Software Product Line Architecture for MRI Machines

Onur DEMİREZEN

Mehmet Arif ŞAHİN

Yunus Kerem TÜRK

Table of Contents

Introduction

A Short Description

This report is to demonstrate the project which aims to provide a software product line for Magnetic Resonance Imaging (MRI) machines as the output of the Software Product Line course.

In this project, we are assuming that we are working on a software product line in the MRI software company. We are given a report that defines the software architecture for MRI machines that is created by the architecture team. Managers stated that they want to create a product family for MRI software and for this purpose, product line principle is used. In accordance with this purpose, our team wrote a report about the software product line for MRI machine software.

Purpose

The purpose of this report is to create a product line for MRI software by analyzing the proposed MRI software architecture report. While creating the product line, the main purpose is to manage variabilities and commonalities of the product family appropriately. To realize this, it is aimed to determine the requirements well. These requirements are also taken into consideration in accordance with the portfolio created in product management. The features to be used in the portfolio and their variability are also tried to be managed well and finally, it was ensured that the variability models are met in architecture.

Method

In this report, a software product line method is used. Software product lines (SPLs), or software product line development, refers to software engineering methods, tools and techniques for creating a collection of similar software systems from a shared set of software assets using a common means of production[1]. The feature model method was used to create a domain variability model. In software development, a feature model is a compact representation of all the products of the Software Product Line (SPL) in terms of "features"[2]. In order to show the variability in the requirements and architecture, an orthogonal variability model is used[3].

[1] Acquired from *https://en.wikipedia.org/wiki/Software_product_line*
[2] Acquired from *https://en.wikipedia.org/wiki/Feature_model*
[3] Klaus Pohl, Günter Böckle, Frank J. van der Linden, *Software Product Line Engineering Foundations, Principles and Techniques*, 2005, ch. 4, pg. 75

Content

Contents of this report includes design method description with proposed layers of the architecture report, market analysis and market focus description for MRI machine software, definition of motivating features, product portfolio derived from features with production timeline, KANO survey and assumption of results, domain variability model that is divided into three main parts in reference to the architecture team. Then three products for each layer have chosen to apply domain variabilities. After definition of the variabilities, domain and product engineering has been done according to them. Traceability matrices are provided for ease of understanding the connection between features, variations and architecture components. Also, artefacts tables are available for tracing artefacts in the product timeline.

Since this is an educational report that assumes a product line architecture, results and learning outcomes are individually discussed in the last chapter by authors.

Presentation of the Architectural Report

Requirements

In this section, requirements identified by the software architecture team are presented. The design goal is to provide extensible software components, which keep up with the evolving requirements of MRI machines.

While identifying requirements, two main important *use scenarios* are discussed. The first one is a scenario consisting of the non-adaptability of former software architectures for MRI machines. According to this scenario, when a groundbreaking algorithm is found, it is difficult to integrate into existing MRI machines and make it available for use by clients.

The second scenario is the following: If a new algorithm or module brings a considerable amount of cost when it is integrated, these pieces of the software become practically unusable when combined with some, therefore our system should be able to acknowledge and use it.

R1. Since the processes which are image processing, feature extracting, classification of images and diagnosis are time-consuming tasks, a module that monitors the performance is required.

R2. Software architecture should be flexible and adaptable enough to integrate future enhancements and newly created modules when it is convenient.

R3. Architecture should provide modularity and required variability for tasks' needs. A module can select the right component for the specific task so that it maximizes utilization and performance. This module also must have knowledge about the system performance for the advantages of these algorithms thus, that can select the appropriate module to complete the task at hand.

R4. This requirement is centered around the security issue of the storage. A module needs to secure storage for the processed images and sensor data as well as future operations which for the purposes of image enhancements and diagnosis. Therefore, the data must be stored without defect and accessible when requested for future operations. Besides these, because the patient's data is private, the module to be implemented also needs to provide storage security.

R5. Since MRI machines' target users are radiology specialists, visualization is a key component for the purpose of a precise diagnosis. User interface should be intuitive for medical doctors; therefore, an optimal user interface module is required. Giving

this module the ability to operate on different types of machines could help standardize a UI for MRI and would make it widely available in the medical industry. Keeping this module separate from the output module makes it open for improvement in future, without interfering with the rest of the components.

R6. An interface connecting hardware to software is needed for handling the data and images provided by MRI machines. Keeping this module separate makes handling future enhancements easy, without interfering with other components.

R7. Provide an infrastructure that supports the other parts of the system.

Architecture Design Method

The proposed design method in the report presented by the software architecture team is *layered architecture*. Components within the layered architecture pattern are organized into horizontal layers, each layer performing a specific role within the application. Although the layered architecture pattern does not specify the number and types of layers that must exist in the pattern, most layered architectures consist of four standard layers: presentation, business, persistence, and database. In some cases, the business layer and persistence layer are combined into a single business layer, particularly when the persistence logic is embedded within the business layer components.[1]

A Short Description of the Proposed Architecture

Based on the *layered architecture* method, three main solution domains which are application modeling and user interface, tracking and control system, system architecture and infrastructure are represented as layers.

Layer 1, Application Modeling focuses on the functionality for the organization that uses it. It refers to the behavior of an application suite that forms a system. Application Modeling indicates how application components like packages, databases and middleware systems communicate with each other. It also defines the interaction between the user and the whole system.

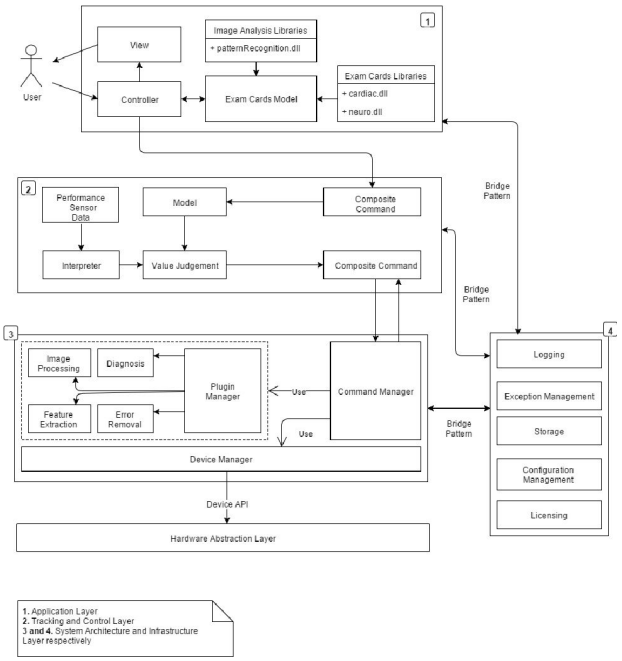
Users perform interactions described in the previous paragraph via "User Interface". Therefore, it should be understandable and efficient. Users should be able to use all features of the application comfortably. According to the report, the quality of the user interface is measured in clarity, concision, familiarity, responsiveness, consistency, aesthetics, efficiency, and forgiveness; all described from the user’s point of view.

The need for layer 2, Tracking and Control System, raises from the requirements. Since the system will support multiple algorithms for different tasks, it should be able to select the proper algorithm for a given task and context while considering performance. The reason to consider performance measures is to decide the execution plan. The need for a control system is identified here. It is required to conceive the whole process to successfully choose the control strategy. The system should be aware of the context and the overall performance and generate an execution plan, address the proper architectural components responsible for image processing, and forward the result to the UI components.

In Layer 3, System Architecture and Infrastructure, Component Based Architecture is used. The motivation behind selecting this architecture is that the lifespan for medical diagnostic equipment is long, the architecture needs to be flexible enough to accommodate future changes. Moreover, the Component Based Architecture inspired by Product Line Architecture, will provide an architecture that will result in increased productivity, a short time to market the product and an easy replacement of existing software components with the enhanced one. The main goal is to avoid monolithic architecture and allow the system to integrate multiple modules for various tasks to improve utilization.

Layers of The Architecture

In the following section, three layers of general architecture demonstrated in below figure and mentioned in the previous section are described in detail.



- **Layer 1: Application Modeling and User Interface**

In this layer, the software architecture report outlines the following invariants:

- An exam cards implementation: The main goal for this implementation is that when the user performs one of the medical procedures, the application should give the appropriate output for the according to the procedure. To succeed this, exam card approach is used. In this approach, the user defines the procedure and the controller requests the appropriate exam card model in the Exam Cards Library. In the Tracking and Control Layer transformations are applied to the model provided by the exam cards model then the MRI machine performs the task. Finally, the output is shown for users on the view component. According to the architecture report, the main advantage of this approach is that it stores the data, relationship and behavior which are reusable, easy to enhance and easy to exchange between medical experts.

- Image Analysis implementation: Image Analysis component is necessary to perform the tasks which are image pattern recognition, false coloring, noise reduction, 3D rendering, enhancement etc. and image processing algorithms like subtraction, superimposition, pixel division, averaging, offset shifting, edge detection etc. Because the implementation of the image analysis is to be considered as a variability, it must be managed well for easy integration.

- User interface implementation: User interface is the connection between doctor and machine. The doctor easily interacts with the application via user interface without struggling with the complexity of the processes performing in the subsystems. In addition to this, the doctor also should be able to see different views of information depending on the procedure he is conducting and the diagnosis he is evaluating. Therefore, the Model-View-Controller architecture pattern is used for coping with this requirement. This design pattern used for the user interface will be adapted for the given task, meaning that the controller component will send commands to the Tracking and Control Layer, the latter replacing the model component. The Controller component will send requests to the Tracking and Control Layer using the command pattern.

- **Layer 2: Tracking and Control System**

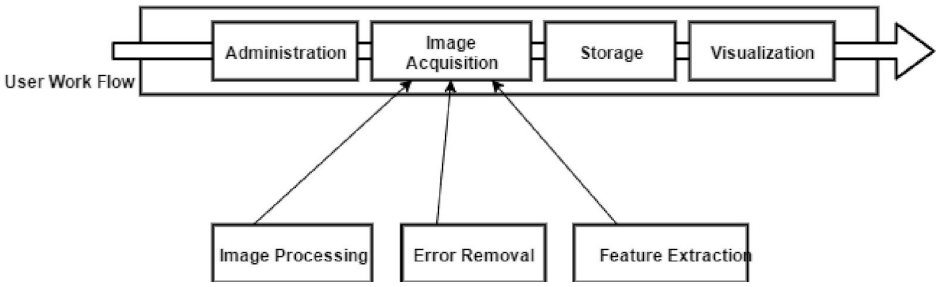
As mentioned in the previous section, tracking and control systems are the second layer of general architecture. This layer is responsible for deciding which action should be taken considering the performance. Decision making is a long and detailed process which can be divided into several components. While addressing the problem several different fields are investigated by the research team and real-time systems are chosen as reference model architecture. It is specially mentioned in the report that real-time

systems are not specified and specific details about how to implement such systems. It only prescribes a hierarchical control model. In this methodology, tasks are divided into finer pieces. These tasks in our case are: (1) Behavior generator component receives input from the upper layer and sends this data to (2) meta model and requests a plan. Meta models generate a series of plans and their costs and forwards this knowledge base to (3) value judgement. At this level (4) performance data is requested and decision is made. Then this decision will be forwarded to the behavior generator. When the behavior generator receives the result, it will send the data to lower layers for processing.

- **Layer 3: System Architecture and Infrastructure**

The discussion of the solution domain for system architecture and infrastructure (Computer Science Domain) is presented in this section. Component based architectures with vertical decomposition is the selected solution domain for the Tracking and Control System layer. It is already inspired from Product Line Architecture (PLA) and it should provide an efficient solution to fulfill the requirements. According to PLA, the focus should not only be on the functionality of the first product but on the whole set of known products for the soon-to-be family.

While the project described in the report that presented focuses on developing an architecture for MRI machines, it can easily be extended for other members of the medical imaging product family. Because the lifespan of medical diagnostic equipment tends to be long (ten to fifteen years), the architecture needs to be flexible enough to omit future changes. This will come with increased productivity, adaptability, and relatively short marketing time. The main goal is to allow the system to integrate multiple modules for various tasks, improving utilization.



To better understand the proposed architecture, it is important to understand that MRI applications are constructed around the same principle as standard desktop applications, they are centered around a host processor running on a desktop operating system. The

specialized peripheral devices are controlled by built-in real time processors. The workflow of the MRI from the end user point of view is shown in the above figure, it provides a broad view of the main components required for MRI software architecture.

Product Management

Market Analysis

We assume that we are asked to perform a global market analysis by our company. We have researched major MRI producers which are all hypothetical. Our observations show that most of these companies develop their own software for integrity and future-proofness.

MRI is a superior technique that uses radio waves, a strong magnetic field and computers to provide remarkably clear pictures. A patient is exposed to short bursts of powerful non-ionizing radio wave energy, while lying inside a massive hollow magnet. MRI uses a large magnet and radio waves to look at organs and structures inside your body. Health care professionals use MRI scans to diagnose a variety of conditions, from torn ligaments to tumors. MRIs are very useful for examining the brain and spinal cord.

The global magnetic resonance imaging market size was valued at USD 5.19 billion in 2019 and is anticipated to expand at a CAGR of 5.9% over the forecast period. Magnetic resonance imaging (MRI) is an efficient diagnostic machine for identifying diseases related to spine lesions, tumors, and stroke; impacting the brain area and blood vessels. Rising prevalence of these health conditions is expected to fuel the market growth. In addition, growing demand for quick and effective diagnostic procedures is expected to drive the adoption of MRI machines. Advancements in diagnostic techniques, such as open MRI, visualization software, and superconducting magnets are fueling the growth of the MRI market. The recent advancements observed in magnetic resonance imaging technology are mainly related to software. These advancements enable faster contrast scans and simplify imaging workflow. In September 2016, GE Healthcare's MAGnetic resonance image Compilation (MAGiC) software received U.S. FDA approval. This software helps deliver eight contrasts in a single acquisition in a fraction of time. Moreover, the development of cardiac pacemaker compatible MRI systems is also expected to propel the demand from the cardiology segment. Various paramagnetic contrast agents such as gadolinium-DTPA are used as an intravenous injection to provide sharp, precise, and accurate images within a shorter time frame. Such advancements are expected to propel the market growth.

Advancements, such as diffusion and diffusion tensor imaging with tractography, neuroimaging including MR spectroscopy, perfusion imaging, and functional imaging using the bold technique are expected to boost the growth of the magnetic resonance imaging market during the forecast period. Rapid development of intraoperative MRI and its usage in neurosurgery and other such applications is expected to propel the market growth. Diffusion-weighted MR imaging is mainly used for detecting stroke within 30

minutes of its occurrence. Therefore, advancements in MRI machines to enhance its usage for various applications is expected to propel the market.

High magnetic frequency MRI systems are gaining pace in terms of demand. Various universities are conducting research or studies on these systems. MRI machines are expected to open new avenues in diagnosis of a wide range of diseases, such as Alzheimer's, heart disease, diabetes, and cancer.

Despite various advantages associated with magnetic resonance imaging systems, the costs incurred in buying and installing these machines is significantly high, which in turn is impacting the market growth; especially in the developing regions. An average cost of a low to mid strength MRI machine is more than USD 1 million and around 70% of this cost comes from software development expenses. [4]

All companies we looked into in the market analysis process are including all the "fundamental" features that trivially make the machine work.

- *Image Processing:* All MRI machines in the market implement image processing algorithms to give an interpretable output. "Imaging" part in the abbreviation of MRI is bound to this feature.

- *Graphical User Interfaces:* Machines must be operable from a distance because of the magnetic field effects, also it must give a graphical output. To manage the machine and its features, a graphical user interface of any kind is needed.

While the features mentioned above are trivial for MRI machines to operate, some features can be seen as basic by users of the machine. We assume that radiology specialists are the user group of focus in this research. Other companies have implemented the features below in if not all, but majority of their products to gain an edge in the specialist's eye.

- *Integration With PACS:* In the market we observed that many of the MRI machines are integrated with the Picture Archiving and Communication System of the hospital.

Market of Focus

In the field of medicine there are a wide variety of diagnosis techniques. In modern days, for both the sake of doctors and patients, non-invasive methods are being preferred. Therefore, medical imagery is frequently used. Because, the techniques other than MRI such as computed tomography, may harm patients with X-rays, also

[4] Acquired from: https://www.grandviewresearch.com/industry-analysis/magnetic-resonance-imaging-market

they do not allow detailed analysis; Magnetic Resonance Imaging is the preferred method of medical imagery. With the current improvement in technology, many useful software features are included in MRI machines. Hence, different companies produce software products in this area. In this report, we focus on this field.

Potential clients are medium or large hospitals whose staff use computers and have well trained technicians/radiologists to perform MRI scans. Others impacted by the system will include: Doctors are actors who request a MRI Scan for a patient and interpret the results in assigning a diagnostic. Radiologist: Actor who uses the MRI machine, visualizes, processes and accepts scans provided by the system in order to answer a doctors request. He is also responsible for later re-processing of the scans. Patient is a person suspected of diseases and whose diagnosis requires an MRI scan.

Motivating Features

1. *Auto diagnosis with image recognition and big data*

Some diseases are so rare that it is not possible for doctors to know and recognize the disease. For such cases, auto diagnosis can diagnose the disease and report it to the doctor via user interface with additional information such as how rare the disease etc. While auto-diagnosis is not in full potential yet, it is a growing area that promises great convenience.

2. *Image processing*

Image processing is a general term that consists of various types of techniques that are used for enhancing the quality of the image, smoothing and sharpening for making some part of image distinguishable etc. Thus it is one of the core features of our software.

3. *Treatment and medicine recommendation system*

An artificial intelligence system that analyzes the results obtained from the auto-diagnosis module in the MRI software and offers possible treatment solutions and medicines, will affect the treatment process very positively.

4. *Adjustable rendering and display resolution*

Resolution measures the number of pixels in a digital image or display. Rendering resolution lets you keep the display resolution the same while adjusting the image is being rendered at. With technology improvements, it is possible to create high quality images but it may end up with performance issues. So we decided that offering a rendering resolution helps to cope up with performance issues while displaying the scan at a given display resolution.

5. *Automatic patient bed adjustment using patient data*

An MRI scanner is a short cylinder that is open at both ends. The patient will lie on a motorized bed that is moved inside the scanner. The patient will enter the scanner either head first or feet first, depending on the part of the body being scanned. In some cases, a frame may be placed over the body part being scanned, such as the head or chest. There are different types of MRI machines in the market[5]. After placing the patient to the bed some adjustments are needed for the perfect scan. For that purpose, a nurse or specialist should configure the bed according to the patient's proportion. We intend to do that with software using the patient's information taken from a remote server (PACS).

6. *Veterinary MRI exam cards model*

In an MRI system to be used specifically by veterinarians, exam cards should be suitable for the animal diseases to be diagnosed. Therefore, exam cards should be implemented and presented to the user for a more convenient scanning process.

7. *Remote control*

The ability of MRI software to be accessible and manipulated remotely for the radiologists is one of the features that motivate the product. The doctor can easily perform the scan procedure from his own room after the patient is being prepared for the scan.

8. *Integration with cloud systems for remote tracking*

The doctor who wants an MRI scan from a patient can easily follow the process by getting the results from a cloud system. Therefore, the doctor examines the necessary data from his/her computer without the need for people to bring physical output to his/her.

9. *Storage encryption module*

Because the patient's data is confidential, data security is mandatory. Therefore, encryption while storing data, increases security. Data can be safely stored to the database using secure encryption algorithms so that this security feature motivates the product.

10. *Biometric authentication*

If hospitals want to increase security in the authentication process in MRI software, biometric authentication modules can be used for this purpose. Different biometric authentication methods can be used.

[5] Acquired from: https://4rai.com/blog/3-types-of-mri-machines

11. Backup software component

The need for patient's data not to be lost increases the importance of backup. So we included that component for backup data.

12. Basic coloring abilities for general use cases.

Although coloring techniques are required for some cases, it is not required for most cases. So it is sensible that listing basic coloring abilities makes sense.

13. Enhanced coloring (false coloring etc.) abilities for a more specific diagnosis

Imaging is a visual process. Coloring allows people examining the scans understand faulty regions more

14. Dynamic monitor configuration

Not all hospitals have the same number of monitors, display screens. Thus, it is important to make the user interface screen parts configurable so that users can drag these interface parts to desired display screens.

15. Exporting scan file

Sometimes a scan file is examined by more than one doctor and more than once throughout time. Thus, a software package that exports scan files to different extensions(pdf etc.) is a feature motivating our products.

16. Comparison with the old scans of patient

Including old scans of patients on-board is a useful feature for medical doctors. They do not need to search from another platform that way, and there will be no disambiguation between scans.

17. Error-removal

Error removal is a process widely used in electronics. Noise is not uncommon in image processing. Cleaning up the noises generated by electronic components can be life saving considering the usage scenarios of MRI machines.

18. Shorter scan duration

Scan durations are crucial for both the quality of image output and comfort of the patient. We can say that these two aspects are connected to each other. If scans take too long, patients will have problems being stationary. Therefore, keeping the scan durations as short as they can be is very beneficial.

19. Faster image processing for future implementation of fMRI

Image recognition is tightly bound to graphics processing technologies which mostly requires specialized hardware. Bleeding edge hardware is required for high outcome.

Implementation should be modular to allow faster and more reliable image processing and recognition. Since fMRI is a high technology product that shows MRI images dynamically, loads of processing power and software optimization is needed.

20. Comparison of organs of the patient with healthy person's organ

It can be said that one of the first steps of automatic diagnosis is analyzing the parts. Comparing abnormal scans with healthy scans of organs is useful for faster diagnosis.

21. Exam cards model

MRI machines should provide predefined sets of instructions that helps to accelerate the process of configuration.

22. Pinning and Commenting

In some cases doctors work together in order to recognize the disease. For better communication we concluded that being able to pin the image and commenting on it may help the doctors to explain themselves better.

23. Mobile Applications for Performance Monitoring and Displaying Scans

It could be very time efficient if doctors would sit in their room and examine scans, while maintainers do the same except analyzing performance data.

24. Performance Data Generation

One of the key tracking features is performance data. Maintainers should be able to read and interpret performance data for shorter scans.

Product Portfolio

While constructing our product portfolio we consider different motivating features for different products. Doing that enables building a product family whose members address different use cases and needs observed in the market. These needs and use cases are taken from the market analysis section, and the report given us.

We define 5 different products in the following section and elaborate the distinctive features forming the final product.

MRI MR-001:

First MRI machine of our product family that has basic capabilities. This product corresponds to a product that is placed in the low-end. In the following section we order features that are included in this product. Although this product has just the basic

capabilities it can be upgraded with other features that we define in the motivating features section.

- Simple and clear user interface module
- Image processing
- Automatic patient bed adjustment using patient data
- Exam cards module
- Performance monitoring module*
- Software for displaying scans
- Obligatory tracking and control system*
- Basic coloring features
- Low-Medium Display Resolution
- Low-Medium Rendering Resolution

*Please refer to the software architecture report for further detail.

MRI MR-SAFE:

This product includes advanced level security modules as well as the features of MRI 001. Storage encryption and biometric authentication modules have been added. It is a software that can be used especially in hospitals, such as military hospitals, where security must be very high. Encryption level and biometric authentication method (fingerprint recognition, iris recognition, face recognition) to be used can be changed according to the customer's request.

- Simple and clear user interface module
- Image processing
- Exam cards module
- Performance monitoring module*
- Software for displaying scans
- Obligatory tracking and control system*
- Basic coloring features
- Automatic patient bed adjustment using patient data
- Low-Medium Display Resolution
- Low-Medium Rendering Resolution
- *Storage encryption*
- *Biometric authentication*

MRI MR-V:

Specialized MRI software for veterinary applications. This product has common features coming from MRI-001 Software. It also includes useful features for animal scanning procedures. These variable features are:

- Simple and clear user interface module
- Image processing
- Specialized exam cards module for veterinary applications that includes limited number of types of animals.
- Performance monitoring module
- Software for displaying scans
- Obligatory tracking and control system
- Low-Medium Display Resolution
- Low-Medium Rendering Resolution
- *Coloring features optimized for animal scans*
- *Faster image processing for the clarity of images captured from animals*
- *Animal relieving sound output*

MRI MR-V-2:

Specialized MRI software for veterinary applications. This product has common features coming from MRI-001 Software. It also includes useful features for animal scanning procedures for a variety of animals. These variable features are:

- Simple and clear user interface module
- Image processing
- Advanced exam cards module for veterinary applications that includes numerous types of animal diseases.
- Performance monitoring module
- Software for displaying scans
- Obligatory tracking and control system
- Low-Medium-High Display Resolution
- Low-Medium-High Rendering Resolution
- *Coloring features optimized for animal scans*
- *Faster image processing for the clarity of images captured from animals*
- *Animal relieving sound output*
- *Integration with veterinary PACS*
- *Auto diagnosis using image recognition*
- *Treatment recommendation system for animals using specialized AI*

MRI MR-002

More advanced model than 001, 002 implements cloud features that allow remote tracking and full integration with the hospital's PACS.

- Simple and clear user interface module
- Automatic patient bed adjustment using patient data
- Image processing
- Enhanced Exam cards module
- Performance monitoring module
- Software for displaying scans
- Low-Medium-High Display Resolution
- Low-Medium-High Rendering Resolution
- Obligatory tracking and control system
- Basic coloring features
- *Integration with PACS*
- *Uploadable scans*
- *Comparison with the old scans of patient*

MRI MR-003

More advanced model than 001, 002 implements modular user interface with advanced user experience. It includes a more advanced piece of software for analyzing scans manually. Also, storage encryption and biometric authentication are provided for advanced level security.

- Image processing
- Advanced Exam cards module
- Automatic patient bed adjustment using patient data
- Performance monitoring module
- Software for displaying scans
- Obligatory tracking and control system
- Integration with PACS
- Uploadable scans
- Comparison with the old scans of the patient
- Storage encryption
- Biometric authentication
- *All types of coloring*
- *Dynamic monitor configuration*
- *Comparison of organ parts of the patient with healthy person's organ parts*
- *Smart pen and smart screen support*

- *Image analysis module equipped with noise reduction, subtraction, enhancement, edge detection*

MRI MR-004

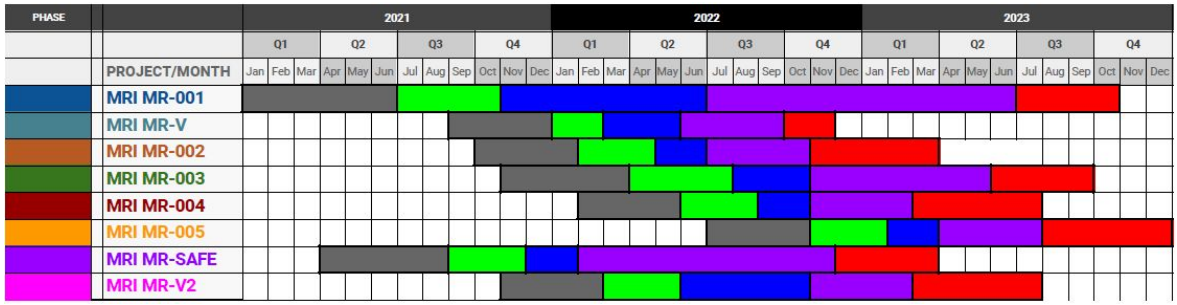
In this product, auto diagnosis modules have been added. Diagnosis process will be automated after that with the image recognition and error-removal features.

- Image processing
- Advanced Exam cards module
- Automatic patient bed adjustment using patient data
- Advanced performance monitoring module
- Software for displaying scans
- Low-Medium-High Display Resolution
- Low-Medium-High Rendering Resolution
- Obligatory tracking and control system
- Integration with PACS
- Uploadable scans
- Comparison with the old scans of patient
- Enhanced coloring (false coloring etc.) abilities for a more specific diagnosis
- *Dynamic monitor configuration*
- *Error-removal using specialized AI*
- *Faster image recognition for future implementation of fMRI*
- *Comparison of organ parts of the patient with healthy person's organ parts*
- *Treatment and medicine recommendation system*

MRI MR-005

- Advanced image processing
- Integration with PACS
- Automatic patient bed adjustment using patient data
- Enhanced coloring (false coloring etc.) abilities for a more specific diagnosis
- Advanced Exam Cards module
- AI-enabled performance monitoring module
- *Mobile apps for controlling and displaying scans*
- *Mobile apps for performance metrics*
- Obligatory tracking and control system

Product Line Timeline



Legend: Gray: Introduction, Green: Growth, Blue: Maturity, Purple: Saturation, Red: Degeneration

KANO Survey

The Kano method [Kano 1984; Kano et al. 1996; Sauerwein 2000] allows choosing a set of product features that yield high customer satisfaction. The key element of this method is the classification scheme for customer requirements.

Below is the KANO survey prepared by the software production line team for determining the categories of the features mentioned above.

1-F) Suppose that MRI Software has the ability of the auto-diagnosis, what would you think about that?

1-DF) Suppose that MRI Software lacks the ability of the auto-diagnosis, what would you think about that?

2-F) Suppose that MRI software has image processing

2-DF) Suppose that MRI software does not have image processing

3-F) Suppose that MRI Software lacks ability to treatment and medicine recommendation systems, what would you think about that?

3-DF) Suppose that MRI Software lacks ability to treatment and medicine recommendation systems, what would you think about that?

4-F) Suppose that MRI software has the ability to adjust rendering and resolution of the scans, what would you think about that?

4-DF) Suppose that MRI software does not have the ability to adjust rendering and resolution of the scans, what would you think about that?

5-F) Suppose that MRI software has the ability to adjust the bed automatically according to patient data, what would you think about that?

5-DF) Suppose that MRI software lacks the ability to adjust the bed automatically according to patient data, what would you think about that?

6-F) Suppose that the MRI Software has a veterinary exam cards model, what would you think about that?

6-DF) Suppose that the MRI Software does not have a veterinary exam cards model, what would you think about that?

7-F) Suppose that MRI software has remote control, what would you think about that?

7-DF) Suppose that MRI software does not have remote control, what would you think about that?

8-F) Suppose that MRI software has integration with cloud systems for remote tracking, what would you think about that?

8-DF) Suppose that MRI software has no integration with cloud systems for remote tracking, what would you think about that?

9-F) Suppose that MRI software has a storage encryption module, what would you think about that?

9-DF) Suppose that MRI software does not have a storage encryption module, what would you think about that?

10-F) Suppose that MRI software has biometric authentication, what would you think about that?

10-DF) Suppose that MRI software does not have biometric authentication, what would you think about that?

11-F) Suppose that MRI software has the ability to backup software components, what would you think about that?

11-DF) Suppose that MRI software lacks the ability to backup software components, what would you think about that?

12-F) Suppose that MRI software has the ability to color the scans, what would you think about that?

12-DF) Suppose that MRI software lacks the ability to color the scans, what would you think about that?

13-F) Suppose that MRI software provides you enhanced coloring abilities for a more specific diagnosis, what would you think about that?

