Software Requirement Specification (SRS)

Automated Quality Inspection Using Computer Vision

Submitted by:
Shikhar Shashank (2301199), Prince Goyal (2301158), Samadrita Mondal (2301185)
Computer Science and Engineering
Indian Institue of Information Technology, Guwahati

Academic Year: 2025–2026

Contents

1 Introduction **2**
- 1.1 Purpose of this Document . . . 2
- 1.2 Scope of this Document . . . 2
- 1.3 Overview . . . 2

2 General Description **2**

3 Functional Requirements **2**

4 Interface Requirements **3**
- 4.1 User Interface . . . 3
- 4.2 Software Interface . . . 3
- 4.3 Hardware Interface . . . 3

5 Performance Requirements **3**
- 5.1 Static Requirements . . . 3
- 5.2 Dynamic Requirements . . . 4

6 Design Constraints **4**

7 Non-Functional Attributes **4**

8 Preliminary Schedule and Budget **4**

9 Appendices **5**

10 Uses of SRS Document **5**

11 Conclusion **5**

1. Introduction

1.1 Purpose of this Document

The main aim of this document is to describe the Software Requirement Specification (SRS) for the Automated Quality Inspection system using Computer Vision. This document defines the purpose, scope, functional requirements, non-functional requirements, constraints, and interfaces of the system. It serves as a reference for developers, testers, project managers, and stakeholders.

1.2 Scope of this Document

This document describes the overall working and main objective of the proposed system. The system aims to automate the inspection of products using computer vision techniques, reducing manual errors and inspection time. It also explains the value provided to customers by improving accuracy and efficiency. The scope includes an overview of development cost and estimated time required to complete the project.

1.3 Overview

This section provides a summary and overall review of the product. The Automated Quality Inspection System captures product images, processes them using computer vision algorithms, and identifies defects automatically. The system enhances manufacturing quality control processes by ensuring consistency and reliability.

2. General Description

This section describes the general functions of the product including user objectives, user characteristics, features, benefits, and importance of the system. It also explains the characteristics of the user community.

The system allows users to upload or capture images of products, analyze them for defects, and generate inspection results. The primary users include operators, quality engineers, and administrators. The system improves inspection accuracy, reduces inspection cost, and minimizes human intervention.

3. Functional Requirements

This section explains the expected behavior of the system and its outcomes. Functional requirements specify the relationship between inputs and outputs of the system and are arranged in ranked order.

- The system shall accept product images as input.
- The system shall preprocess images to improve quality.
- The system shall analyze images to detect defects.
- The system shall classify products as defective or non-defective.
- The system shall highlight defective regions in the image.
- The system shall store inspection results for future reference.

Each requirement specifies the data inputs, valid input ranges, processing actions, and generated outputs.

4. Interface Requirements

This section describes how the software communicates with users and other software components.

4.1 User Interface

The user interface allows users to upload images, view inspection results, and analyze defect regions visually.

4.2 Software Interface

The system interfaces with image processing libraries and machine learning frameworks for defect detection.

4.3 Hardware Interface

The system interfaces with cameras for image acquisition and optional GPU hardware for faster processing.

5. Performance Requirements

This section describes how the system performs under specific conditions.

5.1 Static Requirements

Static requirements include system constraints that do not affect execution behavior, such as supported platforms and image formats.

5.2 Dynamic Requirements

Dynamic requirements specify execution behavior, such as:

- Image processing time shall not exceed 2 seconds per image.
- System accuracy shall be at least 90%.
- Memory usage shall remain within defined limits.

6. Design Constraints

This section describes limitations and restrictions imposed on the design of the system. Constraints include hardware limitations, software dependencies, and standards that must be followed. The system must operate within available computational resources and comply with security and reliability requirements.

7. Non-Functional Attributes

The non-functional requirements include:

- Security
- Reliability
- Scalability
- Portability
- Data Integrity
- Application Compatibility
- Reusability

These attributes ensure better performance and usability of the system.

8. Preliminary Schedule and Budget

This section describes the initial project plan, estimated timeline, and development cost. The project is expected to be completed within the specified academic duration with minimal hardware and software cost due to the use of open-source tools.

9. Appendices

This section includes additional information such as references, definitions of technical terms, acronyms, and abbreviations used throughout the document.

10. Uses of SRS Document

- Development teams use it to build the system according to requirements.
- Test plans are created based on the described external behavior.
- Maintenance teams use it to understand system functionality.
- Project managers use it for planning and estimation.
- Customers rely on it to understand expected product behavior.
- Acts as a contract between developer and customer.
- Serves documentation purposes.

11. Conclusion

Software development requires a well-structured Software Requirement Specification (SRS). The SRS helps stakeholders communicate effectively, guides developers and testers, supports maintenance activities, and assists project management. A clear SRS ensures that the software meets functional and non-functional requirements, resulting in a quality product delivered on time and within budget.