Software Requirements Specification for AortaGeomRecon

Jingyi Lin

September 15, 2023

Contents

Revision History

Date	Version	Notes
2023-02-12	1.0	Notes
2023-03-01	1.01	Modified system context image, coordinate systems, and goal statements.
2023-04-29	1.02	Added requirements, instance models, data definitions
2023-06-05	1.03	Added Traceability Matrices
2023-06-18	1.04	Included the missing sections, modified the equations of DD, and IM. Added 4 new NFRs.
2023-09-15	1.05	Updated Traceability Matrix.

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with existing documentation for 3D Slicer program. The symbols are listed in alphabetical order.

symbol	type	description
m	$\mathbb{N}$	The first dimension of the segmentation volume.
m_i	$\mathbb{N}$	The first dimension of V_{in}.
m_o	$\mathbb{N}$	The first dimension of V_{out}.
n	$\mathbb{N}$	The second dimension of the segmentation volume.
n_i	$\mathbb{N}$	The second dimension of V_{in}.
n_o	$\mathbb{N}$	The second dimension of V_{out}.
p	$\mathbb{N}$	The third dimension of the segmentation volume.
p_i	$\mathbb{N}$	The third dimension of V_{in}.
p_o	$\mathbb{N}$	The third dimension of V_{out}.
$slice$	$\mathbb{R}^{m \times n}$	A slice is a 2 dimensional image view from the superior to inferior direction.
v	$\mathbb{R}$	A voxel reports the intensity of a single point on a grey-scale three-dimensional images.
$HIGH$	$\mathbb{N}$	A high intensity values means 1 on a scale of 0 and 1, or 255 on a scale of 0 to 255.
LOW	$\mathbb{N}$	A low intensity values means 0 on a scale of 0 and 1, or 0 on a scale of 0 to 255.
$Seed\_a$	$\mathbb{N}^3$	The initial ascending aorta centre coordinates.
$Seed\_d$	$\mathbb{N}^3$	The initial descending aorta centre coordinates.
$Start$	$\mathbb{N}^3$	A coordinate indicates the indexes of a starting voxel.
V	$\mathbb{R}^{m \times n \times p}$	Volume formed by a sequence of slice

1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
AortaGeomRecon	Aorta Geometry Reconstructor
DD	Data Definition
DICOM	Digital Imaging and Communications in Medicine
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
T	Theoretical Model

2 Introduction

This document provides an overview of the Software Requirements Specification (SRS) for the AortaGeomRecon. AortaGeomRecon provides a semi-automatically aorta segmentation method, a highly customizable aorta segmentation module, and an interactive user interface to apply the segmentation workflow.

One of the existing methods involves the use ITK-Snap software and its segmentation module. First, the user needs to convert the DICOM data files (or any other file type) to VTK file. Then, the user can load the VTK file to ITK-Snap, and use its segmentation module to perform aorta segmentation. This segmentation method lets user initiate several voxels within the aorta volume and expand with a user's given size in each iteration. After the aorta volume has been filled by the "bubble", the user needs to cut the parts that are not within the aorta.

2.1 Purpose of Document

The main purpose of this document is to provide sufficient information to understand what AortaGeomRecon module does. The goals and theoretical models used in the AortaGeomRecon segmentation module implementation are provided, with an emphasis on explicitly identifying assumptions and unambiguous definitions.

2.2 Scope of Requirements

The scope of requirements only covers for the segmentation of the organ, more specifically the ascending aorta, the aortic curvature and the descending aorta. The requirements assume that the source of the data is accurate, and the user can manipulate (read, change dimensions) the data.

2.3 Characteristics of Intended Reader

The readers of the SRS should have taken the university level introduction to computational mathematic course, and be capable of understand the mathematical notation in the instance model section. The readers might have taken the university level introduction to software engineering course, have learned at least the waterfall software development model, and understands the purpose of the software specification requirement document, and other documents.

2.4 Organization of Document

The organization of this document follows the template for an SRS for scientific computing software proposed by Koothoor (2013) and Smith and Lai (2005). The presentation follows the standard pattern of presenting goals, theories, definitions and assumptions. The goal

statements are refined to the theoretical models, and theoretical models to the instance models. For readers that would like a more bottom-up approach, they can start reading the instance models in Section 4.2.6 and trace back to find any additional information they require.

