

# **Software Requirement Specification**

## **(SRS)**

Automated Quality Inspection Using Computer Vision

**Submitted by:**

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

**Academic Year:** 2025–2026

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Purpose of this Document . . . . .	2
1.2	Scope of this Document . . . . .	2
1.3	Overview . . . . .	2
<b>2</b>	<b>General Description</b>	<b>2</b>
<b>3</b>	<b>Novel Features of the Proposed System</b>	<b>3</b>
3.1	Human-in-the-Loop Inspection . . . . .	3
3.2	Self-Supervised Anomaly Detection . . . . .	3
3.3	Explainability and Trustworthy AI . . . . .	3
3.4	Online Continual Learning . . . . .	3
<b>4</b>	<b>Functional Requirements</b>	<b>3</b>
<b>5</b>	<b>Interface Requirements</b>	<b>4</b>
5.1	User Interface . . . . .	4
5.2	Software Interface . . . . .	4
5.3	Hardware Interface . . . . .	4
<b>6</b>	<b>Performance Requirements</b>	<b>4</b>
6.1	Static Requirements . . . . .	4
6.2	Dynamic Requirements . . . . .	4
<b>7</b>	<b>Design Constraints</b>	<b>4</b>
<b>8</b>	<b>Non-Functional Attributes</b>	<b>5</b>
<b>9</b>	<b>Preliminary Schedule and Budget</b>	<b>5</b>
<b>10</b>	<b>Appendices</b>	<b>5</b>
<b>11</b>	<b>Uses of SRS Document</b>	<b>5</b>
<b>12</b>	<b>Conclusion</b>	<b>5</b>

# **1. Introduction**

## **1.1 Purpose of this Document**

The purpose of this Software Requirement Specification (SRS) document is to describe the functional and non-functional requirements of the Automated Quality Inspection (AQI) system using Computer Vision. This document serves as a reference for developers, testers, project managers, and stakeholders to understand system capabilities, constraints, and expected behavior.

## **1.2 Scope of this Document**

The proposed system automates the inspection of manufactured products using computer vision and machine learning techniques. Unlike traditional inspection systems, the proposed solution incorporates self-supervised anomaly detection, explainable AI, human-in-the-loop feedback, and online continual learning to adapt to evolving defect patterns and reduce dependency on labeled defect data.

## **1.3 Overview**

The Automated Quality Inspection System captures product images using industrial cameras, analyzes them using deep learning-based anomaly detection models, and identifies defective regions. The system allows human inspectors to review results, provide feedback, and improve system performance over time through continual learning.

# **2. General Description**

The system is designed to assist manufacturing quality control by providing accurate, adaptive, and trustworthy defect detection. It supports collaboration between automated inspection models and human inspectors to ensure reliability in real-world industrial environments.

Primary users include:

- Quality control operators
- Manufacturing engineers
- System administrators

The system improves inspection consistency, reduces inspection time, and adapts to new defect types without extensive retraining.

### **3. Novel Features of the Proposed System**

#### **3.1 Human-in-the-Loop Inspection**

The system incorporates a human-in-the-loop mechanism where inspectors can validate detected defects, correct false positives, and mark missed defect regions. This feedback is stored and used to refine the model, enabling continuous improvement and increased trust in automated decisions.

#### **3.2 Self-Supervised Anomaly Detection**

Instead of relying on labeled defect samples, the system learns the visual patterns of defect-free products using self-supervised learning. Any deviation from learned normal patterns is flagged as a potential defect, allowing effective detection even with limited or no defect data.

#### **3.3 Explainability and Trustworthy AI**

The system provides visual explanations such as heatmaps highlighting defective regions and confidence scores for predictions. This improves transparency, enables better human understanding of model decisions, and supports use in safety-critical manufacturing environments.

#### **3.4 Online Continual Learning**

The proposed system supports online continual learning by incorporating validated human feedback into incremental model updates. This allows the system to adapt to new defect types and changing manufacturing conditions without complete retraining.

### **4. Functional Requirements**

- The system shall capture or accept product images as input.
- The system shall preprocess images for noise reduction and normalization.
- The system shall perform self-supervised anomaly detection on input images.
- The system shall identify and localize anomalous (defective) regions.
- The system shall generate visual explanations for detected defects.
- The system shall provide confidence scores for inspection results.
- The system shall allow human inspectors to validate or correct results.

- The system shall store human feedback and inspection history.
- The system shall update its model incrementally using validated feedback.
- The system shall classify products as defective or non-defective.

## 5. Interface Requirements

### 5.1 User Interface

The user interface allows operators to upload or capture images, visualize detected defects with explanations, provide feedback, and review inspection history.

### 5.2 Software Interface

The system interfaces with computer vision libraries, machine learning frameworks, and databases for model inference, training, and data storage.

### 5.3 Hardware Interface

The system interfaces with industrial cameras and optional GPU hardware for accelerated processing.

## 6. Performance Requirements

### 6.1 Static Requirements

- The system shall support common image formats (JPEG, PNG).
- The system shall run on standard workstation or edge devices.

### 6.2 Dynamic Requirements

- Image inspection time shall not exceed 2 seconds per image.
- Defect detection accuracy shall be at least 90%.
- The system shall handle continuous image streams without failure.

## 7. Design Constraints

The system must operate within limited computational resources and utilize open-source software tools. It must ensure data security, reliability, and compliance with standard software engineering practices.

## **8. Non-Functional Attributes**

- Reliability
- Scalability
- Security
- Explainability
- Maintainability
- Portability
- Data Integrity

## **9. Preliminary Schedule and Budget**

The project is planned to be completed within the academic semester. Open-source frameworks and minimal hardware requirements ensure low development and deployment cost.

## **10. Appendices**

This section contains definitions, acronyms, and additional information relevant to the system.

## **11. Uses of SRS Document**

- Serves as a reference for system development and testing
- Guides system validation and verification
- Assists project planning and estimation
- Acts as a formal agreement between stakeholders

## **12. Conclusion**

This SRS defines the requirements for a novel Automated Quality Inspection system that integrates self-supervised learning, explainable AI, human-in-the-loop feedback, and continual learning. The proposed system addresses key limitations of existing inspection solutions and enables adaptive, transparent, and reliable quality control in manufacturing environments.