13-DF) Suppose that MRI software does not provide you enhanced coloring abilities for a more specific diagnosis, what would you think about that?

14-F) Suppose that MRI software provides you a dynamic monitor configuration, what would you think about that?

14-DF) Suppose that MRI software does not provide you a dynamic monitor configuration, what would you think about that?

15-F) Suppose that MRI software has the ability to export scan files to other devices or cloud machines, what would you think about that?

15-DF) Suppose that MRI software lacks the ability to export scan files to other devices or cloud machines, what would you think about that?

16-F) Suppose that MRI Software has the ability of comparing with the old scans, what would you think about that?

16-DF) Suppose that MRI Software lacks the ability of comparing the old scans, what would you think about that?

17-F) Suppose that MRI Software has the ability of removing scan errors, what would you think about that?

17-DF) Suppose that MRI Software lacks the ability of removing scan errors, what would you think about that?

18-F) Suppose that MRI Software has a shorter scan duration, what would you think about that?

18-DF) Suppose that MRI Software does not have a shorter scan duration, what would you think about that?

19-F) Suppose that MRI Software has the ability of faster image processing, what would you think about that?

19-DF) Suppose that MRI Software lacks the ability of faster image processing, what would you think about that?

20-F) Suppose that MRI Software has the ability of comparing healthy organs with the patient's organ scan?

20-DF) Suppose that MRI Software lacks the ability of comparing healthy organs with the patient's organ scan?

21-F) Suppose that MRI Software implements an exam cards model, what would you think about that?

21-DF) Suppose that MRI Software does not implement an exam cards model, what would you think about that?

22-F) Suppose that MRI Software has the ability of pinning and commenting on scans, what would you think about that?

22-DF) Suppose that MRI Software lacks the ability of pinning and commenting on scans, what would you think about that?

23-F) Suppose that MRI Software has mobile applications for performance monitoring and displaying scans, what would you think about that?

23-DF) Suppose that MRI Software does not have mobile applications for performance monitoring and displaying scans, what would you think about that?

24-F) Suppose that MRI Software has the ability of generating performance data, what would you think about that?

24-DF) Suppose that MRI Software lacks the ability of generating performance data, what would you think about that?

Application Assumption of KANO Method

According to our scenario, by applying the questionnaire prepared above, we categorize the features and determine the basic requirements, satisfiers, delighters and indifferent requirements. By doing that, we realize how we should arrange the features to be used in our products. To engage customers to use our products, they must contain basic requirements and also many satisfiers to attract the attention of customers. And delighters provide high satisfaction. Features must be used in the right product at the right time in the product portfolio so that the transition of products to phase “poor dogs*” can be delayed as much as possible. Below survey application shows us which features will be included in which classification. According to the result, the features were distributed logically to the products in the portfolio.

Functional Questions

	Like	Expected	Don't Care	Can Live With it	Dislike
1	X				
2		X			
3	X				
4	X				
5			X		
6		X			
7			X		
8	X				
9	X				
10	X				
11		X			
12		X			
13	X				
14	X				
15		X			
16	X				
17		X			
18	X				
19	X				
20	X				
21		X			
22		X			
23	X				
24		X			

Dysfunctional Questions

	Like	Expected	Don't Care	Can Live With it	Dislike
1				X	
2					X
3				X	
4					X
5			X		
6					X
7				X	
8				X	
9					X
10			X		
11					X
12					X
13					X
14				X	
15					X
16				X	
17					X
18			X		
19			X		
20				X	
21					X
22			X		
23			X		
24					X

Feature Categorization[6]

	Feature Name	Category
1	Auto-diagnosis	Delighter
2	Image Processing	Basic
3	Treatment Rec. System	Delighter
4	Adjustable Resolution	Satisfier
5	Auto Patient Bed Adjustment	Indifferent
6	Veterinary Exam Card	Basic
7	Remote Control	Indifferent
8	Integration with PACS	Delighter
9	Storage Encryption	Satisfier
10	Biometric Authentication	Delighter
11	Backup	Basic
12	Basic Coloring	Basic
13	Enhanced Coloring	Satisfier
14	Dynamic Configuration	Delighter
15	Exporting Scan Files	Basic
16	Comparison of Old Scans	Delighter
17	Error Removal	Basic
18	Shorter Scan Duration	Delighter
19	Faster Image Processing	Delighter
20	Comparison with Healthy Person	Satisfier
21	Exam Cards Model	Basic
22	Pinning and Commenting	Indifferent
23	Mobile Applications	Delighter
24	Performance Data Generation	Basic

[6] Klaus Pohl, Günter Böckle, Frank J. van der Linden, *Software Product Line Engineering Foundations, Principles and Techniques*, 2005, ch. 9, pg. 182

Traceability Matrix

Feature/Product	MR-1	MR-2	MR-3	MR-4	MR-5	MR-v1	MR-v2	MR-S
1 (Auto-diagnosis)				X			X	
2 (Image Processing)	X	X	X	X	X	X	X	X
3 (Treatment Rec. System)				X			X	
4 (Adjustable resolution)	X	XX	XXX	XXX	XXX		X	
5 (Auto Patient Bed Adjustment)	X	X	X	X	X			X
6 (Veterinary Exam Cards)						X	X	
7 (Remote Control)	X	X	X	X	X			X
8 (Integration with PACS)		X	X	X	X		X	
9 (Storage Encryption Module)			X		X			X
10 (Biometric Authentication)					X			X
11 (Backup SC)				X	X			X
12 (Basic Coloring)	X	X	X	X		X		X
13 (Enhanced Coloring)			X	X	X		X	
14 (Dynamic Configuration)			X		X			

15 (Exporting Scan Files)	X	X	X	X	X	X	X	X
16(Comparison of old scans)		X	X	X				
17 (Error Removal)	X	X	X	X	X	X	X	X
18 (Shorter scan duration)				X				
19 (Faster Image Processing for fMRI)				X				
20 (Comparison with healthy person)			X		X			
21 (Exam Cards Model)	X	XX	XXX	XXX	XXX			X
22 (Pinning and commenting)			X		X			
23 (Mobile Applications)					X		X	
24 (Performance Data Generation)	X	X	XX	XX	XXX	X	XX	X

Requirements	Feature #	Feature Description	Architecture Component
R1	14	Dynamic monitor configuration	Monitoring plug-in
R2	16	Comparison with old scans	Comparison component
R3	20	Comparison with healthy person	Comparison component
R4	12/13	Different types of coloring	Coloring component
R5	6/21	Animal/Human exam cards	Exam cards component
R6	4	Adjustable resolution	Resolution component
R8	22	Pinning and commenting	Interaction component
R9	2/17	Image processing and error removal	Image processing component

Traceability Matrix for Application Modelling and User Interface

Requirements	Feature	Feature Description	Architecture Component
R1	24	Performance monitoring	Performance module
R2	15, 16, 23	Scan modularity	Scan acquisition module
R3	1, 20	Efficient automatic diagnosis	Auto-diagnosis plug-in
R4	9	Selectable encryption	Encryption manager
R5	2, 19, 21	Efficient image processing for different use cases	Image processing module

Traceability Matrix for Tracking and Control System

Requirements	Feature	Feature Description	Architecture Component
R1	10	Biometric authentication	Authentication plug-in
R2	2, 19	Image processing, error removal	Image acquisition plug-in
R3	2, 19	Image processing, error removal	Image acquisition plug-in
R4	8	Integration with PACS	Storage manager
R5	9	Storage encryption	Encryption manager
R6	1	Auto-diagnosis	Auto-diagnosis plug-in

Traceability Matrix for System Architecture and Infrastructure

Domain Variability Model

To meet the variability in the product family to be created in our product line, a domain variability model must be created. Because the variability of the product line must be managed, we reduced the complexity by dividing the architecture by 3 and we were able to conduct detailed analysis by narrowing our field of study.

After this section, the studies will be divided into 3 sub-areas and will be carried out as such. Domain division will be handled as the architectural team did in the Software architecture report. According to this separation process, the architecture of our products is divided into 3 main domains: 1) Application modeling and User interface 2) Tracking and control system 3) System architecture and infrastructure. While creating the product line, the architectures of the products will be evaluated from 3 different perspectives, and after the studies of 3 individual subdomains are integrated with each other, the software product line will be created as intended. From this section, the main design rationale of dividing architecture like this is that to be able to work independently from each other as much as possible and reduce the complexity of the analysis further we conduct for creating a product line. A summary of what these subdomains cover is given in "A Short Description of the Proposed Architecture" section under the "Architecture Design Method" heading above.

In this section, we will create a domain variability model to manage the variability of our products in the portfolio we created in the product management section. The variation points in the MRI machine software that we will create in our product line will be determined separately according to the subdomains we assigned in the above paragraph. Afterward, variation instances in these variation points will be found and consistent variability dependencies will be created.

Domain Variability Model for Application Modelling and User Interface

[Yunus Kerem Türk]

In software product line engineering, reusability is essential. We use variability modelling to support the development and the reuse of User Interface components. Thus, in this section our main purpose will be to identify common and variable features. While doing that we will try to answer some questions such as "What does vary in the domain of MRI machine software? ", "How does it vary considering the different versions and models we defined in the portfolio?" As answers to these questions we will find variation points and variants of User Interface components of MRI machine software.

The user interface component will ease the doctor-machine interaction. This is necessary for all software developed to be used outside the computer science domain. Although the MRI user interface does not add new functionality to the MRI machine it is the only way to interact with software. While considering that software architecture team analyzed some patterns and decided on MVC(Model-View-Controller). It is sensible to use in our case too. So, our problem becomes how to display the knowledge gained from Tracking and Control System and ease the doctor while operating and analyzing the scan. From this point we will define variation points and variants of our product line, and bind them with features we defined earlier.

I will start with a feature tree to give a quick look about variation points of User Interface and then give a detailed explanation, use case scenarios, sequence diagrams about that variation point.

Feature Model for MRI Software Application Modelling and User Interface Domain

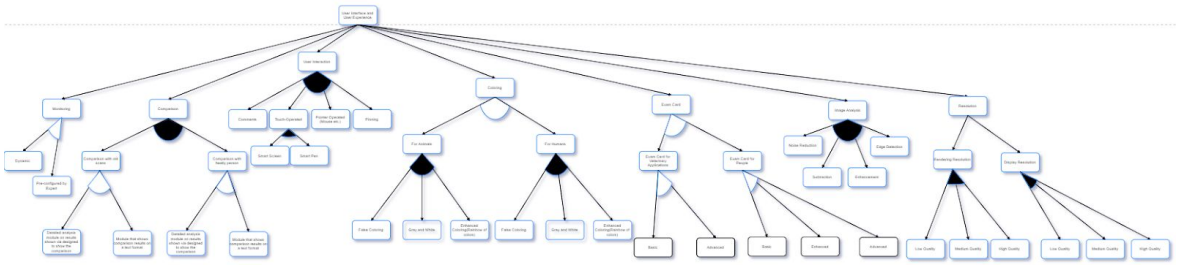


Fig. 3.5

We define 5 different main variation points in Application Modelling and User Interface Domain which are *Monitoring, Coloring, Exam Card, Comparison ,Resolution.* These five variants are derived from motivating features we define in section Product Management.

Except monitoring all variants are divided into more detailed variants. We will analyze these five main variants and their sub-variants in the following section.

Variation point: Monitoring

Variants: Pre-configured, dynamic

While doing our research we concluded that not every doctor wants to see the same screen order. Some of them want to see a false colored version of the scan on the first screen, some of them want to see patient data on the first screen. This difference in opinion results in a variation point for monitoring. Thus, we provide 2 different versions for monitor configuration.

1. Pre-configured version, set for once by experts and users. While designing our user interface, our first aim is to ease the process of taking and analyzing a scan. So, it is sensible that a configuration at first is provided to users. But once the configuration process is done, users cannot configure the display screen according to their current needs.
2. Dynamic version is adaptable for all use cases. With that, doctors can adjust the display screens according to their needs.

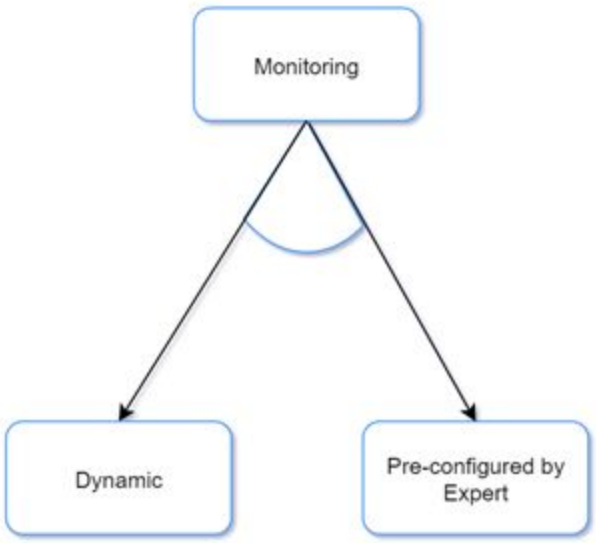


Fig. 3.6

Variation point: Coloring

Variants: False coloring, Gray and White, Rainbow of colors

Different coloring techniques are used for different use cases. Doctor decides on which technique should be applied while taking a scan so that he can see the results clearly. When the process is done and the scan is completed by lower layers of architecture the result is returned to the view. Then these requested results are displayed to the doctor.

We introduced five different models in our portfolio and one of them was for veterinary applications. We concluded that coloring, and the display process will be different for animals and humans. So in feature trees, coloring is divided into two sub features; one of them is for people, and the other is for animals.

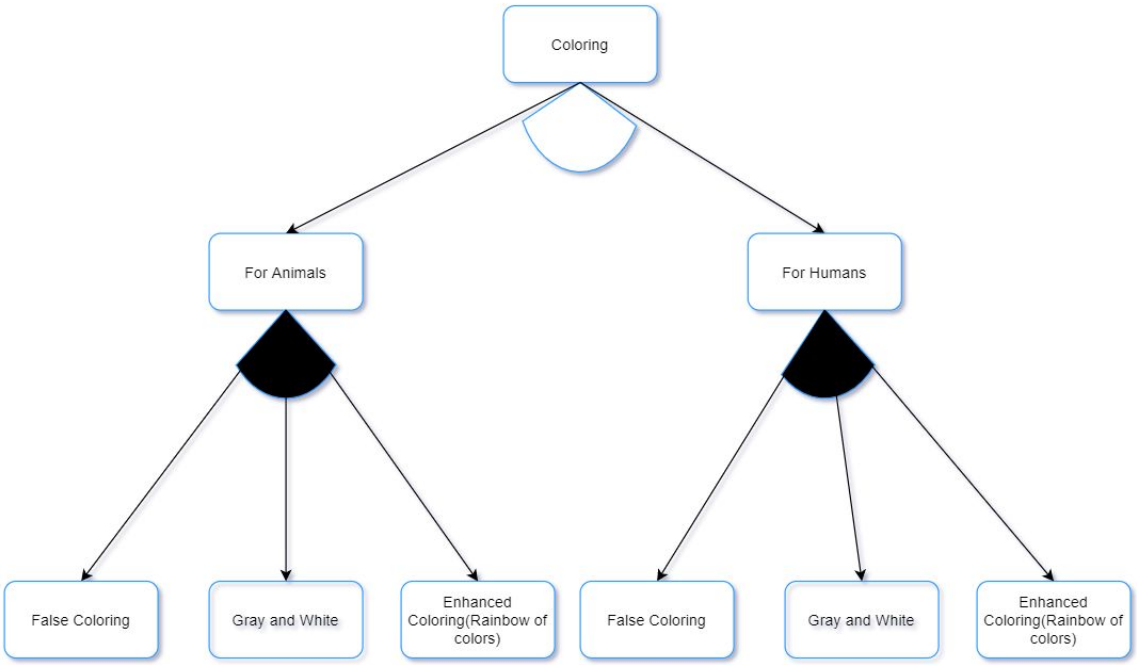


Fig. 3.7

Variation point: Exam Card

Variants: For humans, for animals

Exam card model is introduced by the software architecture team and selected as commonality. But when animal exam cards were introduced, it became a variability. Exam cards model can be described as follows, the user interacts with the controller to define the type of procedure he wants to perform. Depending on that value, the controller then requests the exam cards to model the appropriate exam card. The Exam Cards Library is consulted, and a model is returned to the controller in domain specific language, understood by the control layer. With that result, it sends a command to the Tracking and Control Layer. Next, in the Tracking and Control Layer transformations are applied on the model provided by the exam cards model and the MRI machine performs the task, finalizing with the output being shown for the user on the view component.

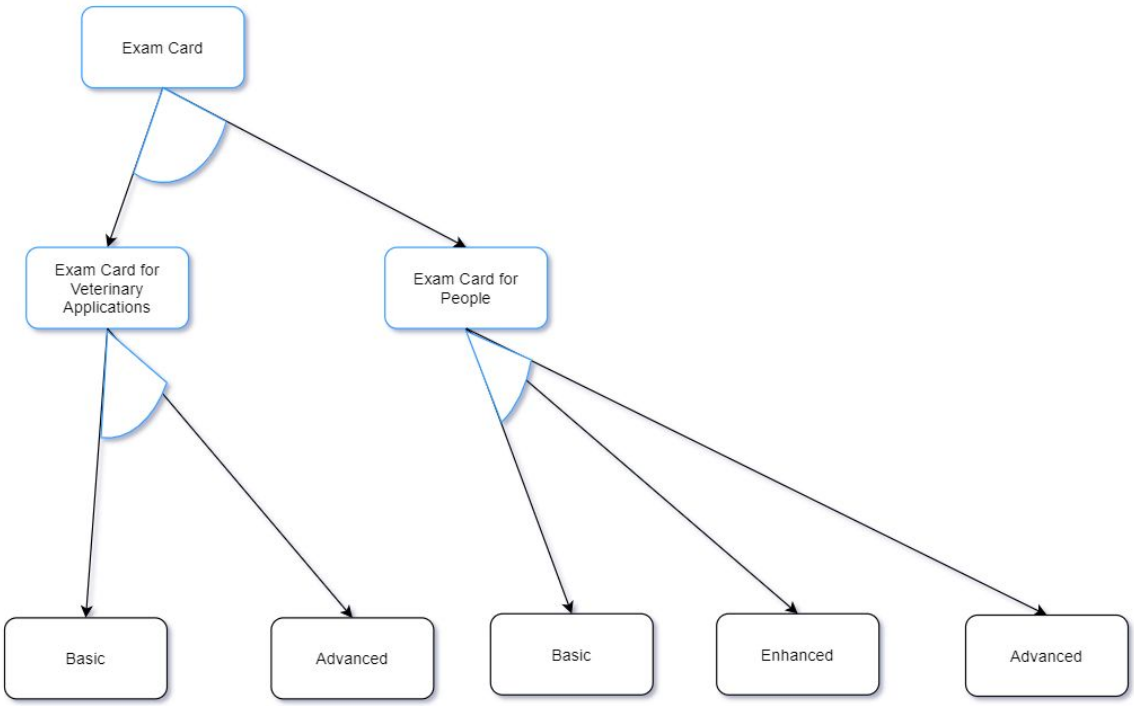


Fig. 3.8

Variation point: Comparison

Variants: Comparison with healthy organs, comparison with past data

In medicine, being able to compare past patient data and normal person's data with current data is crucial and helps to identify the disease. It also helps to measure the progress of the disease. When these comparisons are requested, the data stored in the database is requested and presented along with current scan.

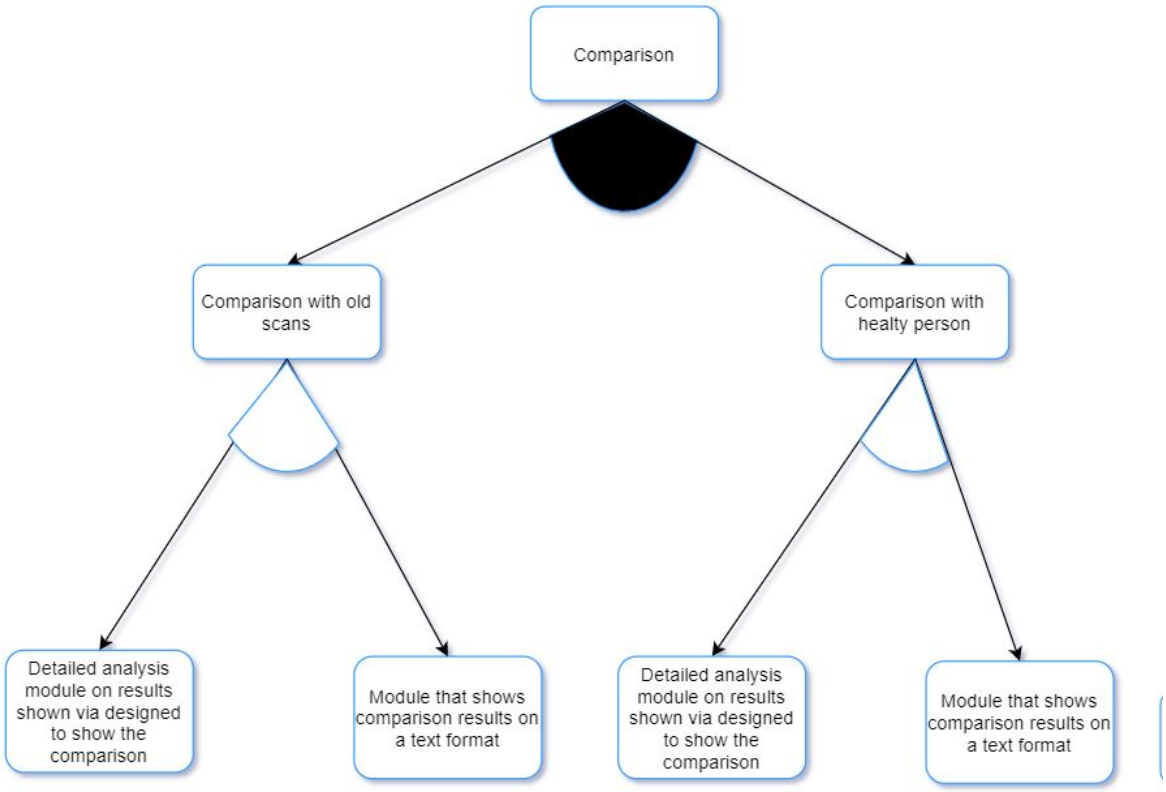
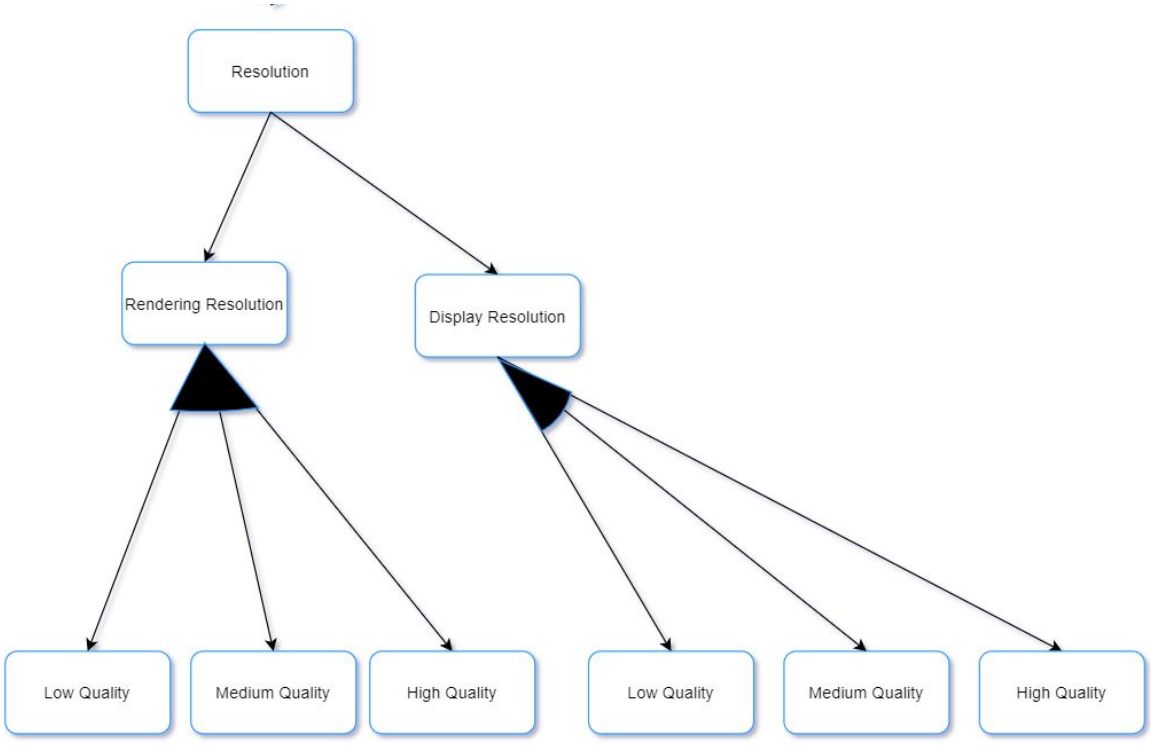


Fig. 3.9

Variation point: Resolution

Variants: Resolution is divided into two section: Rendering Resolution and Display Resolution

Display resolution determines how many pixels are displayed to the user while rendering resolution determines how many pixels are rendered. In some cases, rendering resolution may be lower than display resolution in order to increase GPU performance. For example, 1440*960 pixels can be upsampled to 1920*1080. Although upsampling may cause some deterioration in image, it is acceptable in most cases. Thus, we decided to provide an option for both display resolution and rendering resolution. They are not set once, so it is possible to switch between both rendering resolution and display resolution. It is required because some devices does not support high resolutions and some of them encounter with high waiting time while rendering the original image.

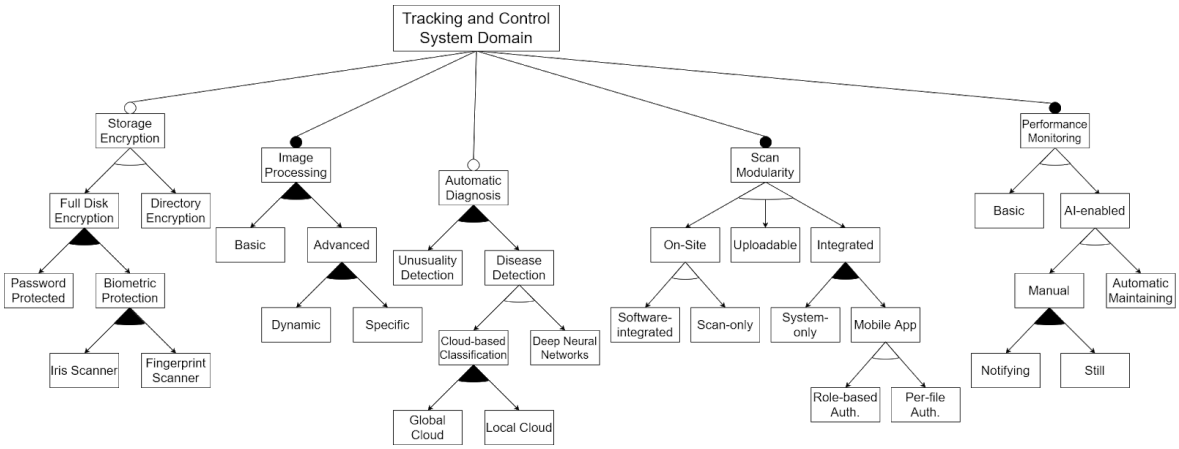


Domain Variability Model for Tracking and Control Systems [Onur Demirezen]

The main goal of tracking and control domain is to use the newly-implemented features without losing performance. Decision making is a very important process in MRI software. Performance should always be considered while making a decision, therefore performance analysis and accordingly making decisions using tracking and control systems make the systems more reliable.

Features would be pointless if users were not able to use them. Hence, determining variation points for features is essential for healthy development of the product family. In this chapter, variation points are explained including their variants and dependencies in detail.

Feature Model for MRI Software Tracking and Control Domain



Variation Point: Image Processing

Variants: Basic, Advanced (Specific, Dynamic)

Image processing is trivial to mention when talking about medical imaging. Signals received from the magnetic field inside the MRI machine must be processed precisely and outputted as human-brain-interpretable images.

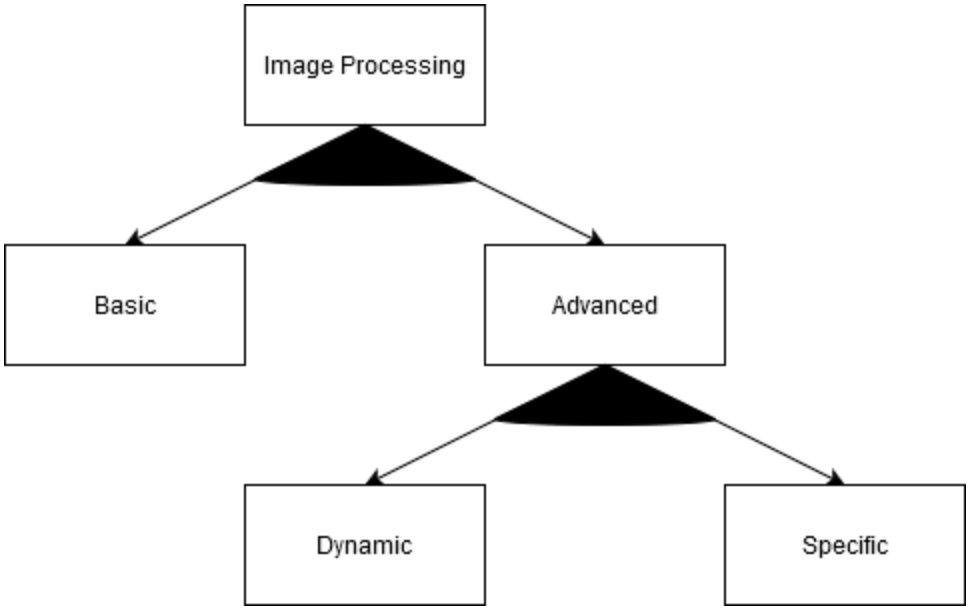
Basic image processing is the traditional method. It gives the images as they are, with intersections. Then they are merged to create a 3D-like image

With advanced image processing, time taken to generate a “basic” image is reduced significantly with noise filtering and optimization algorithms. This allows for more advanced, dense and interpretable images from the tracking and control point of view. Integrated with exam cards models, different image processing algorithms can be used for separate body parts for further time-optimization.

In some cases like veterinary or dentistry specific products, advanced image processing techniques can be further developed and optimized for specific cases. Since animal bodies differ from human parts for veterinary applications, algorithms used in advanced image processing should be adjusted accordingly.

Dynamic processing is generating the images in real-time to observe for instance stomach movements. This technology is very advanced and hard to implement. Significant time optimization is needed for image processing algorithms. One can say that the resolution of the images may decrease for dynamic processing to function.

