USER REQUIREMENTS SPECIFICATION for K00259724 FYP

TUS

Technological University of the Shannon: Midlands Midwest

Ollscoil Teicneolaíochta na Sionainne: Lár Tíre Iarthar Láir

Contents

REVISION HISTORY

Rev.	**Date**	**Approval**	REVISION SUMMARY
A0	05/11/23	__________	Initial Draft – Cormac Farrelly

Document Description:	AI Powered Industrial Automated Part Sorting System.
Status:	For Integration.
Client:	TUS.
Doc Name/Nr:	URS_K00259724_CF

1.0 INTRODUCTION

This User Requirements Specification (URS) has been developed under the guidance of my project supervisor, Jeffery Ryan, to define the foundational requirements for the successful implementation of an AI powered Industrial Part Sorting System. The core purpose of this project is to classify industrial parts, designed and manufactured for testing, into three distinct categories: damaged, undamaged, and unrecognized. These classifications will be made as the parts traverse a conveyor belt driven by a servo motor, integrated with a Siemens S7 1200 PLC and a supporting HMI (Human-Machine Interface). The algorithm, essential to this project, will reside on a Raspberry Pi 4 and will analyze images captured by a Cognex camera or a compatible alternative.

This URS is a critical component of the project, ensuring that the development and integration of these components are carried out in accordance with the specified requirements. It serves as a reference for all project stakeholders, including myself and potential suppliers or collaborators.

The project's operational workflow involves releasing parts onto the conveyor belt while an encoder measures their movement. As each part passes under the camera, an image is captured and sent to the Raspberry Pi for classification. The resulting binary output is transmitted to the PLC, instructing the sorting of parts into their respective collection chambers based on their condition. This process continues until all parts have been processed, which is determined by sensors on the part collection and release mechanisms.

In summary, this URS forms the cornerstone of this final year project, providing clear guidelines and expectations for the development and implementation of the machine learning algorithm and associated hardware components.

2.0 OVERVIEW

The industrial automation and robotics system designed for this final year project is a comprehensive integration of hardware and software components aimed at classifying industrial parts into three distinct categories: damaged, undamaged, and unrecognized. The system is primarily composed of a conveyor belt driven by a servo motor, a Siemens S7 1200 PLC, a Human-Machine Interface (HMI), a Raspberry Pi 4 for machine learning algorithm processing, and a Cognex camera or compatible alternative for image capture.

The operational cycle of this system can be summarized as follows:

- Part Measurement: As the parts move on the conveyor belt, an encoder measures their position.
- Image Capture and Classification: When a part is positioned under the camera, an image is captured and sent to the Raspberry Pi, where the custom machine learning algorithm processes it and classifies the part as damaged, undamaged, or unrecognized.
- PLC Control: The classification result is translated into binary output, which is then communicated to the Siemens S7 1200 PLC.
- Sorting and Collection: Based on the PLC's instructions, the parts are directed to their respective collection chambers, aligned with their condition.
- Continuous Operation: This process repeats until all the parts have been processed, with sensors on the part collection and release mechanisms determining when all parts have been classified and sorted.
- Safety Measures: Any safety breaches or irregularities will trigger an immediate halt in the system's operation.

3.0 OPERATIONAL REQUIREMENTS

3.1 Commodities

Commodity Tables

Industrial Part type	**Color**	**Classification Rate (P/Min)**
3D PRINTED	GREY	2

3.2 Performance

3.2.1 ML algorithm

3.2.1.1 The custom machine learning algorithm, integrated into the industrial automation system, is expected to accurately classify industrial parts in a timely manner. The system will need to check each part's condition as it moves on the conveyor belt. The algorithm's ability to differentiate between "damaged," "undamaged," and "unrecognized" parts will be a critical performance metric. Each part's classification will be vital for directing it to the appropriate collection chamber. This entire process will constitute one cycle of the system.

3.2.2 The robot shall be operable by one person.

3.2.2.1 Safety-related faults shall stop the machine immediately. The machine shall not be allowed to restart without operator intervention in the form of a reset button, which will resetthe fault only when the cause has been eliminated.

