Real-time Machine Vision-based Quality Inspection System for Small and Medium Enterprise

Final Year Project

Session 2017-2021

A project submitted in fulfilment of the
COMSATS University Islamabad Degree
of
BS in Computer Science (CUI)

Department of Computer Science

COMSATS University Islamabad, Lahore Campus

19 January 2021

Evaluation

Project Group Members (To be filled by students)				(To be filled by supervisor)
Sr.#	Reg. #	Student Name	*Signature	Obtained Marks (Total Marks: 10)
(i)	SP17-BCS-079	Abdul Wajid	Abdul Wajid	
(ii)	SP17-BCS-067	Umer Imtiaz	Umer Imtiaz	
(iii)	SP17-BCS-103	M. Abdul Muhib	Abdul Muhib	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others.

Supervisor Name: ______ Dr. Wajahat Mahmood Qazi __ Supervisor Signature: _______________

Remarks (if any): ____

____

____

Abstract

This project aims to develop a real-time machine vision-based quality inspection system for small and medium enterprises. Labelling errors that occur during packaging can damage consumers' perception of quality. To meet the quality and aesthetic expectations of customers, it is important to identify labelling defects before goods reach the market. All industries are in the race of making their products more apparently and qualitatively appealing. In Pakistan, Quality Inspection responsibility is handed over to the human labour but the Production lines works on so high speed that's it is very difficult to catch minor errors in packaging by Humans so our system shall perform Machine Vision-based measures for quality assessment of the products in real-time to ensure the standard of products. Tests include identification of dirt spots, print template analysis, correct seals, torn, damaged and missing labels of packaging. The basic aim behind this project is to promote the fourth industrial revolution and Artificial Intelligence in Pakistan because our country is lacking in both. Deployment of this system will include specialized cameras, a computer with the design and implementation of industry-standard HMI.

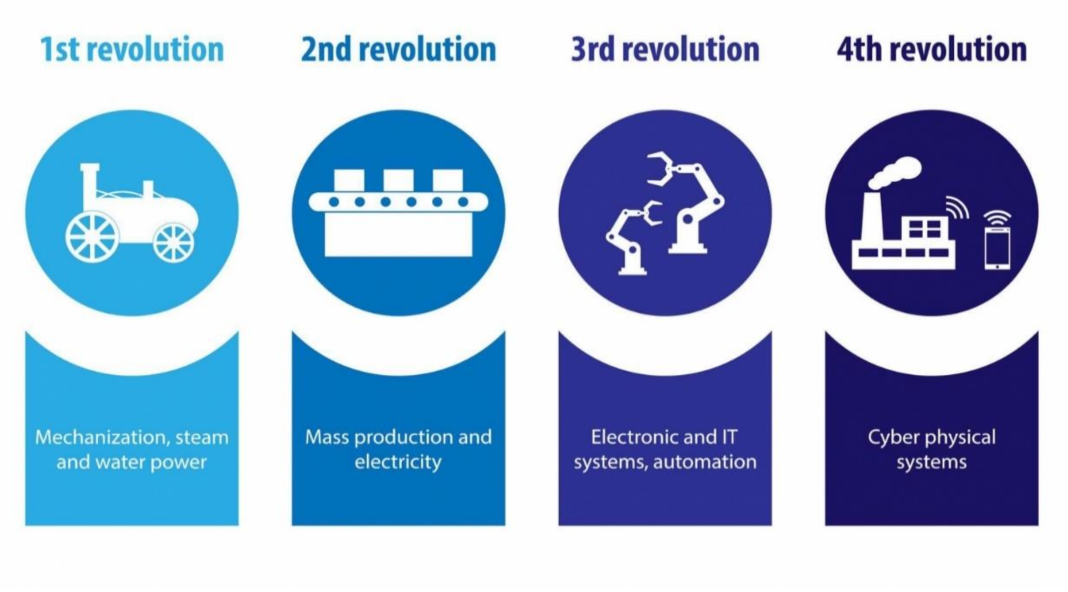
Table of Contents

List of Figures

Chapter 1: Introduction:

1.1 Introduction

Industry 4.0 pursues the goal to optimize the production output through an efficient collaboration of all components of the production process with the adoption of latest technologies like computer vision, machine learning, deep learning and automating the decision making. Cyber-physical systems, the Internet of Things and the Internet of Systems make Industry 4.0 possible and the smart factory a reality. Companies are increasingly investing on their solutions and tools to digitalize their machines, manufacturing processes and products and to integrate all things together in a single integrated network to enhance the productivity and quality of the product and reduce cost [2]. Figure 1 illustrates the revolution in industry 4.0.


1st revolution
2nd revolution
3rd revolution
4th revolution
Mechanization, steam and water power
Mass production and electricity
Electronic and IT systems, automation
Cyber physical systems

Figure# 1: Industry 4.0 revolution [1]

Consumers have raised expectations of the quality and appearance of products. Many companies perform quality inspections on their products using human labour, but they are unable to detect small errors in packaging because of the limitations of humans and the high speed of product line. For example, in the production line of a company, it gives an output of forty-two sachet per minute and there is no mechanism of checking the quality of packaging as shown in figure 2. There can be many issues in packaging like there could be dirt, seals could be damaged, and labels could be missing. All these issues can damage the reputation of a company. For all these reasons, we need quality inspection tools to make our tasks easy and efficient.

Figure# 2: Packaging machine [3]

In industry 3.0 humans use different tools to ensure the quality of packaging. But it still needs the help of a human to ensure the quality. Packaging lines perform the packaging of products at a very high speed. Human labour is unable to perform a quality check on every product. Industry 4.0 lets the machines work independently without the involvement of humans. Machines are itself intelligent to do all these tasks without the supervision of human at very high speed. The machine collects a lot of data and processes it at a very high speed and ensures the quality of the product by triggering actions itself.

The standards that are available for quality control of packaging material include:

- The BRC/IOP is a Global Standard devoted to packaging industries that manufacture packaging materials used in various industries. The purpose behind creating this standard was to provide a set of instructions for packaging industries that ensures consumer's safety and fulfilment of legal requirements. [4]
- Regulation (EC) No 1935/2004): The labelling, presentation and advertising shall not delude the customers. Also, packaging material's traceability should be ensured at all phases to ease control. [4]
- ISO 9001:2015 is a quality management standard that allows the production of ISO quality packaging. It also acts as a core component in operating a quality management system throughout the manufacturing process.

Other developed countries are using autonomous quality inspection tools for a long time. These tools are not fulfilling our Pakistan's industry requirement as they are specially built for their industry requirements and standards. When these technologies are integrated into our industry it becomes very expensive in terms of both time and money. Pakistani Industry is far away from Industry 4.0 and thus or country is lagging in the race of innovation and advancement.

Our basic aim is to develop a real-time vision-based quality inspection system for small and medium industries of our country. It would also be cost-effective.

Based on all these factors we want to purpose a system for quality inspection of ‘packaging material’ with the help of machine vision and artificial intelligence to promote industry 4.0 and automation in our country. Our system will inspect the product packaging in real-time to ensure the quality and standard of the product by performing various tests.

- Identification of dirt spot: The system shall examine the finished product for dirt spot identification. The dirt spots can be either stains or blots on the packaging material. It will have an acceptable range according to the industrial standards, and if the dirt spot exceeds that range the system will fall in unacceptable range and thus will be rejected.
- Print template analysis as per quality standards: The system shall examine the packaging material to see if the printing is done according to the industrial quality inspection standards or not.
- Identification of faulty seals: The system shall examine the seals of the packaging sachet to see if the seals are at their positions and no seal is defective or torn.

Our target will be achieved using cameras computer and cloud technology. Cameras will capture real-time video. After capturing, it sends video to our system for further processing. Our system will perform quality inspection tests on video using intelligent computer vision and machine learning algorithms. After performing all these tests our system will show a response on HMI terminal whether the packaging quality is acceptable or not. HMI application will be of industrial standards. The resulted information would also be stored in our database for later use.

1.2 Objectives

- Develop a real-time Inspection of packaging material.
- Develop a module for dirt spot identification of packaging material.
- Develop a system to entertain the high-speed production of products.
- Develop a system to reduce scrapping in packaging.
- Develop a module to recognize printing mistakes on the packaging.
- Efficient and accurate inspection of packaging.
- Reduce the time of the quality inspection process in packaging.
- Develop a cost-effective system
-

1.3 Problem statement

In industries, the packaging material of the products are inspected by quality inspector that take lots of time. Some of the products go uninspected due to human error. To overcome this issue, the automated inspection of the packaging material will be done that saves lots of time and the products will be inspected with more accuracy.

1.4 Assumptions & constraints

Assumptions:

- Only packaging material’s quality inspection will be performed.
- The system will be fully developed in 3 to 4 months.

Constraints:

- This project will only target Pakistani industries.
- The result of the modules will depend upon the capacity of processing units.
- The results of quality tests will be saved in the database for only 3 months.
- Some limitations of our 3 modules are as under:
    - **Dirt Spot Detection module:** The system will not quantify the intensity of the spot.
    - **Print inspection module:** The module will not quantify the exact displacement of text.
    - **Code structure legibility:** The module will only identify the barcodes.

1.5 Scope of the Project (what to focus and what to ignore)

In other countries, modern technologies are used to ensure the quality of the packaging material. In Pakistan, there are very few industries that are using an autonomous system to inspect the quality of packaging. Most of the local industries are using human resources for quality inspection which is not enough to ensure the quality of packaging and its very time-consuming.

Our goal behind the development of this system is to promote the modern industry 4.0 concept in our country. Our system will enable local small and medium industries to check the quality measures of product packaging at low cost at high speed without any error.

The main motivation towards this project is to contribute towards the promotion of Industrial 4.0 and Artificial Intelligence in Pakistan

Chapter 2: Analysis of the Requirements

2.1 Review of Literature /Study of Existing system

The related work section contains the data from research papers and the details of some products currently working in the industries.

To prevent the damage in packaging material a solution was proposed by Stanley Mamah.In this research work a classification system was developed to assist manufacturing engineer to detect the damages in final product [5].

Convolutional Neural Networks based solution was proposed by Mahdi Bahaghighat, Fereshteh Abedini, Arthur-Jozsef Molnar and Misak S'hoyan to prevent the damage of the bottle caps in drinks manufacturing factory. This paper focuses on the visual inspection of bottle caps in drink factories. 99% accuracy was achieved in this solution based on VGG-19 as an End-to-End deep learning approach. [6]

Optimal approaches to the quality control were proposed by Smith-spark to detect wrong information presented on labels of the Packaging of Various Products. Most of the label errors get undetected due to which the paper suggested the use of software and hardware-based systematic approach. [7]

Distributed Vision Network for Packaging Inspection was proposed by Meliones to deal with the Industrial Packaging Inspection for various products. The Independent Stations that the system had was able to do inspection tasks parallelly such as identification of products, an inspection of product tags and quality control of packaging at high speed and in less time. [8]

A Real-Time Artificial Vision by Aguilar-Torres was proposed to check the quality of the packaging of the industrial products. It performs visual inspection using Pattern Recognition techniques. [9]

Due to continuous advancements in the Internet of Things (IoT) and Sensor Networks, Industry 4.0 having a deep effect on every aspect of the industry, from logistics to quality control. The quality inspection of the products will not deal in a different section. It will be on the production line at the end. The continuous transfer of information from WSNs to the decision-making bodies is the foundation of Industry 4.0. The overall analysis of industry 4.0 on quality inspection is imperative. [10]

Some of the Quality Inspection Tools that are currently available in the market are as follows:

Cognex vision systems inspect food packaging to ensure it is correctly assembled, defect-free, and complete so that only the highest quality products reach customers. Machine vision, barcoding methods and 3d laser profilers are used in their solution. [11]

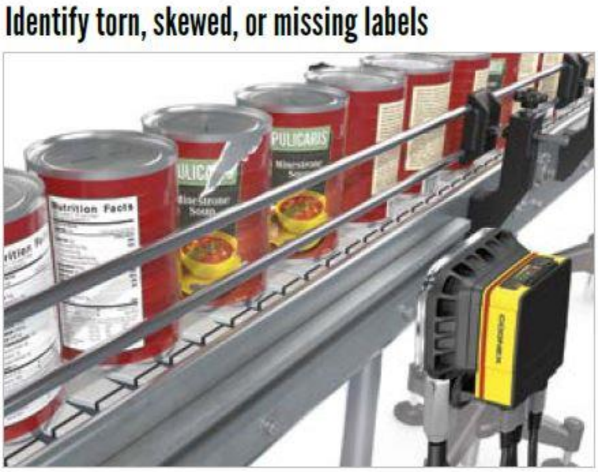


Figure 6: Cognex solution for packaging [11]

The quality inspection tool by Futec company allows to prevent, eliminate and documentation of potential printing defects. In this process the data of packaging print is compared with the master files to prevent the defects. Based on comparison, the product packaging is classified as accepted or rejected. Figure 7 describes the errors caught by their solutions. [12]

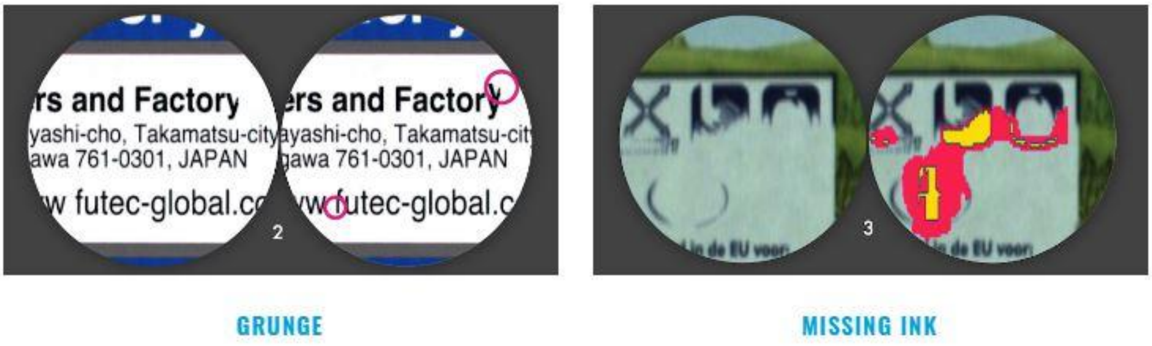


Figure 7: Example of detected errors by Futec tool [12]

Quality Inspection Tool by the EyeC company that provides inspection of print defects and catch artwork before they start generating losses. The machine in figure 8 below is one of their products. [13].