Variability Dependency: Image processing is essential. Customers need to select one of the techniques according to their needs. Variants have an encapsulation relation between them. Advanced image processing already includes basic for instance, and specific image processing includes advanced. Dynamic processing differs in that manner, for it is used in much more specific and advanced applications.



Variation Point: Storage Encryption

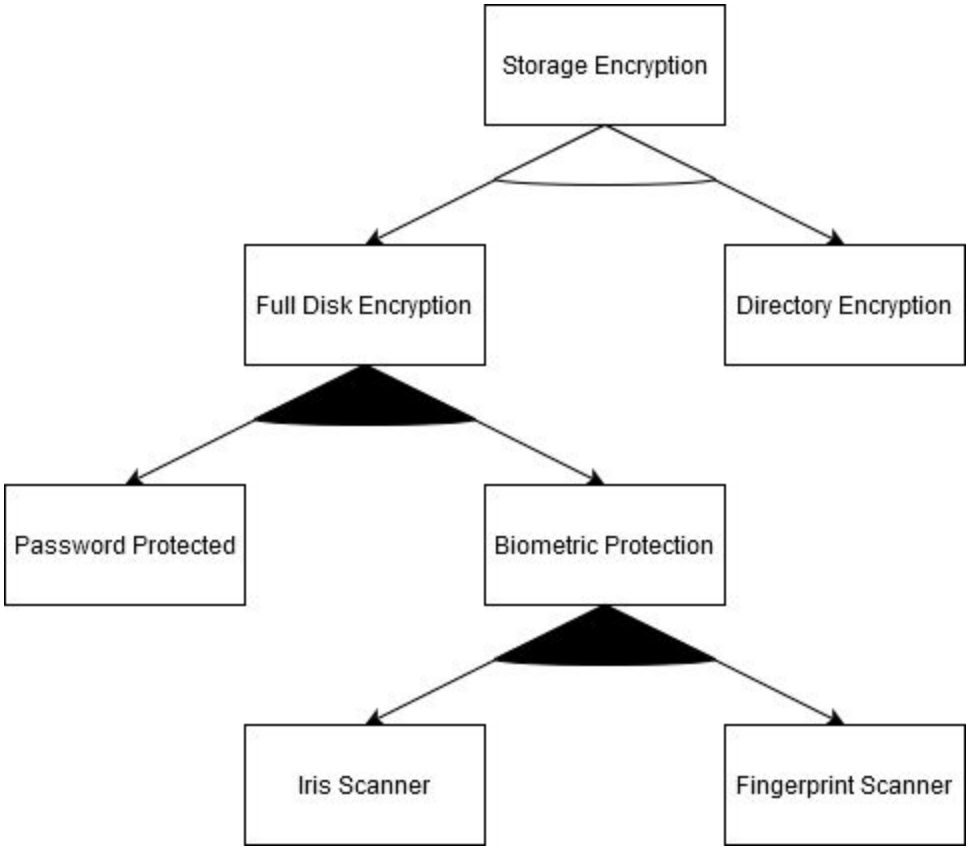
Variants: Full-Disk Encryption (Password Protected, Biometric Protection), Directory Encryption

Storage encryption affects the performance of the disk, therefore it must be considered in the tracking and control domain.

Full disk encryption is, as self-explanatory as it is, the process of encrypting the storage as a whole. This provides the maximum security for the stored data, but writing to and reading from fully encrypted disks takes more time than not encrypted ones. Encryption may delay the big processes machines need to perform while providing loads of security.

Directory encryption separates the patient-related information data with imaging data in separate virtual or, optimally, physical disks. Whereby, large imaging outputs written to disk are not slowed-down by encryption. Relations between disks are managed by software. While not providing full security over all the data, directory encryption secures personal information of patients.

Variability Dependency: Storage encryption is not essential since devices have password protection, but customers concerned with disk security in case of burglary can select one of the variants for their needs but they need to consider the performance differences that can be seen with tracking and control modules. If the customer values security over performance, they can choose to have full encryption. If they want maximum performance, they should have no storage encryption.



Variation Point: Automatic Diagnosis

Variants: Unusuality Detection, Disease Detection (Deep Neural Networks, Cloud Based Classification (Global Cloud, Local Cloud))

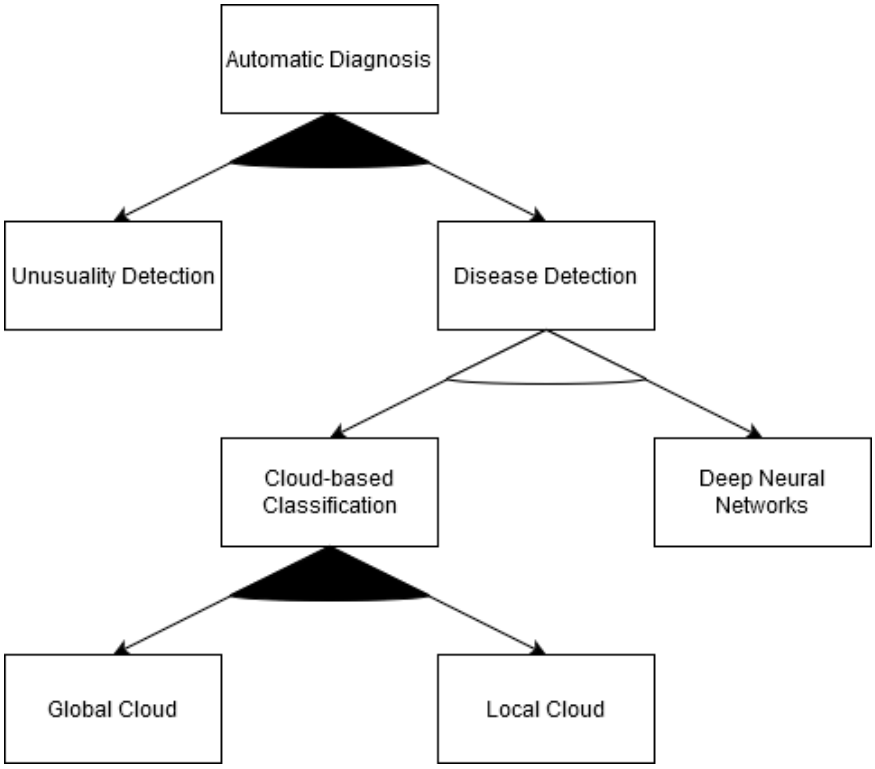
Machine learning algorithms are different from each other in varying use cases. Since images must be analyzed almost pixel-by-pixel, time taken for completion is relatively long. A large dataset must be provided from older scans that are classified correctly by medical doctors for precision. Dataset must contain healthy people's scans along with diseased patients.

With the unusuality detection technique, the system only determines if there is an unusual situation in the scan without providing any further information. With a large enough database and a regression algorithm, the system can provide the doctor with an early detection of anomaly. A prior dataset can be stored in disk. Doctors can label new scans and expand the dataset. Model fits each time the dataset is expanded and makes decisions accordingly. Output is only an area of the image if there is an anomaly, and nothing if there is nothing abnormal.

Use of deep neural networks needs a more advanced dataset with detailed description of the anomalies in scans. For instance, the system can output an area with a tumor and tell that there is a lesion similar to a malignant cancer in the lung pariton. This reduces the diagnosis time significantly. Since some diseases require early diagnosis, lives can be saved.

Cloud based classification is a universal solution for automatic diagnosis. Dataset will be generated by doctors all over the world using the product. MRI machines will be integrated with a cloud containing a dataset, then perform automatic diagnosis using this dataset. It can be described as a shared knowledge base. Software will contain modules for fetching data, diagnosing and adding new scans to the knowledge base for global benefit. This technique is useful for diagnosing rare diseases. And it is time-efficient since everything will run on cloud computers that are very resource-rich. With global cloud based classification, real time data of the global cloud system will always be reachable. Otherwise, local cloud will only provide periodically updated data.

Variability Dependency: Because the doctors are capable of diagnosing by analyzing scans, auto diagnosis is only motivating. Customers can choose to not have an automatic diagnosis solution, or have only one of them.



Variation Point: Scan Modularity

Variants: On-site (Software-integrated, Scan-only), Uploadable, Integrated (Mobile App (Role-Based Authentication, Per-File Authentication), System-Only)

Modularity of scans are somewhat important for performance analysis. Storage speed is an important performance factor for machines. When storage is close to full, performance of the system is impacted due to fragmentation. For traditional hard drives, outer sections that are slower will be used for new data generation. Multiple-disc array systems are a nightmare in terms of fragmentation and speed. Therefore, scan modularity is in the tracking and control domain.

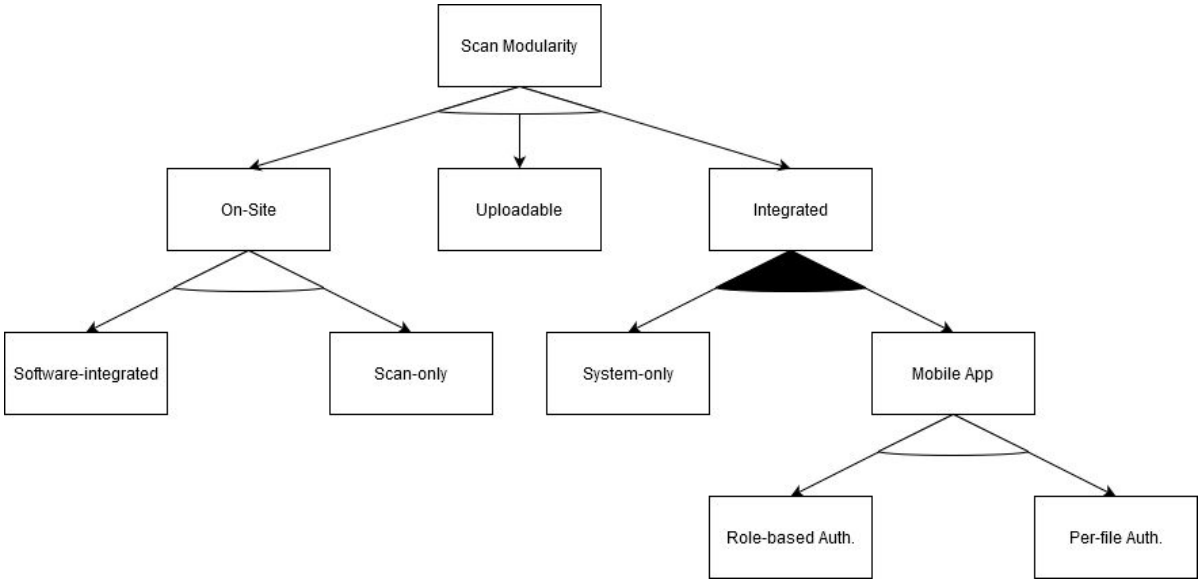
On-site scans only allows customers to manually write scans to a flash drive or a compact disk. Deletion is also manual. Therefore, storage must constantly be checked for space availability for new scans.

Uploadable scans allow uploading scans to the hospital's dedicated patient records system. This process, and deletion of the scans is also manual but no external hardware is needed for sharing the scans.

Integrated modularity is fully automatic and configurable. With this technique, scans are uploaded to PACS automatically with periods. System knows if the storage will overflow, deletes the scans already uploaded to PACS, managing the storage efficiently. Doctors do

not need to check the storage before performing scans, knowing the system is handling everything. If the integration is system-only, scans can only be uploaded to PACS. But if a mobile app is included, every doctor can see the scans if they are authorized.

Variability Dependency: Scan modularity is mandatory. Scans must be transferable to other devices, even to patients'. Customers can select one of the methods they desire.



Variation Point: Performance Monitoring

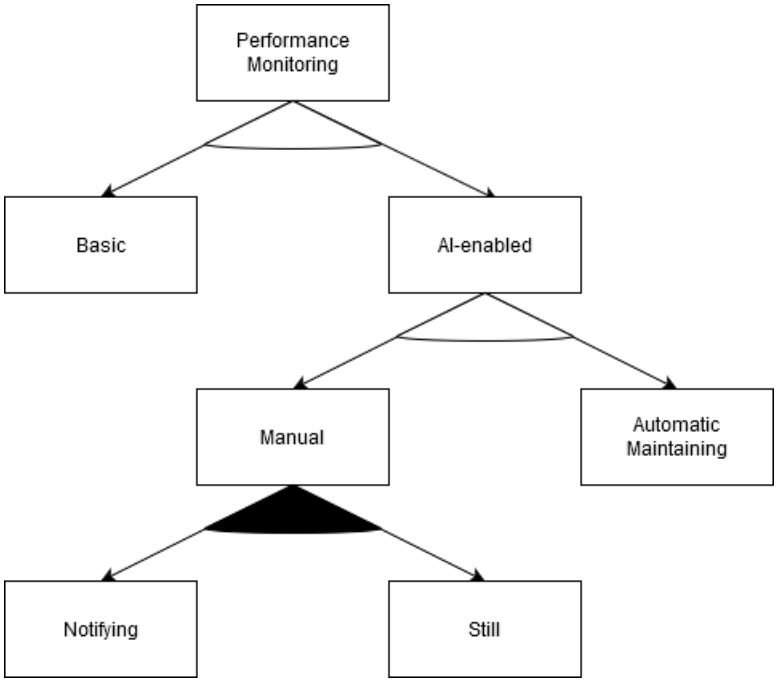
Variants: Basic (On-site, Mobile App), AI-enabled (Manual (Notifying, Still), Auto-maintaining)

Performance monitoring is a very important aspect for the machines to work. With basic monitoring, performance data will be stored in-device or be sent to mobile apps of maintainers to analyze.

With enabling AI, special software will give meaning to performance data and will share insights with maintainers. With manual maintaining, maintainers will choose between getting notified when something is wrong, or manually checking AI-generated insights.

Auto maintaining is the last step that software will alter the machine parameters to keep performance at maximum.

Variability Dependency: Performance monitoring is mandatory. Customers can choose one or more from the variants.


Performance Monitoring
Basic
AI-enabled
Manual
Automatic Maintaining
Notifying
Still

Domain Variability Model for System Architecture and Infrastructure

[Mehmet Arif Şahin]

This domain is organized with easy to integrate and exchange features to provide variability for product families which are MRI machine software. To accomplish this, variability in the product line must be managed well. Therefore, in this section, we create a variability model for the MRI machine solution domain which is system architecture and infrastructure. Variation points were created first during the model creation process. Under these variation points, variants have been determined to meet the diversity of our product portfolio. The relations between variation points and variants(Variability Dependencies) are also shown in detail.

Feature model for MRI Software System Architecture and Infrastructure

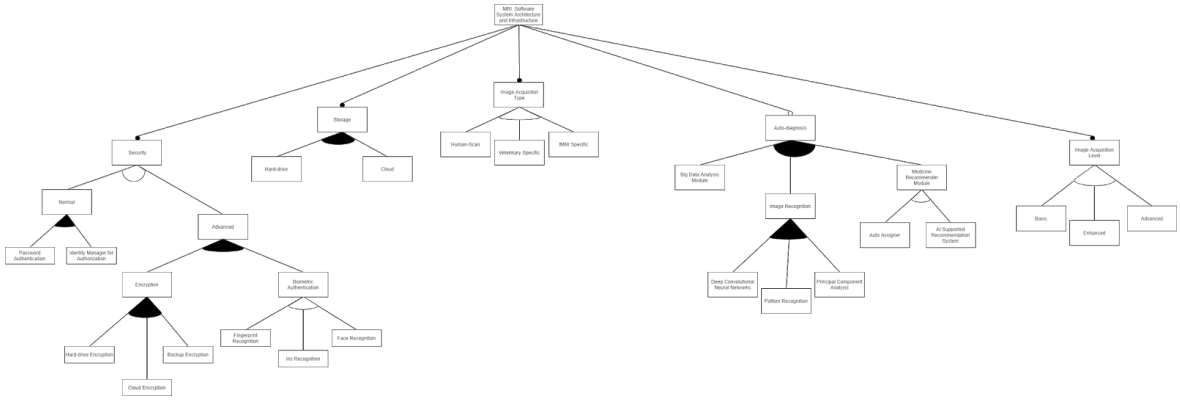


Fig. 3.3

Variants: Normal(password-protected program), Advanced(biometric authentication and encryption)

The importance of patient privacy has made it rational to increase the security of the MRI software when necessary. Confidentiality may be a more sensitive issue, especially for MRI machines to be used in hospitals such as military hospitals. Therefore "Security level for MRI machine software" is appropriate for being a variation point. Customers can choose the level according to their needs. This is an example of external variability but on the other hand, the customer is not aware of how to implement the security in the software. According to customer choice, on the system architecture and infrastructure layer storage

encryption and authentication must be implemented appropriately. The **MRI MR-003** product in our product portfolio has advanced level security.

Basic security level consists of;
Password-based authentication. Therefore, passwords must be hashed then stored. Only authenticated people can reach the patients' data. Also, the identity manager for arranging authorization.

Advanced security level consists of;
Biometric authentication. Developers can choose one of the biometric authentication methods for advanced security. Developers can implement a module that stakeholders approved for biometric authentication methods like face recognition, iris recognition, or fingerprint recognition. This situation requires the safe storage of biometric data. Therefore another security requirement was born here which is encryption. All data, especially the patient's data and biometric data for authentication, is encrypted.

Variability dependency: Hence customers must select one of the variants, there is a mandatory variability dependency between variation point(security level of infrastructure) and variants(normal, advanced).

Of course, each variant for security level needs appropriate user interfaces and applications for authentication methods. Also, advanced security should not increase the processing time therefore appropriate performance monitor modules can be added in Tracking and Control Layer. Therefore some variant constraint dependencies will arise between layers (Variant requires variant -> requires_V_V).
-The basic security level requires a password entry module on the user interface and an appropriate control mechanism on the control layer.
-Advanced security level requires biometric authentication application and user interface on the application layer and appropriate performance monitor module and control mechanism.

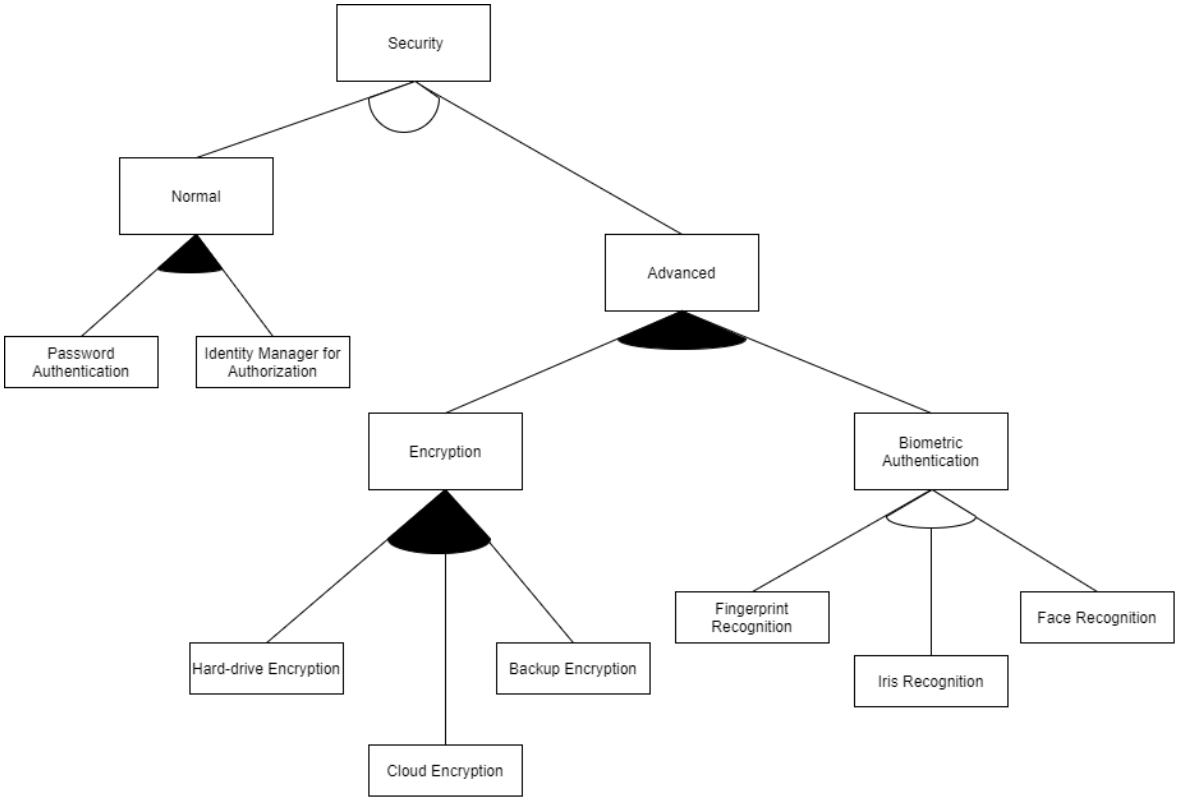


Fig. 3.4

Variation point: Storage for patient's data
Variants: Hard-drive storage, cloud storage

Depending on the customer's request, the software can support the hard-drive storage or cloud storage for the patient's data and scan images. The **MRI MR-002** product in our product portfolio has cloud storage. This product has a remote-tracking mechanism so that necessary data communication is provided through cloud computing.

Variability dependency: There is a mandatory variability dependency between variation points and variants because customers must select at least one of the storage options. Customers also select both of them at the same product. Authentication must be done in both options. To ensure the security of data communication on cloud storage and access, SSL and VPN technologies should be used.

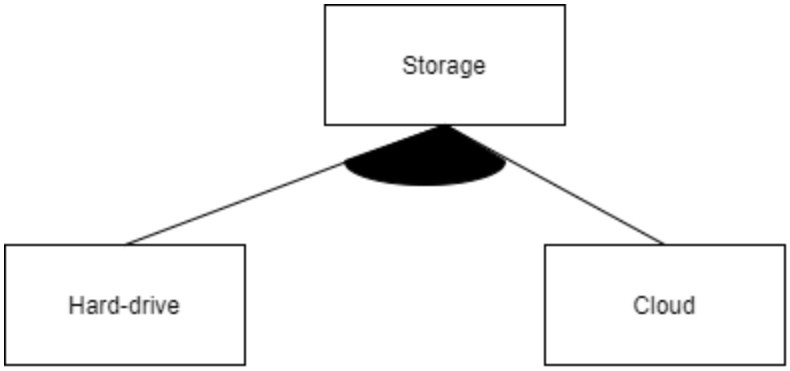


Fig 3.5

Variation point: Image acquisition modules for different types of MRI machines
Variants: Standard Human-Scan MRI Image Acquisition, Veterinary MRI Image Acquisition, Functional MRI Image acquisition

Since the MRI machines can differ according to their usage area, the image acquisition modules must have variability according to the machine types. Our product portfolio, **MR-001** must have standard human-scan image acquisition modules for general and basic scans of patients in hospitals. On the other hand, **MRI MR-V** is the MRI software that will develop specifically used to scan animals. Therefore, image acquisition subprocesses like feature extraction, feature enhancement, error removal differ from standard human-scanning MRI software. Lastly, while developing **MRI MR-004**, fMRI features will be developed. In this stage, functional MRI image acquisition modules should be implemented for future implementations of fMRI. In each case, because of the different image acquisition techniques used, this topic was handled as variability in our software product line and managed appropriately with our product portfolio.

Variability dependency: Each MRI machine software must have exactly one of the image acqıisition modules packaged described above. Therefore, there is a mandatory variability dependency between the variation point (image acquisition modules for different types of MRI machines) and variants(Standart MRI Image Acquisition, Veterinary MRI Image Acquisition, Functional MRI Image acquisition). Because veterinary-specific MRI software only uses “Veterinary MRI Image Acquisition” modules and human-scanning MRI software does not use these modules, these variants exclude each other (excludes_V_V relationship between Standard Human-Scan MRI Image Acquisition and Veterinary MRI Image Acquisition). Also, each variant requires a specific user interface, application model, and control mechanism. Thus, variants require other layer components.

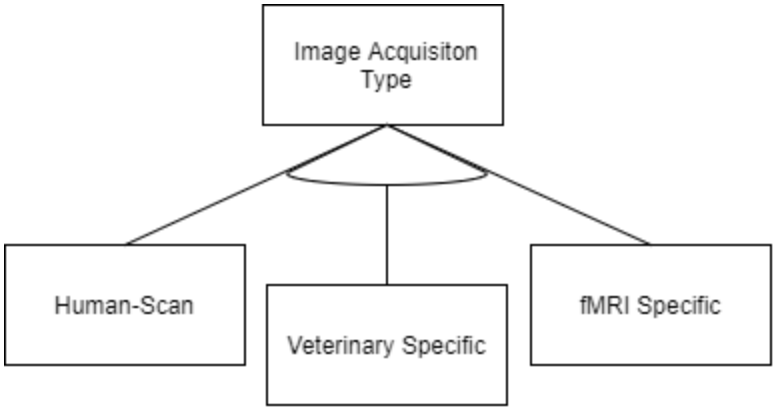


Fig 3.6

Variation point: Image acquisition level
Variants: Basic, enhanced, advanced

Image acquisition operations are the main processes that an MRI machine executes. Image acquisition components can be listed as image processing, feature enhancements, feature extraction, and error removal. The basic image acquisition module consists of elementary modules to acquire the image and process it. Therefore, the acquired images may be lower resolution than the other levels and the details may not appear very sharp. In the enhanced module, the acquired image is cleaner because noises are reduced and higher resolution. The advanced level image acquisition module provides very detailed and clear images. If auto diagnosis is to be performed, it is necessary to use this module to get better results. Customers can choose the appropriate level for their MRI machine according to their usage reason or budget of hospitals.

Variability dependency: Customers must select one of the image acquisition module packages for image acquisition processes. Thus, there is a mandatory variability dependency between variation points and variants. In addition, an enhanced or advanced module is required to use the auto diagnosis module.

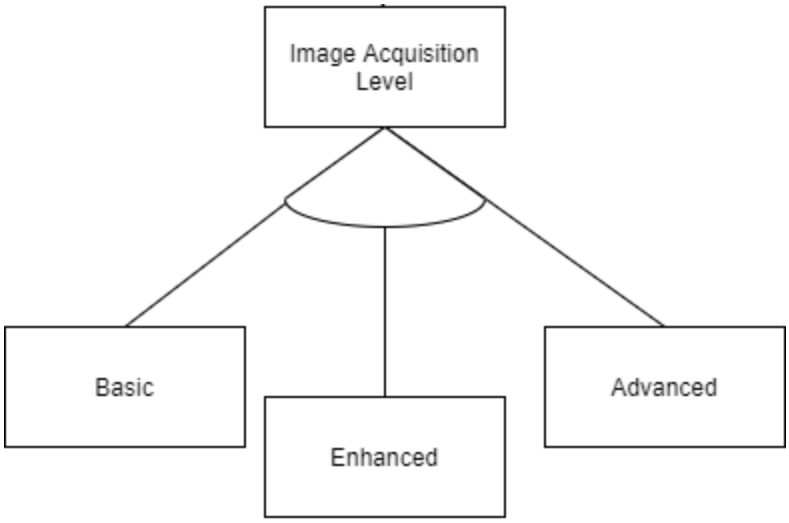


Fig 3.7

Variation point: Diagnosis techniques used on MRI software to diagnose disease
Variants: Image recognition techniques, big data analysis, medicine recommender system

Auto-diagnosis processes used for diagnosing the disease and reporting to the doctor. In this diagnosis process, many machine learning algorithms and approaches are largely used to a more precise diagnosis. The **MRI MR-004** product in our product portfolio will contain an auto-diagnosis module. So many modules can be used for this operation. Therefore, this topic is considered as a variation point and the techniques handled as variants.

Variability dependency: Customers can select zero or more techniques to use for the auto-diagnosis process. Thus, there is an optional variability dependency between variation points (diagnosis techniques used on MRI software to diagnose disease) and variants (different machine learning approaches used for auto-diagnosis). This variability can be handled as external variability. According to the customer's requests, modules can be added to the product.

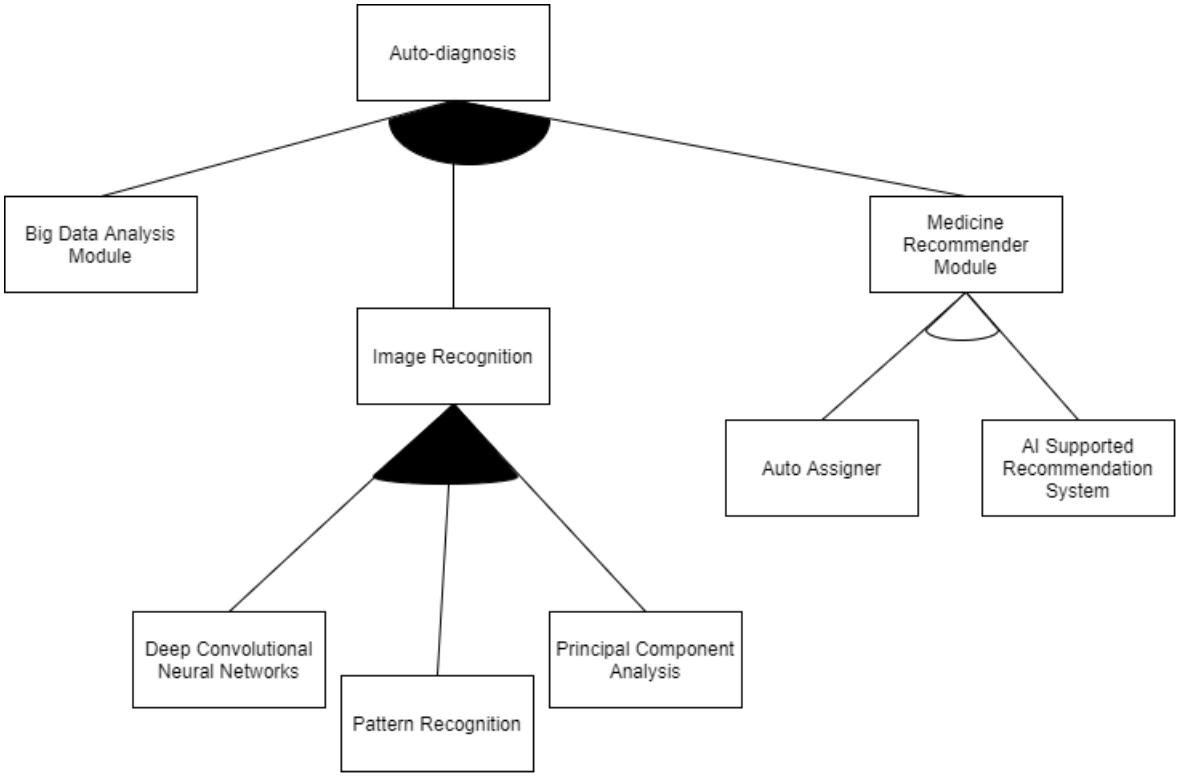

Auto-diagnosis
Big Data Analysis Module
Image Recognition
Medicine Recommender Module
Auto Assigner
AI Supported Recommendation System
Deep Convolutional Neural Networks
Pattern Recognition
Principal Component Analysis

Fig 3.8

Domain Requirements

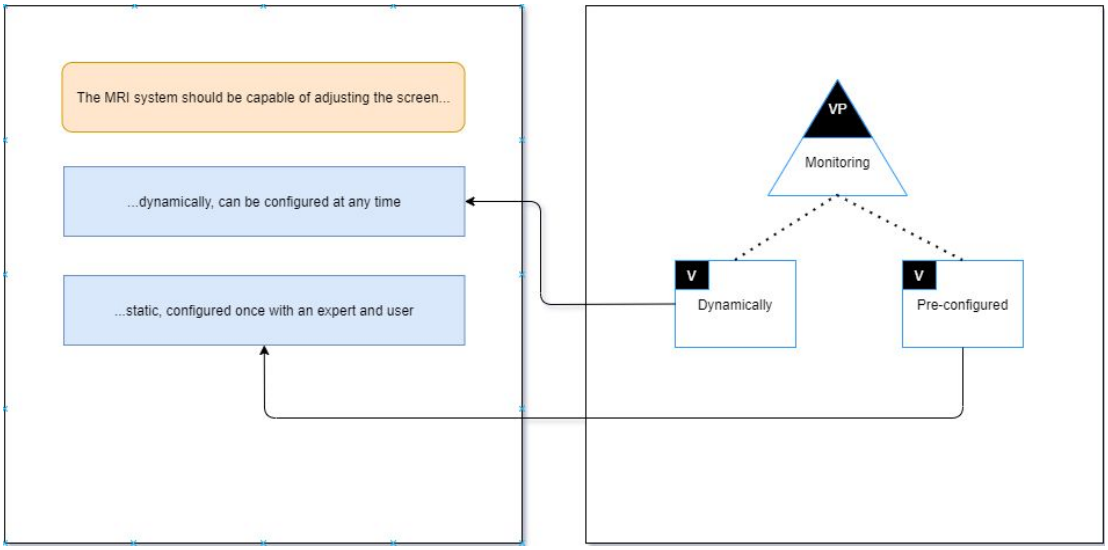
Domain Requirements for Application Modelling and User Interface [Yunus Kerem Türk]

In this section we will try to define domain requirements common to all applications of the software product line as well as variable requirements enabling the creation of different applications.