3   General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

3.1   System Context

Figure 1 shows the system context. A circle represents an external entity outside the software, the user in this case. A rectangle represents the software system itself. Arrows are used to show the data flow between the system and its environment.
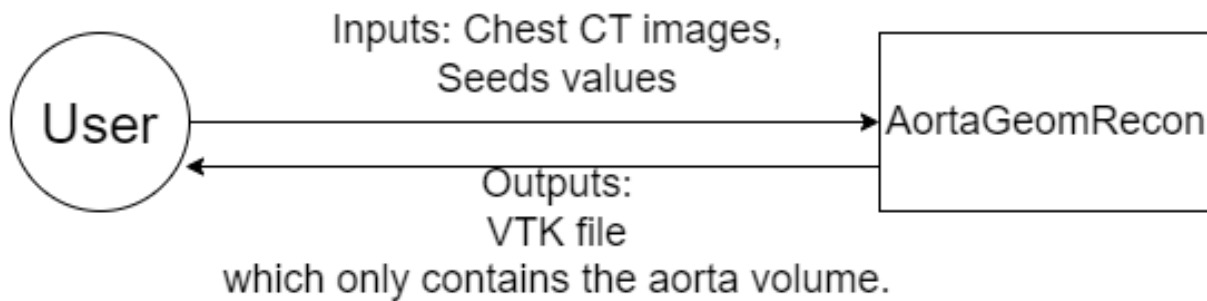

Figure 1: System Context

- User Responsibilities:

    - Provide the input data to the system

    - Ensure the input meets the necessary assumptions

    - Verify the result meets their requirements, otherwise repeat the process with a different seed values.

- AortaGeomRecon Responsibilities:

    - Provide DICOM data reader which can take a path to a folder containing DICOM files.

- Provide crop functionality to easily select a region of interest.
- Provide simple interactions to obtain and store the users' inputs. This includes a data probe to read voxel location which stored as a coordinate, and text inputs for real numbers.
- Provide visualization on the result data.

3.2 User Characteristics

The end user of AortaGeomRecon should have taken the university level anatomy introduction course, and be capable of finding the center of the descending aorta and the ascending aorta.

3.3 System Constraints

There are no system constraints of AortaGeomRecon.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

The main purpose of AortaGeomRecon is to semi-automatically segment a 3D aorta geometry from a chest CT scan.

4.1.1 Organ Segmentation

Organs are the body's recognizable structures (for example, the heart, lungs, liver, eyes, and stomach) that perform specific functions. Figure 2 below shows all the organs within a human body. The organ segmentation or the organ boundary segmentation is useful for orientation and identification of the regions of interests inside the organ during the diagnostic or treatment procedure. The aorta segmentation is important for aortic calcification quantification and to guide the segmentation of other central vessels. Villa-Forte (2022)
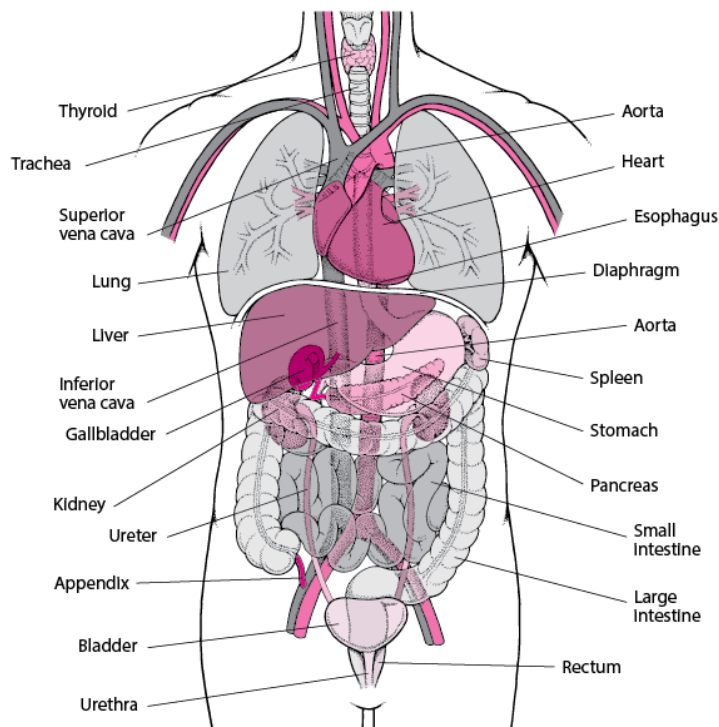
Figure 2: Human organs Villa-Forte (2022)