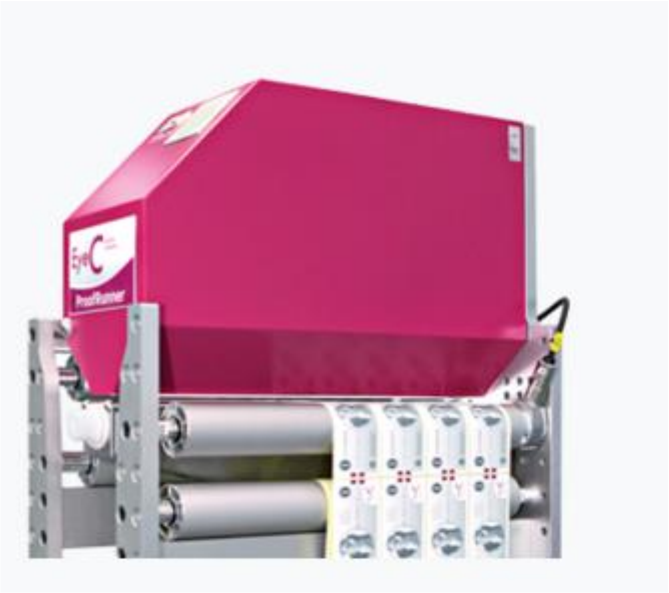


Figure 8 : A Tool by EyeC company [13]

The TriVision IML Packaging Inspector is offering quality control of packaging products that are produced using IML technologies [14].

Quality Inspection Solution by YUZEN is focusing on quality inspection and control of plastic packaging. Their main customers are in food, drug packaging, packaging industry, plastic and rubber packaging. [15]

2.2 List of Stakeholders (Actors)

The stakeholders are:

- **Industry:**
  Industry is one of the stakeholders, which is responsible for providing us basic requirements they need, the packaging material needed for performing quality inspection tests.

- **Supervisor:**

Supervisor is one of the stakeholders who is in charge of allocating job to Inspectors, so that Inspectors can perform quality inspection on packaging material.

- **Inspector:**
  Inspector is one of the stakeholders who is in charge of performing quality inspection tests on packaging material.

- **Admin:**
  Admin is one of the stakeholders who is in charge of managing the inspectors and supervisors account in the system. Admin is responsible for assigning Inspectors and Supervisors their ID's and password so that they can have access to the system.

2.3 Elicitation of Requirements

2.3.1 Functionality Requirements

FR1: Login

Req. #.	**Type of Requirements: Functionality Requirements**
FR1-1	The system shall enable the operator, manager and admin to login in the system by accessing their User ID/Email and their Password.
FR1-2	The system shall enable the operator, admin and manager to access the overall functionalities of the system after providing correct User ID/Email and Password.
FR1-3	Once all the information is filled the system will validate the user by checking the record in the database.
FR1-4	The user will be able to access the system login.

FR2: Logout

Req. #.	**Type of Requirements: Functionality Requirements**
FR2-1	The operator, manager and the admin shall be enabled by the system to logout from the system.
FR2-2	When the user is logged out, he/she will be informed about this on the next page that is displayed.

FR3: Password Change

Req. #.	Type of Requirements: Functionality Requirements
FR3-1	The system shall enable all the users (Admin, supervisor and Inspector) to change their password.
FR3-2	When the user clicks on Change Password Button, the user can change their password.

FR4: Capture Video (using camera)

Req. No.	Functional Requirements
FR4-1	Cameras should be placed 100cm away both on front and back from the packaging material.
FR4-2	The system shall allow the camera to take video of the packaging material.
FR4-3	These videos must be sent to the processing unit for further inspection.

FR5: Start Inspection

Req. #.	Type of Requirements: Functionality Requirements
FR5-1	The Inspector will click on “Start inspection” button to start inspection of packaging material.
FR5-2	The result of inspection performed will be displayed as “Accepted” or “Rejected” in all three Quality inspection tests.
FR5-3	The result of inspection performed will be sent to database.

FR6: Dirt spot inspection

Req. #.	Type of Requirements: Functionality Requirements

FR6-1	The system must have video of the package that is being inspected.
FR6-2	The system shall identify dirt spot if any on packaging material.
FR6-3	The system must Accept the packet if there is no dirt spot on it.
FR6-4	The system must Reject the packet if there is dirt spot on it.
FR6-5	The system shall use semantic segmentation for this module.
FR6-6	The system shall send the results of dirt spot inspection to Database and display it on Inspector’s Window.

FR7: Semantic segmentation:

Req. #.	**Type of Requirements: Functionality Requirements**
FR7-1	The system shall enable the dirt spot detection module to detect dirt spot using deep fully convolutional network (FCN).
FR7-2	The system shall demarcate the video of the packaging material that is being inspected.
FR7-3	The system shall enable the convolutional to layers down the input video of the packaging material, while the deconvolutional layers restore the original video size of packaging material by continuous sampling and combining with the corresponding convolutional layers at each level.
FR7-4	The resultant video shall have label map comprising a fixed number of labels that will show the detected dirt spot on the packaging material (if any) as specified by the ground truth during the training process.

FR8: Print Inspection

Req. #.	**Type of Requirements: Functionality Requirements**
FR8-1	System shall perform print inspection on packaging material.
FR8-2	The system must have the image of the packaging material that is being inspected before this requirement
FR8-3	System shall identify whether label is placed on its actual position.

FR8-4	The system shall also identify if the label may get displaced from its actual location and the part of label from the front panel may be printed on the back panel of the packing or the other way around.
FR8-5	The system shall inspect this problem by registering a query image against a reference template image using SIFT approach.
FR8-6	The system will approve the packaging material, if the label is placed on its actual position.
FR8-7	The system will reject the packaging material if it the label is displaced.
FR8-8	System shall send the results of print inspection to the HMI.

FR9: Key Point Matching

Req. #.	**Type of Requirements: Functionality Requirements**
FR9-1	The system shall allow the print label inspection module to extract the SIFT key points from the reference label video of the packaging material.
FR9-2	The system shall allow these key points to be matched against the video of the packaging material that is being inspected, and the shift in sense video printing will be measured based on offset between the matched points, which shall be achieved by identifying their nearest neighbours.
FR9-3	The system shall check if the product’s printing is displaced.

FR10: Code legibility inspection

Req. #.	**Type of Requirements: Functionality Requirements**
FR10-1	The system must have video of the package that is being inspected.
FR10-2	The system shall perform barcode detection and verification of its legibility.
FR10-3	The system shall use the YOLOv3 tiny weights for localizing the barcode on product packing.
FR10-4	The system must Accept the packet if the barcode on packet is legible.

FR10-5	The system must Reject the packet if the barcode on packet is not legible.
FR10-6	The system shall send the results of code legibility inspection to the HMI.

FR11: Object detection

Req. #.	**Type of Requirements: Functionality Requirements**
FR11-1	The barcode shall be bounded by identifying rectangle by the YOLO model.
FR11-2	The system shall identify barcode by scanning across a horizontal profile of the barcode bounded video in the packaging material.
FR11-3	The system shall translate the scanned profile into an ID value according to the underlying coding scheme used for generating the barcode in the industry.
FR11-4	The system will compare the translated ID data with the pre-specified actual ID value of the packaging material.
FR11-5	The system shall declare code legibility test as a fail in case of a mismatch between values and that packaging material will be rejected for quality inspection.

FR12: Show Results

Req. #.	**Type of Requirements: Functionality Requirements**
FR12-1	System must generate qualitative results of the inspections.
FR12-2	The result will be sent on Inspector’s window.
FR12-3	The HMI must show the accepted and rejected range of the tests and the qualitative result of your video.
FR12-4	If the result of the inspection lie in the accepted range green indicator will turn on, otherwise red indicator will be turned on.

FR13: Store in database

Req. #.	**Type of Requirements: Functionality Requirements**

FR13-1	The system shall allow the results to be stored in the database.
FR13-2	The system shall enable the admin to control all the functionalities of the databases.

FR14: Decision of batch status

Req. #.	**Type of Requirements: Functionality Requirements**
FR14-1	The system must count the number of rejected packages.
FR14-2	The system should compare the number of rejected packages in a batch.
FR14-3	The system will reject a batch if the number of rejected packages in a batch exceeds industrial policy of faulty products.

FR15: Add Supervisor

Req. #.	**Type of Requirements: Functionality Requirements**
FR15-1	The system shall enable Admin to add Supervisor by entering Supervisor’s ID, Name, Email, and password.
FR15-2	The system will add new Supervisor in both dashboard and database.
FR15-3	The system will display the new Supervisor on dashboard.

FR16: Add Inspector

Req. #.	**Type of Requirements: Functionality Requirements**
FR16-1	The system shall enable Admin to add Inspectors by entering Inspector’s ID, Name, Email, and password.
FR16-2	The system will add new Inspector in both dashboard and database.
FR16-3	The system will display the new Inspector on dashboard.

FR17: Delete Supervisor

Req. #.	Type of Requirements: Functionality Requirements
FR17-1	The system shall enable Admin to Delete Supervisor from system by providing Supervisor’s ID.
FR17-2	The system will delete Inspector from both dashboard and database.
FR17-3	The system will display the other Inspectors on dashboard.

FR18: Delete Inspector

Req. #.	Type of Requirements: Functionality Requirements
FR18-1	The system shall enable Admin to Delete Inspector from system by providing Inspector’s ID.
FR18-2	The system will delete Inspector from both dashboard and database.
FR18-3	The system will display the other Inspectors on dashboard.

FR19: Modify Supervisor’s Information

Req. #.	Type of Requirements: Functionality Requirements
FR19-1	The system shall enable Admin to Modify any information of Supervisor in the system.
FR19-2	The system will update Supervisor’s Information in both database and dashboard.
FR19-3	The system will display the updated information on dashboard.

FR20: Modify Inspector’s Information

Req. #.	Type of Requirements: Functionality Requirements
FR20-1	The system shall enable Admin to Modify any information of Inspector in the system.

FR20-2	The system will update Inspector’s Information in both database and dashboard.
FR20-3	The system will display the updated information on dashboard.

FR22: Batch Entry

Req. #.	Type of Requirements: Functionality Requirements
FR21-1	The system shall enable Supervisor to Add Batch information by providing its Batch ID and Description.
FR21-2	The system will save the data in both system and Database.

FR22: Job Allocation

Req. #.	Type of Requirements: Functionality Requirements
FR22-1	The system shall enable Supervisor to Allocate job to Inceptors.
FR22-2	The system shall enable the Supervisor to select a Conveyor Line and Batch to allocate it to Inspector.
FR22-3	The system will save the data in both system and Database.
FR22-4	The system will display the Allocated jobs on Supervisor’s Window.

2.3.2 Non-Functionality Requirements

NFR1: Usability

Req. #.	Type of Requirements: Non-Functionality Requirements
NFR1-1	The system Should be intuitive and simple in the way it displays all relevant data.
NFR1-2	The system must be accessible quickly.
NFR1-3	The system must be easy to use for new users.

NFR2: Reliability

Req. #.	Type of Requirements: Non-Functionality Requirements
NFR2-1	The system must provide accurate results for all quality inspections.
NFR2-2	The system should provide details of inspection to user if it is completed correctly. If it is not completed correctly then provide a reason for incomplete inspection.
NFR02-03	The system should not update database for any incomplete or failed quality inspection.

NFR3: Graphical user Interface

Req. #.	Type of Requirements: Non-Functionality Requirements
NFR3-1	System should provide an easy interface for operator to start inspection.
NFR3-2	The results displayed on HMI should be interactive and easy to understand for the user
NFR3-3	System should offer a simple and interactive way of viewing the reports of inspection.

NFR4: Performance

Req. #.	Type of Requirements: Non-Functionality Requirements
NFR4-1	Cameras should capture images of packaging material in less than 5 seconds.
NFR4-2	The system should perform each inspection in less than 5 seconds.
NFR4-3	The system should display results of inspection on HMI in less than 20 seconds.

NFR5: Security

Req. #.	Type of Requirements: Non-Functionality Requirements
NFR5-1	No unauthorized person should able to access the system.
NFR5-2	Unauthorized user must not have access to database so he cannot view or update it.

NFR6: Documentation

Req. #.	Type of Requirements: Non-Functionality Requirements
NFR6-1	Help documentation must be complete in providing information about each and every module and functionality provided by the system.
NFR6-2	Help manual should be available in hard document form.
NFR6-3	Technical terms included in the help manual should be well defined in the document.

NFR7: Defect Maintenance

Req. #.	Type of Requirements: Non-Functionality Requirements
NFR7-1	System should be regularly checked for any kind of bugs.
NFR7-2	If user reports a bug or defect in the system, it should be remove/resolve immediately for ease of users.

2.4 Use case descriptions

2.4.1 UC1: Description of Login

ID: UC-1	
Name: Login	
Priority: High	
Included Actors: Admin, Manager, operator	
Summary	The Admin, Manager, operator will login into the system.
Preconditions:	Users must have their account created by Admin in the system.
Flow of Events	**Alternative Flow**
1. Start case: User will enter their UserID and password to login.	
2. User will click on Login button	
3. User will redirect to their window.	
4. End Case	
Exceptions or Alerts	
If the user has entered the wrong UserID/ Email or password, an alert message of "Incorrect UserID or password" will be displayed to user.	
Postconditions	
Step no:	**Description of Steps**
1.	The user has logged in into the system.
Cross References	
Use Case includes	*None*
Use Case extends	Verification

Table# 1-Login

2.4.2 UC2: Description of Logout

ID: UC-2	
Name: Logout	
Priority: High	
Included Actors: Admin, Manager, operator	
Summary	The Admin, Manager, operator will logout from the system.
Preconditions:	The user is already logged in into the system.
Flow of Events	**Alternative Flow**
1. Start case: The user will click on Logout button to logout their accounts.	