Customers and users typically prefer talking about concrete sequences of actions that describe system usage rather than talking about abstract models of a system. [7] For this reason, scenarios and schemas are widely used in requirements engineering. Scenarios describe concrete sequence of actions related to the intended feature.

Requirements

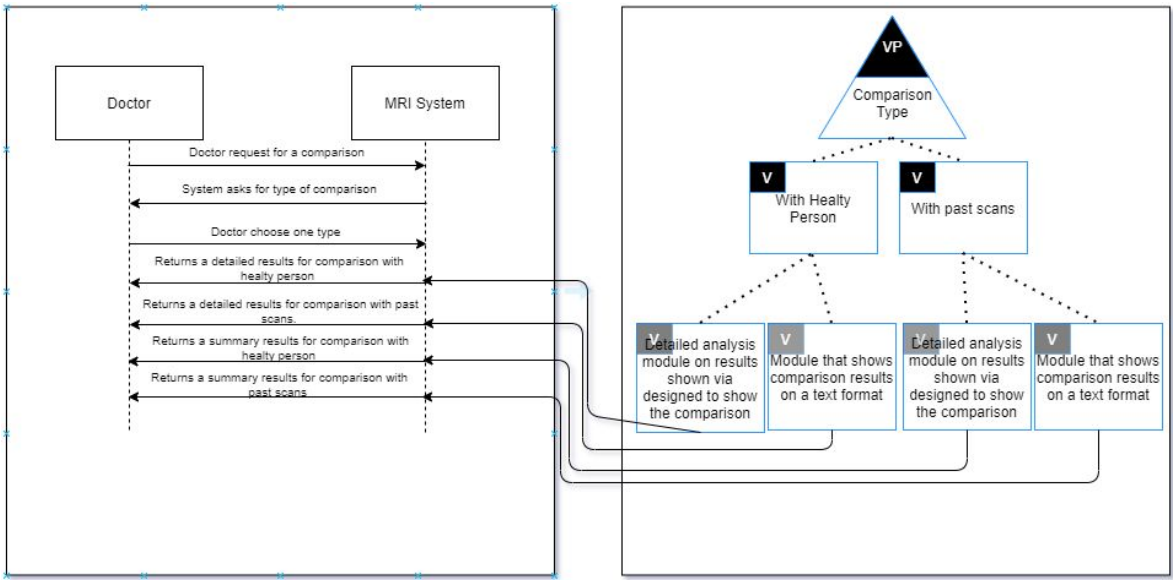
1) MRI machine user interface should provide a configuration for monitoring the scans. This configuration should include adaptability for different numbers of display screens, providing a setting to move the display screen parts to the desired location(to different screens, to different parts of the screen etc.).



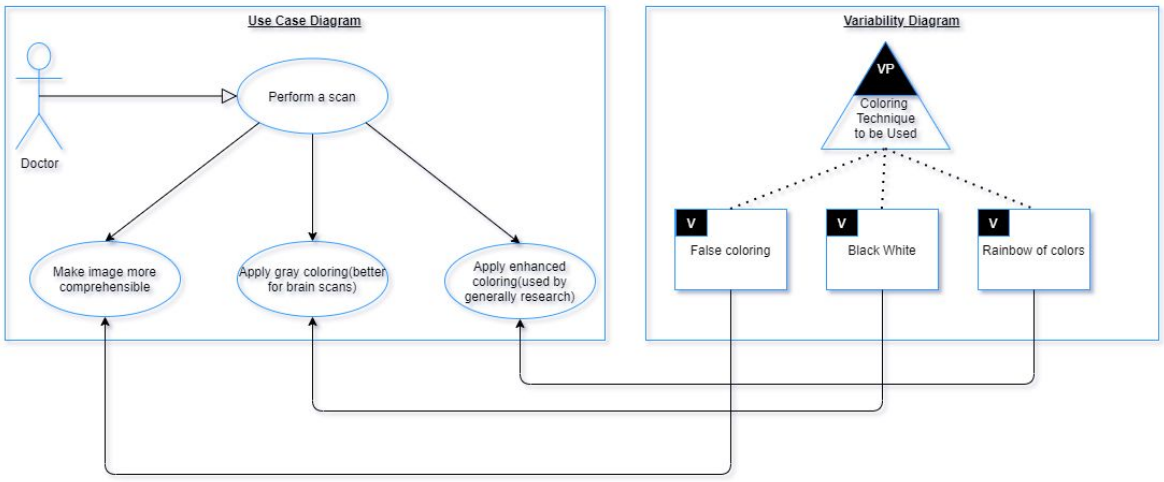
2) To track the process of disease, being able to compare current results with old results is necessary. MRI machine interface should provide a clear interface for comparison scan results with old scans.

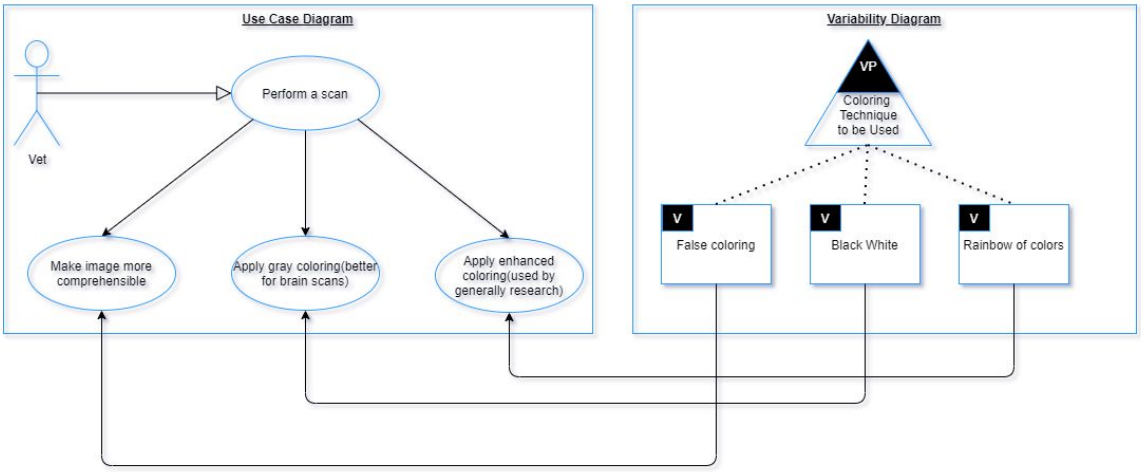
[7] Carroll 2000; Weidenhaupt et al. 1998

3) In medicine, comparing the results of normal and present play a crucial role to diagnose the disease. So, MRI machine interface should provide a clear interface for comparison scan results with a healthy person

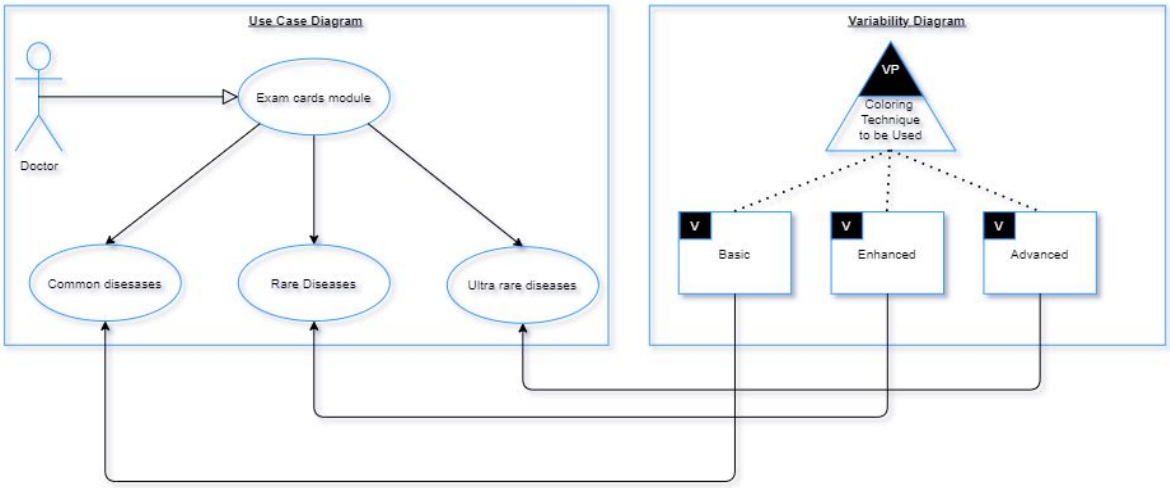


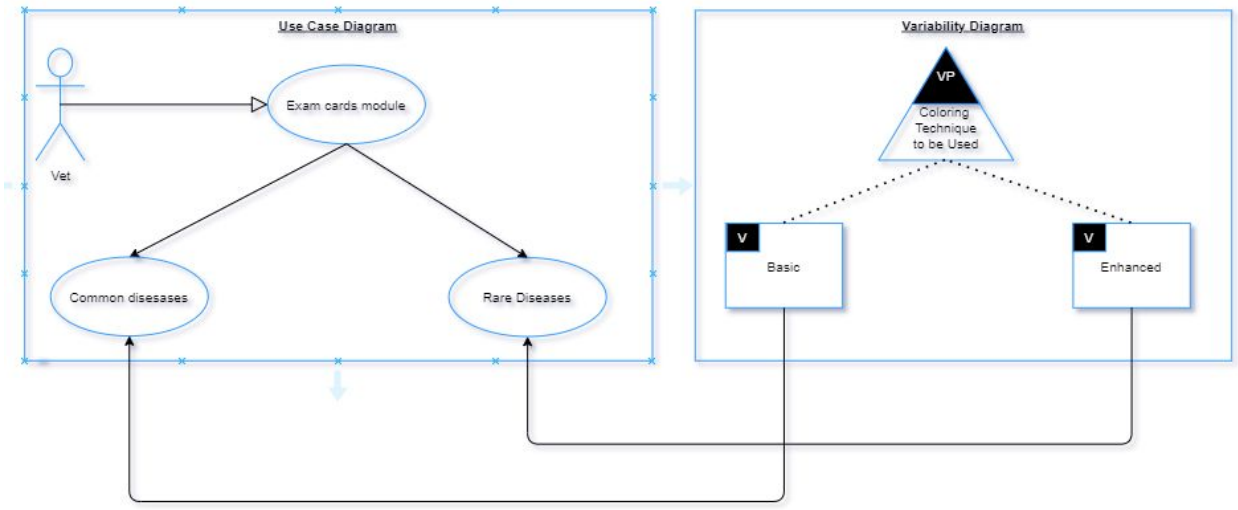
4) Different coloring techniques are the key component to precise diagnosis.Thus, coloring capability for different kinds of coloring techniques is required for both animal MRI machines and human MRI machines.



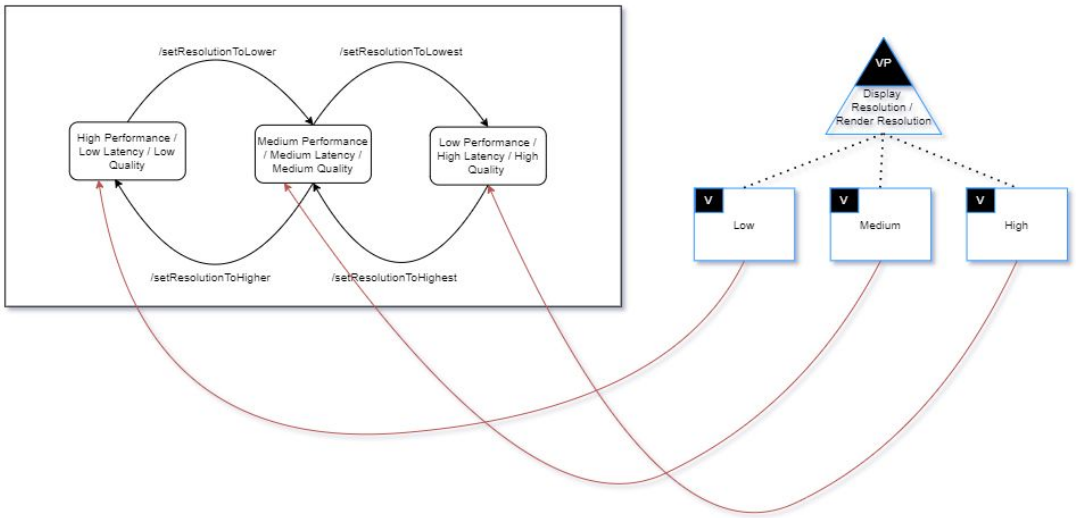


5) MRI machine software should provide an exam cards module that is upgradable and adaptable for both humans and animals.

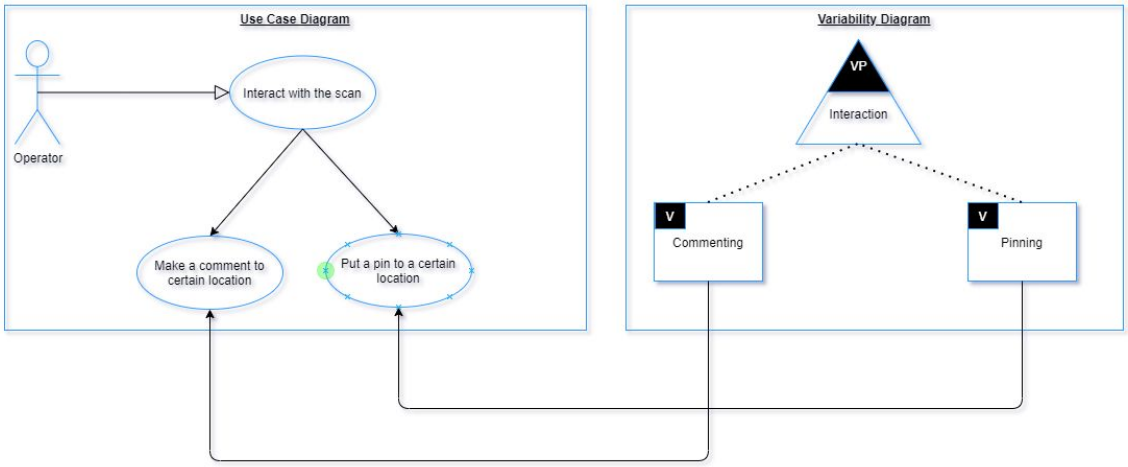




6) MRI machine software should provide a setting for adjusting the rendering and display resolution to the user for better experience.



7) The whole user interface module has to handle different types of MRI machines for displaying results. By keeping this module separate, the system can be improved without touching the rest of the components.
8) How the user interacts with the system is one of the most important things for user interface and experience design. Thus, being able to interact with the system via

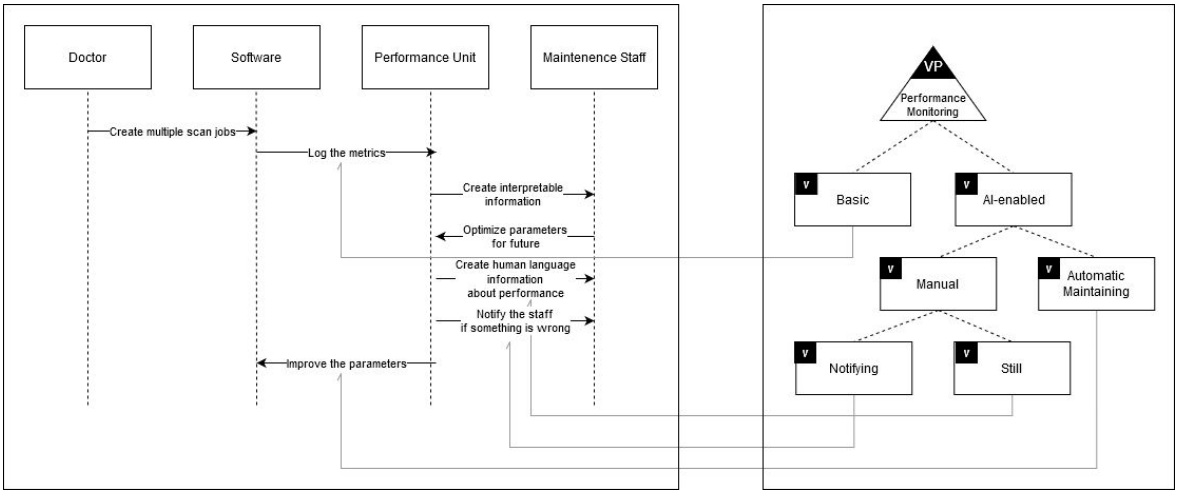


different channels should be provided.

9) Different types of algorithms may be needed for different use cases. Thus a library that provides image analysis techniques should be provided.

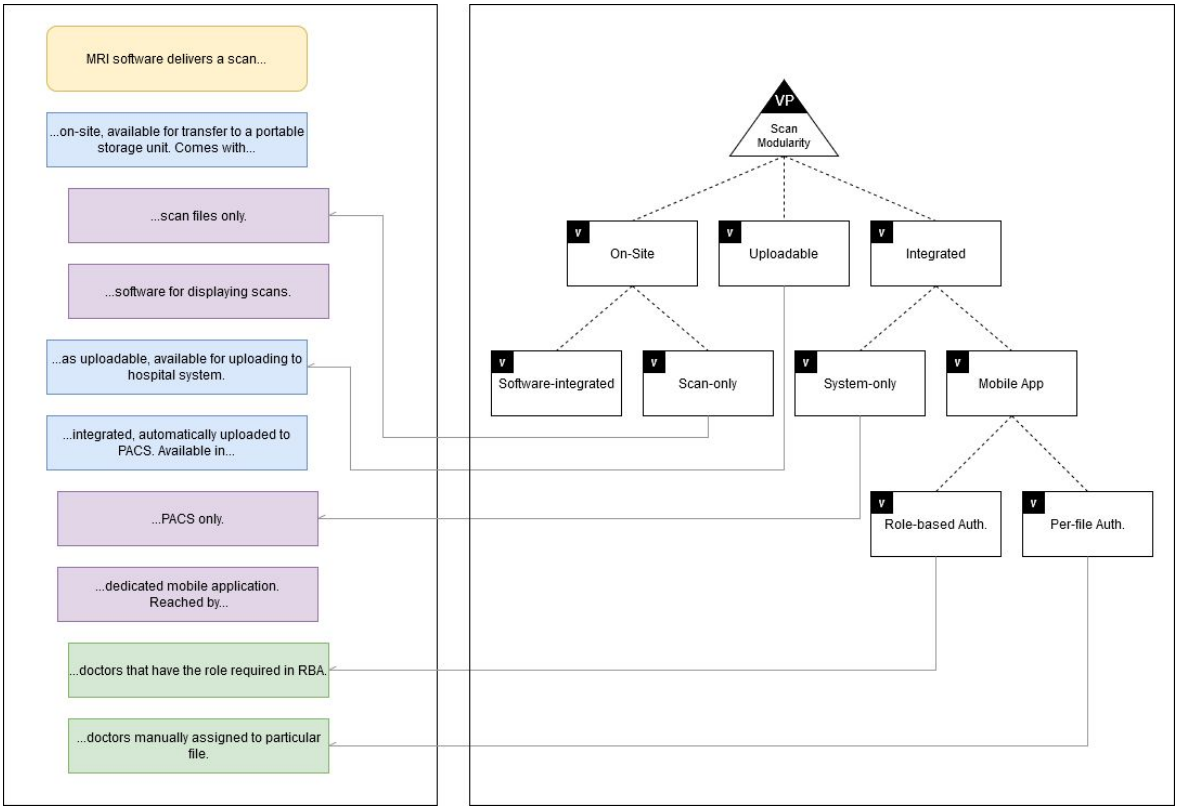
Domain Requirements for Tracking and Control System [Onur Demirezen]

1) MRI system software must generate meaningful performance data that is interpretable by maintainer staff or it must improve its performance automatically.

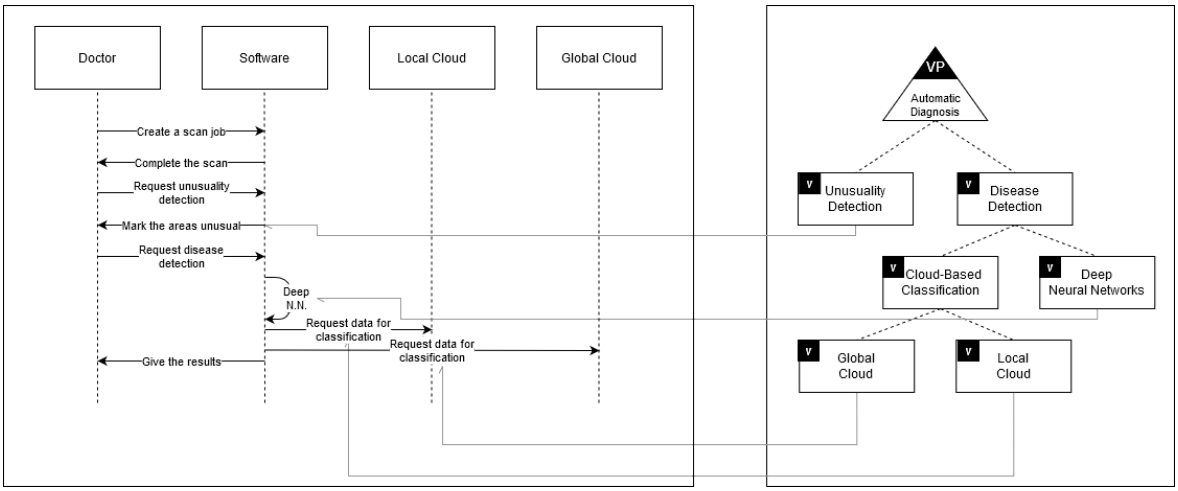


Performance data should be used for efficiency in various operations.

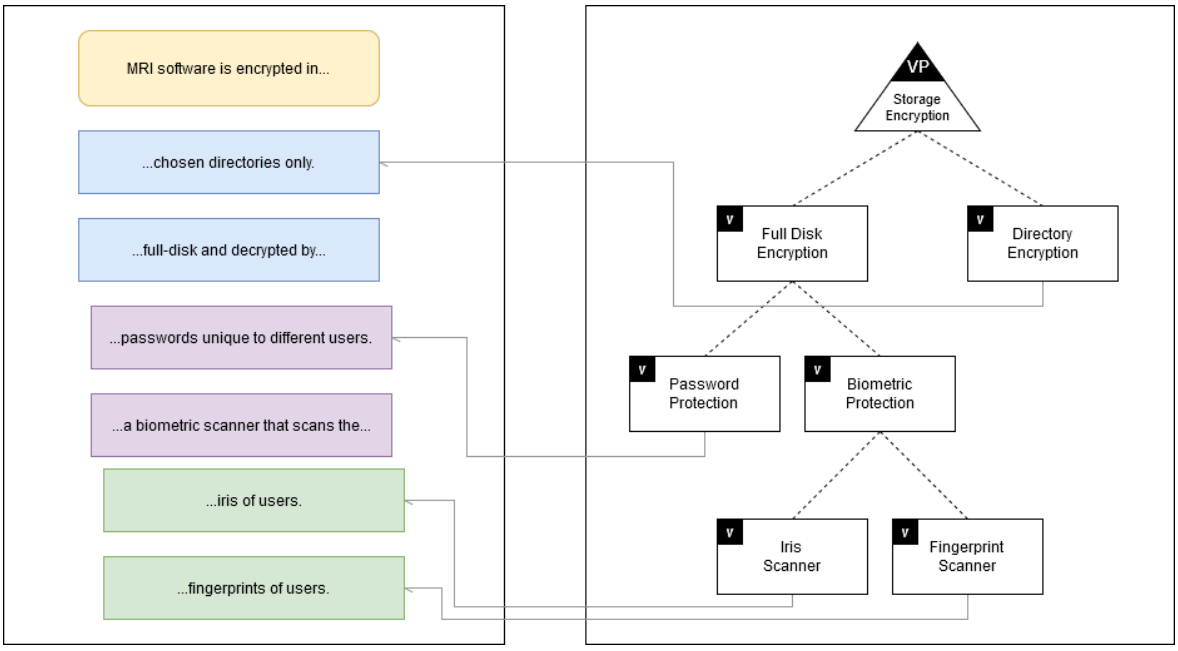
2) MRI software has to make the scans available out-of-machine in a way preferred by customers.



3) MRI software should give automatic diagnosis results efficiently and accurately. Users should be able to choose to auto-diagnose or not. Machine should decide which procedure it will use.

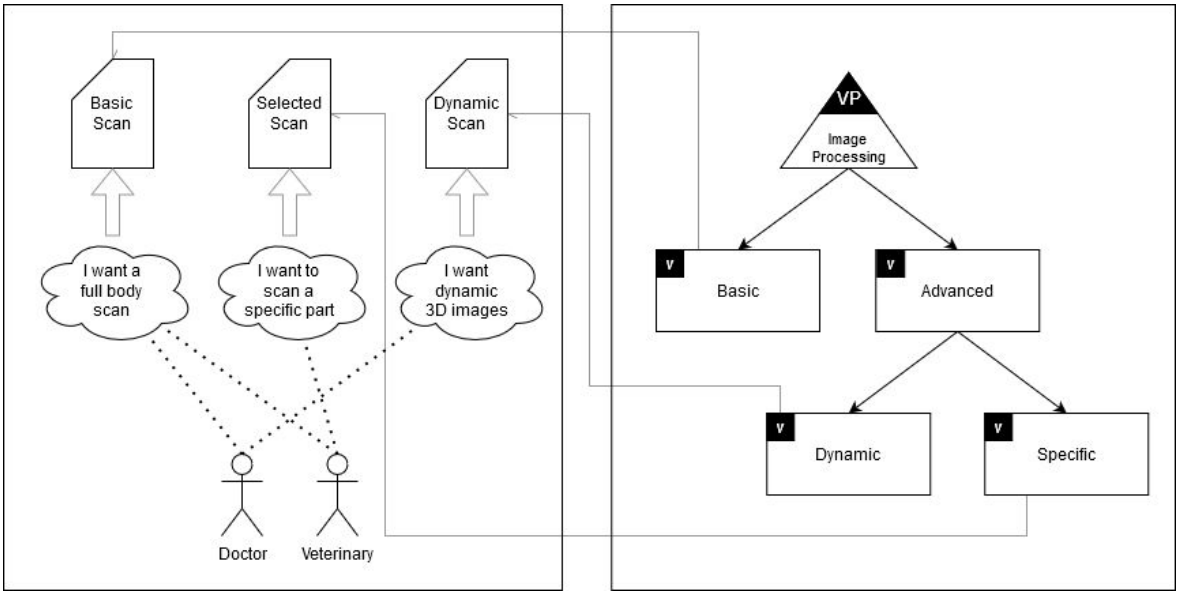


4) MRI software should give options for storage encryption because of the performance impacts. As described in variation point definition, disk encryption has negative impact on performance, hence should not be mandatory.



5) Essential image processing can be done in different ways for certain requirements. If image processing algorithms are optimized for specific use cases, it will have a positive impact on performance. For instance, an image processing algorithm specialized for veterinary applications will scan animals faster than a general-purpose algorithm. Or even an algorithm optimized for brain scans can help

the process of scanning the human brain. Then there is a dynamic scan option that can be considered a 3-D motion picture.

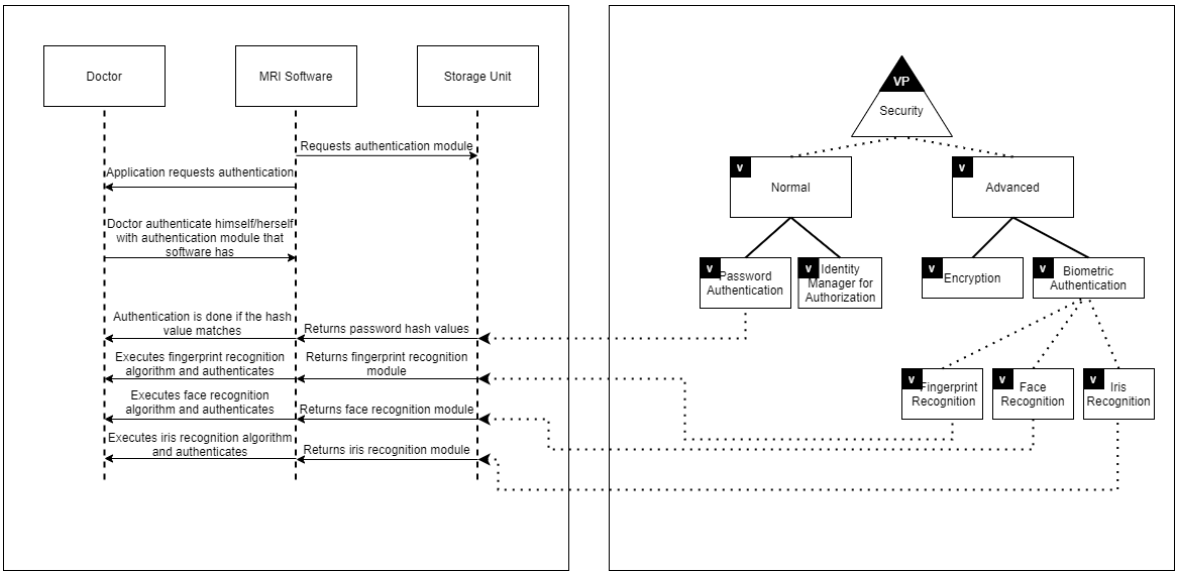


Domain Requirements for System Architecture and Infrastructure [Mehmet Arif Şahin]

In this section, the requirements for the system architecture section of the domain are defined. When defining requirements for the domain, product line variability is considered as well as the common requirements which formed our product line. Therefore, domain requirements must satisfy the variation points and their variants which are determined in the previous section. Scenarios and use cases are used to clearly express the requirements for the usage of MRI software. When documenting requirements, textual requirements, and orthogonal variability models are largely used.