4.1.2 Coordinate Systems

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements. Nejad (2017)

While working with medical images, it is necessary to be familiar with the different coordinate systems of the medical literature and how data (voxels' orientation) is interpreted in different medical and nonmedical software. Each coordinate system uses one or more numbers (coordinates) to uniquely determine the position of a point (in the medical context, we refer to each point as a voxel). The purpose of this section is to introduce some coordinate systems related to the medical imaging. There are different coordinate systems to represent data. A knowledge of the following coordinate systems is needed to work with the medical images.

Cartesian Coordinate System A Cartesian coordinate system is a coordinate system that specifies each point uniquely in a 2D plane by a pair of numerical coordinates or in a 3D space by three numerical coordinates. We assume a right-hand Cartesian coordinate system throughout this document.

World Coordinate System   World Coordinate System (WCS) is a Cartesian coordinate system that describes the physical coordinates associated with a model such as an MRI scanner or a patient. While each model has its own coordinate system, without a universal coordinate system such as WCS, they cannot interact with each other. For model interaction to be possible, their coordinate systems must be transformed into the WCS. Figure 3 shows the WCS corresponding space and axes.
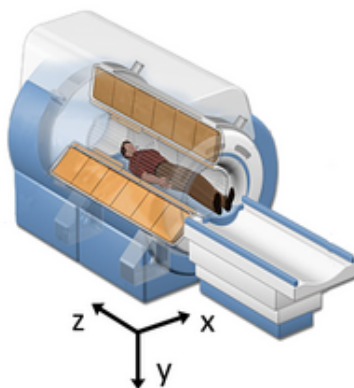
Figure 3: World Coordinate System Space and Axes sli (2014)

Anatomical Coordinate System   Anatomical coordinate system, also known as patient coordinate system, is a right-handed 3D coordinate system that describes the standard anatomical position of a human using the following 3 orthogonal planes:

- Axial / Transverse plane: is a plane parallel to the ground that separates the body into head (superior) and tail (inferior) positions.

- Coronal / Frontal plane: is a plane perpendicular to the ground that divides the body into front (anterior) and back (posterior) positions.

- Sagittal / Median plane: is a plane that divides the body into right and left positions.

Figure 4 shows this coordinate system.

(a) Anatomical Coordinate System
(b) Space and Axes
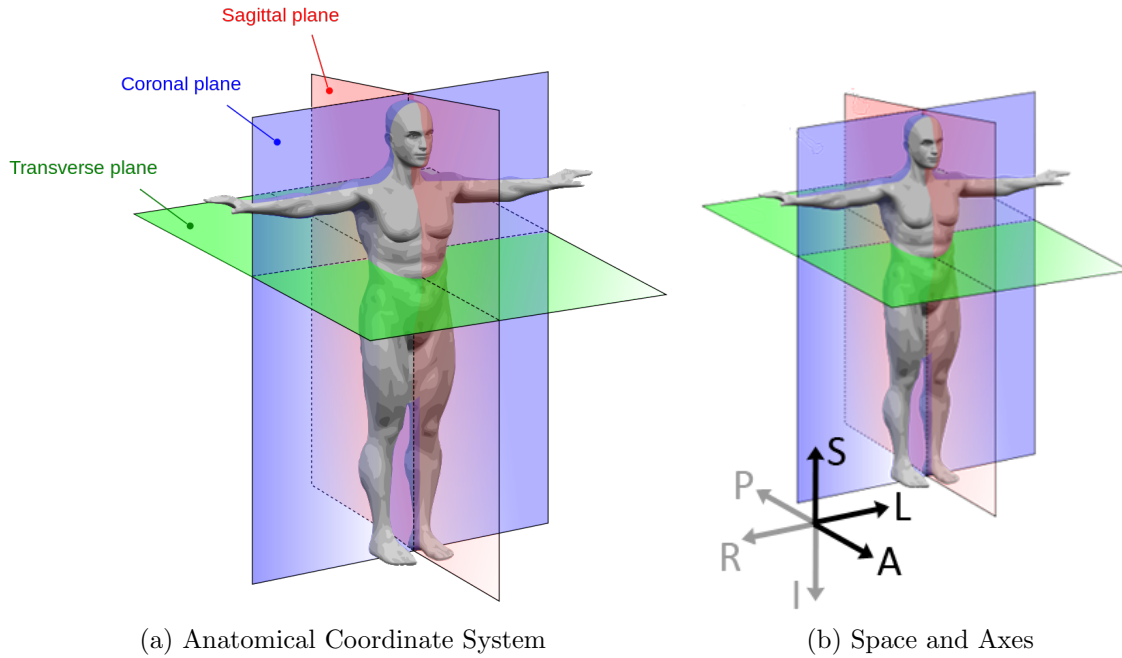
Figure 4: Anatomical Coordinate System Space and Axes sli (2014)