2. End Case	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	Session will expire.
Cross References	
Use Case Includes	*Session Expired*
Use Case Extends	None

Table# 2-Logout

2.4.3 UC3: Description of Password Change.

ID: UC-3	
Name: Password Change	
Priority: High	
Included Actors: Admin, Manager, operator	
Summary	The Admin, Manager, operator will change their account's password.
Preconditions:	The user has an account present in the system which password he wants to change.
Flow of Events	**Alternative Flow**
1. Start case: The users will click on "**Change Password**" button to change their account password.	
2. User will enter their account's old password.	
3. User will enter new password for their account.	
4. User will confirm new password for their account.	
5. User will enter "**Change**" button.	
6. End case.	
Exceptions or Alerts	
If the user has entered wrong new password, then an error message of "Incorrect Password" will be displayed to user.	
Postconditions	
Step no:	**Description of Steps**

1	The User will be redirected to the login screen to again login into the system.
Cross References	
Use Case includes	*Verification, Notify, Enter Password*
Use Case extends	None

Table# 3-Password change

2.4.4 UC4: Description of Capture Video

ID: UC-4	
Name: Capture Video	
Priority: High	
Included Actors: Cameras	
Summary	The camera will capture Video of packaging material.
Preconditions:	The packaging material is placed Infront of the camera.
Flow of Events	**Alternative Flow**
1. Start case: Inspector will click on "**Capture Video**" to capture video of packaging material.	
2. End case.	
Exceptions or Alerts	
Post Conditions	
Step no:	**Description of Steps**
1	The captured video is displayed on Inspector's Window.
Cross References	
Use Case includes	*Place Packaging material*
Use Case extends	None

Table# 4-Capture Video

2.4.5 UC5: Description of Trigger Inspection

ID: UC-5		
Name: Trigger inspection		
Priority: High		
Included Actors: Inspector		
Summary	Quality inspection of the packaging material will be performed.	
Preconditions:	The system has a video of packaging material captured by Inspector.	
Flow of Events	**Alternative Flow**	
1. Start case: The Inspector will click on "**Start Inspection**" to perform quality inspection test on packaging material.		
2. End case.		
Exceptions or Alerts		
Post Conditions		
Step no:	**Description of Steps**	
1	Result of packaging material's Inspection is displayed in Inspector's Window.	
Cross References		
Use Case includes	*Dirt spot detection, Print inspection, Code legibility*	
Use Case extends	Start Inspection	

Table# 5-Start inspection

2.4.6 UC6: Description of Dirt Spot Inspection

ID: UC-6	
Name: Dirt Spot Inspection	
Priority: High	
Included Actors: Quality Inspection agent	
Summary	Dirt spot is detected on packaging material.
Preconditions:	The system must have images taken from the cameras of packaging material.
Flow of Events	**Alternative Flow**
1. Start case: The Quality Inspector Agent will perform dirt spot detection using semantic segmentation on captured video.	
2. The Quality Inspector Agent will identify dirt spot on packaging material.	
3. The Quality Inspector Agent will "**Accept**" the packet if there is no dirt spot on it.	**3a.** The Quality Inspector Agent will "**Reject**" the packet if there is dirt spot on it.
4. The Quality Inspector Agent will display result of dirt spot detection performed.	
5. The Quality Inspector Agent will send result of dirt spot detection performed to Database.	
6. End case.	
Exceptions or Alerts	
1. If the Dirt spot range is greater than the given industrial policy, then the quality inspection agent will display error message on HMI**.**	
Postconditions	
Step no:	**Description of Steps**
1	The Quality inspection agent will perform remaining quality inspection tests.

Cross References	
Use Case includes	*None*
Use Case extends	None

Table# 6-Dirt spot inspection

2.4.7 UC7: Description of Semantic segmentation

ID: UC-7	
Name: Semantic segmentation	
Priority: High	
Included Actors: Quality Inspection agent	
Summary	Semantic segmentation will be used on dirt spot detection module.
Preconditions:	The system must retrieve industrial policy before inspection.

Flow of Events	**Alternative Flow**
1. Start case: The Quality inspection agent shall start the dirt spot detection.	
2. The quality inspection agent demarcate the video of the packaging material that is being inspected.	
3. The quality inspection agent will enable the convolutional to layers down the input video of the packaging material, while the deconvolutional layers restore the original video size of packaging material by continuous sampling and combining with the corresponding convolutional layers at each level.	
4. The resultant video shall have label map comprising a fixed number of labels that will show the detected dirt spot on the packaging material (if any) as specified by the ground truth during the training process.	
5. End case.	
Exceptions or Alerts	
1. If the Dirt spot range is greater than the given industrial policy, then the quality inspection agent will display error message on HMI**.**	
Postconditions	
Step no:	**Description of Steps**

1	The Quality inspection agent will perform remaining quality inspection tests.
Cross References	
Use Case includes	*Demarcate Capture Video, layer down video, label dirt spot*
Use Case extends	None

Table# 7-Semantic segmentation

2.4.8 UC8: Description of Print inspection

ID: UC-8	
Name: Print Inspection	
Priority: High	
Included Actors: Quality Inspection agent	
Summary	Print inspection is performed on packaging material.
Preconditions:	The system must have video of packaging material.

Flow of Events	**Alternative Flow**
1. Start case: The quality inspection agent will perform print inspection using key point matching on captured video.	
3. The quality inspection agent will identify if the packaging material's label is on actual position or not.	
4. The quality inspection agent will also identify if the label is displaced from its actual position and the part of label from front panel is printed on the back panel of packaging material or other way around.	
5. The quality inspection agent will inspect by registering a query video against a reference template with the use of SIFT approach.	
6. The quality inspection agent will approve the packaging material if the label is in its actual position.	**6a.** The quality inspection agent will reject the packaging material, if the label is not in its actual position.
7. The quality inspection agent will send the results to HMI.	
8. End case.	
Exceptions or Alerts	

1. If the label on packaging material is displaced, then the quality inspection agent will trigger error message on HMI**.**	
Postconditions	
Step no:	**Description of Steps**
1	The Quality inspection agent will perform remaining quality assurance tests.
Cross References	
Use Case includes	*Capture Image, extract key points, detect key points, match key points, retrieve policy*
Use Case extends	Store results

Table# 8-Print inspection

2.4.9 UC9: Description of Key point matching

ID: UC-9	
Name: Key point matching	
Priority: High	
Included Actors: Quality Inspection agent	
Summary	Key point matching will be used on print inspection module.
Preconditions:	The system must retrieve industrial policy before inspection.
Flow of Events	**Alternative Flow**
1. Start case: The print label inspection module will extract the SIFT key points from the reference label image of the packaging material.	
2. The key points will be matched against the image of the packaging material that is being inspected, and the shift in sense image printing will be measured based on offset between the matched points, by identifying their nearest neighbours.	
3. The shift value beyond the pre-defined policy will be calculated by the system that will result in rejection of packaging material.	
4. End case.	
Exceptions or Alerts	

Postconditions	
Step no:	**Description of Steps**
1	

Cross References	
Use Case includes	*extract key points, match key points.*
Use Case extends	None

Table# 9-Key point matching

2.4.10 UC010: Description of Code legibility inspection

ID: UC-10	
Name: Code legibility inspection	
Priority: High	
Included Actors: Quality Inspection agent	
Summary	Code legibility inception will be performed on packaging material.
Preconditions:	The system must have images taken from the cameras of packaging material.

Flow of Events	**Alternative Flow**
1. Start case: The Quality Inspector Agent will perform code legibility inspection using object detection on captured video.	
2. The Quality Inspector Agent will perform barcode detection and verification of its legibility.	
3. The Quality Inspector Agent will Accept the packet if the barcode on packet is legible.	**3a.** The Quality Inspector Agent will Reject the packet, if the barcode on packet is not legible.
4. The Quality Inspector Agent will display result of Code legibility Inspection performed.	

5. The Quality Inspector Agent will send result of Code legibility Inspection performed to Database.	
6. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	The Quality Inspector Agent will perform quality inspection test on new packet.
Cross References	
Use Case includes	*None*
Use Case extends	None

Table# 10-Code legibility inspection

2.4.11 UC11: Description of Object Detection

ID: UC-11	
Name: Object Detection	
Priority: High	
Included Actors: Quality Inspection agent	
Summary	Object detection will be used on code legibility inspection module.
Preconditions:	The system must retrieve industrial policy before inspection.
Flow of Events	**Alternative Flow**
1. Start case: The system will identify barcode by scanning across a horizontal profile of the barcode bounded image in the packaging material	
2. The scanned profile will be translated by the system into an ID Value according to the underlying coding scheme used for generating the barcode in the industry.	

3. The translated ID data will be compared by the system with the pre-specified actual ID value of the packaging material, that shall be retrieved from the policy.	
7. End case.	
Exceptions or Alerts	
1. If there is a mismatch between values, then the packaging material will be rejected for quality inspection.	
Postconditions	
Step no:	**Description of Steps**
1	
Cross References	
Use Case includes	*Capture image, identify bar code, compare bar code, retrieve policy*
Use Case extends	Store results

Table# 11-Object detection

2.4.12 UC12: Description of Show results

ID: UC-12	
Name: Show results	
Priority: High	
Included Actors: Quality Inspection agent	
Summary	Qualitative results of inspections will be displayed.
Preconditions:	Quality inspection must be performed on packaging material to display results.
Flow of Events	**Alternative Flow**
1. Start case: The Quality Inspector Agent will generate qualitative results of each quality inspection test performed.	

2. The system will display results on Inspector’s Window.	
3. The system will display status and result of packaging material.	
4. The system will give a green signal if the product is accepted.	**4a.** The system will give a red signal if the product is rejected.
5. End case.	
6. End case.	
Exceptions or Alerts	
Post Conditions	
Step no:	**Description of Steps**
1	Next product’s quality inspection is performed
Cross References	
Use Case includes	*None*
Use Case extends	Give alert

Table# 12-Show results

2.4.13 UC13: Description of Store in database

ID: UC-13	
Name: Store in database	
Priority: High	
Included Actors: Quality Inspection agent, Admin	
Summary	Store results in database.
Preconditions:	Quality inspection test should be performed on packaging material to store results in database.
Flow of Events	**Alternative Flow**
1. Start case: The system will store the results in database.	
2. Database functionalities will be controlled by the admin.	
3. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	A report of results will be generated.

Cross References	
Use Case includes	*None*
Use Case extends	None

Table# 13-Store in database

2.4.14 UC14: Description of Decision of batch status

ID: UC-14	
Name: Decision of Batch status	
Priority: High	
Included Actors: Quality Inspection Agent	
Summary	Decision about batches will be done.
Preconditions:	System have Information about number of packets accepted and rejected.
Flow of Events	**Alternative Flow**
1. Start case: The quality inspection agent will count the number of rejected packages.	
2. Quality inspection agent will compare the number of rejected packages with retrieved policy.	
3. If the number of rejected batches exceeds industrial policy then quality inspection agent will reject the batch.	**3a.** If the number of rejected batches does not exceed industrial policy, then quality inspection agent will accept the batch.
2. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	Result of rejected batches will be displayed on HMI.
Cross References	
Use Case includes	*count rejected products*
Use Case extends	None

Table# 14- Decision of Batch status

2.4.15 UC15: Description of Add Supervisor

ID: UC-15
Name: Add Supervisor

Priority: High	
Included Actors: Admin	
Summary	Admin will add a new supervisor in the system by proving Supervisor’s information.
Preconditions:	The Supervisor don’t have an account in the system.
Flow of Events	**Alternative Flow**
1. Start case: The Admin will click on “**Add New**” button to add new Supervisors in the system.	
2. A form will appear, the Admin will enter Supervisor’s ID, Name, Email and Password.	
3. After entering all the information, Admin will click on “**Add**” Button.	
4. New Supervisor will be added in the system and database.	
5. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	New Supervisor added is displayed on Dashboard.
Cross References	
Includes	*Store in database*
Extends	None

Table# 15- Add Supervisor

2.4.16 UC16: Description of Add Inspector

ID: UC-16	
Name: Add Inspector	
Priority: High	
Included Actors: Admin	
Summary	Admin will add a new Inspector in the system by proving Inspector’s information.
Preconditions:	The Inspector don’t have an account in the system.
Flow of Events	**Alternative Flow**
1. Start case: The Admin will click on “**Add New**” button to add new Inspectors in the system.	

2. A form will appear, the Admin will enter Inspector's ID, Name, Email and Password.	
3. After entering all the information, Admin will click on "**Add**" Button.	
4. New Inspector will be added in both system and database.	
5. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	New Inspector added is displayed on Dashboard.
Cross References	
Includes	*Store in database*
Extends	None

Table# 16- Add Inspector

2.4.17 UC17: Description of Delete Supervisor

ID: UC-17	
Name: Delete Supervisor	
Priority: High	
Included Actors: Admin	
Summary	Admin will delete supervisor from the system by entering Supervisor's ID.
Preconditions:	The Supervisor have an account in the system.
Flow of Events	**Alternative Flow**
1. Start case: The Admin will click on "**Delete**" button to delete Supervisors from the system.	
2. A form will appear, and the Admin will enter Supervisor's ID they want to delete.	
3. Admin will click on "**Delete**" button in form.	
4. If the entered Supervisor ID is valid, then the Supervisor will be deleted from the system.	**4a.** If the entered Supervisor ID is invalid, then no Supervisor will be deleted from the system.
5. End case.	