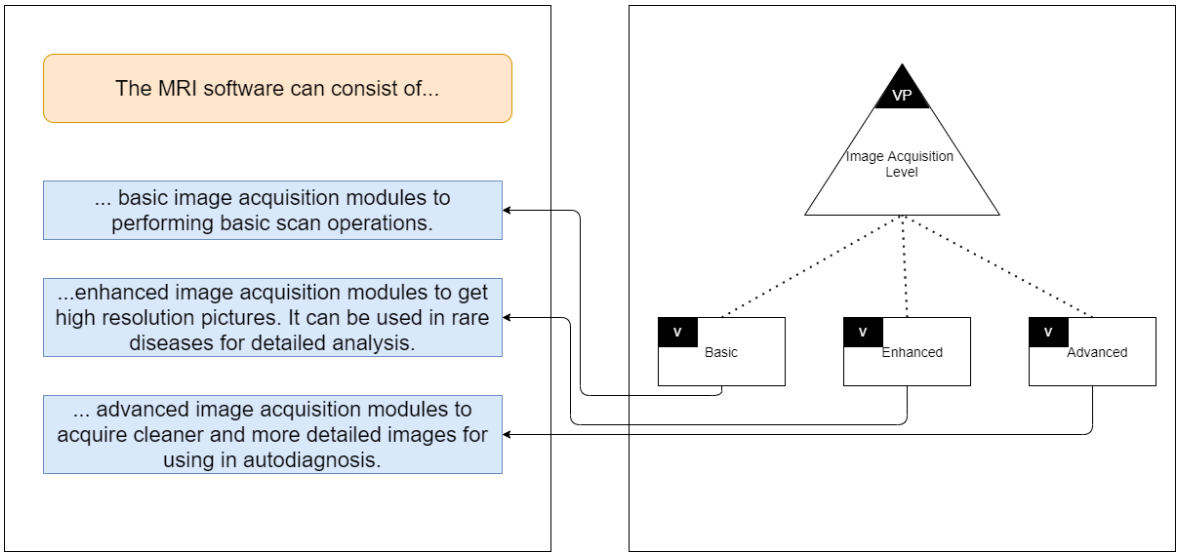
Requirements

1) All products in our product line must authenticate the user for the usage of the software. So, an authentication module must be provided for only authorized people to use the software. This authentication operation can be made in various ways (password authentication, biometric authentication methods). Therefore, different recognition modules can be provided. Different password data are obtained from these different methods, so there will be also different storage options.

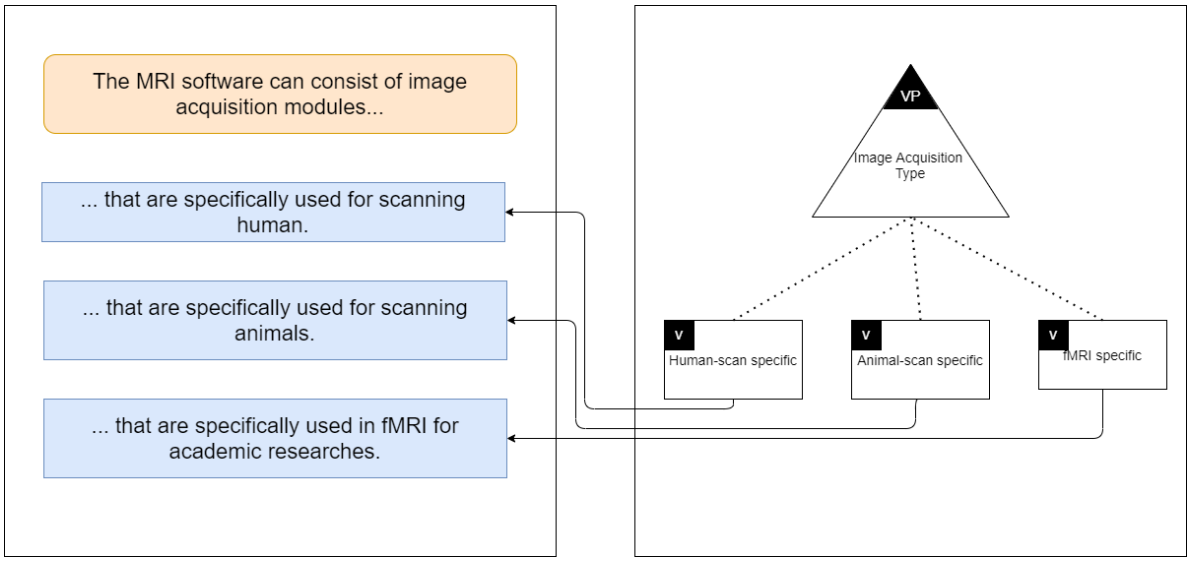


2) Image acquisition is the most fundamental operation that MRI machine software performs. All products in our product line must have image acquisition modules like image processing, feature extraction, error removal. Therefore, the need for image acquisition modules can be considered as a domain requirement. Of course,

more advanced or basic modules can be used according to the usage area or how detailed the image is desired. This situation creates variability in the domain.

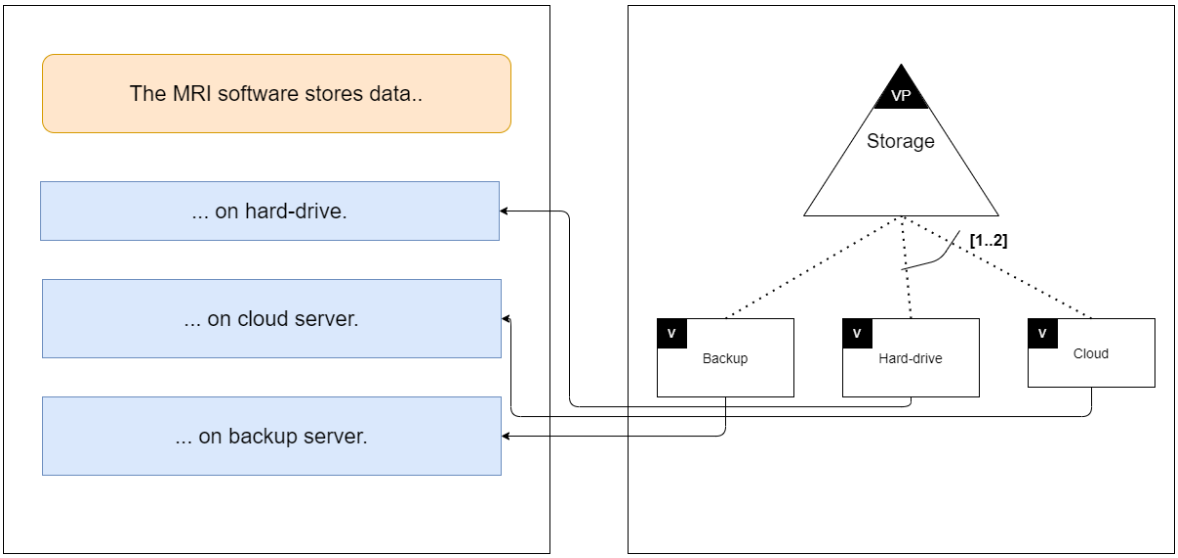


3) As stated in the previous requirement, each product contains image acquisition modules. There are different types of MRI machines that are used for different purposes. In our product line, there are three main different MRI software types which are human-scan MRI, animal-scan MRI, and fMRI. Depending on which MRI type uses our software product, image acquisition modules can be varied.

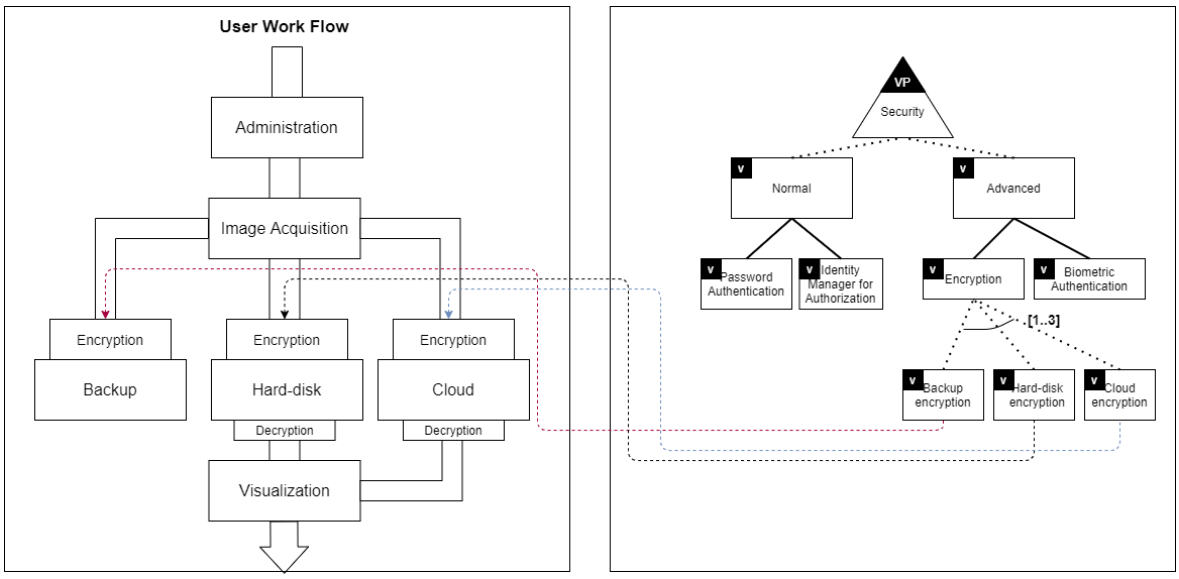


4) Storage is one of the essential requirements for the domain. The software should always use one of the storage options to store the data and access the desired data.

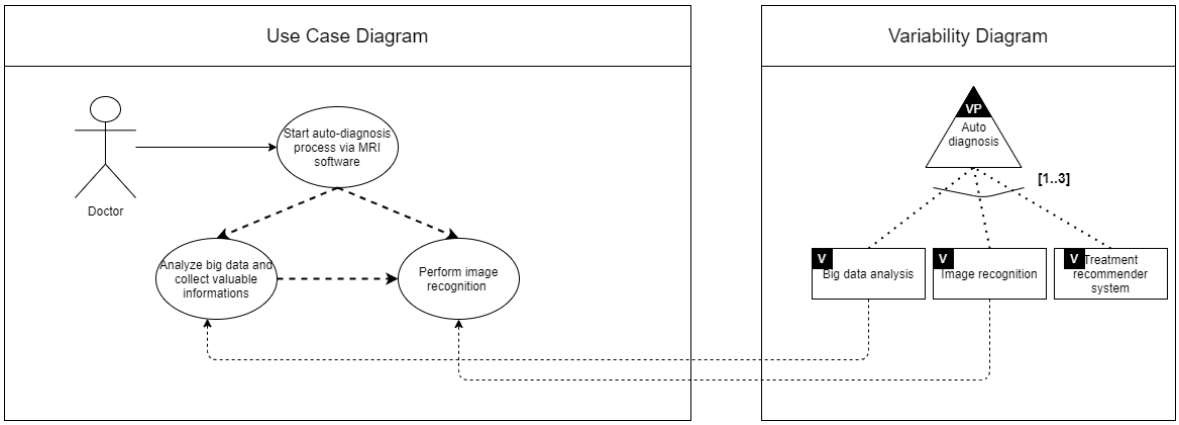
Storage can be provided with a hard-drive or cloud-based. Backups can also be made.



5) Because the patient's data is confidential, customers can add an encryption module to their desired storage unit for more security.



6) Auto-diagnosis modules are required for the domain because they are used for many products to automatically diagnose diseases. Therefore, reusable auto diagnosis modules must be in the domain.



Product Requirements

Product Requirements for Application Modeling and User Interface [Yunus Kerem Türk]

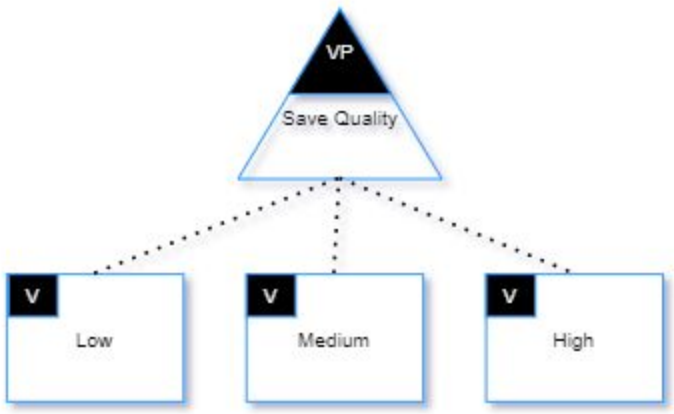
In this section, we will try to define application requirements for three different applications selected from the product portfolio. Application requirements are requirements that completely specify a particular product line application. The three different products for this report will be MRI MR-002, MRI MR-V and MRI MR-003.

MRI MR-002

In this product different modules of product line are used which are:

1. Pre-configured monitor setting
2. Enhanced exam cards module
3. Display and Rendering resolution settings
4. Gray-white and False coloring techniques

- Additional to domain requirements stakeholder wants to upload the scans to the cloud at desired display quality. While taking render the desired quality was chosen, but when uploading the file it can be changed to higher settings.



New variation point for this product

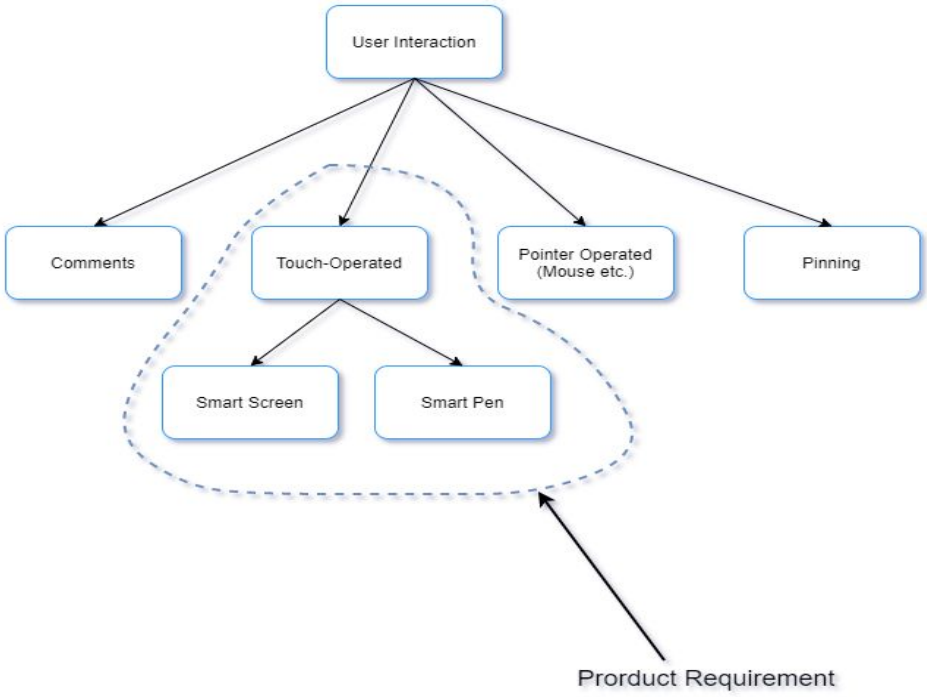
- This product needs to be integrated with PACS (Picture Archive and Communication System). To achieve this integration a user interface module for displaying PACS data is required.

MRI MR-003

This product's main purpose is to be used in academic research, so some additional requirements particular to this product are needed. Before that we define the requirements taken from the product line.

1. Advanced exam cards module
2. Dynamic monitor configuration
3. All types of coloring techniques
4. All types of comparisons

- Additional to these requirements some application specific requirements are needed too. Normally our application is interacted by a mouse or some kind of pointer. But stakeholder requires that touch operated analysis, and using a smart pen should be enabled for teaching purposes. So this adds a new variant to our defined variation points.



MRI MR-V:

The last product I choose is a MRI machine designed for veterinarian applications. Although machines core and hardware features are similar, many changes are applied on software. Such as, specialized exam cards data for veterinary applications and specialized coloring features are needed which were defined in the product line.

- Pre-configured or dynamic monitor configuration
- Exam cards module specialized for veterinary applications
- Display and Rendering resolution settings
- Coloring features

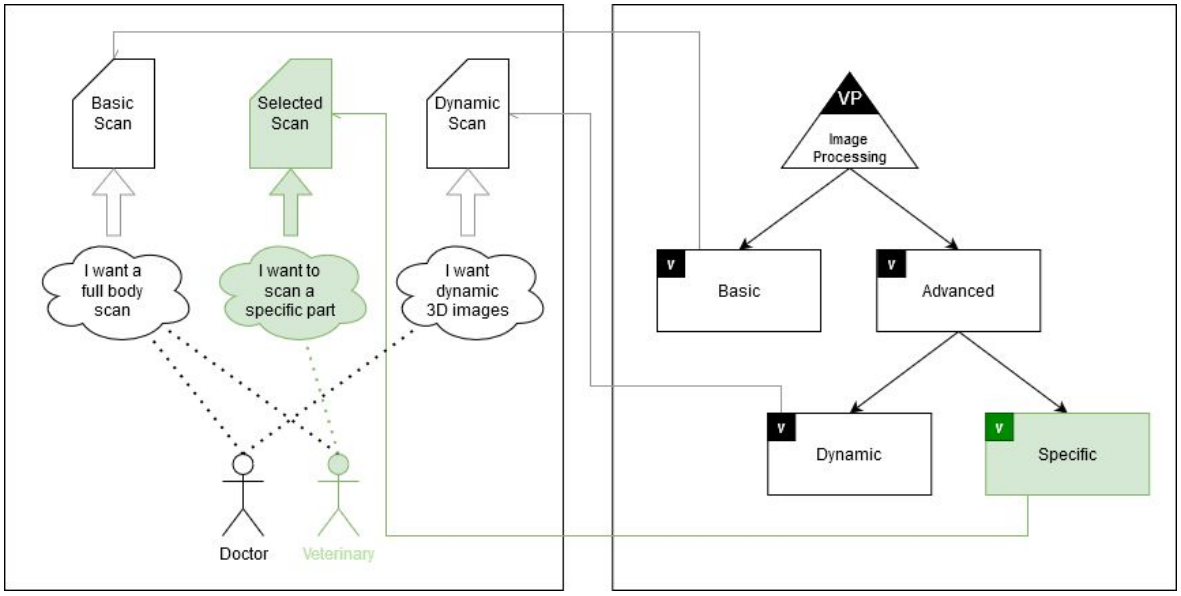
- Veterinary applications for MRI machines are different from human applications, because there is more than one species and for one species there are many types and all of them have different characteristics. So we reminded the stakeholder that prior knowledge and information should be presented to the user via interface for a better scan.

Product Requirements for Tracking and Control System [Onur Demirezen]

MRI MR-V2

This product is designed for advanced veterinary applications, like veterinary faculties that give education and treat various types of animals. The requirements from the Tracking and Control Domain are as follows:

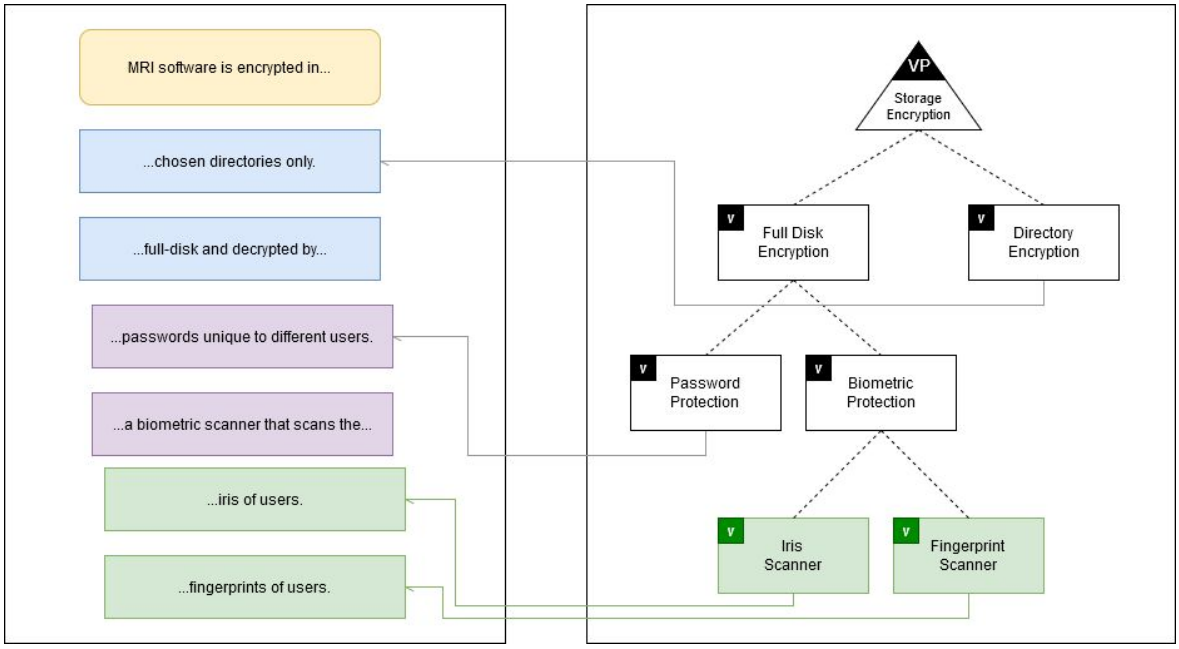
1) Specialized image processing modules for veterinary applications
2) Basic performance monitoring module
3) On-site and integrated scans
4) Auto-diagnosis using neural networks



MRI MR-004

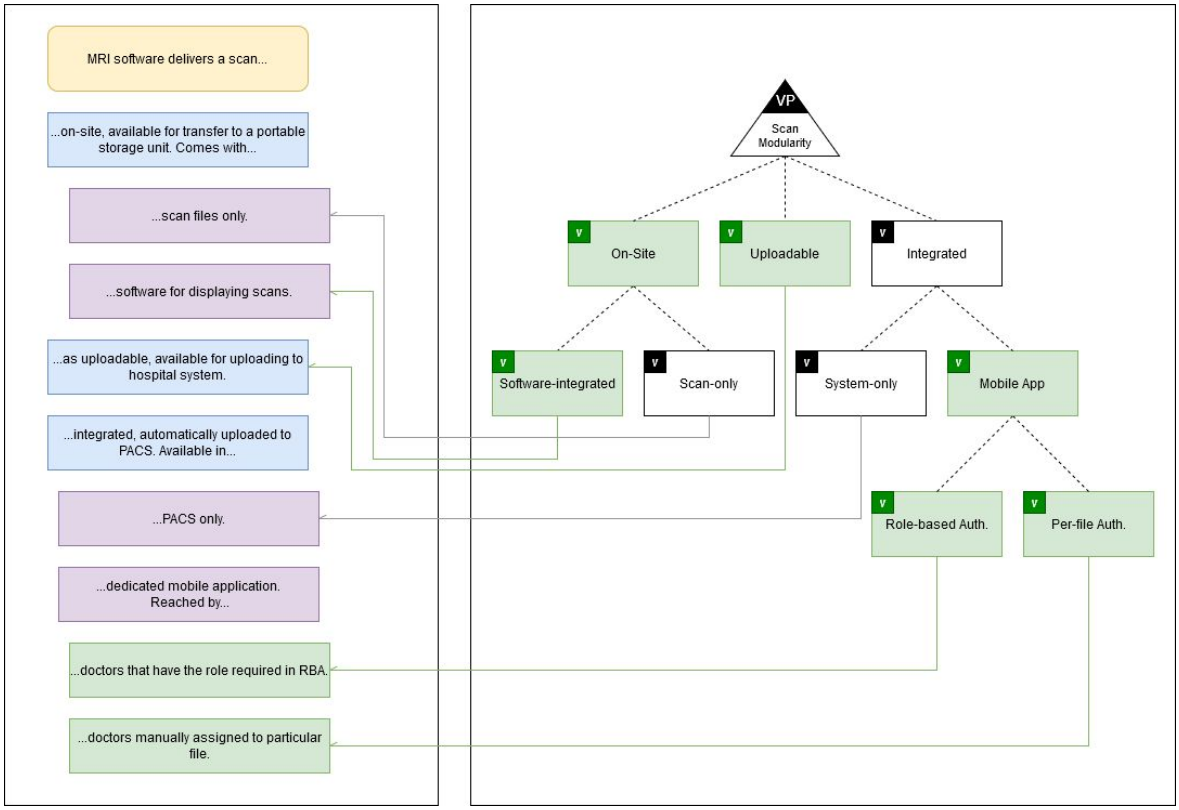
This product is mainly designed for research purposes. fMRI system is used for dynamic scans and it is a very costly process.

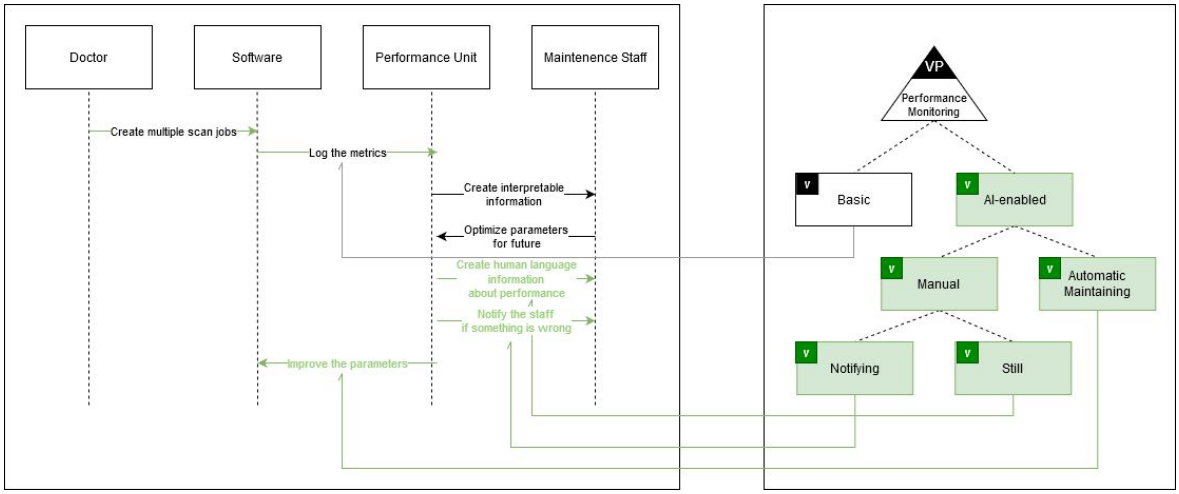
1) Advanced performance monitoring module
2) On-site and uploadable scans
3) Storage encryption with biometric authentication
4) Advanced image processing with dynamic processing



MRI MR-005

1) Advanced image processing with support for specific body parts
2) Uploadable and integrated scan modularity with mobile application support
3) Performance monitoring module with AI enabled
4) Automatic diagnosis with global/local classification


MRI software delivers a scan...
...on-site, available for transfer to a portable storage unit. Comes with...
...scan files only.
...software for displaying scans.
...as uploadable, available for uploading to hospital system.
...integrated, automatically uploaded to PACS. Available in...
...PACS only.
...dedicated mobile application. Reached by...
...doctors that have the role required in RBA.
...doctors manually assigned to particular file.
VP
Scan Modularity
On-Site
Uploadable
Integrated
Software-integrated
Scan-only
System-only
Mobile App
Role-based Auth.
Per-file Auth.


Doctor
Software
Performance Unit
Maintenence Staff
Create multiple scan jobs
Log the metrics
Create interpretable information
Optimize parameters for future
Create human language information about performance
Notify the staff if something is wrong
Improve the parameters
VP
Performance Monitoring
Basic
AI-enabled
Manual
Automatic Maintaining
Notifying
Still

Product Requirements for System Architecture and Infrastructure [Mehmet Arif Şahin]

In this section, the application requirements of 3 products in our product portfolio are listed. Every one of them has specific requirements as well as the common requirements coming from the domain. Accordingly, the requirements from the domain and product-specific requirements of MRI MR-V2, MRI MR-SAFE, and MRI MR-005 products are listed below.

MRI MR-V2:

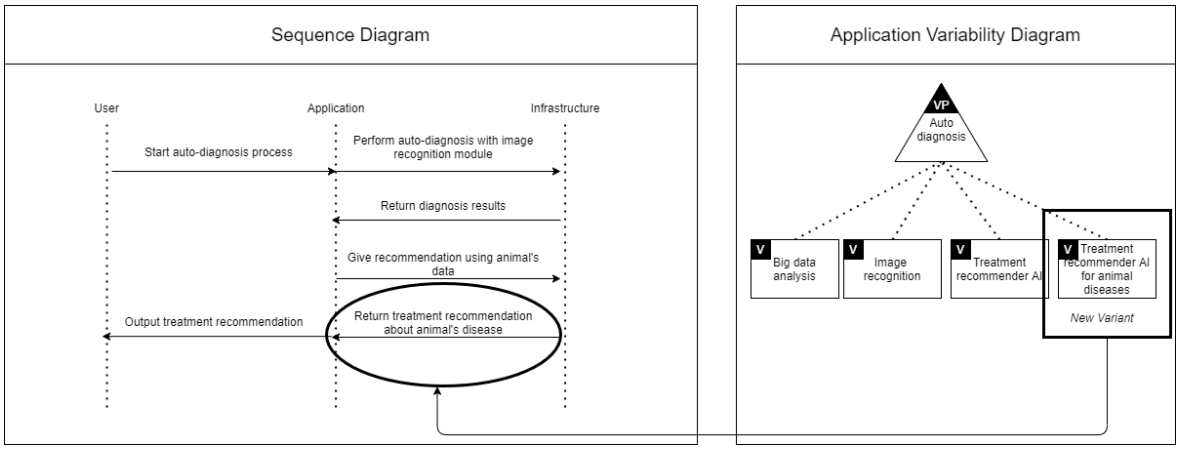
In this advanced veterinary MRI software, the requirements from the domain for the system architecture and infrastructure layer are listed as follows:

1) Veterinary specific image acquisition modules
2) Enhanced level image acquisition modules
3) Password protected authentication
4) Hard-drive and cloud(integration with veterinary PACS) storage
5) Auto diagnosis with image recognition

In addition to these, this product has a specific feature for the system architecture and infrastructure layer which is

6) The treatment recommendation system for diagnosed animal disease using specialized AI.