Medical applications follow an anatomical coordinate system to store voxels in sequences. Depending on how the data is stored, this coordinate system can be divided into different bases. The most common ones are:

- LPS Coordinate System:

  The LPS coordinate system is used in DICOM images. In this system, voxels are ordered from left to right in a row, rows are ordered from posterior to anterior, and slices are stored from inferior to superior. LPS stands for Left-Posterior-Superior which indicates the directions that spatial axes are increasing.

- RAS Coordinate System:

  The RAS coordinate system is the preferred basis for Neurological applications such as 3dfim+, and 3D Slicer. RAS stands for Right-Anterior-Superior is similar to LPS with the first two axes flipped.

Image Coordinate System To specify locations in an image we need to know to which coordinate system it is referenced. Different software may use different orders as their index convention.
Each of the coordinate systems mentioned above are used by different systems.

4.1.3 Physical System Description

We do not study the physical system for the images or how the data is actually generated.

4.1.4 Goal Statements

Given the DICOM image that includes patient's chest, the descending aorta center voxel coordinate, and the ascending aorta center voxel coordinate, the goal statements are:

GS1: Extract the three-dimensional segmentation of the aorta.

4.2 Solution Characteristics Specification

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the theoretical model [T], general definition [GD], data definition [DD], instance model [IM], or likely change [LC], in which the respective assumption is used.

A1: The 3D image provided by the user must contain a visually distingushable aorta volume [IM1].

A2: User should select a valid region of interest [IM2].

A3: User should input a singular volume (3 dimensional image) even if the data format supports the 4th dimension (time) [IM1].

4.2.2 Theoretical Models

There are no theoretical models used in this document.

4.2.3 General Definitions

There are no general definition used in this document.

4.2.4 Data Definitions

This section collects and defines all the data needed to build the instance models.

Number	DD1
Label	**Voxel**
Symbol	$v : \mathbb{R}$
SI Units	-
Equation	-
Description	A slice (DD2) consists of $n \times n$ voxels. A real number is assigned to each voxel to reports the intensity on a grey-scale image.
Sources	Nejad (2017)
Ref. By	DD2

Number	DD2
Label	**Image/Slice**
Symbol	$slice : \mathbb{R}^{m \times n}$
SI Units	-
Equation	-
Description	A visual representation that is using only two spatial dimensions with a sequence of arrays where a voxel (DD1) represents the color or intensity. Each move in the transverse plane (Figure 4) is considered as one slice
Sources	Nejad (2017)
Ref. By	DD3

Number	DD3
Label	**Volume**
Symbol	$V : \mathbb{R}^{m \times n \times p}$
SI Units	-
Equation	-
Description	A three-dimensional image is a sequence of some images/slices (DD2).
Sources	-
Ref. By	IM1

4.2.5 Data Types

There are no additional data types used in this document.

4.2.6 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.4 to replace the abstract symbols in the models. There are no theoretical models or general definitions used in this document.

The goals GS1 are solved by finding IM1 and perform IM2 on the aorta.