Exceptions or Alerts	
If the Supervisor ID is invalid and alert message will be displayed that "User ID does not exit".	
Postconditions	
Step no:	**Description of Steps**
1	The dashboard and database are updated.
Cross References	
Includes	*Remove from database*
Extends	None

Table# 17- Delete Supervisor

2.4.18 UC18: Description of Delete Inspector

ID: UC-18	
Name: Delete Inspector	
Priority: High	
Included Actors: Admin	
Summary	Admin will delete Inspector from the system by entering Inspector's ID.
Preconditions:	The Inspector have an account in the system.
Flow of Events	**Alternative Flow**
1. Start case: The Admin will click on "**Delete**" button to delete Inspectors from the system.	
2. A form will appear, and the Admin will enter Inspector's ID they want to delete.	
3. Admin will click on "**Delete**" button in form.	
4. If the entered Inspector ID is valid, then the Inspector will be deleted from the system.	**4a.** If the entered Inspector ID is invalid, then no Inspector will be deleted from the system.
5. End case.	
Exceptions or Alerts	
If the Inspector *ID is invalid and alert message will be displayed that "User ID does not exit".*	
Postconditions	
Step no:	**Description of Steps**
1	The dashboard and database are updated.
Cross References	
Includes	*Remove from database*

Extends	None

Table# 18- Delete Inspector

2.4.19 UC19: Description of Modify Supervisor's Information

ID: UC-19	
Name: Modify Supervisor's Information	
Priority: High	
Included Actors: Admin	
Summary	Admin will modify any information of Supervisor they want from the system.
Preconditions:	The Supervisor have an account in the system.

Flow of Events	**Alternative Flow**
1. Start case: The Admin will click on "**Modify**" button to modify Supervisor's Information from the system.	
2. Admin will click on any information (ID, Name, Email, Password) of Supervisor on the Record's Table.	
3. Admin will modify the information and press Enter.	
4. The entered information will be updated in both Dashboard and Database.	**4a.** If the entered Inspector ID is invalid, then no Inspector will be deleted from the system.
5. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	The Modified information is displayed on Dashboard.
Cross References	
Includes	*Modified database*
Extends	None

Table# 19- Modify Supervisor's Information

2.4.20 UC20: Description of Modify Inspector's Information

ID: UC-20
Name: Modify Inspector Information

Priority: High	
Included Actors: Admin	
Summary	Admin will modify any information of Inspector they want from the system.
Preconditions:	The Inspector have an account in the system.
Flow of Events	**Alternative Flow**
1. Start case: The Admin will click on "**Modify**" button to modify Inspector's Information from the system.	
2. Admin will click on any information (ID, Name, Email, Password) of Inspector on the Record's Table.	
3. Admin will modify the information and press Enter.	
4. The entered information will be updated in both Dashboard and Database.	
5. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	The Modified information is displayed on Dashboard.
Cross References	
Includes	*Modified database*
Extends	None

Table# 20- Modify Inspector's Information

2.4.21 UC21: Description of Batch Entry

ID: UC-21	
Name: Batch Entry	
Priority: High	
Included Actors: Supervisor	
Summary	Supervisor will enter Batch Information (Batch ID and Description) to add it in system.
Preconditions:	The Supervisor must have batch information.
Flow of Events	**Alternative Flow**
1. Start case: The Supervisor will click on "**Add Batch**" to add Information of batch.	

2. A form will appear; the Supervisor will enter Batch ID and its Description.	
3. Supervisor will click on “**Add**” Button.	
4. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	The entered information will be saved in both System and Database.
Cross References	
Includes	*Store in database*
Extends	None

Table# 21- Batch Entry

2.4.22 UC22: Description of Job Allocation

ID: UC-22	
Name: Job Allocation	
Priority: High	
Included Actors: Supervisor	
Summary	Supervisor will Allocate Job to Inspector.
Preconditions:	Batch ID, Conveyor Line, Inspector information is available in the system.
Flow of Events	**Alternative Flow**
1. Start case: The Supervisor will select Conveyor Line, Batch ID, and Inspector to allocate job to Inspector.	
2. After selecting, Supervisor will click on “**Allocate**” Button.	
3. The information will be saved in both System and Database.	
4. End case.	
Exceptions or Alerts	
Postconditions	
Step no:	**Description of Steps**
1	The job allocated will be displayed on Supervisor’s Window.
Cross References	

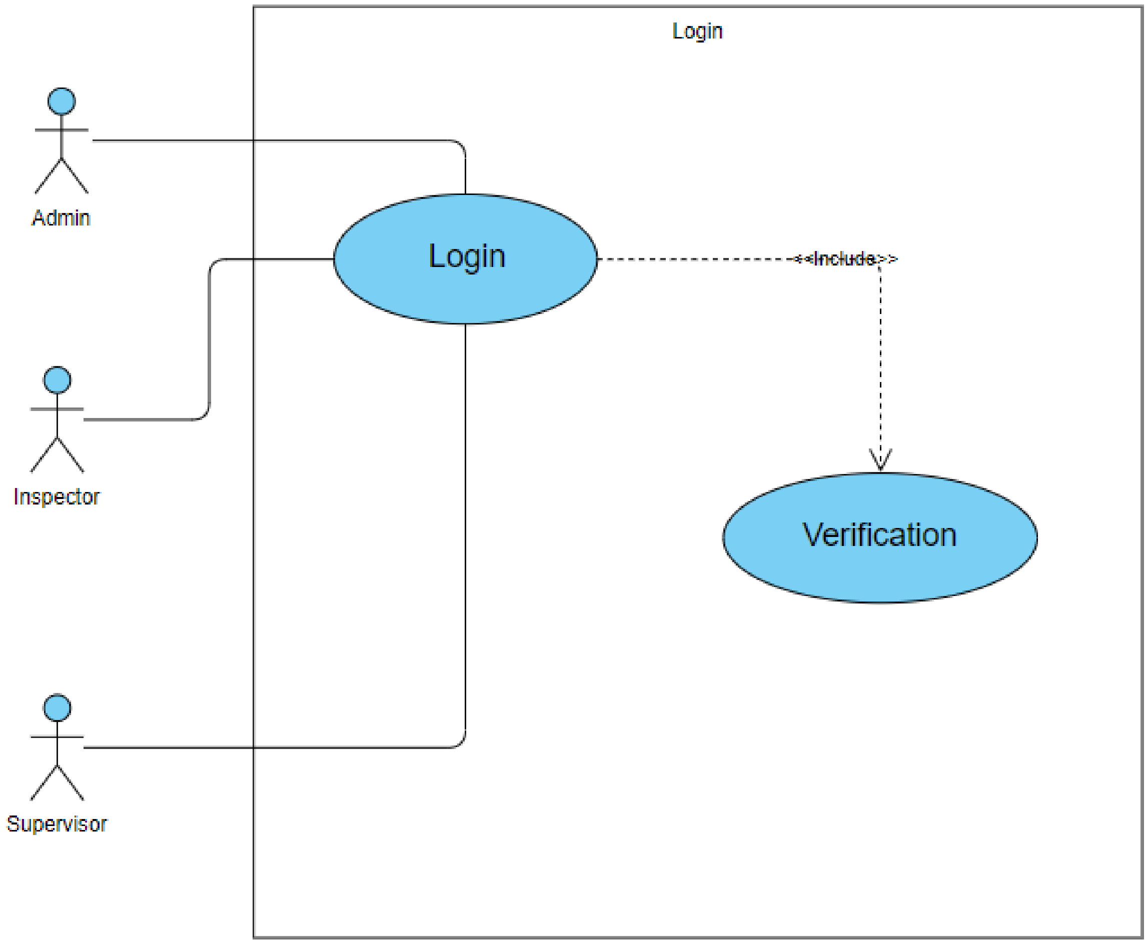
Includes	*Store in database*
Extends	None

Table# 22- Job Allocation

2.5 Use case design

Login

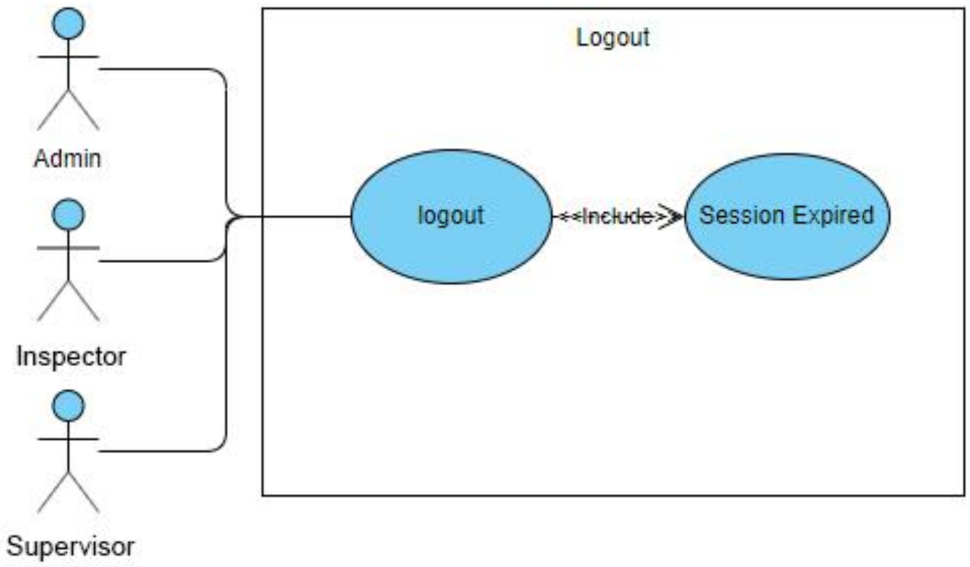
In this use case Admin, Inspector and Supervisor are actors and the login include verification.



Figure# 1: UC Login

Logout

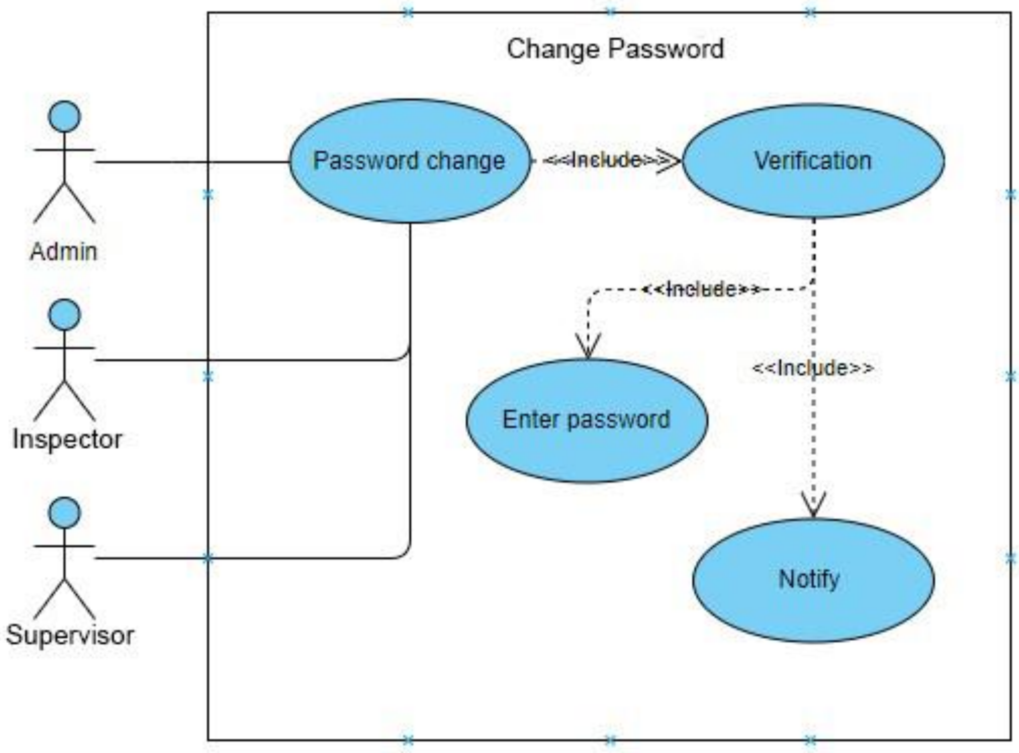
In this use case Admin, Inspector and Supervisor are actors and the logout include session expired.



Figure# 2: UC Logout

Change Password

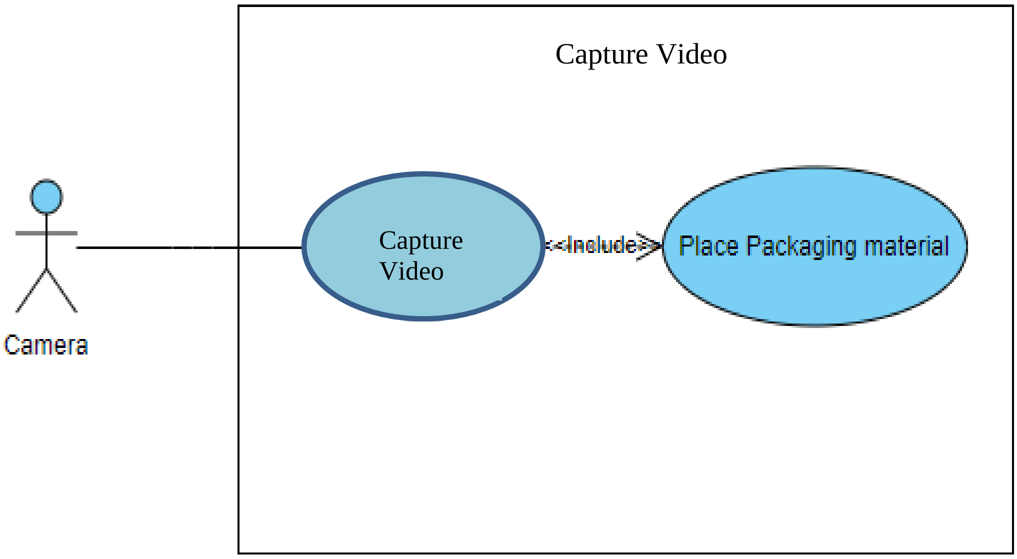
In this use case Admin, Inspector and Supervisor are actors and the password change include verification and verification includes enter the old password and notifying that the password is being changed.