The AI system that is trained for human diseases can be reusable, so it is kept in the domain. However, the AI system to be created for the purpose of giving recommendations about the treatment of animal diseases is product specific. The application variability diagram can be expressed as in the figure below using the orthogonal variability model.



MRI MR-SAFE:

In this advanced security level MRI software, the requirements from the domain for the system architecture and infrastructure layer are listed as follows:

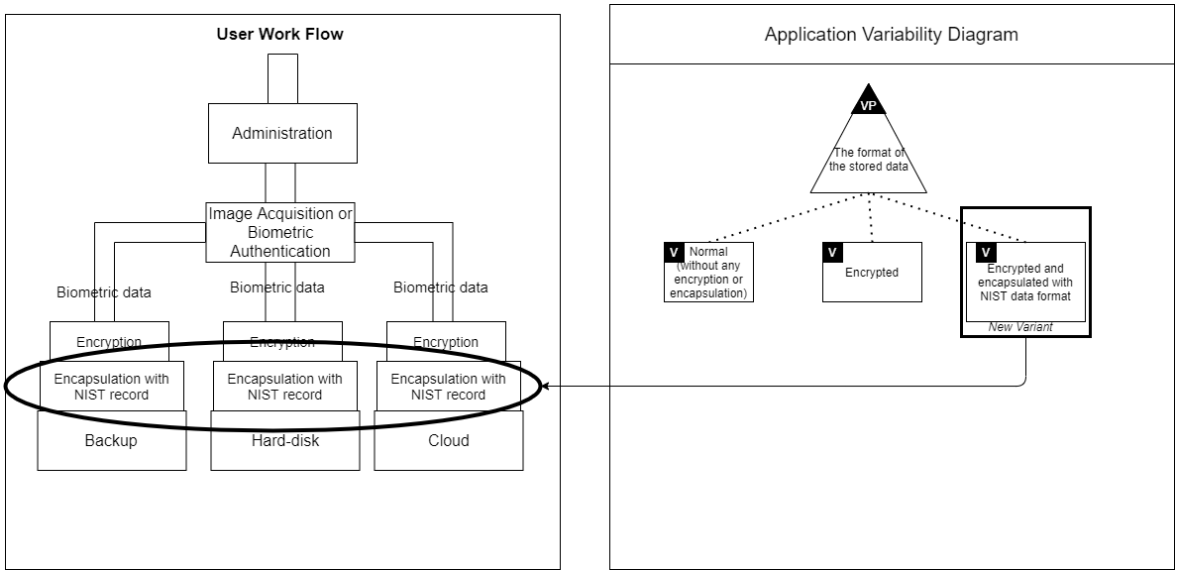
1) Human-scan image acquisition modules
2) Basic level image acquisition modules
3) Biometric authentication (It can be fingerprint recognition, iris recognition, or face recognition. Customers can select one of the authentication methods in the installation phase.)
4) Hard-drive or cloud storage
5) Storage encryption

In addition to these, this product has a specific feature for the system architecture and infrastructure layer which is,

6) The use of the NIST standard for the transportation and storage of biometric data is a special requirement for this product.

Doctors' biometric data which can be a fingerprint, iris, or face data that is used for authentication and patients' data are encapsulated into a record that is created according to the NIST data format for the interchange of biometric information[8]. Data must be in this format when it is stored or sent through the internet.

[8] "American National Standard ANSI/NIST-ITL 1-2007, Data ...." 27 Nis. 2007, https://www.nist.gov/document/approved-std-20070427-2pdf.



MRI MR-005:

In this mobile app supported MRI software, the requirements from the domain for the system architecture and infrastructure layer are listed as follows:

1) Human-scan image acquisition modules
2) Enhanced level image acquisition modules for enabling enhanced coloring abilities
3) Password authentication
4) Hard-drive and cloud storage because of the integration with PACS

In addition to these, this product has a specific feature for the system architecture and infrastructure layer which is,

5) Back-end server for mobile application

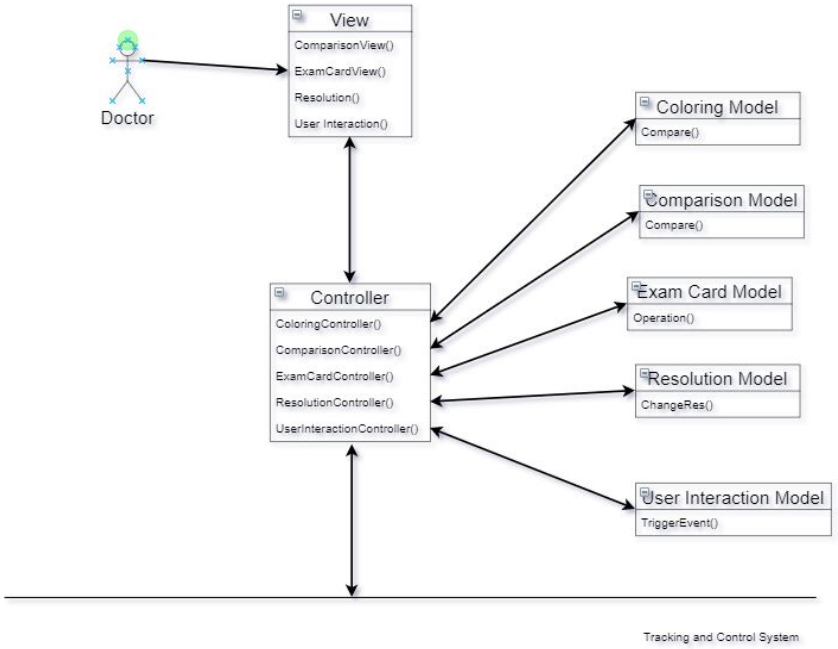
Domain Architecture

Domain Architecture for Application Modelling and User Interface [Yunus Kerem Türk]

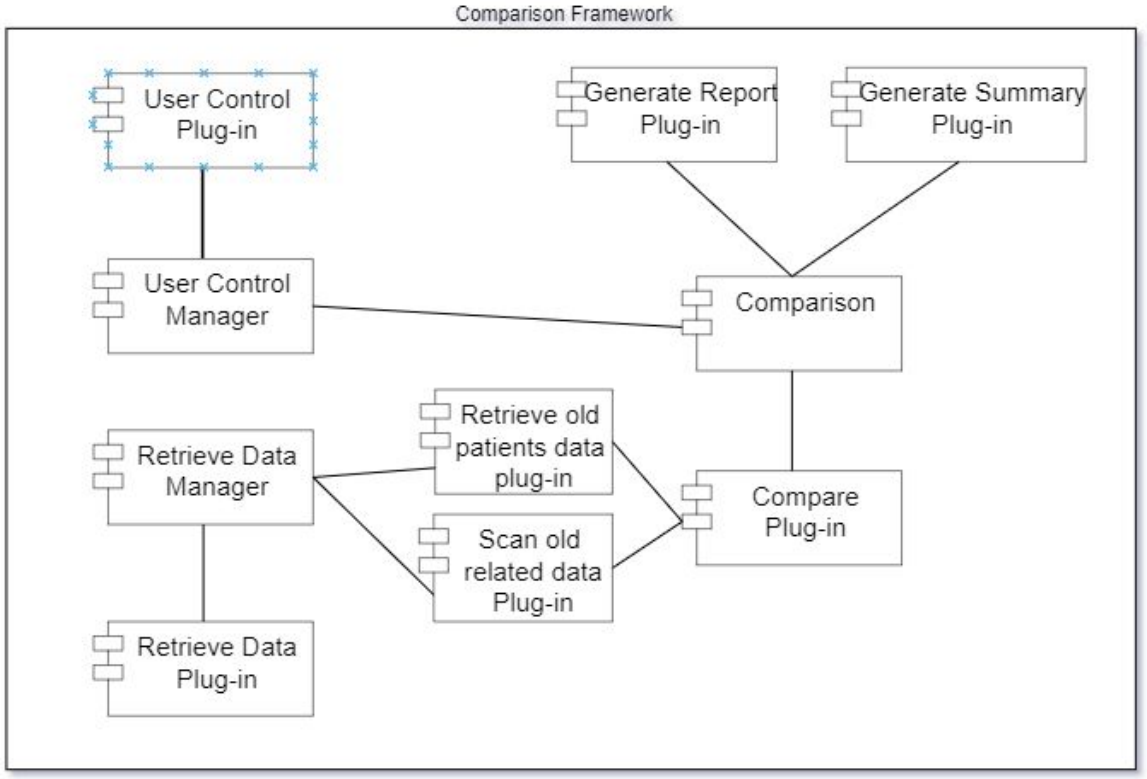
In this part of the report the main goal is to produce reference architecture which defines the main software structure and the texture of the application modeling and user interface. While producing the reference architecture flexibility, modularity, evolvability should be taken into account.

In the given report (DSA Report) MVC is chosen as an architecture. Because, it separates the information displayed to the user from computerized data. The design pattern used for the user interface will be adapted for the given task, meaning that the controller component will send commands to the Tracking and Control Layer, the latter replacing the model component. The Controller component will send requests to the Tracking and Control Layer using the command pattern.

The reference architecture typically consists of a large number of components, named component frameworks, that can be connected through interfaces. Component frameworks restrict the number of component configurations. I will define the general architecture first to give a better understanding of what the frameworks are, and then move on to the component frameworks in detail.

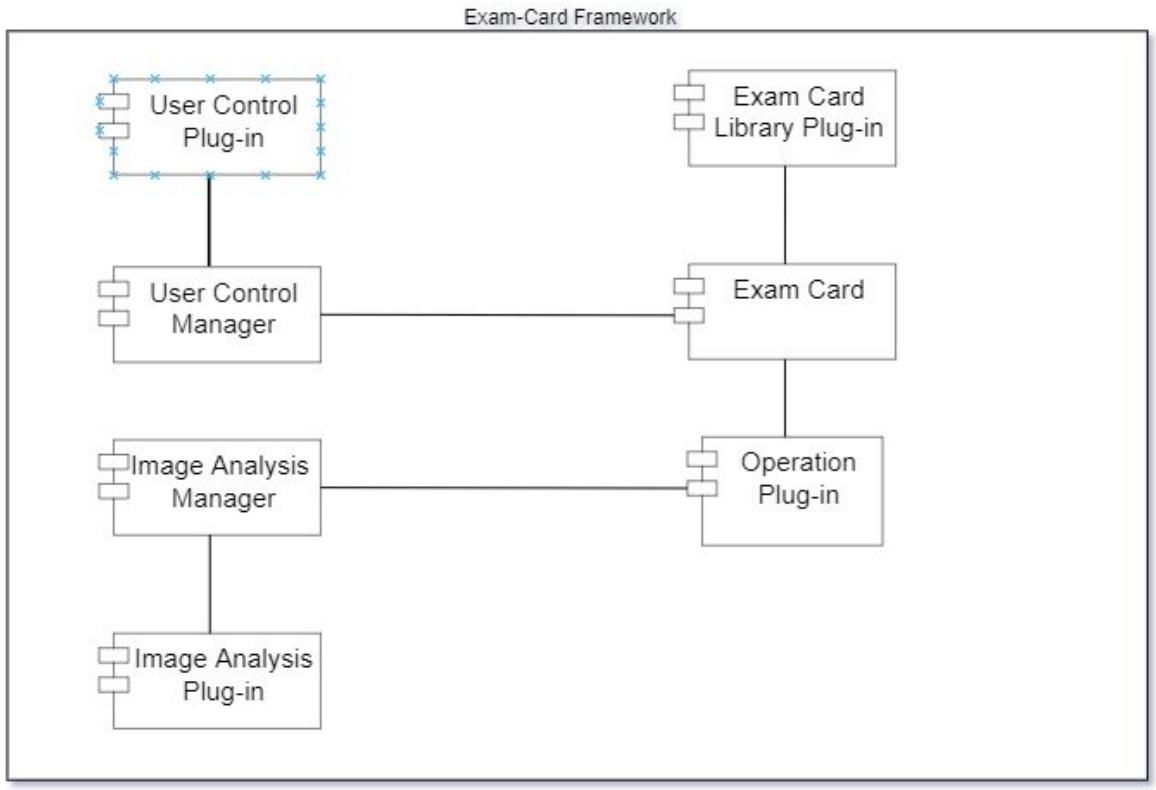


I will define three different frameworks that basically correspond to the features defined in domain variability model and domain requirements. Other than frameworks, the image analysis component is explained in detail because it may help the reader to understand the features it contains.



Comparison framework is used to compare current scan with the past scan result of the patient and healthy person's scan result. This component is used for three of our products which are MRI MR-002, MRI MR-003, MRI MR-004, MRI MR-005. But for all these products components are adjusted according to features of the product.

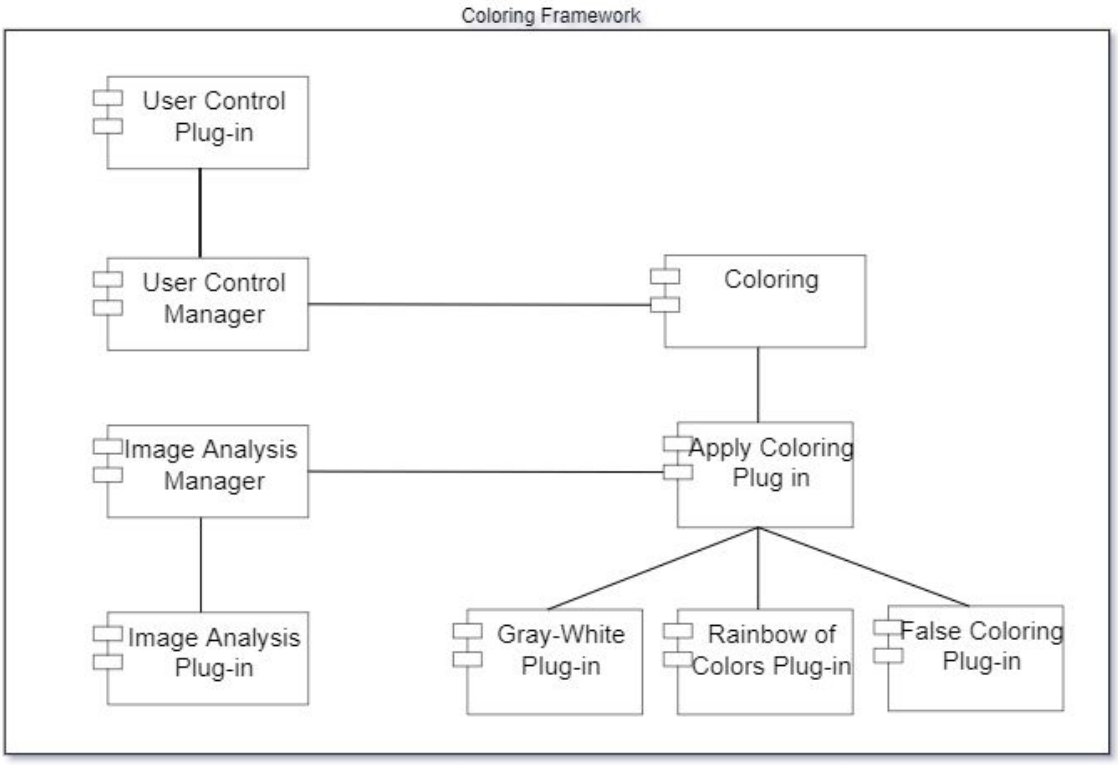
Comparison framework consists of two main parts: 1)Compare 2)Generate report. To be able to compare two different results there should be a sub-component that should retrieve the requested data (old scan etc.), which corresponds to the first main part. And there should be a sub-component generating results.



Exam card implementation is one of the key features of our product. While designing the exam card component I refer to the DSA report and observe how they put the exam card model to the singular system or architecture.

The exam card component has a library that includes defined procedures for a scan, and an operation plug-in which actually does the job. Since both image analysis and exam card libraries can be extended to use cases, it perfectly fits the software product line.

There are different coloring techniques to use for different cases. So it was sensible to support this feature with “use what you want” understanding. Just like the exam card component, the coloring component uses the power of image analysis plug-in too.



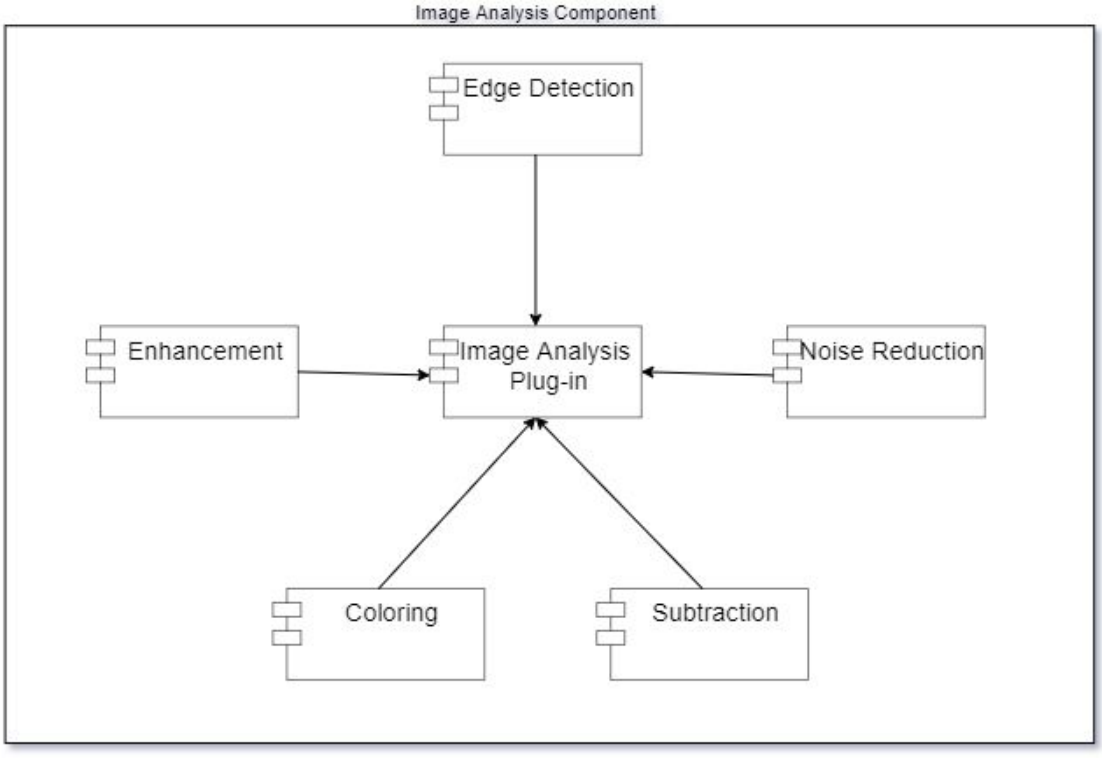
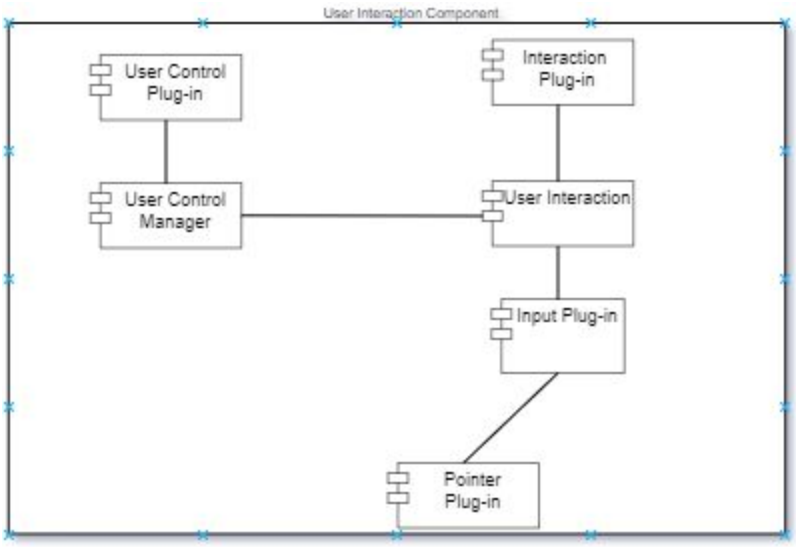
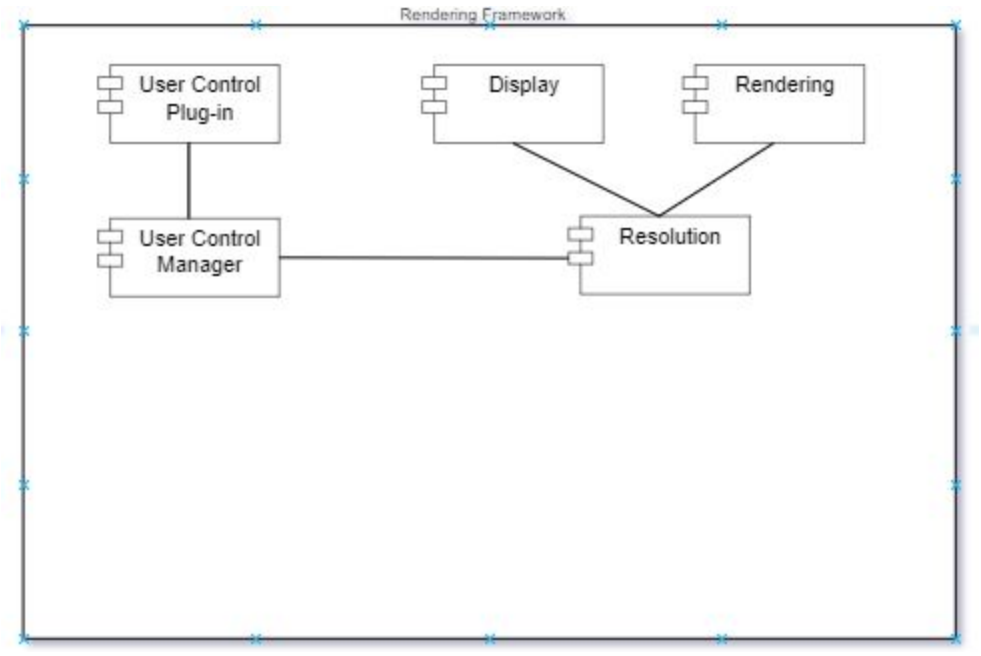


Image analysis is a component that includes different algorithms and techniques to be used in a scan. It is by definition an extendible component by adding new libraries or new algorithms.



How users interact with the whole system is important just like any other system. In our case we provide an interaction plug-in which can be extended over time, but now has two branches which are 1-)Commenting, 2-)Pinning.

Input plug-in is responsible for taking inputs correctly, and delivering it to the process layer.(Button pressed etc.)

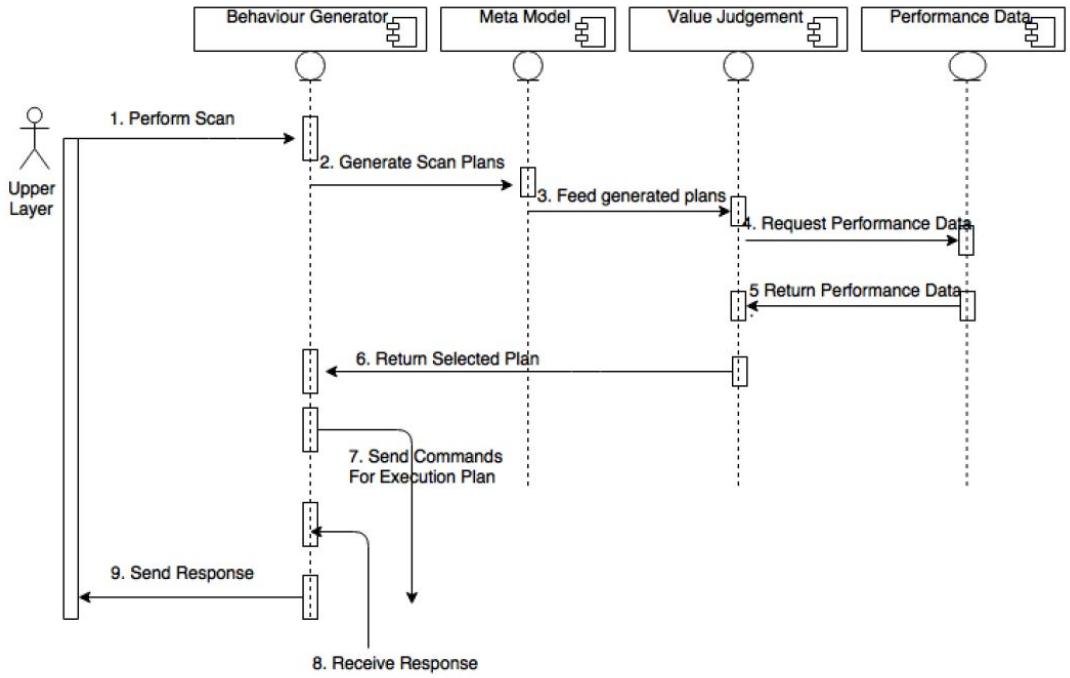


Resolution component has two main branches named, display and rendering each is responsible to adjust the rendered scan to desired resolution.

Domain Architecture for Tracking and Control Systems [Onur Demirezen]

A reference architecture model for tracking and control domain is described in this section. Since the report gathered from the architecture team states that component based architecture is used in this layer, we will use component based architecture with vertical decomposition in that manner. That enables all the products in the product line to be compatible with what the architecture team has to offer.

Below is the interaction diagram for tracking and control control system, created by the architecture team.

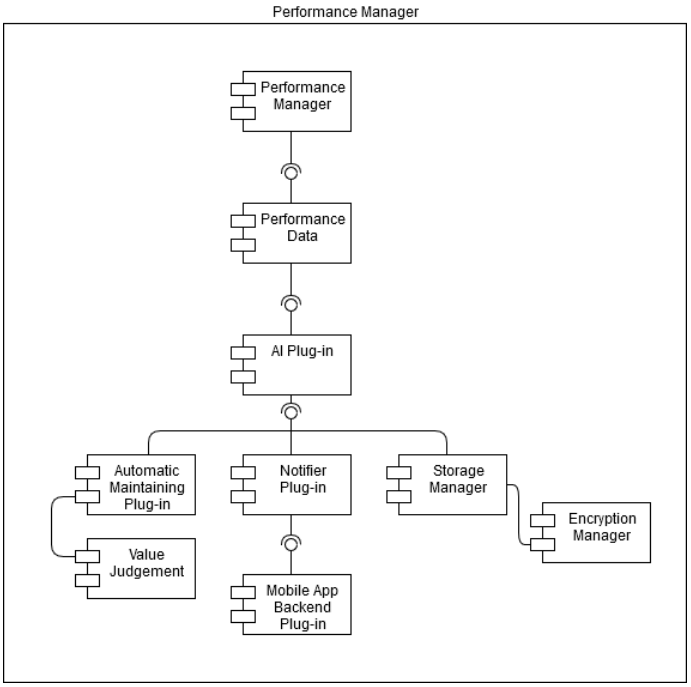


There are 4 main components in this architecture. First is the *Behaviour Generator* that handles different types of scans, second is the *Meta Model* component that defines the scans that handled by Behaviour Generator, third is the *Value Judgement* component that analyses scan model and performance data and last is the *Performance Data* component that is responsible for collecting performance data. According to the diagram, when a user performs a scan, the behaviour generator handles the request and sends the scan plans to the meta model component. Meta model feeds the plans to value judgement, then requests performance data about the model from the performance data component. After the value judgement interprets data, returns the model at hand to the behaviour generator for execution of the plan.

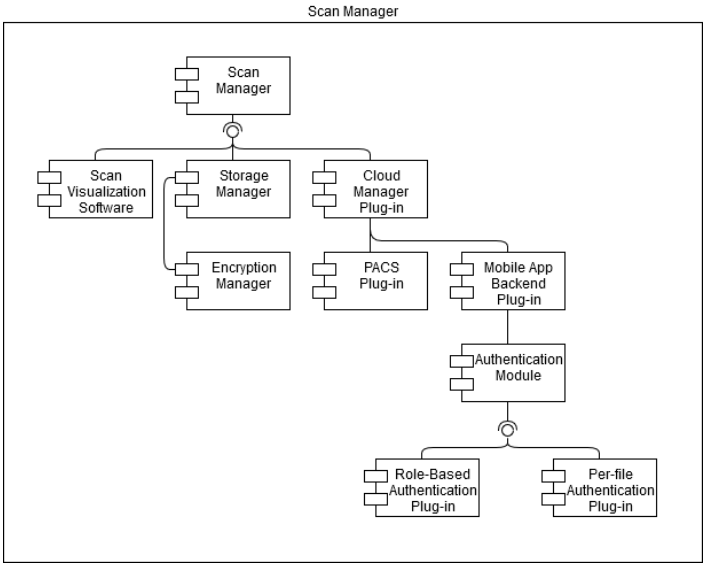
Variabilities come when different types of performance analysis are needed. Since the main purpose of tracking and control systems is improving performance, it should not be

stuck in analyzing scan models. Also, performance analysis has been improved in some variabilities by enabling AI etc.

Let's first look at how we handled the Performance Monitoring variation point.



Performance manager collects the performance data through the Performance Data module. Then sends the data to the AI plug-in if applicable. AI plug-in decides to notify staff, auto-maintain or just store the interpreted results. If notification is selected, it will be sent to the backend plug-in. Automatic Maintaining Plug-in sends the results to Value Judgement if applicable.



