**COMSATS University Islamabad,**

**Abbottabad Campus**

**SOFTWARE REQUIREMENTS SPECIFICATION   
(SRS DOCUMENT)**

**for**

**Liver Tumor Segmentation in CT Scan images Using Light Weight Deep Learning Model**  
Version 1.0

***By***

**Faizan CIIT/FA21-BSE-011/ATD**

**Fawad Iqbal CIIT/FA21-BSE-012/ATD**

***Supervisor*Dr. Mubashir Ahmad**

***Bachelor of Science in Computer Science (2024-2025)***

Table of Contents

[1. Introduction 5](#_Toc186322165)

[1.1 Purpose 5](#_Toc186322166)

[1.2 Scope 5](#_Toc186322167)

[2. Overall description 5](#_Toc186322168)

[2.1 Product perspective 5](#_Toc186322169)

[2.2 Operating environment 6](#_Toc186322170)

[2.3 Design and implementation constraints 6](#_Toc186322171)

[3. Requirement identifying technique 6](#_Toc186322172)

[3.1 Use case diagram 7](#_Toc186322173)

[3.2 Use case description 8](#_Toc186322174)

[Functional Requirements 12](#_Toc186322175)

[Functional Requirement X 13](#_Toc186322176)

[Non-Functional Requirements 16](#_Toc186322177)

[Usability 16](#_Toc186322178)

[Performance 16](#_Toc186322179)

[Reliability 16](#_Toc186322180)

[Scalability 17](#_Toc186322181)

[Security 17](#_Toc186322182)

[Maintainability 17](#_Toc186322183)

[Compatibility 17](#_Toc186322184)

[References 17](#_Toc186322185)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

**Application Evaluation History**

|  |  |
| --- | --- |
| **Comments (by committee)**  **\*include the ones given at scope time both in doc and presentation** | **Action Taken** |
|  |  |
|  |  |

**Supervised by**

**<Supervisor’s Name>**

Signature\_\_\_\_\_\_\_\_\_\_\_

# 1. Introduction

The **Software Requirements Specification (SRS)** document provides an organized framework for understanding the requirements and design of our liver tumor segmentation system. This system automates tumor detection in 2D CT scans using advanced medical image segmentation architectures. The project is to be accomplished with the help of deep learning techniques, specifically a hybrid approach of convolutional layers and attention mechanisms, to achieve accurate and efficient segmentation [1].

This document aims to facilitate communication among stakeholders, including developers, researchers, and potential users, ensuring that the system meets the specified requirements.

## 1.1 Purpose

The purpose of this project is to develop a liver tumor segmentation system that improves diagnostic processes by automating tumor detection. By using the nnFormer architecture [1], the system captures both local and global features [1], offering a more precise solution for liver tumor analysis.

## 1.2 Scope

The system will process 3D CT scans by converting them into 2D images to manage computational complexity. The architecture includes one core model: for liver and tumor segmentation. The frontend will allow doctors to upload images, view segmented outputs, and analyze tumor-related statistics.

# 2. Overall description

This section gives an overview of the system, explaining its environment, context, and any constraints we’ve considered during development.

## 2.1 Product perspective

Our liver tumor segmentation system is a standalone tool that combines a smart backend for image analysis and a user-friendly frontend for displaying results. The backend uses state-of-the-art architectures and deep learning models [1], which is specially chosen for its ability to accurately segment liver and tumor regions from CT images. To make things simple and easy to use, the frontend allows users to upload images, view segmentation results, and analyze important details like tumor size.

Since 3D processing is demanding, we’ve adapted the system to handle 2D image slices, making it more efficient while still maintaining good accuracy.