Figure# 3: UC Password Change

Capture Video

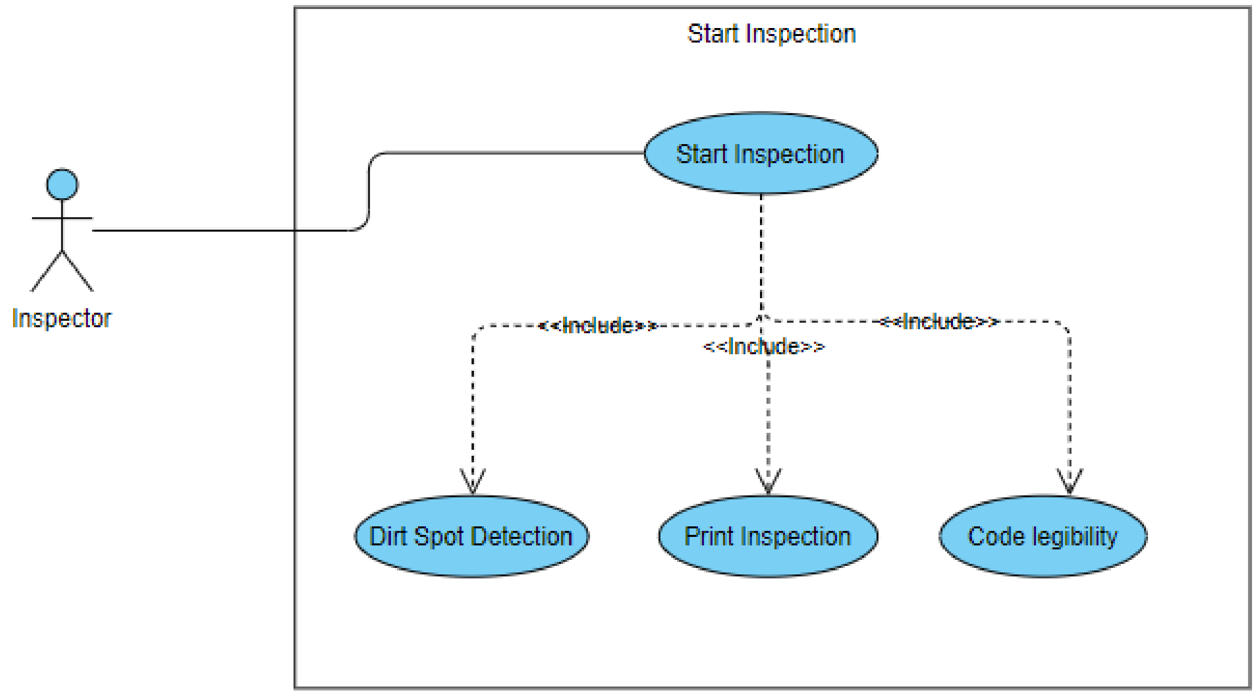
Camera is the actor in this use case camera will be placed in front of the packaging material and camera will capture the video of the packaging material.



Figure# 5: UC Capture Video

Start Inspection

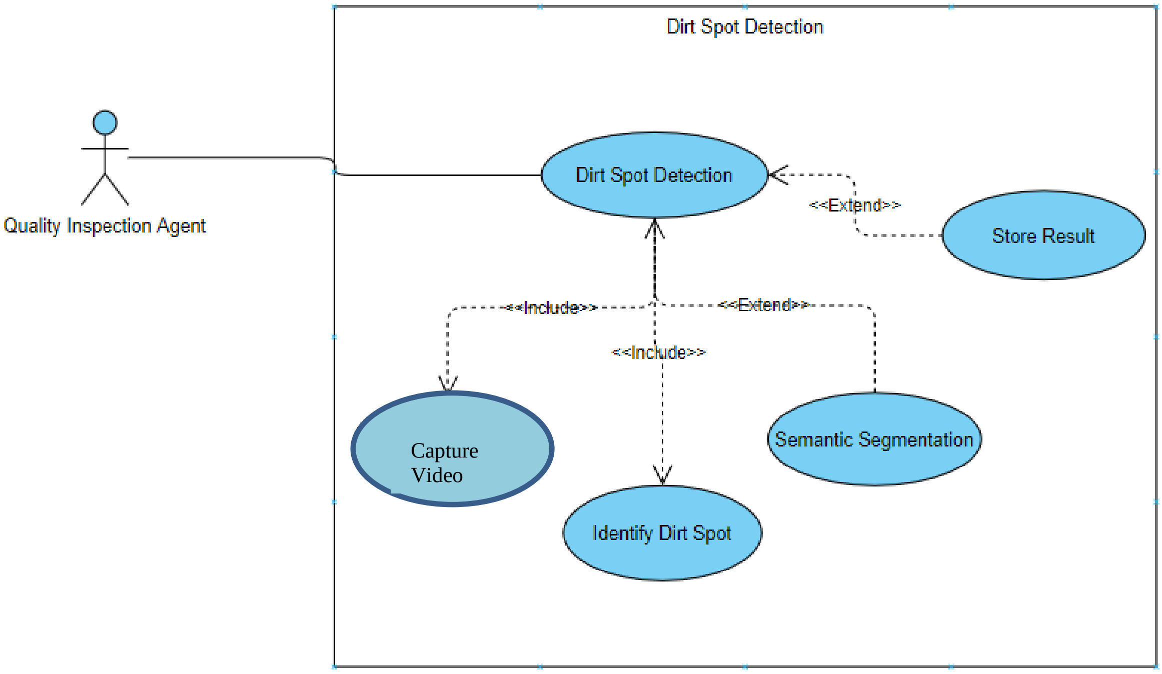
The Inspector will trigger the inspection and the inspection will be started and the inspection includes the dirt spot detection, code legibility, print inspection.



Figure# 6: UC Start Inspection

Dirt spot detection

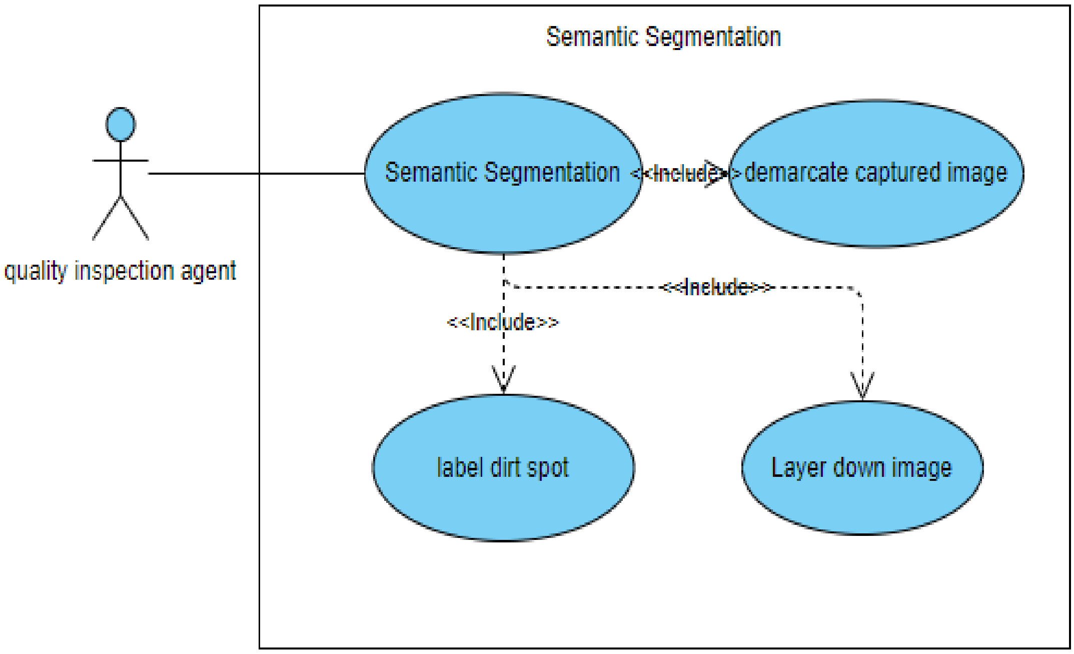
Dirt spot detection will identify the dirt spots on the packaging material which is performed by quality inspection agent. The dirt spot detection will include the video of packaging material and after detecting it will show the results of inspection.



Figure# 7: UC Dirt Spot Detection

Semantic segmentation

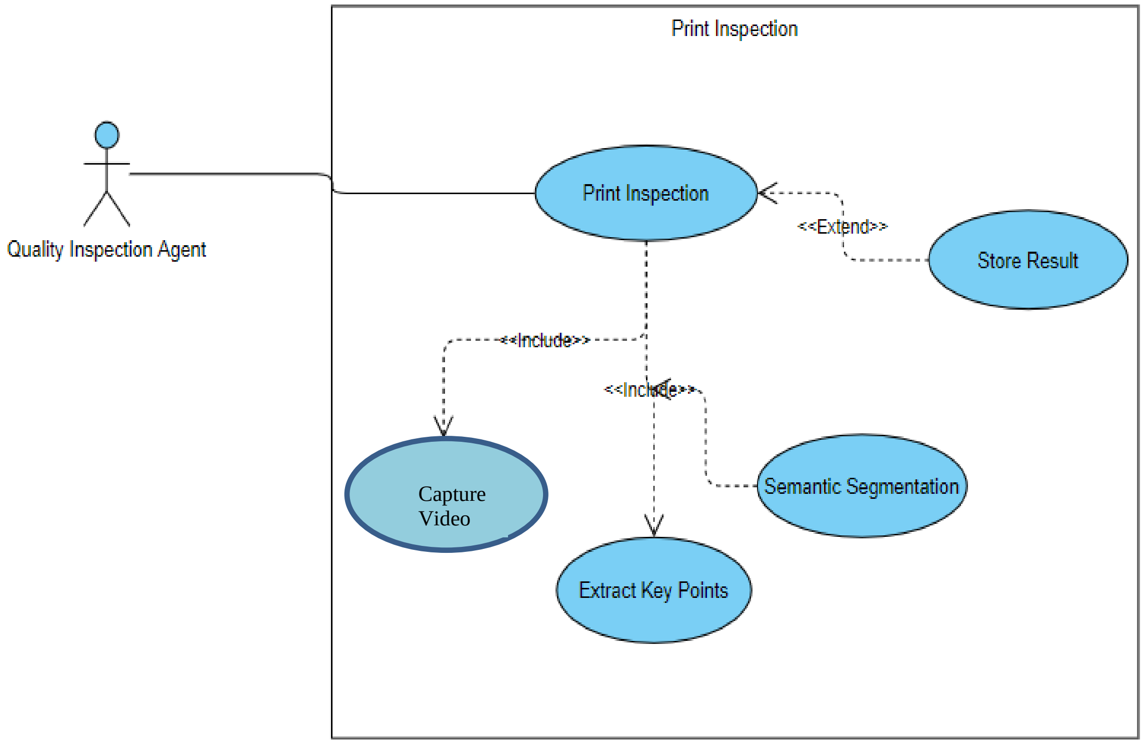
In this use case of semantic segmentation, the actor is quality inspection agent. Semantic segmentation includes demarcate captured image, labelling of dirt spot and the layering down of image.



Figure# 8: UC Semantic Segmentation

Print inspection

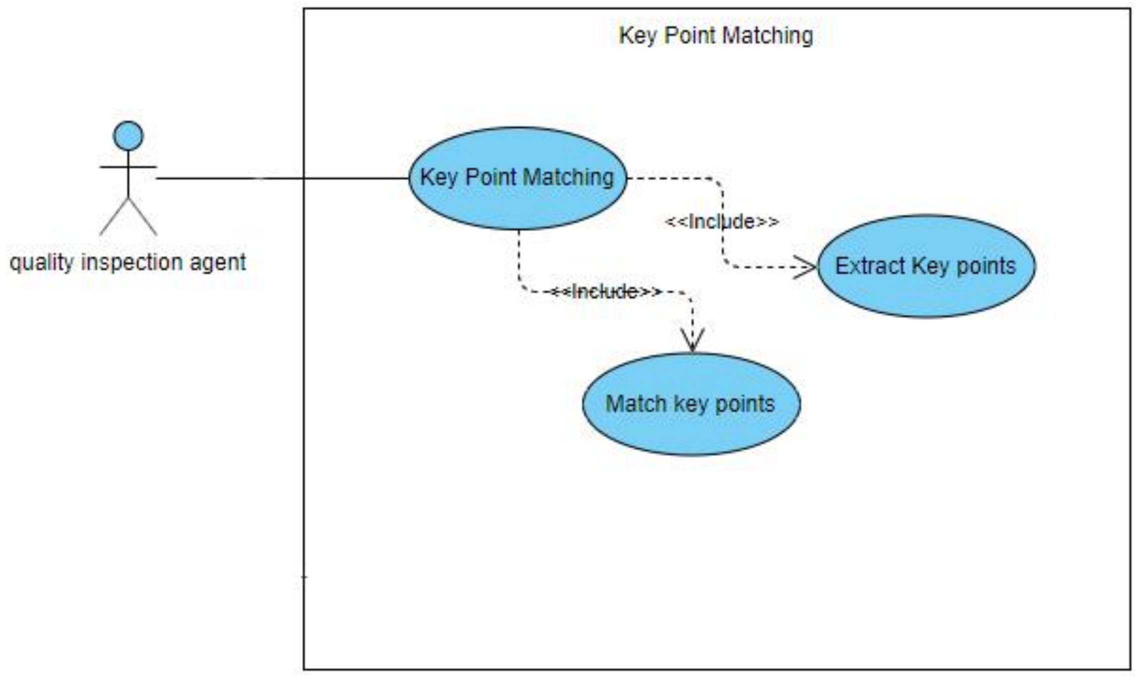
In this use case of print inspection, the actor is quality inspection agent. Print inspection includes capture video, semantic segmentation, extract key points and excludes store result.