Number	IM1
Label	**Region of interest**
Inputs	$V_{\text{in}} : \mathbb{R}^{m_i \times n_i \times p_i}$, $Start : \mathbb{N}^3$, $m_o, n_o, p_o : \mathbb{N}$, with the following constraints: $$0 \leq Start[0] < (m_i - 1)$$ $$0 \leq Start[1] < (n_i - 1)$$ $$0 \leq Start[2] < (p_i - 1)$$ $$0 < m_o \leq (m_i - Start[0])$$ $$0 < n_o \leq (n_i - Start[1])$$ $$0 < p_o \leq (p_i - Start[2])$$
Output	$V_{\text{out}} : \mathbb{R}^{m_o \times n_o \times p_o}$ such that $$\forall (i, j, k : \mathbb{N} \mid$$ $$i \in [Start[0]..Start[0] + m_o] \wedge$$ $$j \in [Start[1]..Start[1] + n_o] \wedge$$ $$k \in [Start[2]..Start[2] + p_o] :$$ $$V_{\text{out}}[i][j][k] = V_{\text{in}}[i][j][k])$$
Description	The regions of interest is a subset (shaped like a box) of the 3D V_{out}. This subset contains the anatomical structure that the users wants to read, process or extract.
Sources	
Ref. By	IM2

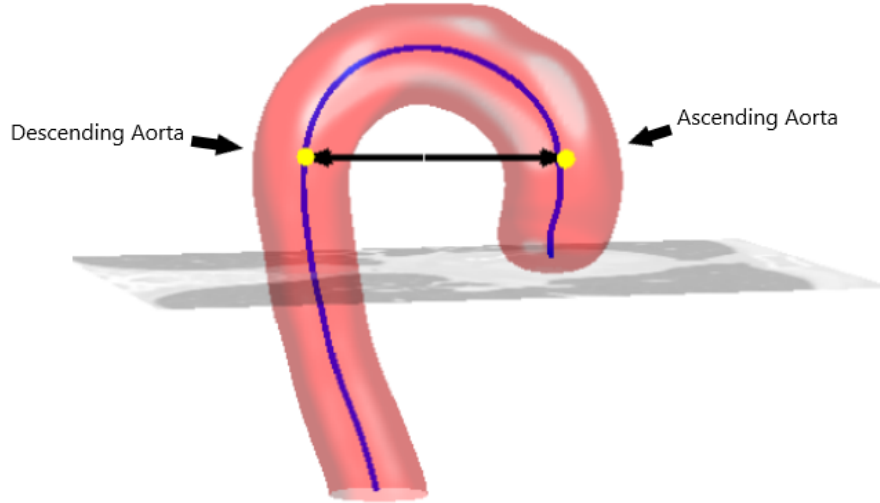| Number	IM2
Label	**Segmentation**
Input	$V_{\text{in}} : \mathbb{R}^{m \times n \times p}$, $Seed\_a : \mathbb{N}^3$, $Seed\_d : \mathbb{N}^3$
Output	$V_{\text{out}} : \mathbb{R}^{m \times n \times p}$ such that $$\forall (i, j, k : \mathbb{N} \mid$$ $$i \in [0..m-1] \land j \in [0..n-1] \land k \in [0..p-1]:$$ $$(V_{\text{in}}[i,j,k] \in \text{structure} \implies V_{\text{out}}[i,j,k] = HIGH \mid$$ $$V_{\text{in}}[i,j,k] \notin \text{structure} \implies V_{\text{out}}[i,j,k] = LOW))$$ The inputs $Seed\_a$ and $Seed\_d$ are used to determine whether a given element of V_{in} is in structure or not.
Description	The process of extract an anotomical structure from the original 3D volume. The extracted anotomical structure is represented with high intensity pixel value. The rest of the image should have a lower intensity pixel value. The segmentation needs the region of interset from IM1 to process less noise data. A seed is what the algorithm needed as the inputs to perform segmentation, the type of seed is different among different algorithm. The seeds in this section are the centre coordinate of the descending aorta and the ascending aorta. The yellow dots shown in Figure 5 are the example of the seed.
Sources	
Ref. By	R3, LC1

Figure 5: Aorta Seeds Kurugol et al. (2012)

4.2.7   Input Data Constraints

The only software constraint is the input volume data. It must be an acceptable file type to the system processing the data. For example, using ITK-Snap software to perform organ segmentation, the input data must be of VTK file.

4.2.8   Properties of a Correct Solution

A correct solution cannot be measured, but it can be confirmed by visually comparing the intersection of the extracted anotomical structure and the original volume.

5   Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1   Functional Requirements

R1: Input the following functions, data and parameters:

symbol	description
V	CT Scans volume (DD3)
$Seed\_a$	The seed of ascending aorta centre coordinate (IM2)
$Seed\_d$	The seed of descending aorta centre coordinate (IM2)

R2: Use the volume in R1 to create a second volume, the region of interest (IM1) that contains all voxels of the aorta.