3.2.2.2 The vision system must be capable of collecting and sending images to the RaspberryPi accurately, regardless of their order on the conveyor belt. This ability will ensure the correct sorting of parts into their respective categories and enable the smooth operation of the entire system.

3.3 Functions

3.3.1 Image Capture and Processing Functions:

3.3.1.1 Capture Image: Capture an image of the industrial part when it reaches a specific location on the conveyor belt and send it to the RaspberryPi.

3.3.1.2 Image Preprocessing: Apply necessary preprocessing techniques to enhance the quality of captured images within the algorithm.

3.3.1.3 Feature Extraction: Extract relevant features from the captured image for classification.

3.3.2 Machine Learning Functions:

3.3.2.1 Model Training: Train the custom machine learning algorithm (Resnet50V2) using TensorFlow and Keras based on the pre-processed image data.

3.3.2.2 Classification: Use the trained model to classify each industrial part into one of three categories: damaged, undamaged, or unrecognized.

3.3.3 Communication and Control Functions:

3.3.3.1 Data Transfer: Transmit the classification output as binary data to the Siemens S7 1200 PLC using C on Pi.

3.3.3.2 PLC Control: Control the movement and sorting of parts based on the classification result within the part collection station.

3.3.3.3 Part Collection: Coordinate the collection of parts into their respective chambers.

3.3.4 System Operation Functions:

3.3.4.1 Conveyor Movement: Control the conveyor belt, ensuring that parts move under the camera at the appropriate time by making use of an encoder.

3.3.4.2 Encoder Monitoring: Monitor the position of parts on the conveyor using an encoder.

3.3.4.3 Sensor Integration: Integrate sensors to detect when the part collection and release areas are full or empty.

3.4 Power Failure and Recovery

3.4.1 System Recovery on Power Failure:

3.4.1.1 Safe State on Power Failure: In the event of a power failure, the system will transition into a safe state.

3.4.1.2 Stop All Motion: All system motion, related to the conveyor and image capture, will cease immediately.

3.4.1.3 Reset Required: To restart the system after power failure, a manual reset operation is necessary.

3.4.1.4 Prevent Damage: The system is designed to ensure that no damage occurs due to the transition into a safe state.

3.4.2 Power Restoration:

3.4.2.1 Operator Input Required: The system will not automatically restart upon power restoration; operator input is essential.

3.4.3 Data and Program Retention:

3.4.3.1 Program Retention: All equipment shall be designed to retain the machine learning algorithm and PLC program in case of power loss.

3.4.3.2 Minimal Operator Actions: Recovering from power loss shall require minimal operator actions.

3.5 Emergency Stop

3.5.1 Emergency Stop Buttons

3.5.1.1 Emergency Stop Buttons: Emergency stop buttons shall be provided within reach for the operator.

3.5.1.2 Activation Consequences: When activated, the emergency stop shall immediately halt all system motion.

3.5.1.3 No Damage: Activation of the emergency stop shall not result in any damage to the system.

3.5.1.4 Operator Intervention: The system cannot restart without operator intervention and a reset operation.

3.5.1.5 Device Power Disconnect: All emergency stops shall be hardwired to disconnect device power.

3.5.1.6 HMI Alarm: An alarm will be displayed on the HMI to inform the operator of the emergency stop activation.

3.6 Alarms and Warnings

3.6.1 System Alarms

3.6.1.1 Image Capture Error: An alarm will be triggered if there's an issue with image capture.

3.6.1.2 Algorithm Fault: An alarm will be activated if there's a fault in the machine learning algorithm.

3.6.1.3 Conveyor Belt Error: An alarm for conveyor belt issues.

3.6.1.4 PLC Communication Error: Alarm for communication faults between the Raspberry Pi and PLC.

3.6.2 Operator Messages

3.6.2.1 Operator Guidance: Display messages on the HMI to inform the operator about alarms and provide instructions for resolution.

3.7 Security

3.7.1 User Access Control

3.7.1.1 User Authorization: Define user roles with appropriate authorities to prevent unauthorized access.

3.7.1.2 Default User Access: The default user at system startup will have access to essential controls and information screens.

3.7.2 Locking Capability

3.7.2.1 External Locking: Control panels housing critical components shall have the capability of being locked externally to prevent tampering.

