Quality Analysis using Image Processing of Real Time Data in Industry 4.0

**Software Design Specification**

Prepared by: Zohair Lokhandwala & Sibte Abbas

Table of Contents

1. Introduction 1

1.1 Purpose of this document 1

1.2 Scope of the development project 1

1.3 Definitions, acronyms, and abbreviations 1

1.4 References 1

1.4 Overview of document 1

2. System architecture description 1

2.1 Section Overview 1

2.2 General Constraints 1

2.3 Data Design 1

2.4 Program Structure 2

3. Detailed Description of Components 2

4. User Interface Design 3

4.1 Section Overview 3

4.2 Interface Design Rules 3

4.3 GUI Components 3

4.4 Detailed Description 3

5. Reuse and relationships to other products 3

6. Design decisions and tradeoffs 3

7. Pseudocode for components 3

8. Appendices 3

**Software Design Specification**

**1. Introduction**

**1.1 Purpose of this document**

The aim of this document is to a specify a high-level view of the architecture of our system, and the interaction between the user and the system. It also focuses on detailing a low-level view of each component of the software and how the components interact with each other.

**1.2 Scope of the development project**

The following system will be developed to make this whole process to be made automated, accurate and fast image processing system which could identify surface defects of gear, select gear with required dimensions (Gold plating) and count the number of teeth.

**1.3 Definitions, acronyms, and abbreviations**

N/A

**1.4 References**

N/A

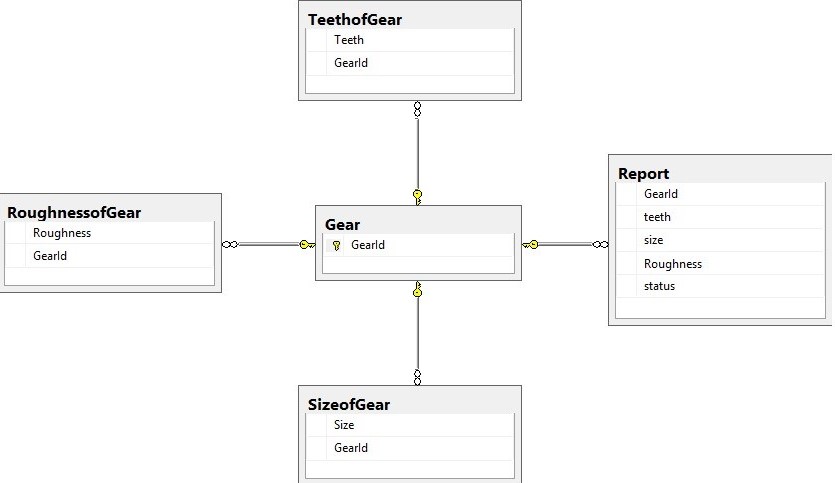
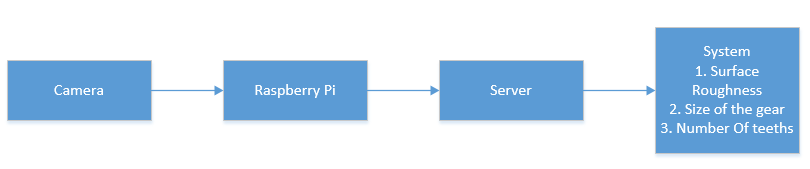
**1.5 Overview of document**

System architecture section provides an overview of the system's major components and architecture, as well as specifications on the interaction between the system and the user. The Detailed description of components section will be the main focus.  It will describe lower-level classes, components, and functions, as well as the interaction between these internal components. The user interface section will provide the GUI’s of the actual product and the appendices section will contain all the UML diagrams of the system.

**2. System architecture description**

* 1. **Section Overview**  
     This section provides the overall summary of our system architecture.
  2. **General Constraints**

This system will be used by the operator of the gear hobbing machine to check the quality of each gear in real-time. This system will be developed on Python using the OpenCV library. It will require a computer and an internet connection to operate. The functionality of our system is constrained by data provided via the cameras on the machine. The internet should be up and running all the time so the data can be uploaded to the sever. Our System relies on the cameras functioning properly and would shut down if that were not the case.