## 2.2 Operating environment

***OE-1****: The system shall operate correctly on general-purpose computers equipped with GPUs to handle computationally intensive tasks.*

***OE-2****: The backend shall function on Python 3.12 with PyTorch for deep learning operations and segmentation tasks.*

***OE-3****: The frontend shall be compatible with modern web browsers, including Google Chrome (all versions), Mozilla Firefox (latest versions), and Microsoft Edge (latest versions).*

***OE-4****: The system shall support the use of medical datasets, specifically formatted in .nii files, ensuring compatibility with the LITS17 dataset for both training and testing.*

## 2.3 Design and implementation constraints

***CO-1****: The system shall utilize nnFormer architecture for segmentation tasks, requiring adaptation from 3D CT scans to 2D slices to address resource constraints.*

***CO-2****: The backend shall strictly use PyTorch libraries due to their flexibility and compatibility with GPU-accelerated training and inference.*

***CO-3****: Data privacy must be ensured, limiting the use of external servers for sensitive medical information and processing data locally when possible.*

***CO-4****: Due to limited hardware resources, the system shall only support processing 2D image slices instead of full 3D volumes, maintaining acceptable accuracy levels.*

***CO-5****: All implementations shall comply with data security regulations, ensuring the confidentiality of medical datasets used during training and inference*

# 3. Requirement identifying technique

The process of identifying requirements for the liver tumor segmentation system involves a well-structured approach to ensure all functional and non-functional aspects are thoroughly captured and addressed. This method ensures no critical detail is overlooked, enabling the development of a robust and efficient system.

## 3.1 Use case diagram

A diagram of a company

Description automatically generated

*Figure 1: Use case diagram the Liver Tumor Segmentations using deep learning techniques, this diagram shows how doctors and technicians access the system.*

## 3.2 Use case description

#### Table 1: Login

|  |  |
| --- | --- |
| **Use Case ID** | **UC-1** |
| **Use Case Name** | Login |
| **Primary Actor** | Doctor |
| **Secondary Actor** | Technician |
| **Description** | This use case allows the Doctor or Technician to log into the system using their credentials. |
| **Trigger** | The user opens the system and initiates the login process. |
| **Preconditions** | 1. Users must be registered. 2. The system must be online. |
| **Postconditions** | 1. User gains access to the dashboard. 2. User session is initiated. |
| **Normal Flow** | 1. User opens the login page. 2. System displays the login form. 3. Users enter their username and password. 4. System verifies the credentials. 5. Upon success, user is redirected to the dashboard. |
| **Alternative Flows** | **1.0.E1 Invalid credentials entered:**   1. System informs the user of incorrect credentials. 2. User retries or chooses “Forgot Password”. |
| **Exceptions** | **1.0.E2 System maintenance:**   1. System informs users of unavailability |
| **Business Rules** | **BR-1:** Passwords must be encrypted. |
| **Assumptions** | 1. Users have valid credentials |

#### Table 2: Upload CT scan

|  |  |
| --- | --- |
| **Use Case ID** | **UC-2** |
| **Use Case Name** | Upload CT scan |
| **Primary Actor** | Doctor |
| **Secondary Actor** | Technician |
| **Description** | This use case enables the user to upload CT scan images for processing. |
| **Trigger** | The user selects the "Upload Image" option. |
| **Preconditions** | 1. User is logged in. 2. The CT scan file is accessible. |
| **Postconditions** | 1. Image is uploaded successfully. 2. The image is queued for analysis. |
| **Normal Flow** | 1. User clicks the "Upload CT Scan" button. 2. System opens a file picker. 3. User selects a file. 4. System validates the file format. 5. File is uploaded successfully. |
| **Alternative Flows** | **2.0.E1 Invalid file format:**   1. System notifies the user of unsupported format. |
| **Exceptions** | **2.0.E2 Network error:**   1. System notifies the user of upload failure. |
| **Business Rules** | **BR-1:** Only DICOM and NIFTI formats are accepted. |
| **Assumptions** | 1. User has the correct file format. |