Scan modularity is one of the key variation points. Above is the architecture diagram for the scan manager module. According to needs, the module chooses to do actions. Scan modularity requires interaction of many different parts. Since every data stored in disk goes to encryption manager, let us look into the architecture of this component,

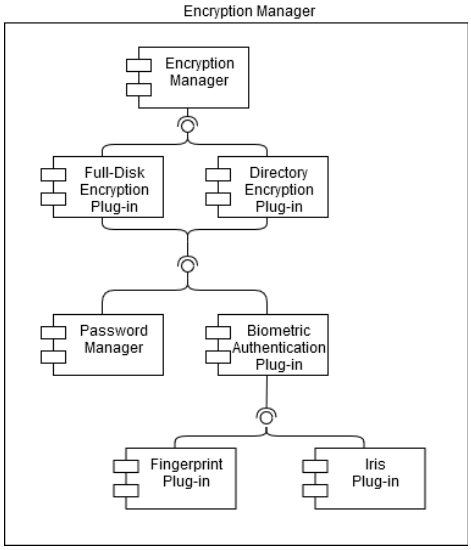
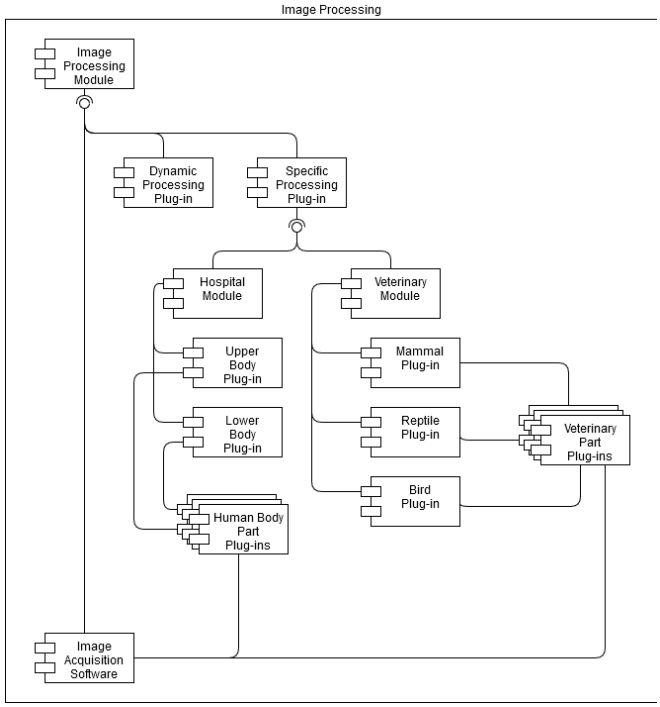


Image processing in the domain is important since optimization of the algorithms impact performance. Therefore, different architectural components are provided for ease of image processing in tracking and control domain. Specialized plug-ins for body parts have a crucial impact on performance of the scans.



Domain Architecture for System Architecture and Infrastructure [Mehmet Arif Şahin]

In this section, a reference architecture model will be created for the system architecture and infrastructure layer. The report prepared by the architectural team stated that Component-based architecture was used for this layer and it was said that this solution domain was inspired by the product line architecture. Component-based architecture with vertical decomposition is determined as the solution domain in our product line. This is because it is an architecture that can keep pace with the flexibility of the product line and make updates and changes easily.

The architecture formed by the architectural team has been used and in addition, internal variability has been provided by making additions for the newly introduced features. Reference is made to the architectural report for the parts where the same components are used. Therefore, the domain architecture includes variation points and variants documented in the variability model. To indicate this situation, the components and the frameworks in this layer are shown with diagrams, and an analysis of the commonality and variability of the domain created by these pieces of software is done. To better understand the domain architecture of the system architecture and infrastructure layer, the diagram of the architecture of this layer is shown in the figure below.

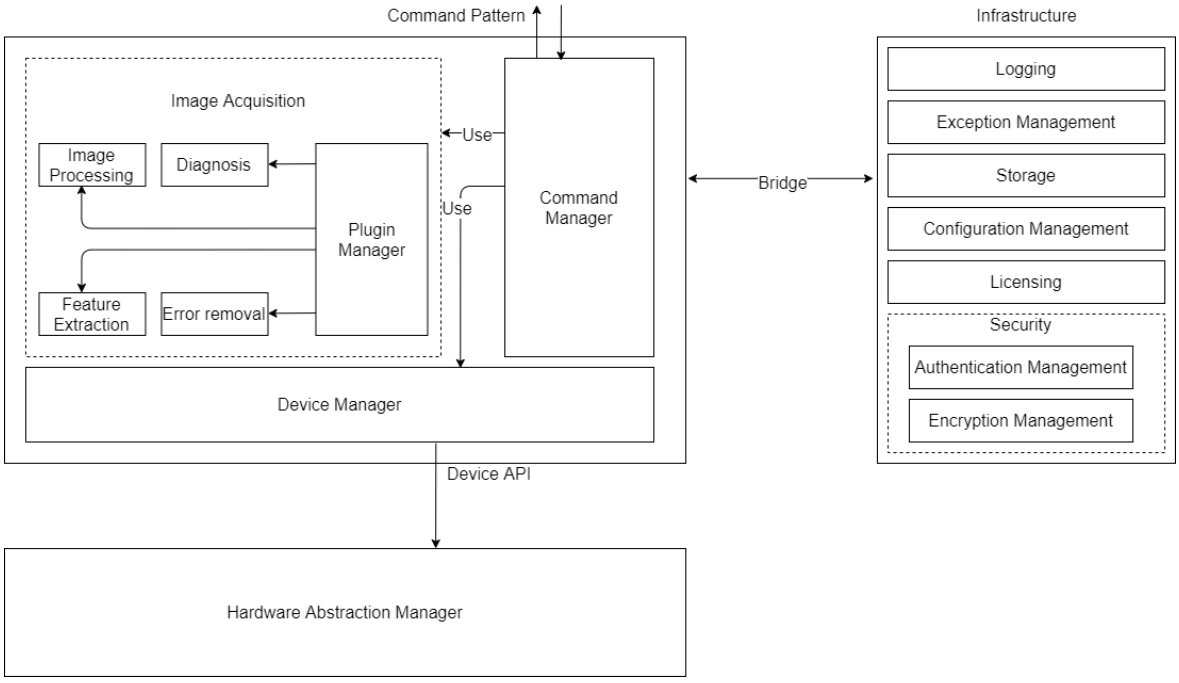


Figure: System Architecture & Infrastructure Architecture Diagram

Structurally, it remained almost identical to the architecture created by the architectural team. In addition, the authentication manager and encryption manager have been added to the infrastructure. The place where variability is provided is inside the components and usage of the plugin manager.

Let's first look at how the image acquisition framework, which does the most fundamental operations of the MRI software, is designed.

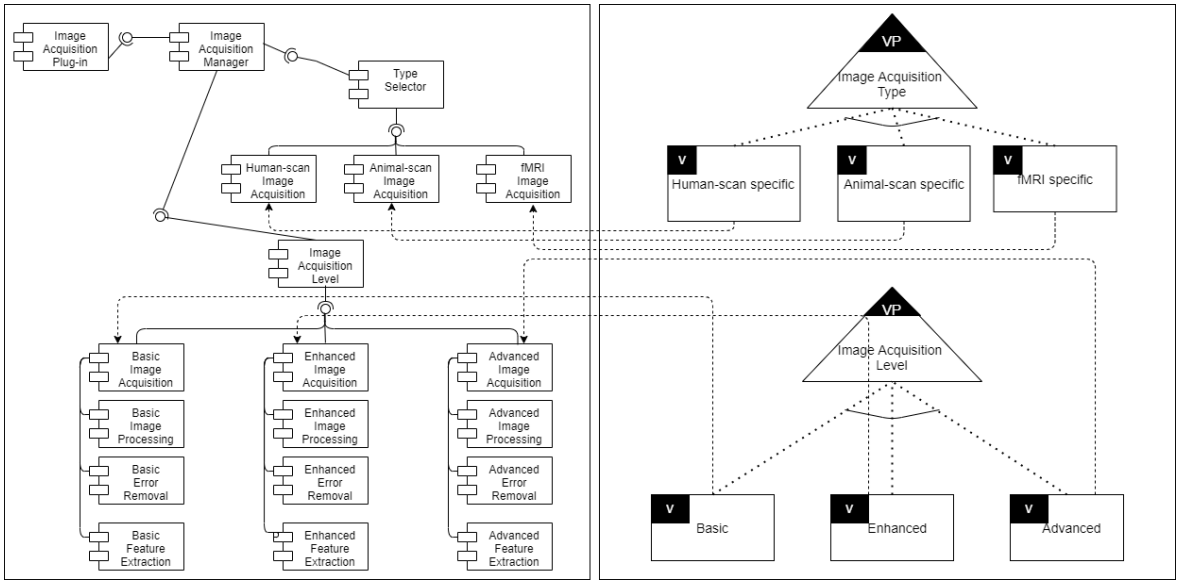
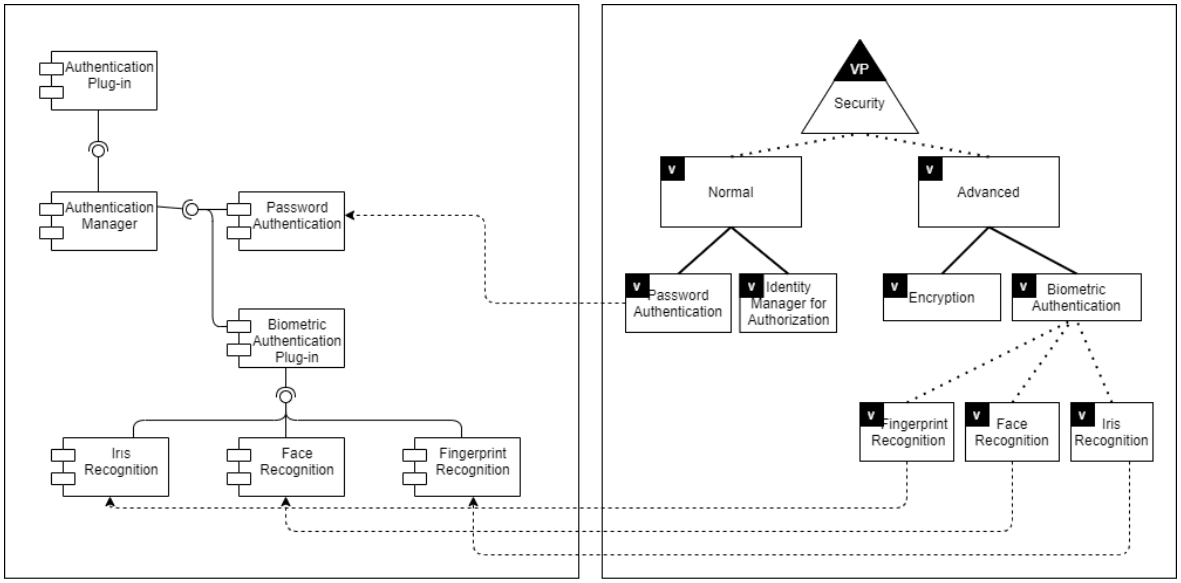


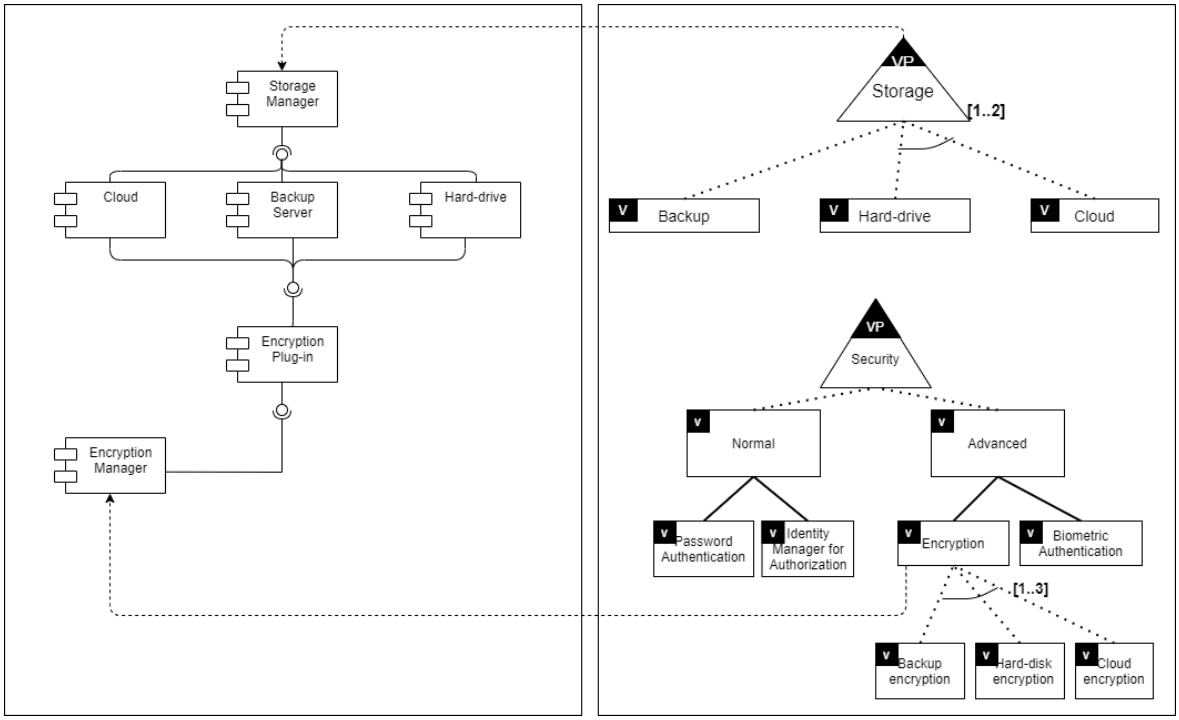
Figure: Image Acquisition Framework

The image acquisition modules vary depending on the level and type of machine to be used. Therefore, the necessary modules are added to the software during the installation with the image acquisition plug-in, according to the machine to be used.

Security is one of the variation points and you can see how authentication and encryption are done in the domain architecture figures below. Also, the domain has different storage options. Since the plugin system is used, the modules to be added in the architecture (internal variability) can be adjusted more easily according to the user's choices (external variability).

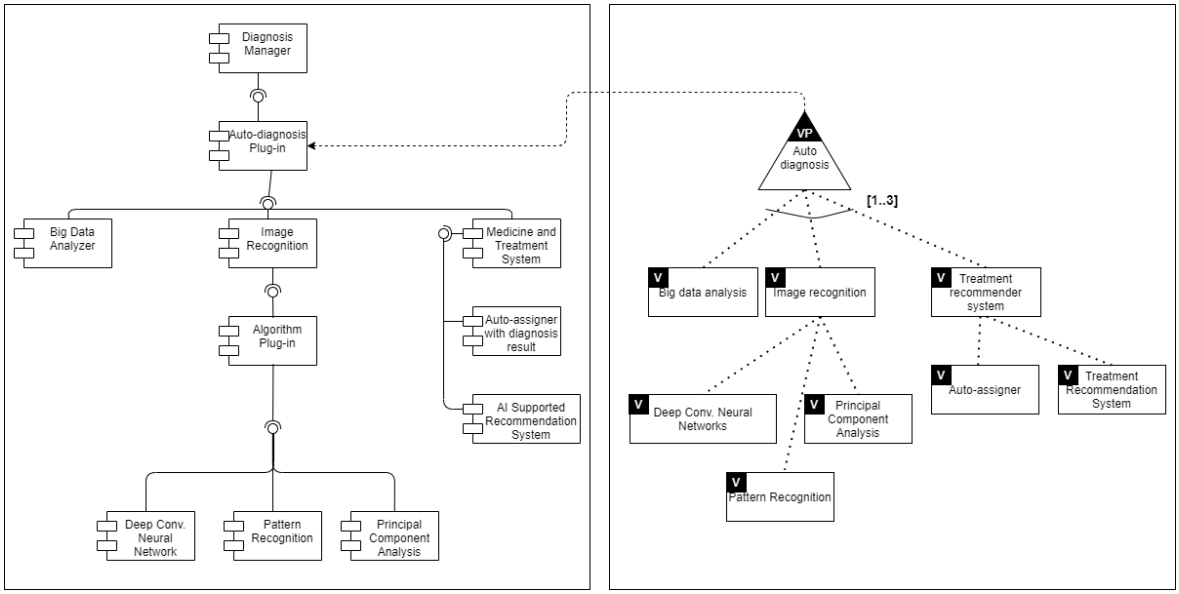


Authentication Framework



Storage and Encryption Framework

Finally, the diagnosis component is shown below. With the help of the auto-diagnosis plug-in, machine learning algorithms and modules can be added to the software at the desired level.



Auto-diagnosis Framework

Product Engineering

Product Engineering for Application Modelling and User Interface [Yunus Kerem Türk]

Main purpose of this part is to elicit and to document the requirements artefacts for a particular application at the same time reuse, as much as possible, the domain artefacts. The reuse of domain requirements artefacts for each application supports the overall goal of obtaining a high degree of domain artefact reuse.

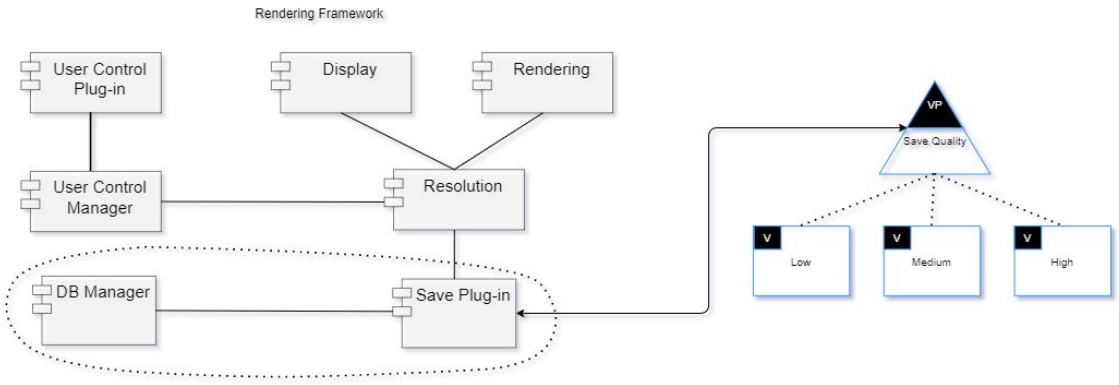
In this part of the report, domain architecture components will be applied to products that were chosen on the products that we described in product requirements. In that part we defined product specific requirements. These products are MRI MR-002, MRI MR-003, and MRI MR-V.

MRI MR-002

Reusable components coming from domain can be listed as follows:

1. Exam cards component
2. Rendering component
3. Image analysis component
4. Coloring component

Although listed components are used in this product, all components will be configured according to its requirements. Such as coloring; in the coloring component we have three different sub-modules, each responsible for specialized coloring techniques.
Additional to these reusable domain components, derived from the domain requirements; users requested that uploading scans at desired resolution should be provided. To achieve desired functionality we introduce new component works as an upload scan interface.
We had described the rendering component, that renders the scan at desired resolution. It should also be possible to save the file at desired resolution.



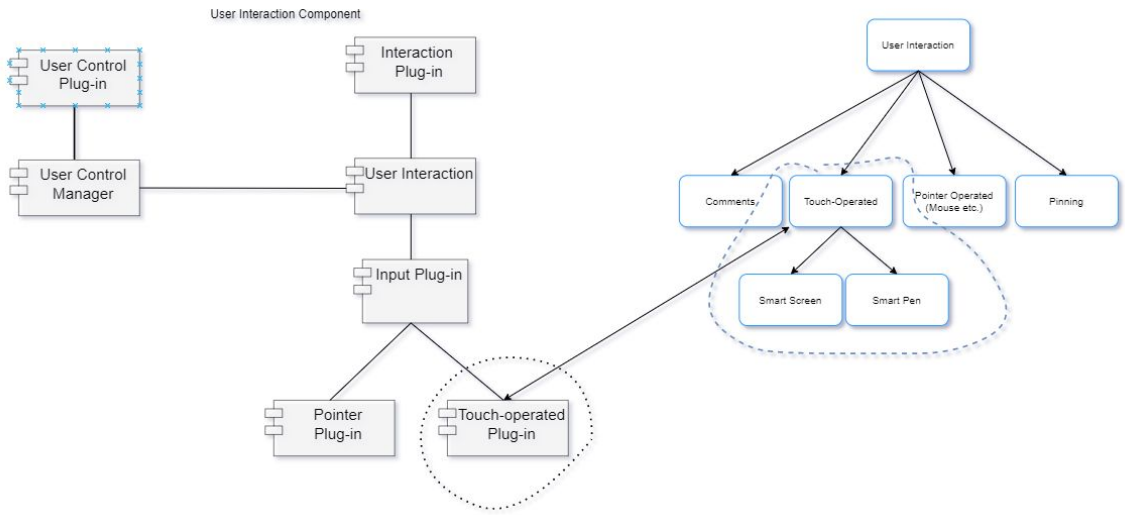
Save Plug-in are introduced in the rendering component to achieve the functionality of saving at desired resolution. In the process, already defined components are heavily used. So it is sensible to describe the extra functionality within this component.

MRI MR-003

This model of our product line is designed for academic purposes. Therefore it will be used in university hospitals etc. On this product, reusable features are:

1. Exam cards component
2. Comparison component
3. Image analysis component
4. Coloring component
5. User interaction component

Although it has a user interaction component, for this product there is a newly introduced feature in the user interaction component, which is touch operated user screen.

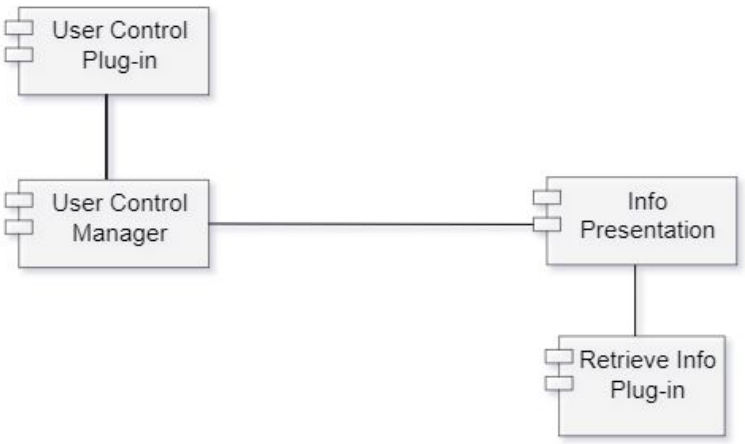


MRI MR-V

We defined an MRI machine specialized for veterinary applications in our product portfolio, and named it with code MR-V. In this type of MRI machine, many of the components remain the same, but some of them need a configuration, such as exam cards. So the components that are taken from domain components are:

1. Exam cards component specialized for veterinary applications
2. Rendering component
3. Coloring component
4. Image analysis component

Other than these components, we want to provide a user interface that shows the prior knowledge about the species should be provided. For that purpose, we define a new component that is responsible for this task.



Product Engineering for Tracking and Control Systems [Onur Demirezen]

Engineering under the scope of Tracking and Control Domain for three previously selected products that are MR-V2, MR-004 and MR-005, is described in this section. While most are common, there are specific components to products.

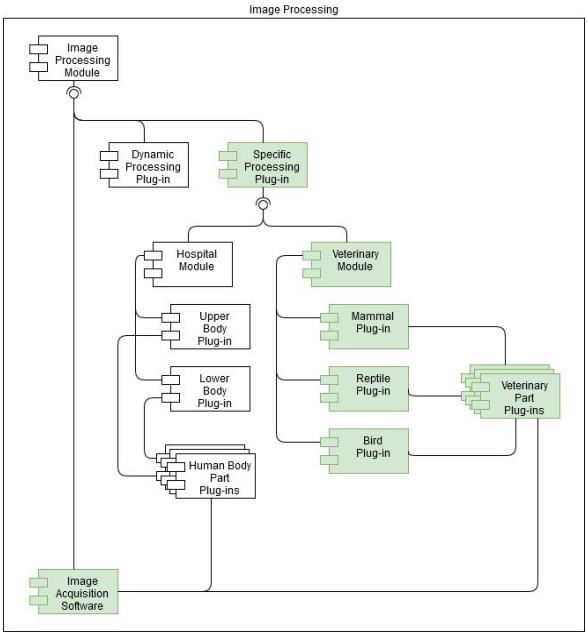
MRI MR-V2

Below is a list of reusable components coming from domain. Detailed descriptions and orthogonal variability models can be found in previous two chapters.

1) Performance monitoring module without AI plug-in that enables basic local performance data storage and interpretation
2) Scan manager with Cloud Manager, Veterinary PACS plug-in that enables automatic upload of scans
3) Auto diagnosis module with Neural Network plug-in

Product is specialized by how the common components are used along with one product-specific plug-in. Performance monitoring module's AI plug-in is not included here. Also, scan modularity module's mobile application plug-in is not used, and auto-diagnosis has only Neural Network Plug-in enabled. Cloud based classification is not enabled, therefore not available. The product specific plug-in added to MR-V2 is as follows

4) Image Processing Module with Specific Processing Plug-in that includes a module designed for veterinary applications. Veterinary Part Plug-ins are also included.



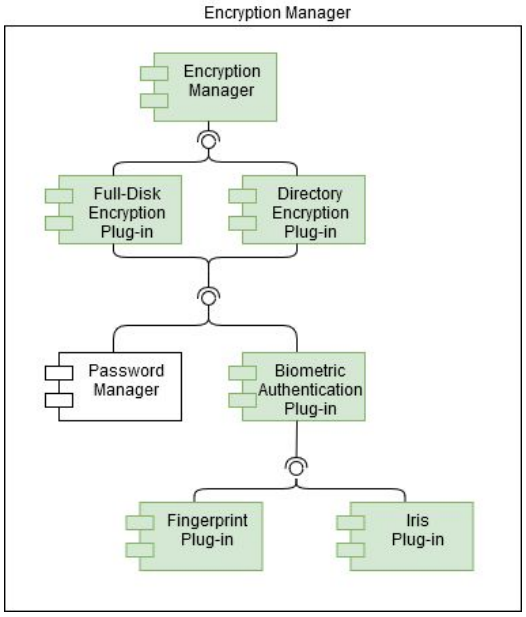
MRI MR-004

Common components used in MR-004 are:

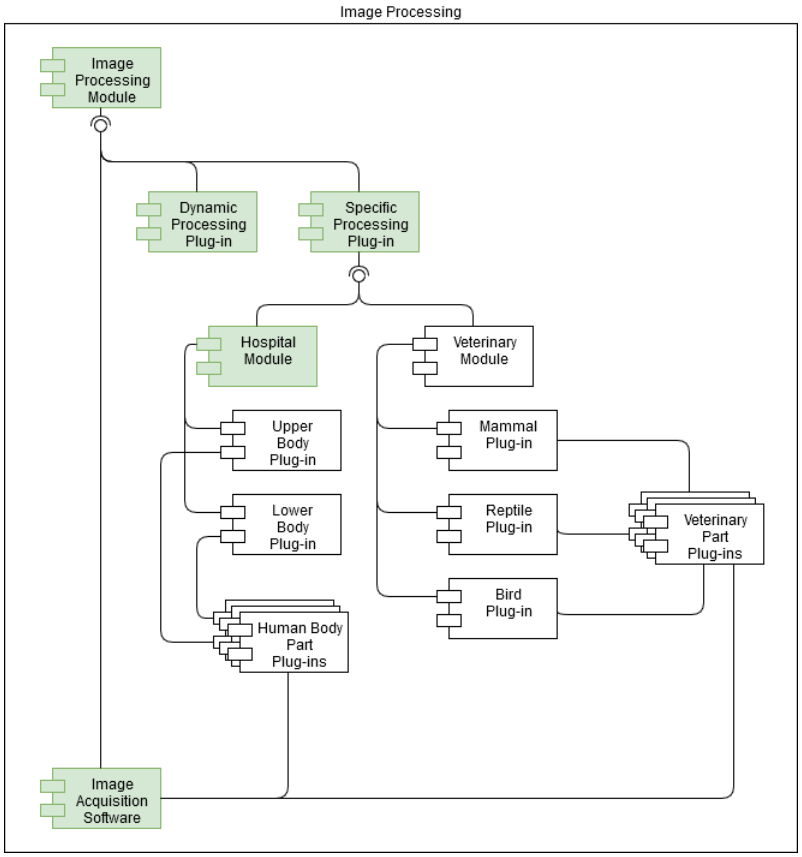
1) Performance monitoring module with AI Plug-in reporting to Storage Module
2) Scan Manager connected to Storage and Cloud Managers with PACS Plug-in

Common components include customized plug-ins for specific purposes. In MR-004, Scan Manager is connected to Storage Manager for on-site scans, and Cloud Manager for PACS integration. Performance monitoring module has AI Plug-in enabled but not connected to notifiers. AI gives insights just in storage, therefore it is connected to the storage manager. Special components for MR-004 are as following:

3) Active Encryption Manager with full-disk and directory encryption, Biometric Authentication Plug-in is enabled



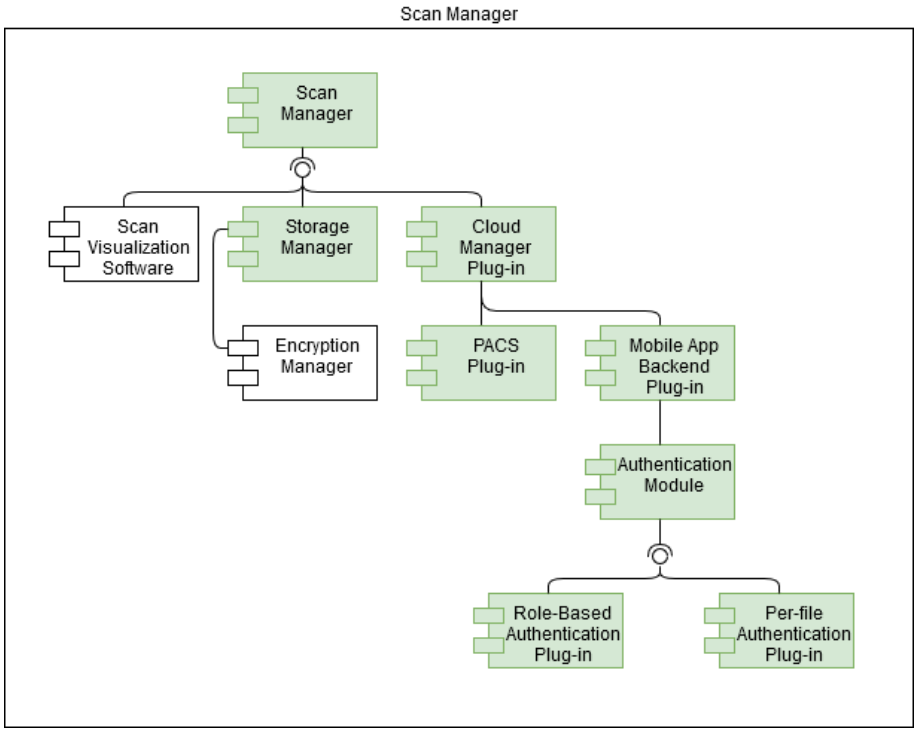
4) Dynamic Image Processing Plug-in is enabled and unique to this product



MRI MR-005

While MR-005 has all the common components from other two products in the scope described above, it has many unique features, therefore components.

1) Human Body Parts Plug-ins for faster scans of humans
2) Mobile Application Backend Plug-in for both Performance Data and Scan Modularity
3) Automatic performance optimization with Automatic Maintaining Plug-in for Performance Data
4) Automatic Diagnosis Module with Classification Manager for cloud-based classification


Scan Manager
Scan Manager
Scan Visualization Software
Storage Manager
Cloud Manager Plug-in
Encryption Manager
PACS Plug-in
Mobile App Backend Plug-in
Authentication Module
Role-Based Authentication Plug-in
Per-file Authentication Plug-in

Automatic Diagnosis

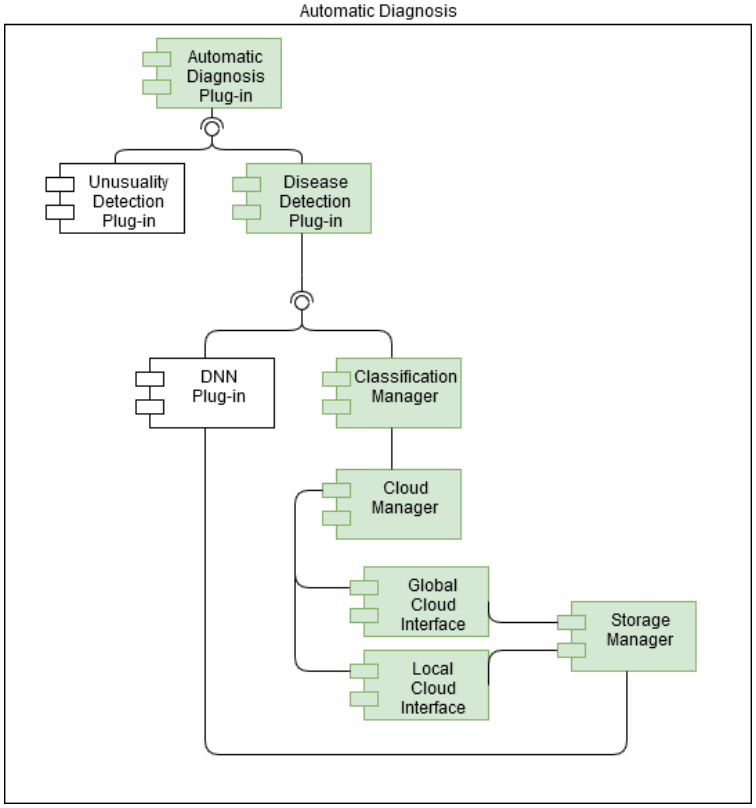
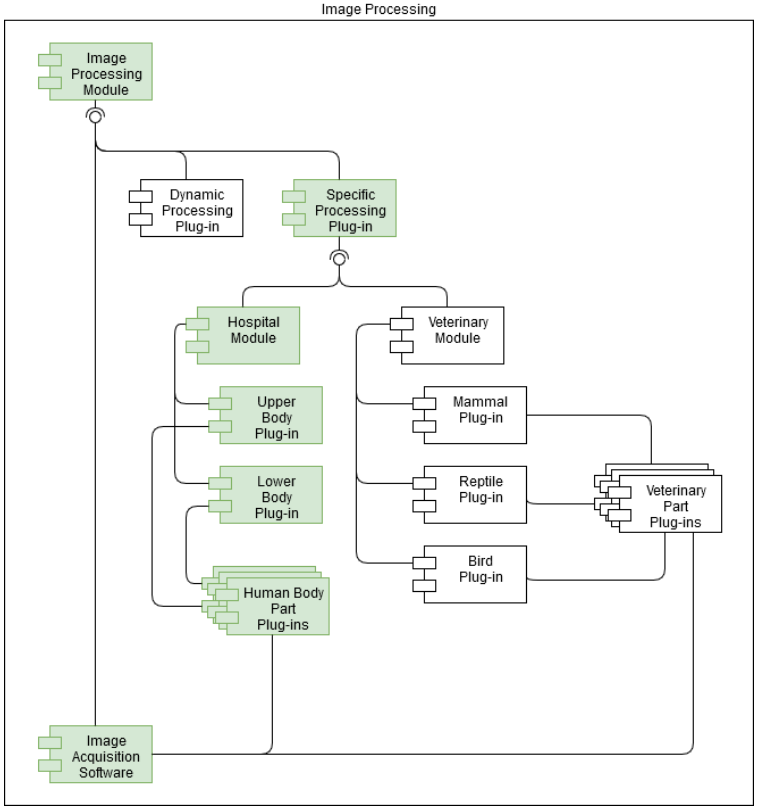


Image Processing



Product Engineering for System Architecture and Infrastructure [Mehmet Arif Şahin]

In this section, product engineering is applied for 3 products selected from the product family which are MRI MR-V2, MRI MR-SAFE, and MRI MR-005, and the application variability model of the products are created for the system architecture and infrastructure layer. Since there are products created on the product line, most of the components of the products which are reusable come from the domain. Therefore, firstly the reusable component, frameworks coming from the domain are indicated and referenced to domain reusable artefacts. Then the product variability models of the product-specific components are given.

MRI MR-V2

Reusable components coming from domain can be listed as follows (you can check the domain architecture for detailed descriptions for this software components and their orthogonal variability models),

1) In the image acquisition framework, type selector plug-in enables the animal-scan image acquisition module and image acquisition level plug-in enables the enhanced level image acquisition modules.
2) Authentication manager selects the password authentication module.
3) Storage manager handles storage for both hard-drive and cloud server.
4) Authentication plug-in enables the image recognition module for automatic diagnosis.

At this point, product-specific module is added to software which is

5) Specialized AI for animal-specific treatment recommendation system module is added to software with a diagnosis manager.

You can see below the application variability model for this new variant and corresponding software module.

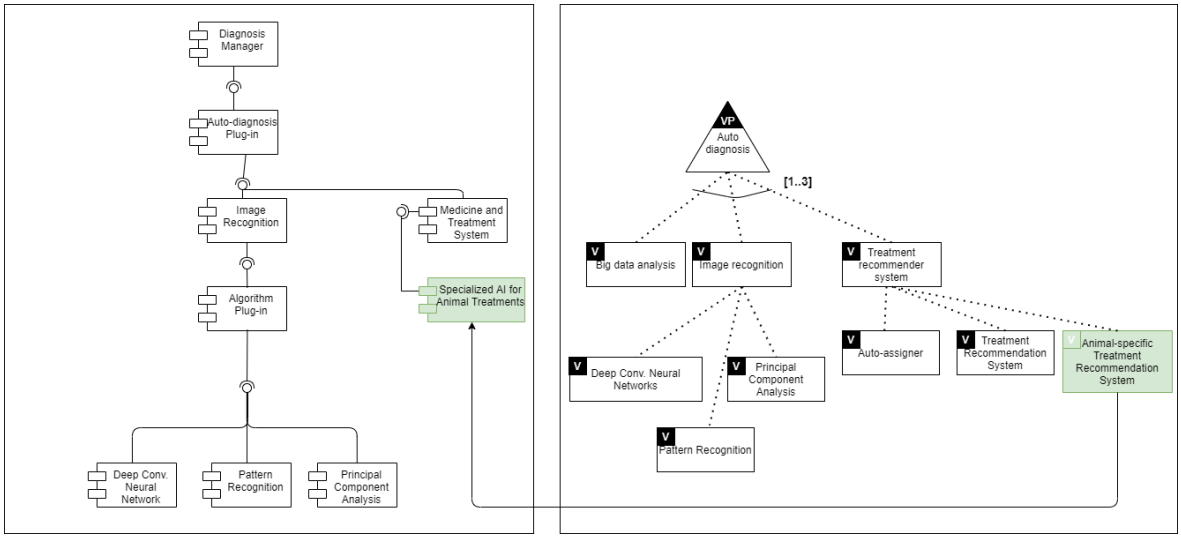


Figure: MRI MR-V2 Application Specific Variability Diagram for System Architecture and Infrastructure

MRI MR-SAFE

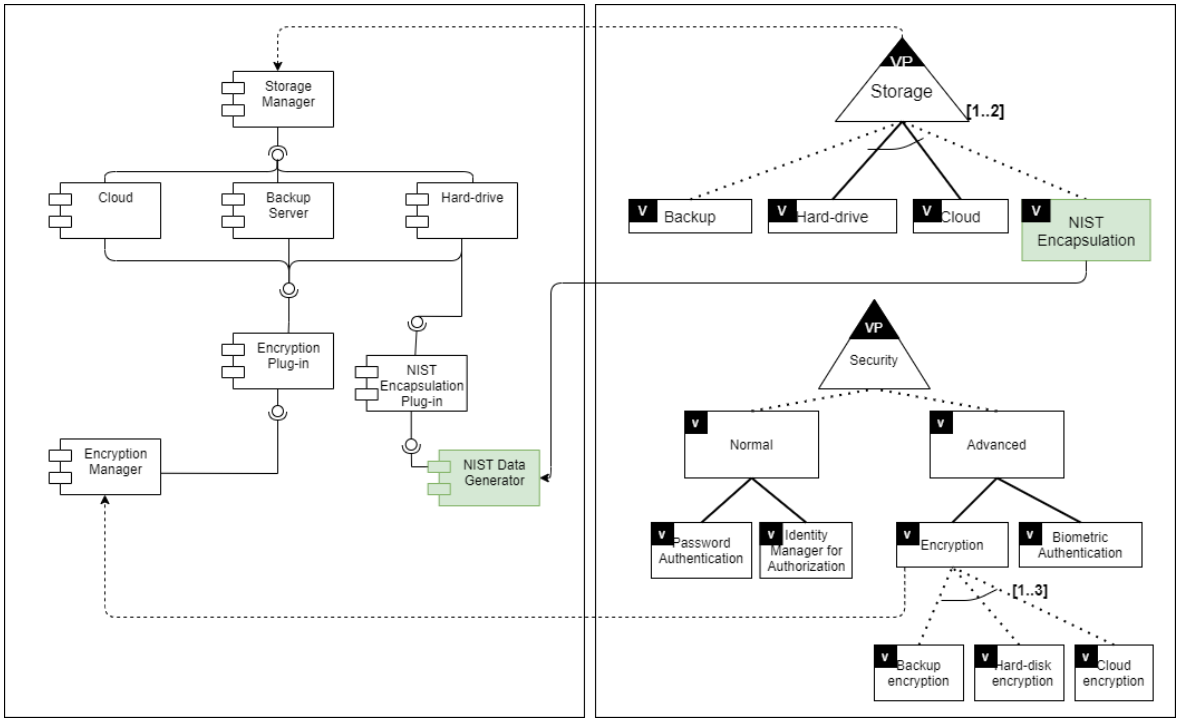
Reusable components coming from domain can be listed as follows (you can check the domain architecture for detailed descriptions for this software components and their orthogonal variability models),

1) In the image acquisition framework, type selector plug-in enables the human-scan image acquisition module and image acquisition level plug-in enables the basic level image acquisition modules plugged.
2) Authentication manager selects one of the biometric authentication modules with a biometric authentication plug-in (In installation time customers select one of them then the corresponding modules are added to software).
3) Storage manager handles storage for both hard-drive and cloud server.
4) Storage encryption must be provided. Therefore, the encryption manager handles which storage modules are encrypted with the encryption plug-in.

At this point, product-specific module is added to software which is

5) Biometric data is encapsulated with NIST standard then stored. Therefore, the NIST data generator module encapsulates the data before storage.

You can see below the application variability model for this new variant and corresponding software module.



MRI MR-SAFE Application Specific Variability Diagram for System Architecture and Infrastructure

MRI MR-005

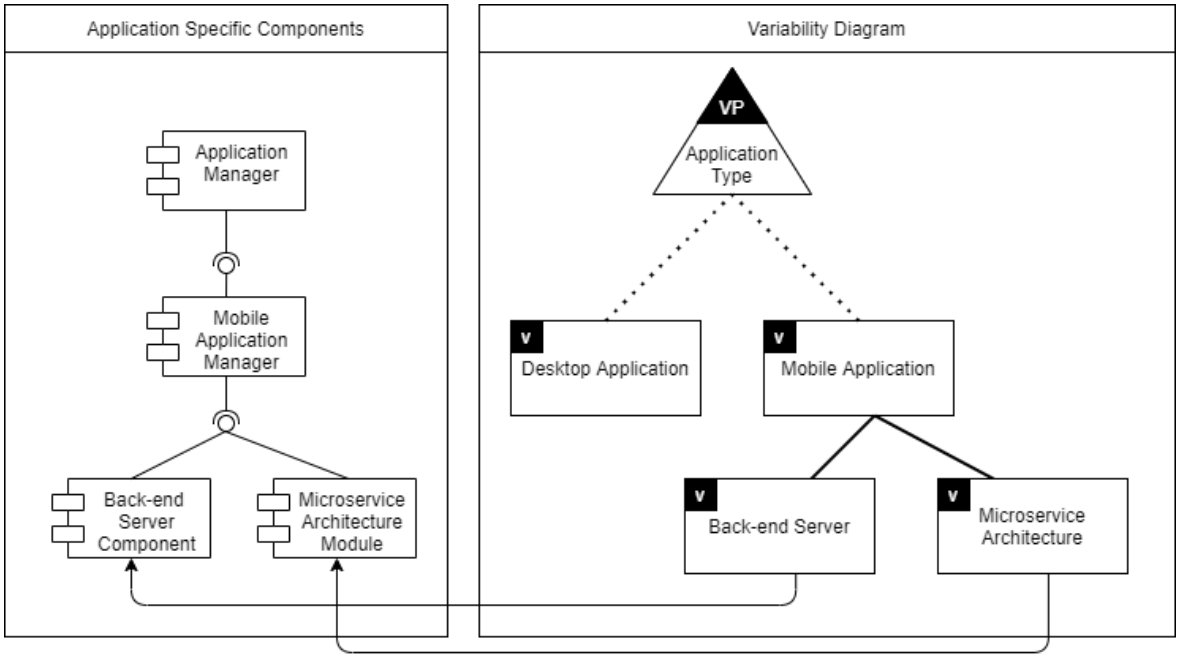
Reusable components coming from domain can be listed as follows (you can check the domain architecture for detailed descriptions for this software components and their orthogonal variability models),

1) In the image acquisition framework, type selector plug-in enables the human-scan image acquisition module and image acquisition level plug-in enables the enhanced level image acquisition modules plugged.
2) Authentication manager select password authentication module.
3) Storage manager handles storage for both hard-drive and cloud server.

At this point, product-specific module is added to software which is

4) Back-end server module is added for mobile application and microservice architecture.

You can see below the application variability model for this new variant and corresponding software module.

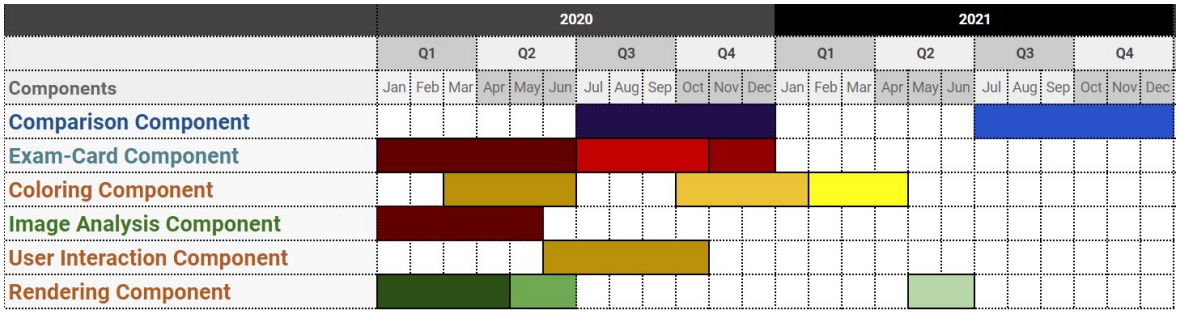


MRI MR-005 Application Specific Variability Diagram for System Architecture and Infrastructure

Artefact Table

Artefact Table for Application Modelling and User Interface [Yunus Kerem Türk]

System Architecture and Infrastructure Artefact List	Programming Time (Person*Month)
Comparison Component	72 P*M
Exam-Card Component	72 P*M
Coloring Component	66 P*M
Image Analysis Component	30 P*M
User Interaction Component	30 P*M
Rendering Component	30 P*M



It should be noted that implementation starts in 2020 which is hypothetical. While planning this timeline This timeline is based on the products that I chose for product requirements and product engineering, and these products are MRI MR-002, MRI MR-003, MRI MR-V.

Since our first product MRI MR-001 will be released in 2021 January, all the necessary parts of the domain artefacts should be provided. Therefore, I assume that implementation efforts start off 2020 January.

Usually a component has different capabilities. In my case,

- Exam cards has three version: Basic, Enhanced, Advanced
- Comparison has two branch: Old data, Healthy data
- Coloring component: Basic, Enhanced, Advanced
- Rendering Component: Low, Medium, High

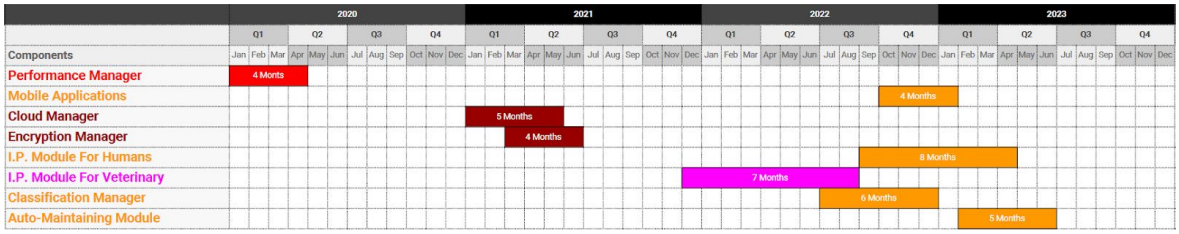
Although it is not shown in the above figure, product specific functionalities should be implemented too. For that purpose,

- First three months of 2021 should be preserved for product specific functionality for MRI MR-002. Remaining months should be preserved for testing the application.
- First and second quarter of 2021 should be preserved for product specific functionality for MRI MR-003. Remaining months should be preserved for testing the application
- First two months of 2021 should be preserved for product specific functionality for MRI MR-V. Remaining months should be preserved for testing the application.

Artefact Table for Tracking and Control Systems [Onur Demirezen]

This section includes the table and development timeline of the artefacts in the architecture. We have an assumption that *Person*Month* metric is based on calendar years and an employee has 12 Person*Month effort.

Artefact	**Programming Time (Person*Month)**
Performance Manager AI	60
Mobile Applications	40
Cloud Manager	50
Encryption Manager	20
Specific Image Processing Module for Humans	120
Specific Image Processing Module for Veterinary	140
Classification Manager	60
Automatic Maintaining Module	50



Since the performance manager is a crucial component and used in development, we start early and finish as soon as possible. Other modules have matching colors with the products in the product timeline, and their development times are determined by it.

Artefact Table for System Architecture and Infrastructure [Mehmet Arif Şahin]

In this section, the software parts required for the creation of the system architecture and infrastructure layer are listed. The programming times of these software parts are determined hypothetically. In addition, the production plan has been made to correlate with the portfolio. In my assumption, the system architecture and infrastructure development department is divided into 3 teams and each team has 30 members. First team takes the task of creating image acquisition modules. There are 3 different image acquisition module packages which are human-scan, animal-scan and fMRI-scan specific modules. Each package has different levels of image acquisition modules which are basic,enhanced advanced. Second team takes the tasks related to security. Firstly, they will develop authentication components which include password-protected authentication and biometric authentication modules(iris recognition, face recognition, fingerprint recognition). Third team takes the tasks of handling storage and developing auto-diagnosis modules (big-data analysis,image-recognition, treatment and recommendation system).

Programming time metric -> 1 corresponds to one developer's efforts within one month.

System Architecture and Infrastructure Artefact List	**Programming Time**
Image Acquisition Modules for human-scan	150
Image Acquisition Modules for animal-scan	150
Image Acquisition Modules for fMRI	300
Authentication Components (password-protected and biometric)	270
Encryption Component	60
Storage Manager	90
Auto-diagnosis Components	450

Table: Artifacts Table

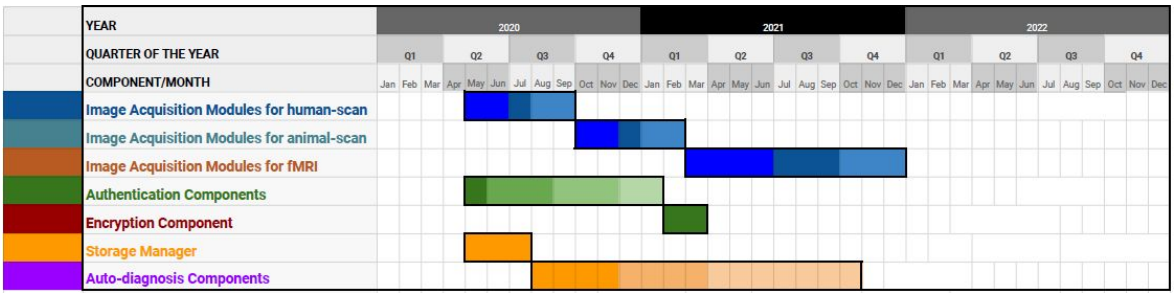


Figure: Artefact Production Plan Timeline

Legend: Blue: Team 1, Green: Team 2, Yellow : Team 3

-Shades of blue corresponds to the level of image acquisition modules (basic, enhanced, advanced).

-Shades of green corresponds to the authentication components (password-protected authentication, iris recognition, face recognition, fingerprint recognition).

-Shades of yellow corresponds to the auto-diagnosis modules (big data analysis, image recognition module, treatment and recommendation system).

Results and Discussions

Results and Discussions for Application Modelling and User Interface

[Yunus Kerem Türk]

In the following section, I will share the gained knowledge of software product line design during the process designing the product line for MRI machines.

It is known that companies are trying to make their own products and therefore product lines. For the purpose of creating the product line, market analysis should be done.

In the market analysis process, I learned how to analyze the market and how to gain knowledge from the market about how the products should be. After market analysis, we defined the motivating features. While defining the features, I learned that it is important to know the products that are already released and the domain. From using these features, we defined different kinds of products that aim to reach different types of markets such as academic, veterinary applications.

To assess the features, we learned and used the KANO survey which is like a feature filter. From the results of this survey, the delighters are sorted out and placed in product specific requirements.

From that point, domain and product artefacts are created. I learned how to create a feature tree that describes the functionalities seen by the user. After the feature tree is created, I learned how to define the variation points and variants. Based on the features, I listed the requirements for the domain and product. There were different types of techniques, schemas to list the requirements and many of them were learned at the process.

After all the requirements and features are listed, I started to write the domain architecture and product specific architecture. While there were many techniques about this issue, we tried to keep it small because of limited time.

Besides from the content of the report, I learned how important the design rationale is.

Encountered Difficulties While Creating a Product Line

Main difficulty that I encountered was that I had no prior knowledge about neither medical software nor software engineering. Because of that, I had to do a lot of research before beginning to write and report.

In the architecture section it was hard for me to divide the system into proper and manageable components.

Results of the Report

Requirements

1) Introduction that describes the aim and content of the report.
2) Product management that includes the analysis of the market, KANO survey, motivating features, product portfolio
3) Domain variability model that defines the feature tree, variabilities, variation points, and relationship between variants and variation points.
4) Requirements that describe the customer's and domain's needs.
5) Architecture for both domain and products
6) Artefact table

Evaluation of the Requirements

When the report is traced from beginning to end, it is seen that there is a consistency between product-requirement-feature-architectural component. First of all, the aim, task and content of the report was given at Introduction. The given report was analyzed and the architectural requirements and design of the architecture were examined in the sections [A Short Description of the Proposed Architecture] and [Requirements]. Based on this design, the architecture was divided into 3 subsystems and I applied the product line processes for the application modelling and user interface. The portfolio was created according to the motivating features obtained as a result of the analysis and the KANO application in the section [Product Management]. A domain variability model was created to meet the variability of the features in the section #heading=h.2et92p0[Domain Variability Model for Application Modelling and User Interface]. Requirements were determined by considering commonality and variability in the section [Domain Requirements for Application Modelling and User Interface]. To meet these requirements, components, and frameworks were created in the [Domain Architecture for Application Modelling and User Interface] section.

Therefore all requirements have been met.

Results and Discussions for Tracking and Control Systems [Onur Demirezen]

Learning Outcome

In this section, explanations of learning outcomes from designing a product line for Tracking and Control Systems are discussed.

Among everything, first to mention is the realization of the benefits of creating a product line architecture for software is the most important learning outcome of this project. Project-based approach is outdated in the era we are in, therefore modularization is mandatory. Second important realization is capability of understanding a product's software that I have never used, or even see someone use. With only a supposed architecture team report at hand, hopefully we were able to create a product line architecture for the proposed software architecture.

While developing a product line is important, the method is crucial. Design process should be done step-by-step and must be iterative. Iterative working is also a learning outcome of this report. I have learned to interpret a comprehensive report, then extract the information useful for our purpose. Then I have created products with a team for various purposes, according to features also determined by us. Since MRI machines were very far from our scope, we properly learned how to analyze a market and user stories for generating features. Implementation of the features with variation points was an enjoyable and educational process. I started to realize how the daily software products I use are generated. My point in vision in product features has changed. By generating products with common and distinctive features, I am now able to understand an actual software product's motivating features, what is included and what is not etc. I learnt designing an architecture according to requirements and engineering products according to the architecture. All sums up into iterative thinking design, which is the third most important general outcome that I have to mention.

Encountered Difficulties

The main difficulty encountered is MRI machine software itself. As stated in the previous section, MRI machine software was very far from our scope. We are familiar with various software products like web applications, video games, operating systems, database applications, communication software etc. but MRI software is not one of them. Hence, definition of the requirements and features were very difficult to overcome. We have consulted a friend that is an MD candidate and somewhat experienced in MRI machines to overcome this unfamiliarity.

An important difficulty was understanding and adapting the software architecture. Having no prior knowledge about the architecture was difficult to overcome.

Results

In this section, I defined the requirements of the report and discussed if the report satisfies them or not.

Requirements of the Report

1) Proper definition of the purpose and method
2) Properly stating the assumptions
3) Proper analysis of the architecture report
4) Proper generation of the product portfolio
5) Proper division of the domains
6) Ease of traceability
7) Proper definition of the domain variabilities
8) Proper definition of the domain and product requirements
9) Proper definition of the domain and product architecture
10) Properly stating artefacts

Evaluation of the Requirements

From the beginning of the report to the end, when tracing is done, there is a consistency between all contents of the sections. Almost all sections reference the previous ones. First, method and purpose has been defined in Purpose and Method sections[1]. Then, the architecture report has been analyzed and described in Architecture Design Method[3]. In reference to the architecture report, according to market analysis and market of focus, a product portfolio has been defined in the Product Portfolio by considering Motivating Features[4]. Domain Variability Model has been defined individually by dividing the domain into three main parts in reference to the architecture report and Traceability Matrices has been generated for the ease of understanding[5][6][7]. According to the domain variability model and product portfolio, Domain and Product (after selecting three from portfolio) requirements has been defined, then architectural components and design has been generated for Domain and Products[8][9]. After the engineering of the products, used artefacts have been stated in Artefact Table[10]. Hence, the report satisfies the requirements.

Results and Discussions for System Architecture and Infrastructure

[Mehmet Arif Şahin]

Learning Outcome

During this period, a lot of information was gained about software product line architectural design. It is observed that a product line is an inevitable process to create a product family. 2 key concepts are crucial for product development in the product line. These are platform and mass customization concepts. For mass customization, the domain must be created correctly and the variability must be managed correctly. Therefore, a product line is created systematically. The process of creating a product line is iterative. Reorganizations and improvements can be made continuously. Also, problem domain and solution domain should be well separated from each other.

First of all, in the product management section, the focused market area is determined and the market is analyzed. Features that will motivate the products will be extracted from these analyzes. A consistent portfolio is created from these features and the distribution of the features is done correctly with the KANO application. To manage the variability, domain variability model is created, the variation point and variants are determined, and variability dependencies are indicated. Domain requirements are determined by using these models and portfolio. Also, commonalities are determined and added to the domain. Domain architecture is created to meet these domain requirements. The product line is separated into two sub-processes. Above processes are done in the domain engineering sub process. The other sub-process is application engineering. In the application engineering subsections, the same domain sections are done specifically for the product. In product engineering, besides the reusable features coming from the domain, product-specific requirements are defined and an implementation is made for special requirements. In order for these processes to be processed properly, the features of the requirements and the features must meet the architectural parts (traceability is one of the most important concepts). All these processes can prevent tangling and scattering in software products.

Encountered Difficulties

Some difficulties were encountered while creating the product line. First of all, it was a systematic and challenging task to make the portfolio-requirement-feature-component trace correctly. There is no normal start-to-end report writing process. The report is written iteratively. This extended the report writing time because a detailed analysis is required. Besides, MRI software is an area we haven't studied, so creating variability is a bit

challenging because we had to distinguish which pieces of software could be used in MRI software.

Results of the Report

Requirements

1) Report should be able to analyze the given architecture report well.
2) The domain should be divided into 3 side-systems.
3) Market analysis should be done for the focused market area, motivating features should be determined and a portfolio should be created accordingly.
4) Domain variability models should be created.
5) Domain requirements should be determined.
6) Product requirements should be determined for 3 distinct products in the product family.
7) Domain architecture should be created.
8) Product engineering should be applied for 3 distinct products.
9) Artifacts table should be created.

Evaluation of the Requirements

First of all, the proposed architectural report was analyzed and the architectural requirements and design of the architecture were examined in the sections [A Short Description of the Proposed Architecture] and [Requirements]. Based on this design, the architecture was divided into 3 side-systems and I applied the product line processes for the system architecture and infrastructure layer. The portfolio created according to the motivating features obtained as a result of the analysis and the KANO application in the section [Product Management]. A domain variability model was created to meet the variability of the features in the section [Domain Variability Model for System Architecture and Infrastructure[Mehmet Arif Şahin]]. Requirements were determined by considering commonality and variability in the section [Domain Requirements for System Architecture and Infrastructure[Mehmet Arif Şahin]]. To meet these requirements, components, and frameworks were created in the [Domain Architecture for System Architecture and Infrastructure[Mehmet Arif Şahin] section. On the other hand, the product-specific requirements and artefacts of the 3 products selected from the portfolio were indicated in the sections [Product Requirements for System Architecture and Infrastructure[Mehmet Arif Şahin]] and [Product Engineering for System Architecture and Infrastructure[Mehmet Arif Şahin]]. A traceability matrix was created to trace all these parts [Traceability Matrix]. In the [Artefact Table for System Architecture and Infrastructure[Mehmet Arif Şahin]] section, an artefacts list is prepared. Eventually, this is how all the requirements for this report were met under the subsections.