* 1. **Data Design**  
     
  2. **Program Structure**  
     

1. **Detailed description of components**

|  |  |
| --- | --- |
| Identification | Camera |
| Type | Hardware |
| Purpose | Capture Images of the Gear |
| Function | Capture and store images in raspberry pi |
| Subordinates | ------------------------------------------------------------------------------------------------ |
| Dependencies | The camera should be connected to the power and the raspberry pi to store the captured images. |
| Interfaces | ------------------------------------------------------------------------------------------------ |
| Resources | Camera and the Raspberry Pi |
| Processing | ------------------------------------------------------------------------------------------------ |
| Data | Image Files |

|  |  |
| --- | --- |
| Identification | Raspberry Pi |
| Type | Hardware |
| Purpose | Store Images of the Gear |
| Function | Store the captured images received from the camera |
| Subordinates | ------------------------------------------------------------------------------------------------ |
| Dependencies | It should be connected to the power and the camera to store the captured images and also to the server to send image files to the system. |
| Interfaces | ------------------------------------------------------------------------------------------------ |
| Resources | Camera, Raspberry Pi, Server |
| Processing | ------------------------------------------------------------------------------------------------ |
| Data | Image Files |

|  |  |
| --- | --- |
| Identification | Server |
| Type | Hardware |
| Purpose | Send Data |
| Function | Receive the data from the raspberry pi and send it to the system through server |
| Subordinates | ------------------------------------------------------------------------------------------------ |
| Dependencies | It should be connected to the power and the raspberry pi |
| Interfaces | ------------------------------------------------------------------------------------------------ |
| Resources | System, Raspberry Pi, Server |
| Processing | ------------------------------------------------------------------------------------------------ |
| Data | Image Files |

|  |  |
| --- | --- |
| Identification | System |
| Type | Hardware, Desktop Application |
| Purpose | Process the images |
| Function | The images received from the server will be processed |
| Subordinates | ------------------------------------------------------------------------------------------------ |
| Dependencies | It should be connected to the power and the server to receive the images of the gear. |
| Interfaces | ------------------------------------------------------------------------------------------------ |
| Resources | System, Server |
| Processing | ------------------------------------------------------------------------------------------------ |
| Data | Image Files |

1. **User Interface Design**
   1. **Section Overview**  
      This section provides the detailed description about user interface and its contraints.
   2. **Interface Design Rules**  
      N/A
   3. **GUI Components**  
      N/A
   4. **Detailed Description**  
      We have chosen Python language with PyCharm IDE and SQL database to implement our project because of its more interactive support and save the reports. The user will interact with a keyboard and a mouse if he is using a computer to type information and move the cursor. User can also interact using a touch screen if he/she is using a tablet or a mobile phone. Camera will be used for capturing images. Raspberry Pi acts as an intermediate device for sending the data collected from these units to the server for further processing.