#### Table 3: Analyze Image

|  |  |
| --- | --- |
| **Use Case ID** | **UC-3** |
| **Use Case Name** | Analyze Image |
| **Primary Actor** | Doctor |
| **Secondary Actor** | None |
| **Description** | This use case processes the uploaded CT scan image to identify and segment the liver and tumor regions. |
| **Trigger** | User clicks the “Analyze” button after uploading an image. |
| **Preconditions** | 1. Image must be uploaded successfully. 2. System analysis module is operational. |
| **Postconditions** | 1. Image is processed, and segmented regions are generated. 2. Segmentation masks are stored. |
| **Normal Flow** | 1. User clicks "Analyze" on the dashboard. 2. System retrieves the uploaded image. 3. Analysis module performs segmentation. 4. Results are stored and marked as complete. |
| **Alternative Flows** | **3.0.E1 Insufficient resources:**   1. System informs the user of processing delay. |
| **Exceptions** | **3.0.E2 Model failure:**   1. System notifies the user of analysis failure. |
| **Business Rules** | **BR-1:** Only valid uploaded images are processed. |
| **Assumptions** | 1. System has sufficient computational resources. |

#### Table 4: View Results

|  |  |
| --- | --- |
| **Use Case ID** | **UC-3** |
| **Use Case Name** | View Results |
| **Primary Actor** | Doctor |
| **Secondary Actor** | None |
| **Description** | This use case allows users to view segmented liver and tumor regions along with statistical data. |
| **Trigger** | User selects the "View Results" option after analysis is complete. |
| **Preconditions** | 1. Image analysis must be complete. 2. Results must be available in the database. |
| **Postconditions** | 1. Segmented image and related data are displayed to the user. |
| **Normal Flow** | 1. User clicks "View Results." 2. System fetches segmentation results. 3. System displays the segmented image and statistical data. |
| **Alternative Flows** | **4.0.E1 No results available:**   1. System informs the user that results are unavailable. |
| **Exceptions** | **4.0.E2 Database error:**   1. System notifies the user of an internal error. |
| **Business Rules** | **BR-1:** Results must include both visual and textual representations. |
| **Assumptions** | 1. User has completed the analysis step. |

# 4. Functional Requirements

This section focuses on detailing the functional requirements of the liver tumor segmentation system. Each functional requirement is written in a simple and clear natural language, organized by feature. The goal is to capture the software's capabilities that are essential for implementing each feature, ensuring that users can perform desired operations effectively.

## 4.1 Functional Requirement X

#### Table 5: FR-001 Preprocess Input CT scans

|  |  |
| --- | --- |
| **Identifier** | FR-001 |
| **Title** | Preprocessing Input CT scans |
| **Requirement** | The system shall preprocess 3D CT scan images into 2D slices of size 512x512 for segmentation using PyTorch. |
| **Source** | Research Workflow Design |
| **Rationale** | To ensure the usage of state-of-the-art methodologies, which shall help to reduce the computational complexity |
| **Business Rule** | We shall preserve the Global spatial data to ensure the model is trained on enough features for better results. |
| **Dependencies** | None |
| **Priority** | High |

#### Table 6: FR-002 Liver Segmentation Model

|  |  |
| --- | --- |
| **Identifier** | FR-002 |
| **Title** | Liver and Tumor Segmentation Model |
| **Requirement** | The system shall use a single nnFormer model to perform simultaneous segmentation of liver regions and tumours from 2D CT slices. |
| **Source** | Research Objective |
| **Rationale** | To simplify the segmentation workflow and improve efficiency by combining liver and tumor segmentation tasks into one model. |
| **Business Rule** | The combined segmentation model must achieve a Dice Similarity Coefficient (DSC) of at least 90% for liver and 85% for tumor segmentation. |
| **Dependencies** | FR-001 |
| **Priority** | High |

#### Table 7: FR-003 Patch-Based Analysis

|  |  |
| --- | --- |
| **Identifier** | FR-003 |
| **Title** | Patch-Based Analysis |
| **Requirement** | The system shall divide CT images into patches and adapt them for segmentation input to optimize performance on available hardware. |
| **Source** | Research Workflow Design |
| **Rationale** | To align with nnFormer’s patch-based processing and reduce GPU requirements. |
| **Business Rule** | Patches with no liver or tumor regions shall be excluded during segmentation. |
| **Dependencies** | FR-002 |
| **Priority** | High |

#### Table 8: FR-004 Display Segmentation Results

|  |  |
| --- | --- |
| **Identifier** | FR-004 |
| **Title** | Display Segmentation Results |
| **Requirement** | The system shall display the segmented liver and tumor regions on a graphical dashboard for the end-user. |
| **Source** | User Requirement |
| **Rationale** | |  | | --- | |  |  |  | | --- | | To provide a clear visualization of segmentation results. | |
| **Business Rule** | Segmentation outputs must be displayed within 10 seconds for a single scan. |
| **Dependencies** | FR-002 |
| **Priority** | High |

#### Table 9: FR-005 Analytical Insights

|  |  |
| --- | --- |
| **Identifier** | FR-005 |
| **Title** | Analytical Insights |
| **Requirement** | The system shall calculate, and display tumor size, location, and percentage of liver volume affected. |
| **Source** | Research Objective |
| **Rationale** | To provide quantitative insights for medical analysis. |
| **Business Rule** | Analytical results must be accurate within ±2% of actual measurements. |
| **Dependencies** | FR-002 |
| **Priority** | Medium |

# 5. Non-Functional Requirements

#### 5.1 Usability

***USE-1:*** *The system shall allow users to upload a 2D CT image and receive segmented liver and tumor masks within 10 seconds.****USE-2:*** *The UI shall provide intuitive navigation between image upload, result display, and analysis dashboard.****USE-3:*** *The system shall provide clear error messages for invalid image uploads, enabling easy recovery.****USE-4:*** *The system shall allow zooming on segmented results with at least 90% clarity.*

#### 5.2 Performance

***PER-1:*** *The system shall process and return segmentation results for a 512x512 CT image within 3 minutes.****PER-2:*** *95% of user interactions (e.g., uploading, result display) shall be completed within 5 seconds.****PER-3:*** *The system shall support at least 3 concurrent image uploads without performance degradation.****PER-4:*** *The segmentation process shall be optimized to complete 95% of tasks within 10 minutes.*

#### 5.3 Reliability

***REL-1:*** *The system shall maintain 99.5% uptime with no downtime exceeding 3 hours per month.****REL-2:*** *In case of system failure, the process shall automatically recover without data loss.****REL-3:*** *The segmentation model shall provide consistent results with less than 5% variance.*

#### 5.4 Scalability

***SCA-1:*** *The system shall support 100 concurrent users without noticeable degradation in performance.****SCA-2:*** *The infrastructure shall handle a 50% increase in processing requests during peak times.*

#### 5.4 Security

***SEC-1:*** *All data (CT images and results) shall be encrypted during transmission and storage.****SEC-2:*** *The system shall comply with relevant data protection regulations (e.g., HIPAA, GDPR).****SEC-3:*** *Only authorized users shall access the system’s image upload and result viewing features.*

#### 5.5 Maintainability

***MAI-1:*** *The system shall be modular to facilitate easy updates to the segmentation model and UI.****MAI-2:*** *The system shall include comprehensive documentation for the codebase and AI models.****MAI-3:*** *The system shall include automated tests for segmentation performance and UI stability.*

#### 5.6 Compatibility

***COM-1:*** *The system shall be compatible with the latest versions of Chrome, Firefox, Safari, and Edge.****COM-2:*** *The system shall support Windows, macOS, and Linux platforms.****COM-3:*** *The system shall support devices with a screen resolution of at least 1280x720 pixels.*

# 6. References

[1]. <https://arxiv.org/pdf/2109.03201v6>