3.8 Interfaces

3.8.1 Operator Interface

3.8.1.1 HMI Control: The Human Machine Interface (HMI) shall provide control over machine learning and automation functions.

3.8.1.2 Alarm Display: Display alarms and warnings related to the machine controls.

3.8.1.3 Operator Guidance: Provide explanatory notes on HMI functions for better operator understanding.

3.8.1.4 System State: Display the current system state (RUN, STOP, or FAULT) on the HMI.

3.9 NON-FUNCTIONAL REQUIREMENTS

3.10 Performance

3.10.1 Algorithm Efficiency

3.10.1.1 The machine learning algorithm should demonstrate a high accuracy rate in classifying industrial parts into damaged, undamaged, and unrecognized categories.

3.10.1.2 The algorithm should be capable of processing and classifying a significant number of parts in a specified time frame, such as at least two parts per minute.

3.10.1.3 Performance Testing: Evaluation of the algorithm should include testing with a large dataset to assess accuracy and efficiency in real-world conditions.

3.11 Scalability

3.11.1 System Adaptability

3.11.1.1 The system's architecture should be designed with modularity and flexibility, allowing for the integration of additional image capture devices or improvements in the machine learning model.

3.11.1.2 Consideration for Future Expansion: The system's design should account for potential future enhancements to accommodate evolving requirements.

3.12 Availability

3.12.1 System Reliability

3.12.1.1 The system shall maintain a high level of availability, ensuring uninterrupted operation during specified testing periods.

3.12.1.2 Fast Startup: The system should have a quick startup time, not exceeding 2 minutes from power-on to being fully operational, including the initialization of servo motors and image capture devices.

3.13 Recoverability

3.13.1 Fault Tolerance

3.13.1.1 The system should be capable of recovering from errors, alarms, hardware, or software failures.

3.13.1.2 Mechanical Failures: In case of mechanical issues or component failures, the system should have mechanisms for automatic recovery or alerting the operator for intervention.

3.14 Maintainability

3.14.1 System Maintenance

3.14.1.1 The system's components and software should be designed for ease of maintenance.

5.0 COMPATIBILITY AND UTILITIES

5.1 System Compatibility

5.1.1 Software Compatibility

5.1.1.1 The system shall be meticulously crafted to work with commonly used software tools in the development of machine learning algorithms. This includes Python, TensorFlow, and Keras, ensuring that these essential components interact flawlessly.

5.1.1.2 The embedded software shall adhere to industry best practices in software development. It will be designed to ensure compatibility with standard libraries and guidelines for machine learning in Python.

5.1.2 Preferred Manufacturers

5.1.2.1 The system's design will prioritize compatibility including cameras and sensors suitable for industrial image capture and processing.

5.2 Utility Requirements

5.2.1 Power Supply

5.2.1.1 The system's energy needs shall be met through the utilization of

standard electrical services. This includes a 230-volt, 1-phase, 60 Hz power supply with proper grounding, ensuring a stable and reliable power source.

6.0 AVAILABILITY

6.1 Operational Schedule

6.1.1 System Usage

6.1.1.1 The system is designed for educational and research purposes. It is anticipated to be operational during designated periods, facilitating experiments and research.

7.0 PROCEDURAL CONSTRAINTS

7.1 Regulatory Compliance

7.1.1 Safety and Regulatory Standards

7.1.1.1 Adherence to safety and quality standards will be part of the system's design and operation, ensuring it complies with established regulatory norms like GAMP, cGMP, GMP.

8.0 GLOSSARY

Table 1.0 Glossary	
Acronym	**Definition**
cGMP	common Good Manufacturing Practices
GAMP	Good Automation Manufacturing Practices
HMI	Human Machine Interface
PLC	Programmable Logic Controller
URS	User Requirement Specification
AI	Artificial Intelligence

9.0 APPROVAL

This document has been reviewed by the Project Supervisor and approved for use by the Supplier.

Printed/Typed Name	Signature	Date

This document has been reviewed by the Suppliers / Developers and approved for use by the Supplier Project Team.

Printed/Typed Name	Signature	Date