**5.0 Reuse and relationships to other products**

N/A

**6.0 Design decisions and tradeoffs**

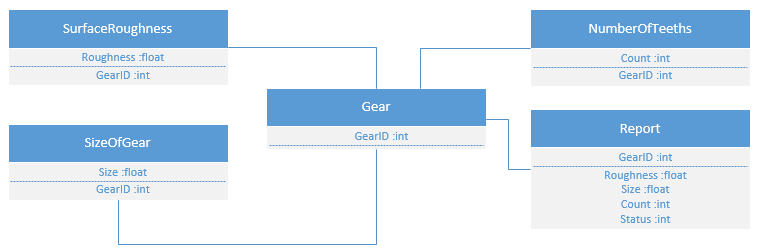
N/A

**7.0 Pseudocode for components**

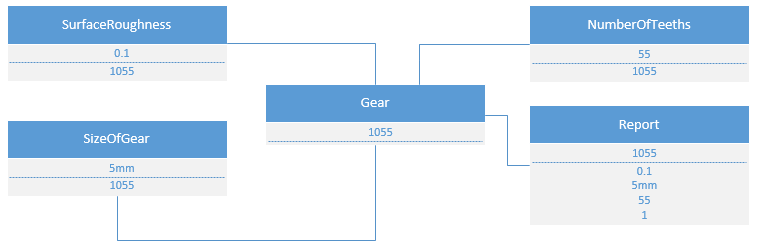
**N/A**

**8.0 Appendices**

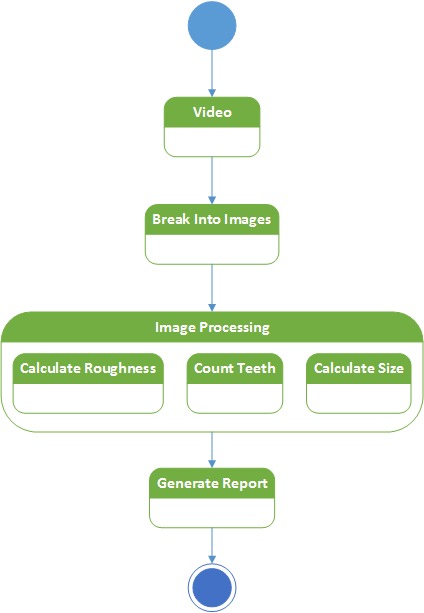
Class Diagram:



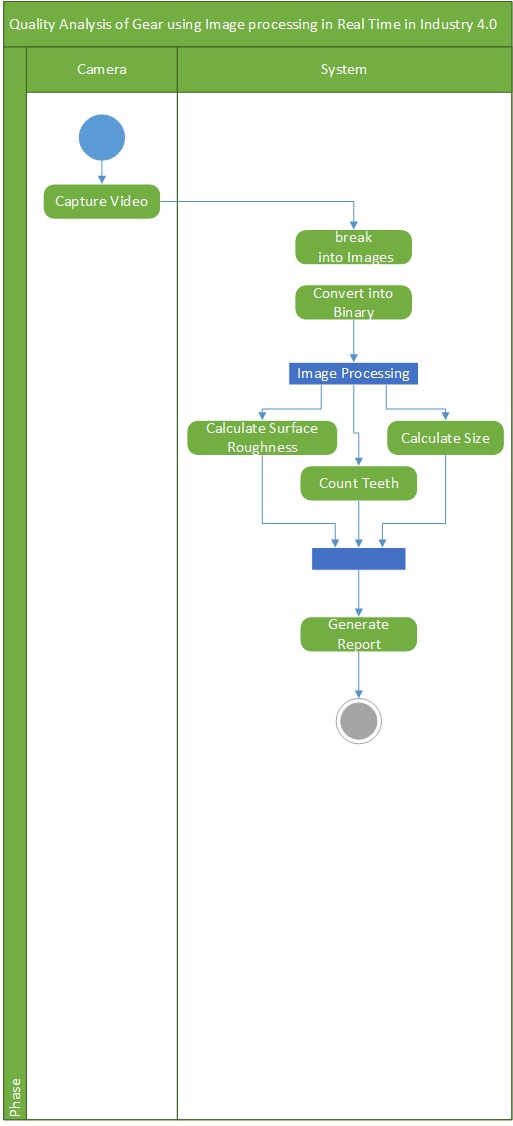
Object Diagram:



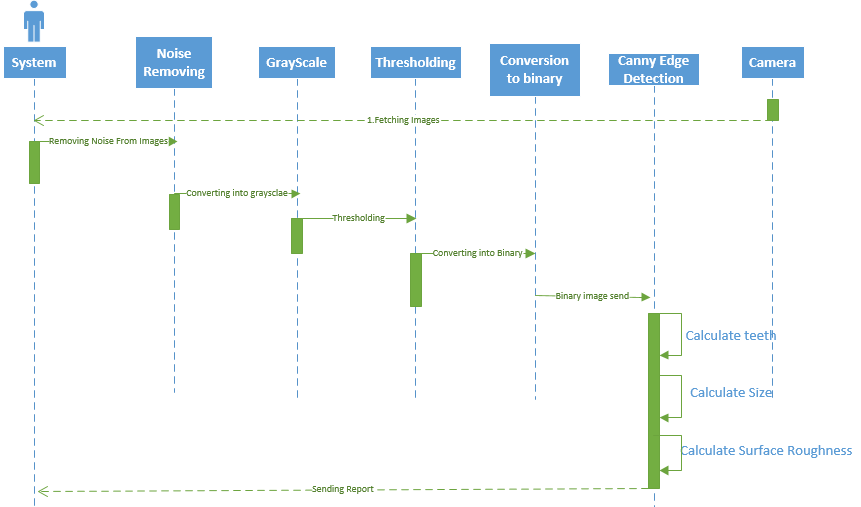
State Chart Diagram:



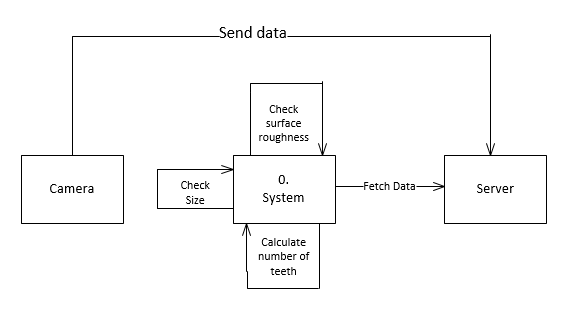
Activity Diagram:



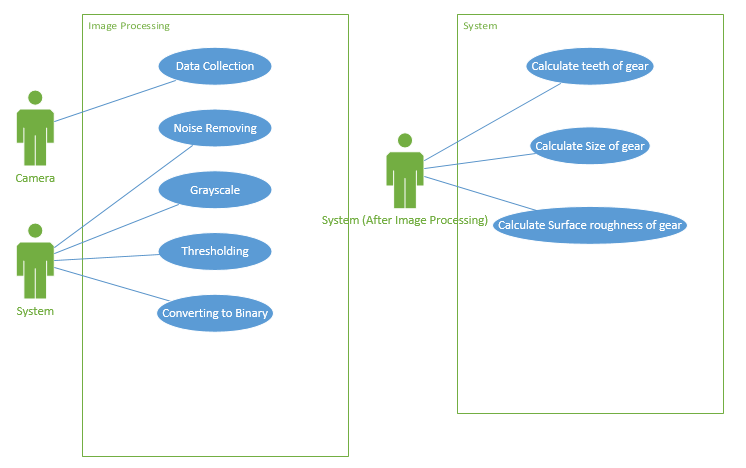
Sequence Diagram:



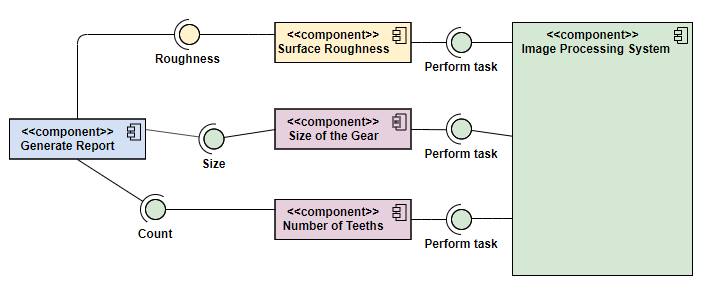
Collaboration Diagram:



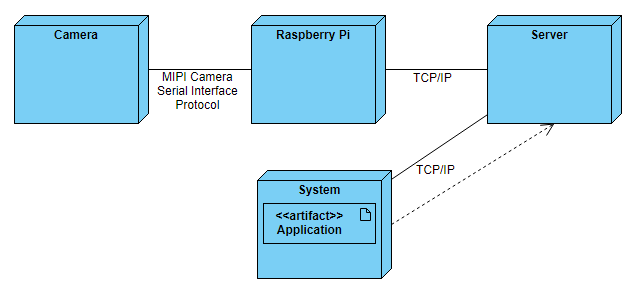
Use Case Diagram:



Component Diagram:



Deployment Diagram:



Block Diagram:

