNN layer with the number of feature maps equal to the number of segments desired.

UNet uses a rather novel loss weighting scheme for each pixel such that there is a higher weight at the border of segmented objects. This loss weighting scheme helped the U-Net model segment cells in biomedical images in a discontinuous fashion such that individual cells may be easily identified within the binary segmentation map.
First of all pixel-wise softmax applied on the resultant image which is followed by cross-entropy loss function. So we are classifying each pixel into one of the classes. The idea is that even in segmentation every pixel have to lie in some category and we just need to make sure that they do. So we just converted a segmentation problem into a multiclass classification one and it performed very well as compared to the traditional loss functions.

Data Preparation

The data from which the experiment was conducted on consists of 54 CT scans. On these scans, the color intensity values of the heart tissues are larger. The challenge is not only those tissues have that property. To overcome this issue, first the heart tissues have to be segmented manually. Traditional computer vision algorithms such as thresholding and region growing were used for this task. In the following 2 paragraphs, this algorithms will be presented in detail.

The goal of binary thresholding is to separate object pixels from background pixels. The challenge is to select the optimal threshold value. There are many existing algorithms for that, however, since this project requires domain specific knowledge for finding the heart tissues, we choose to set the threshold values by hand.

Region growing starts with the selection of seed points. The regions start to grow from these points depending on some neighbourhood metric. The process stops when no more points are being added to the regions.

Another difficulty with the dataset is that the resolution of the images is different: there are 64x64x64 images, and images of size 128x128x128. In addition, some of the images are only focused on the chest area, or directly on the heart. Only the images of size 64x64x64 was kept. This way, the size of the dataset was reduced to 21. This is clearly not enough for training a deep model. To overcome this data augmentation was applied to generate additional data. Data augmentation applies transformations such as shifting and rotation on the original files to create new samples. For the segmentation, ITK-SNAP, an open source tool was used.

Experiment

The experiment was conducted using Google Colab, an online environment to run Jupyter notebooks. The environment grants 33 Gigabytes free space and 13 Gigabytes of RAM. The data was uploaded to Google Drive, and mounted to the online environment. To deal with the DICOM medical image file format, the python package dicom was used, and the package nibabel to read the segmented images. The three dimensional version of the UNet architecture was built using Keras. The cost function was binary cross entropy and the ADAM optimizer was used to find the optimal parameters

Result

Due to a lack of computational power and too little memory to store the 90,296,069 network parameters in RAM we could not train the U-Net architecture. An attempt in google Colab resulted in an out of memory error. We have also created 50 annotated images of heart CT-Scans, which we are not allowed to share publicly due to confidentiality.

Conclusion

Our presented work makes it easy for future projects to test out a 3D U-Net architecture. It will therefore remain publicly available so that it can ease the work for others. It can be copied from our GitHub repository: https://github.com/JordiSpranger/3D-U-Net. We suggest to train and test out our U-Net with medical 3D image data.

Annex I – Terminology

- **CR,** Computed Radiography
- **CT,** Computed Tomography scan
- **DICOM,** Digital Imaging and Communications in Medicine (Standard)
- **DX,** Digital X-Ray
- **ECG,** Electrocardiogram
- **HIS,** Hospital Information System
- **MRI,** Magnetic Resonance Imaging
- **NM,** Nuclear Medicine
- **PACS,** Picture Archiving and Communication System
- **PET**, Positron Emission Tomography
- **ROI,** Region Of Interest
- **RTSTUCT,** Radiotherapy Structure Set
- **SC,** Secondary Capture
- **SPECT / SPET,** Single Photon Emission Computed Tomography
- **SUV,** Standardized Uptake Value
- **XA,** X-Ray Angiography
- **VOI,** Volume Of Interest

Annex II – Bibliography

Scholar:

- Gábor Jakab, Tamás Huszár, Balázs Csébfalvi: Iterative CT Reconstruction on the GPU. In: VI. GRAFGEO Conference. Budapest, Hungary, 02.21. 2012-02.22. 2012. pp. 124-131.
- 3: Gábor Jakab, Attila Rácz, Tamás Bükki, Gábor Németh: Fully GPU Based Real Time Corrections and Reconstruction for Cone Beam
- Gabor Jakab, Laszlo Szirmay-Kalos: Hybrid Monte Carlo CT Simulation on GPU. In: Lecture Notes of Computer Science (LSSC'13), 2013

General:

- Interview Fusion, multimodality image processing workstation for clinical applications. http://www.mediso.hu/uploaded/INTF_1014_web.pdf
- DICOM standard, https://www.dicomstandard.org/
- SonicDICOM PACS, https://docs.sonicdicom.com/install-manual/dicom-communication.html
- Interview Fusion, https://www.omniagmd.com/product/interview%E2%84%A2-fusion
- Interview Fusion, http://www.mediso.com/products.php?fid=1%2C10%2C6&pid=67