R3: Perform segmentation (IM2) on the volume created in R2.

R4: Visualize a volume (DD3).

5.2   Nonfunctional Requirements

NFR1: **Usability** AortaGeomRecon allows a user that meets the user characteristics (Section 3.2) to import any DICOM files, input the required parameters, and begin the segmentation effortlessly. The number of steps it takes using AortaGeomRecon should be at least 30% less than the number of steps it takes by using ITK-Snap (bubble method mentioned in Section 2).

NFR2: **Safety** For a valid image, the AortaGeomRecon provides a correct solution, or no answer.

NFR3: **Learnability** The user interface and documentation should allow a user that meets the user characteristics (Section 3.2) to learn how to do an aorta segmentation in at least 30% of the time it takes to learn and use ITK-Snap (bubble method mentioned in Section 2).

NFR4: **Accuracy** For a given image the segmentation found by AortaGeomRecon should match that found by an expert using ITK-Snap. Whether to two segmentations match is something that would be judged by a medical imaging expert.

NFR5: **Consistency** The coordinate system may be modified through the calculations, but any transformations will not alter the meaning of the data.

Other NFRs that might be discussed in the future include verifiability, and reusability.

6 Likely Changes

LC1: IM2 There are various segmentation algorithms, each has a different procedure and inputs.

7 Unlikely Changes

UC1: IM1 The method to retrieve a region of interest from a volume is fixed.

8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 2 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 4 shows the dependencies of instance models, requirements, and data constraints on each other. Table 3 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

The purpose of the traceability graphs is also to provide easy references on what has to be additionally modified if a certain component is changed. The arrows in the graphs represent dependencies. The component at the tail of an arrow is depended on by the component at

the head of that arrow. Therefore, if a component is changed, the components that it points to should also be changed.

	DD1	DD2	DD3	IM1	IM2
DD1	X				
DD2	X	X			
DD3		X	X		
IM1			X	X	
IM2				X	X

Table 2: Traceability Matrix Showing the Connections Between Items of Different Sections

	A1	A2	A3
DD1			
DD2			
DD3			X
IM1	X		X
IM2		X	X
LC1	X	X	X
UC1			X

Table 3: Traceability Matrix Showing the Connections Between Assumptions and Other Items

	DD1	DD2	DD3	IM1	IM2	R1	R2	R3	R4	NFR1	NFR2	NFR3	NFR4	NFR5
DD1	X													
DD2	X	X												
DD3		X	X											
IM1			X	X										
IM2				X	X									
R1			X	X	X	X								
R2			X	X			X							
R3			X		X			X						
R4			X						X	X				
NFR1		X	X	X	X	X	X	X	X	X		X		
NFR2				X	X	X	X	X			X			
NFR3				X	X	X			X	X		X		
NFR4	X	X	X	X	X								X	
NFR5	X	X	X	X	X		X	X	X					X

Table 4: Traceability Matrix Showing the Connections Between Requirements and Instance Models

9 Referance

References

Coordinate systems, June 2014. URL https://www.slicer.org/wiki/Coordinate_systems.

Nirmitha Koothoor. A document drive approach to certifying scientific computing software. Master's thesis, McMaster University, Hamilton, Ontario, Canada, 2013. URL http://hdl.handle.net/11375/13266.

Sila Kurugol, Raul San Jose Estepar, James Ross, and George R. Washko. Aorta segmentation with a 3d level set approach and quantification of aortic calcifications in non-contrast chest ct. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2343–2346, 2012. doi: 10.1109/EMBC.2012.6346433.

Mojdeh Sayari Nejad. A case study in assurance case development for scientific software. Master's thesis, McMaster University, Hamilton, ON, Canada, July 2017. http://hdl.handle.net/11375/23075.

W. Spencer Smith and Lei Lai. A new requirements template for scientific computing. In J. Ralyté, P. Ȧgerfalk, and N. Kraiem, editors, *Proceedings of the First International Workshop on Situational Requirements Engineering Processes – Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP'05*, pages 107–121, Paris, France, 2005. In conjunction with 13th IEEE International Requirements Engineering Conference.

Alexandra Villa-Forte. Tissues and organs. https://www.merckmanuals.com/en-ca/home/fundamentals/the-human-body/tissues-and-organs, 2022.