Figure# 9: UC Print Inspection

Key point matching

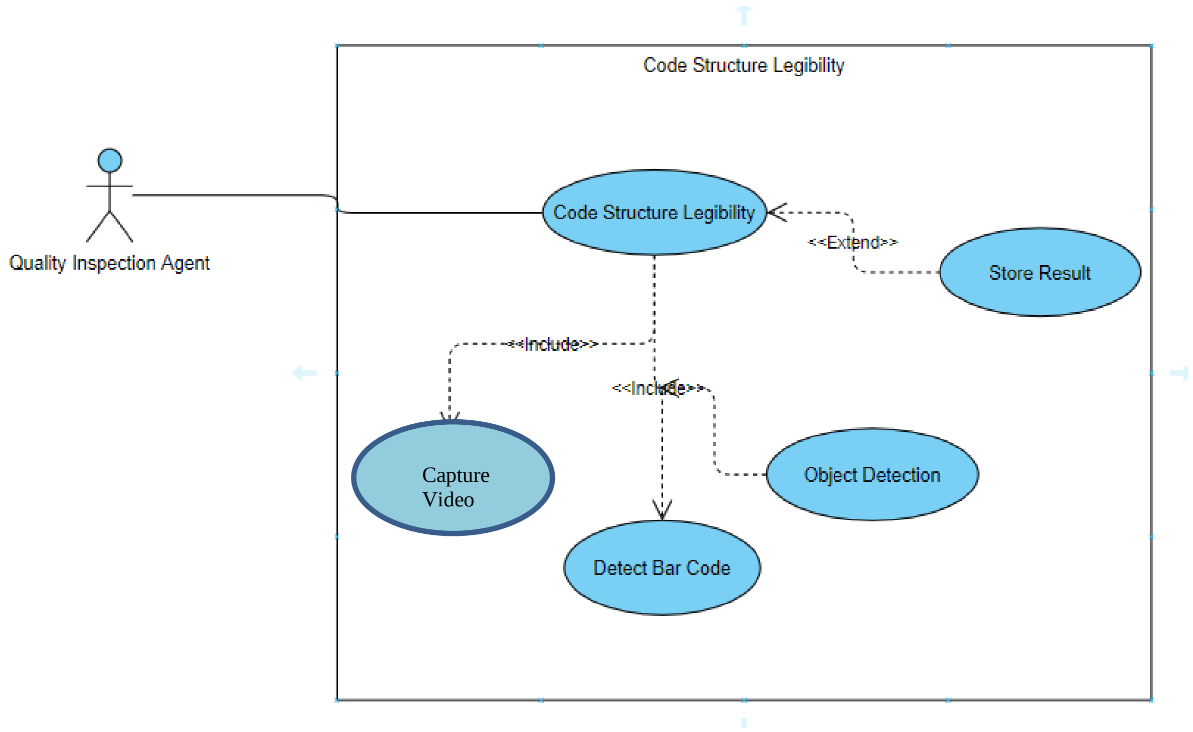
In this use case of key point matching, the actor is quality inspection agent. Key point matching includes extract key points and match key points.



Figure# 10: UC Key Point Matching

Code legibility inspection

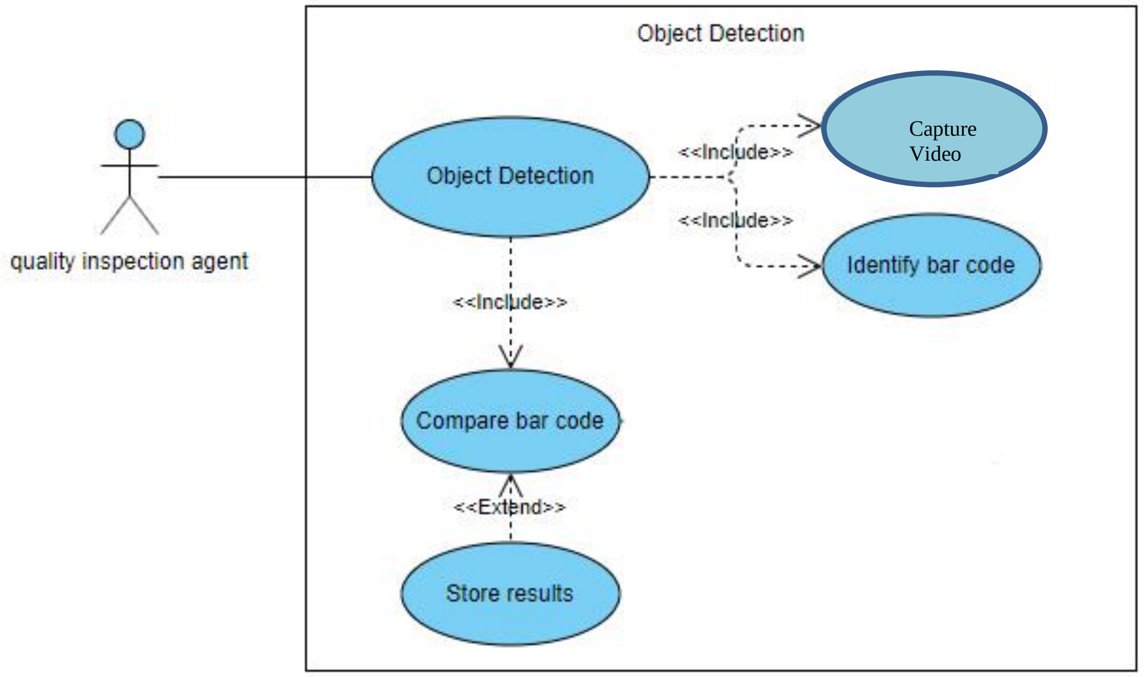
In this use case of code legibility inspection, the actor is quality inspection agent. code legibility inspection includes capture video, detect barcode, object detection and excludes store result.



Figure# 11: UC Code Legibility Inspection

Object detection

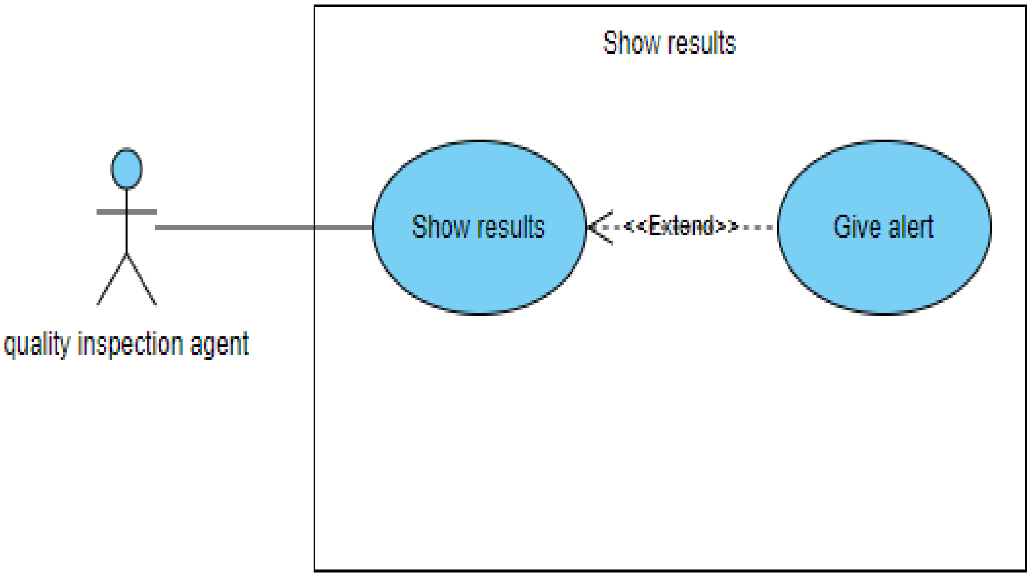
In this use case of object detection, the actor is quality inspection agent. Object detection includes capture video, identify barcode and compare barcode while compare barcode extends store result.



Figure# 12: UC Object Detection

Show results

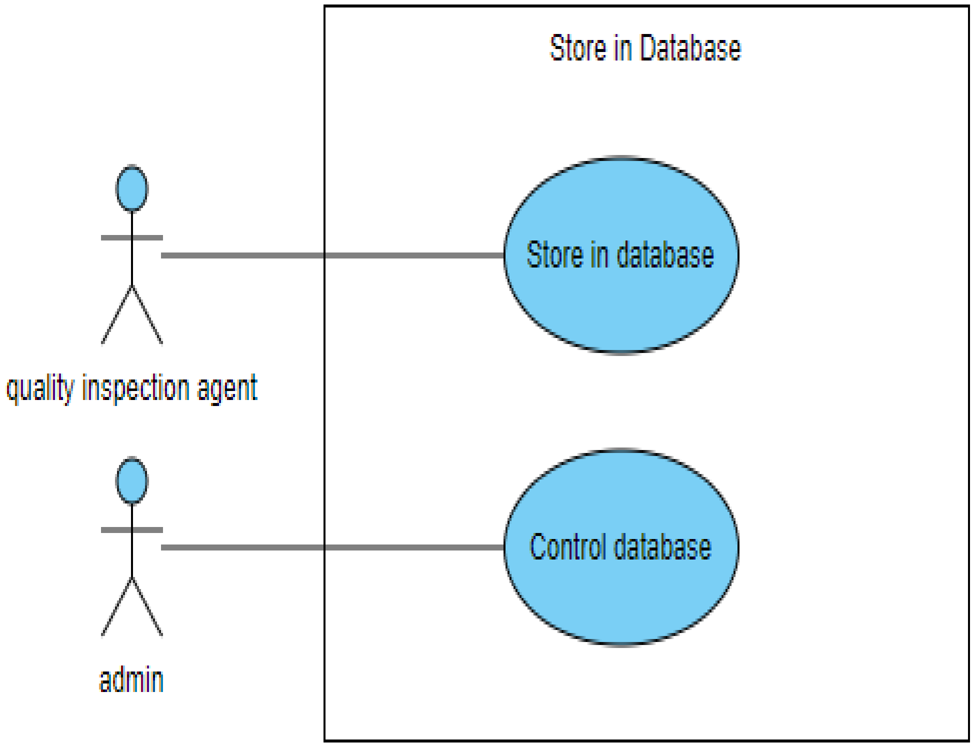
In this use case of show results, the actor is quality inspection agent. Store results extends give alert.



Figure# 13: UC Show Results

Store in database

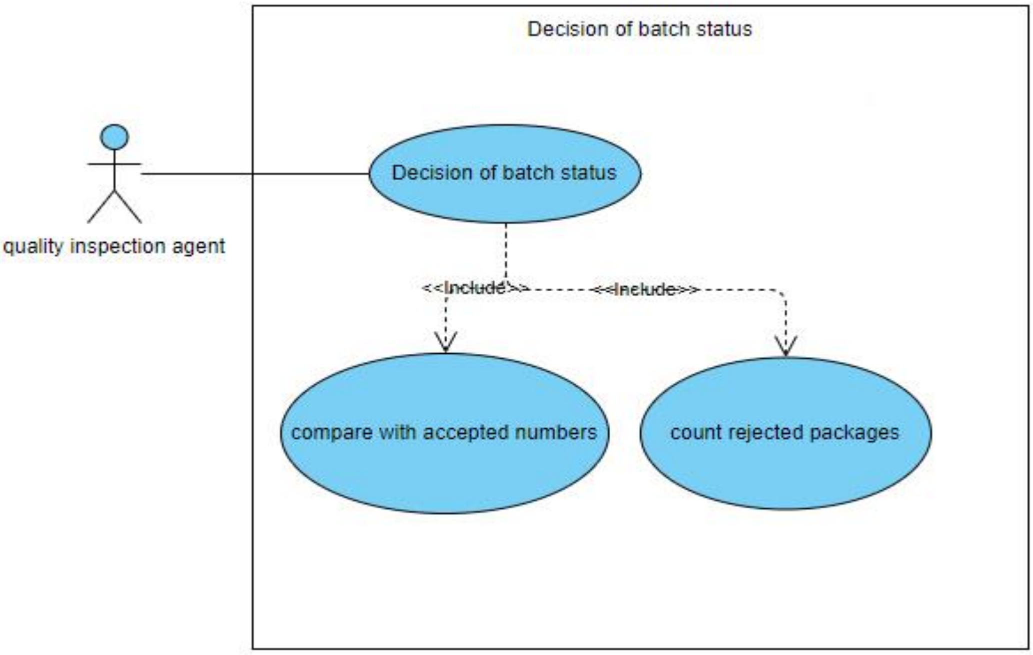
In this use case the actors are quality inspection agent and admin where quality inspection agent stores data in database and admin control database.



Figure# 14: UC Store in Database

Decision of batch status

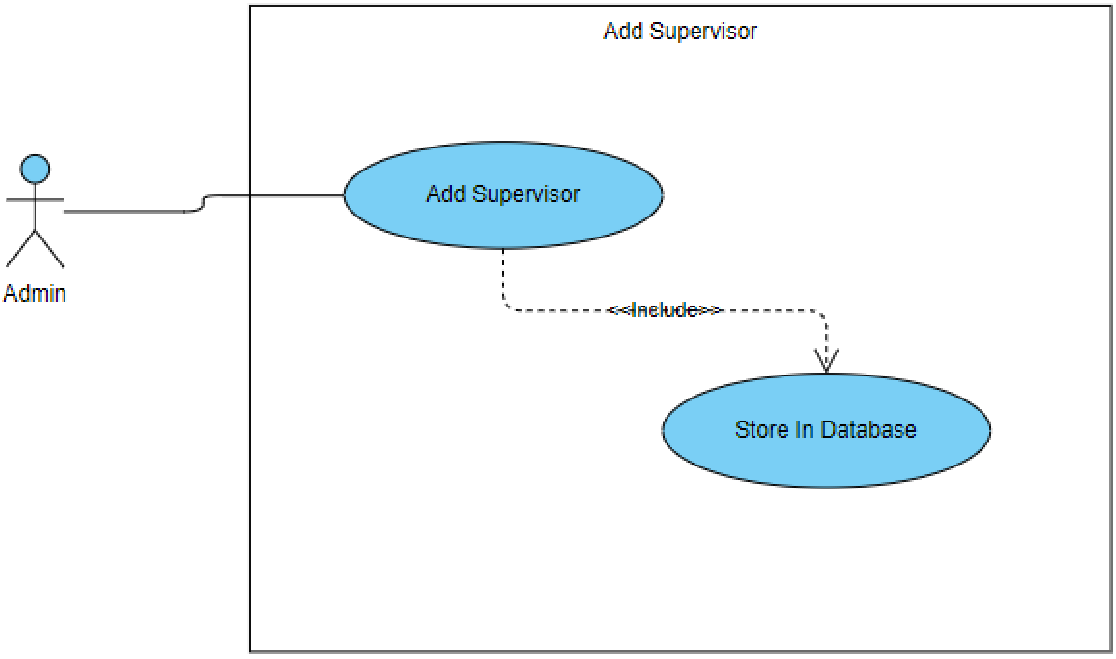
In this use case the actor is quality inspection agent.Decision of batch status includes compare with accepted numbers and count rejected packages.



Figure# 15: UC Decision of Batch status

Add Supervisor

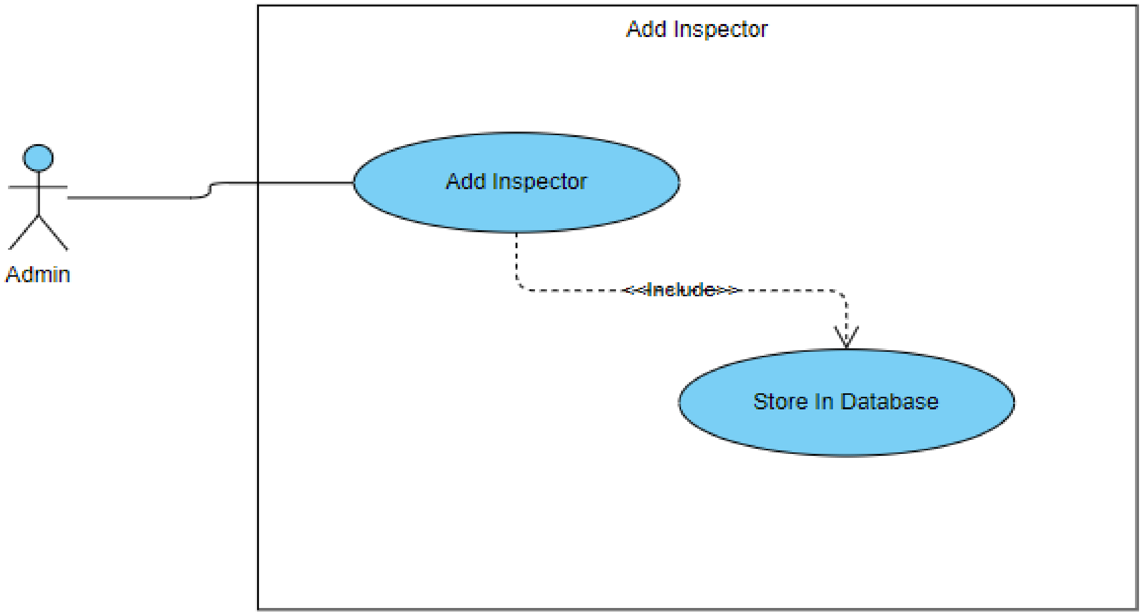
In this use case the actor is admin.Add supervisor include store in database.



Figure# 16: UC Add Supervisor

Add Inspector

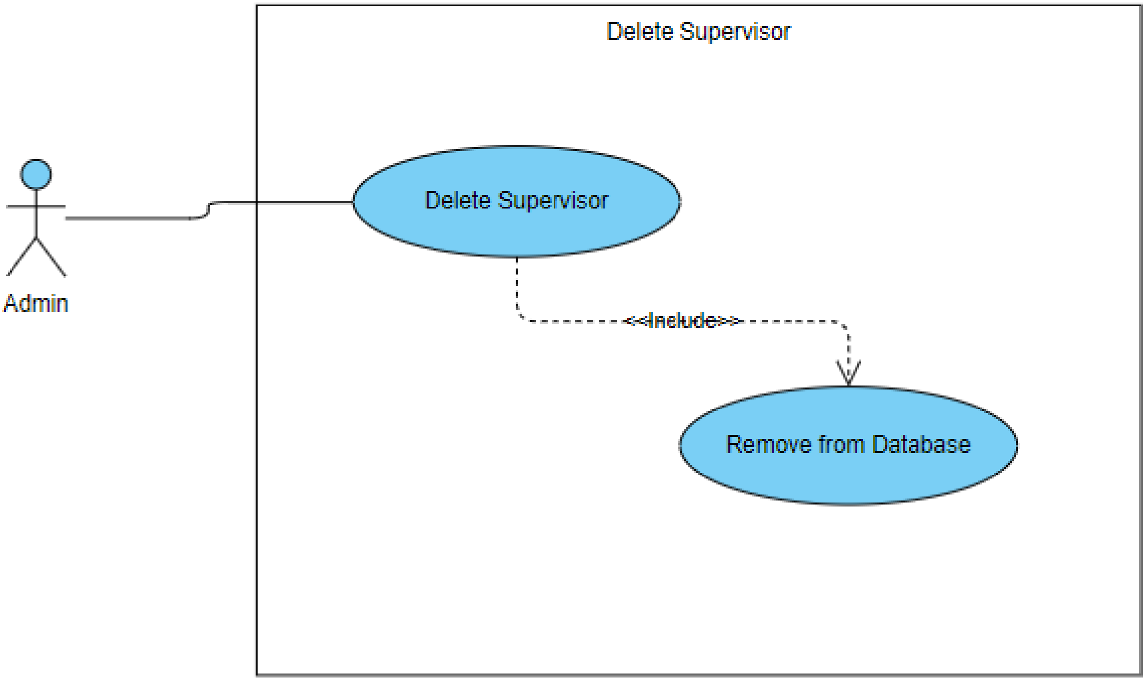
In this use case the actor is admin.Add Inspector include store in database.



Figure# 17: UC Add Inspector

Delete Supervisor

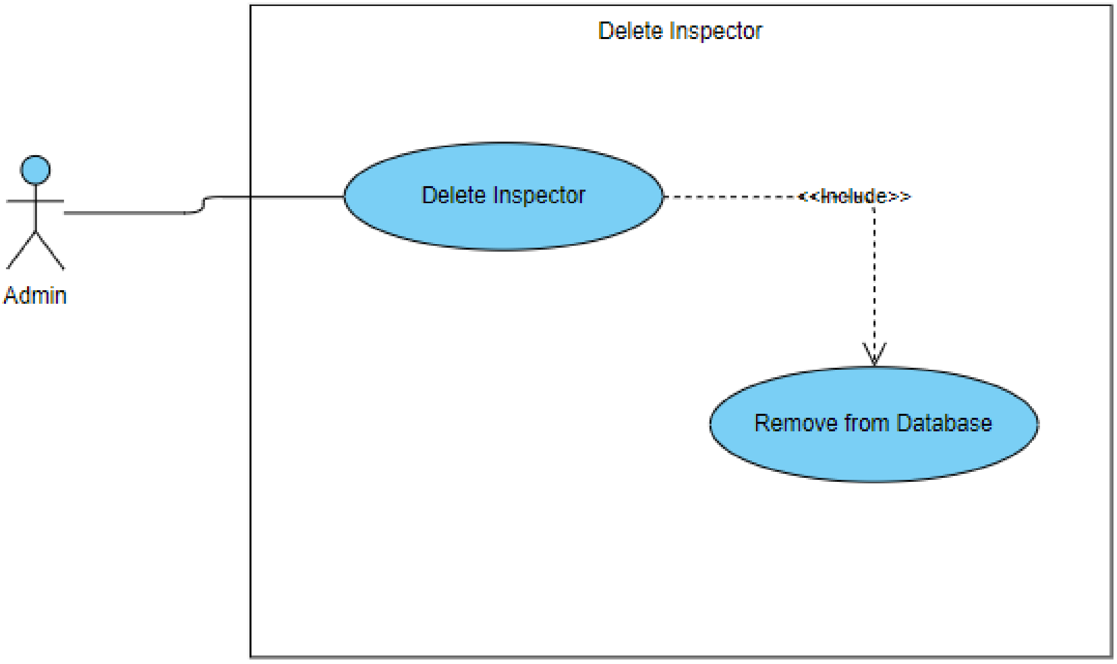
In this use case the actor is admin.Delete supervisor include delete from database.



Figure# 18: UC Delete Supervisor

Delete Inspector

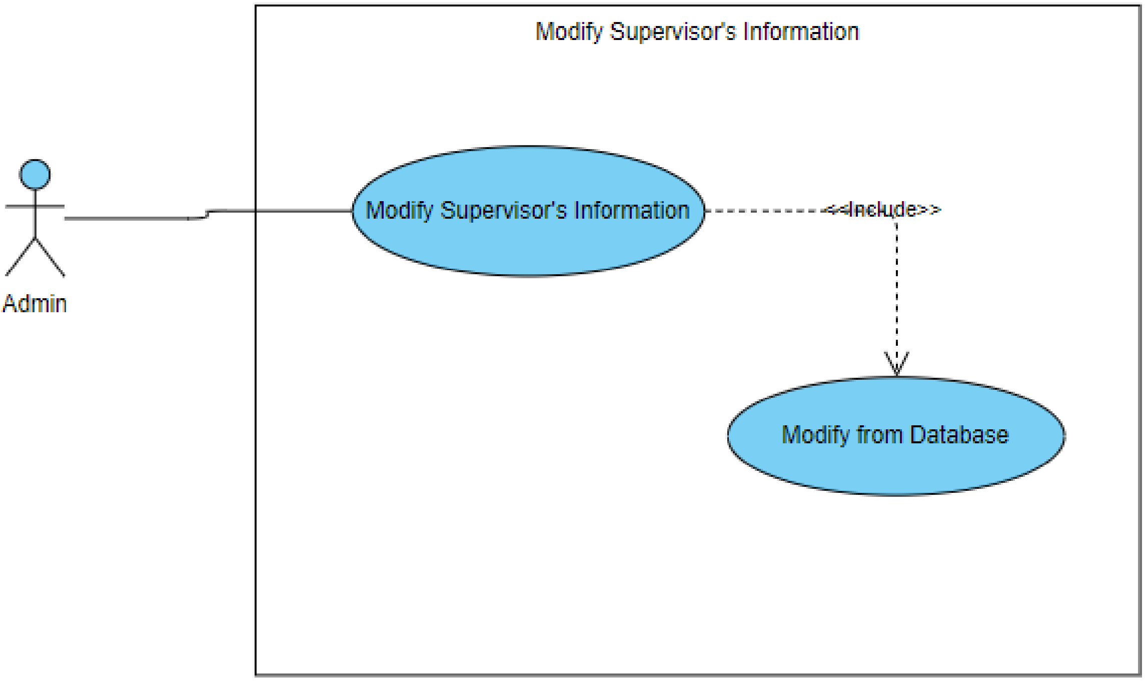
In this use case the actor is admin.Delete inspector include delete from database.



Figure# 19: Delete Inspector

Modify Supervisor’s Information

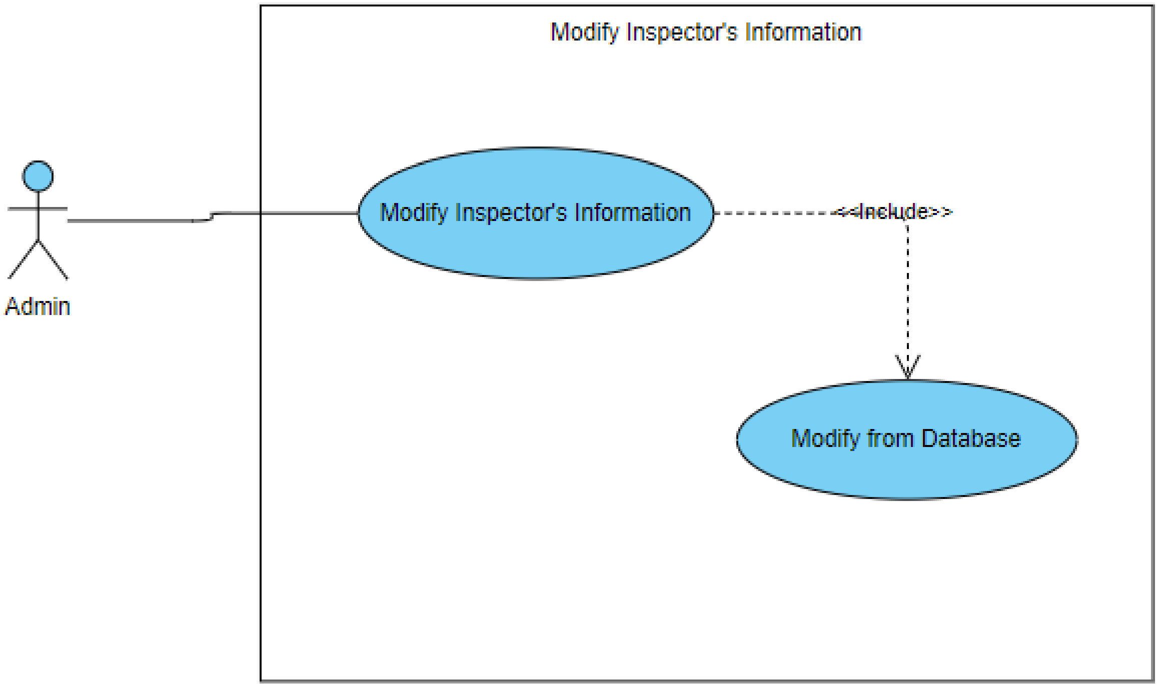
In this use case the actor is admin.Modify supervisor’s information include modify from database.


Modify Supervisor's Information
Admin
Modify Supervisor's Information
<<Include>>
Modify from Database

Figure# 20: UC Modify Supervisor's Information

Modify Inspector’s Information

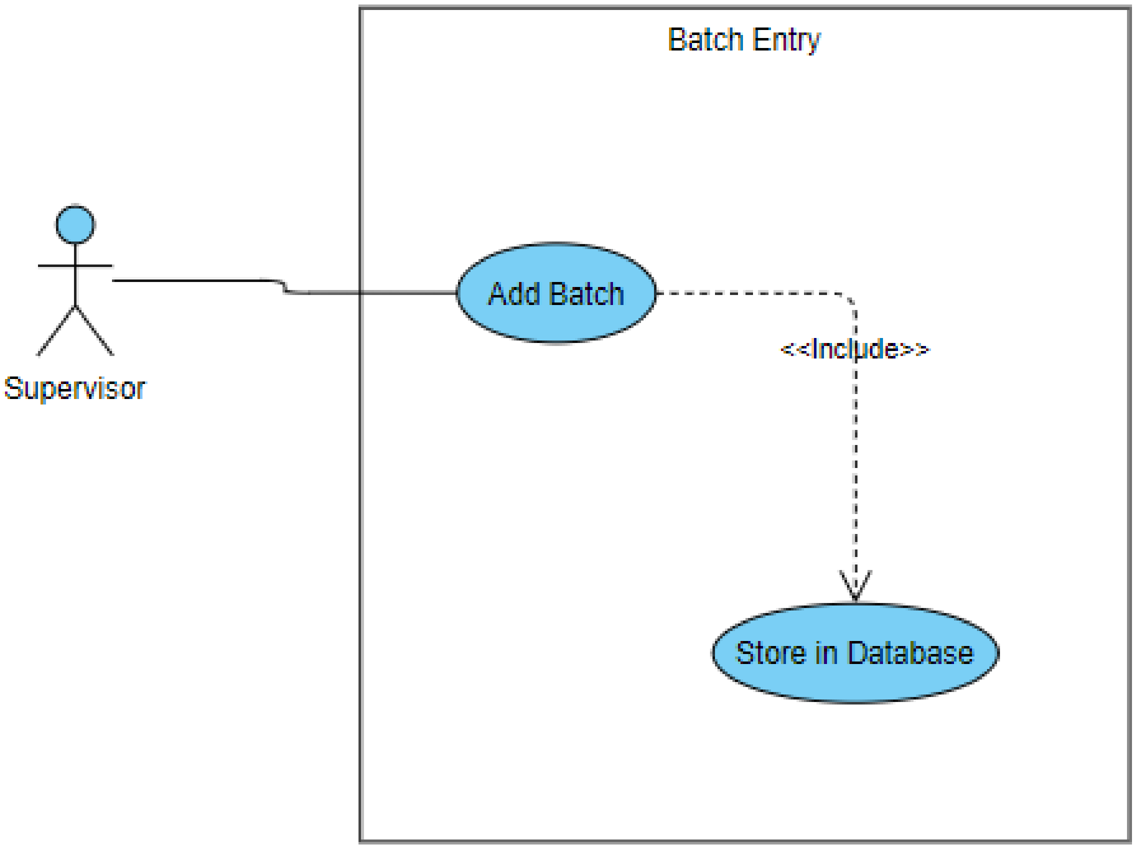
In this use case the actor is admin.Modify inspector’s information include modify from database.


Modify Inspector's Information
Admin
Modify Inspector's Information
<<Include>>
Modify from Database

Figure# 21: UC Modify Inspector's Information

Batch Entry

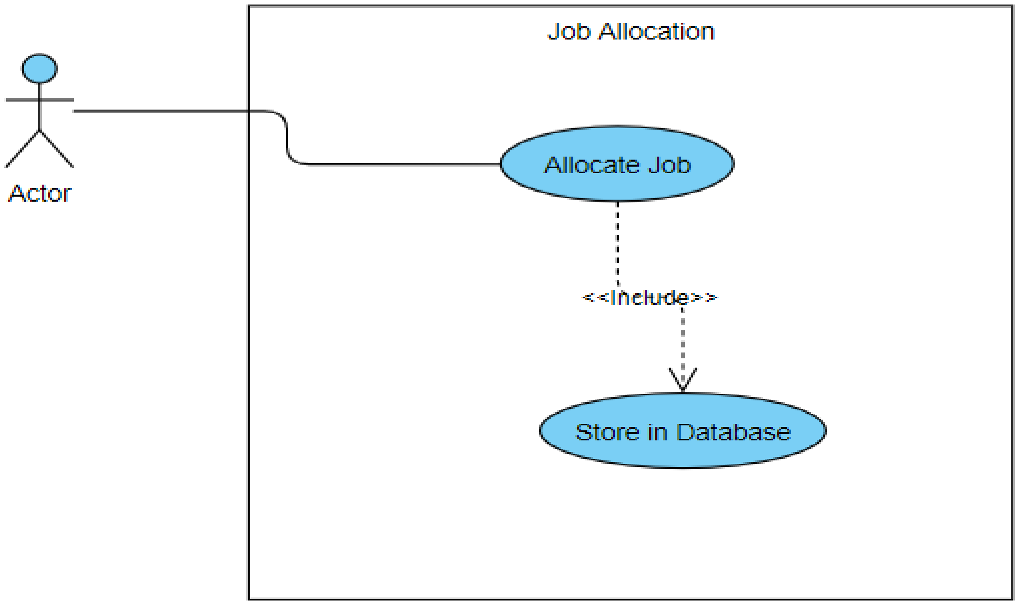
In this use case the actor is supervisor. Add Batch include store in database



Figure# 22: UC Batch Entry

Job Allocation

In this use case the actor is admin. Allocate job include store in database



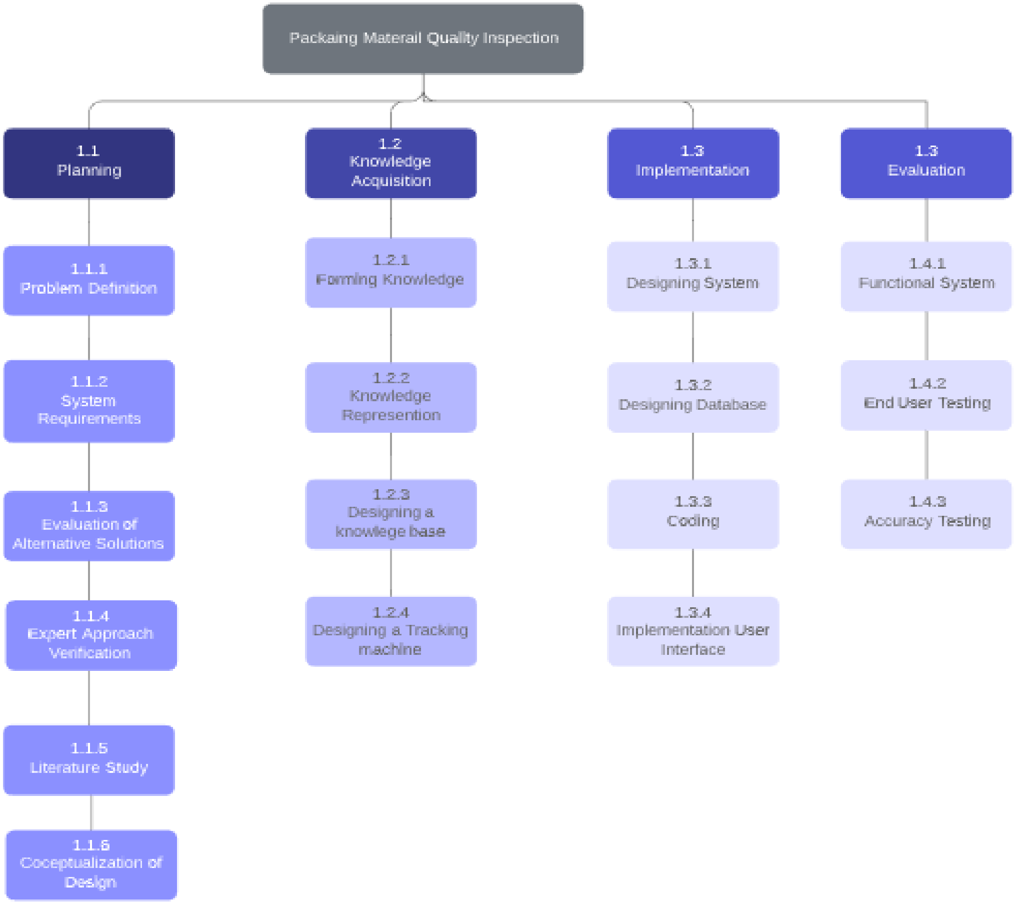
Figure# 23: UC Job Allocation

2.6 Software Development Life Cycle (SDLC) Model (justification on the consideration of the proposed model)

We decided to use Expert System Development Life Cycle (ESDLC) for our project. It is a computer based decision making system that shall use both heuristics and facts to solve complex decision making problems. We are using this development life cycle because our system is an AI software that shall use knowledge stored in a knowledge base to solve the problems, and that usually require a human expert because the problems are complex. The data in the knowledge base shall be added by humans that are experts of a particular domain, the software shall be use by non-exepert users (like operator) to acquire information. Another reason for using this development life cycle is that it improves the decsion quality and also helps us to get accurate and fast answers.

Chapter 3: Design of the System

3.1 Structure of Work breakdown (SWB)

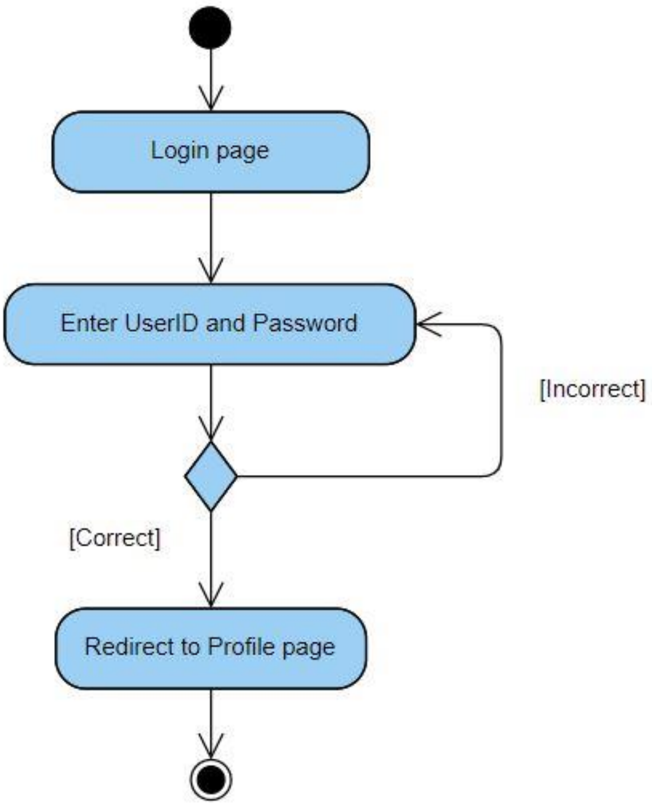

Packaing Materail Quality Inspection
1.1 Planning
1.1.1 Problem Definition
1.1.2 System Requirements
1.1.3 Evaluation of Alternative Solutions
1.1.4 Expert Approach Verification
1.1.5 Literature Study
1.1.6 Coceptualization of Design
1.2 Knowledge Acquisition
1.2.1 Forming Knowledge
1.2.2 Knowledge Represention
1.2.3 Designing a knowlege base
1.2.4 Designing a Tracking machine
1.3 Implementation
1.3.1 Designing System
1.3.2 Designing Database
1.3.3 Coding
1.3.4 Implementation User Interface
1.3 Evaluation
1.4.1 Functional System
1.4.2 End User Testing
1.4.3 Accuracy Testing

Figure# 4: WBS

3.2 Activity Diagrams

3.2.1 Login:

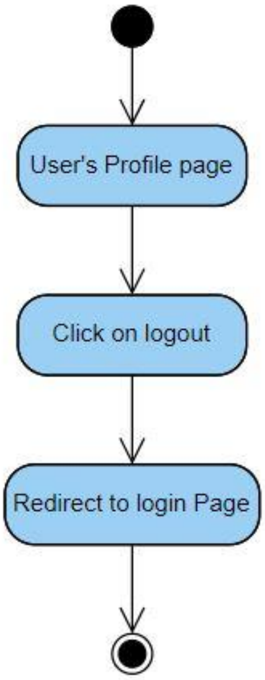
The User will see the Login page at the start then he will enter his User ID and password if it is incorrect then he will see then login page again with some alert else he will be redirected to Profile Page.



Figure# 5: Activity Login

3.2.2 Logout:

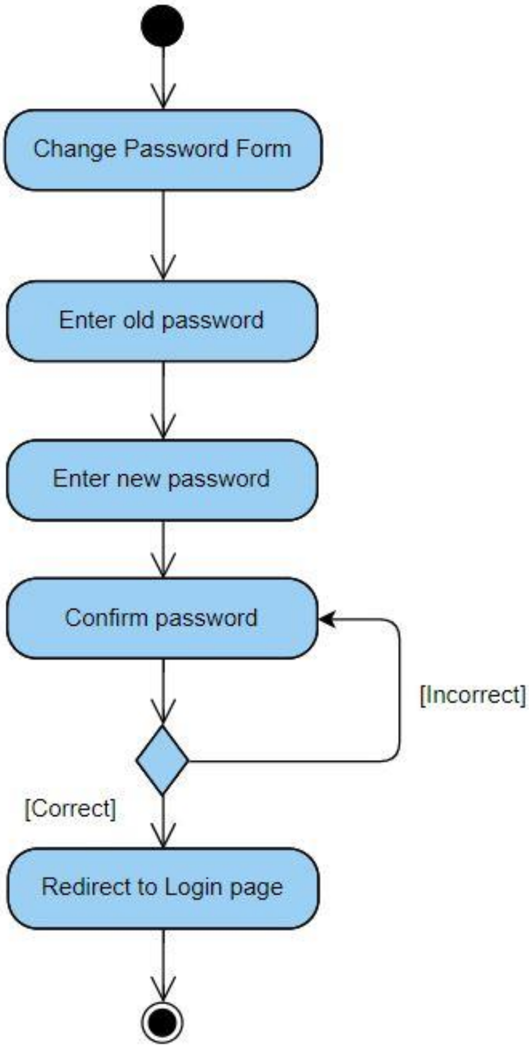
On User Profile Page there is a logout button at the left bottom of the window the user just simple click on the logout button and confirmation dialogue will appear if he clicks yes then he will be redirected to login page.



Figure# 6: Activity Logout

3.2.3 Change Password:

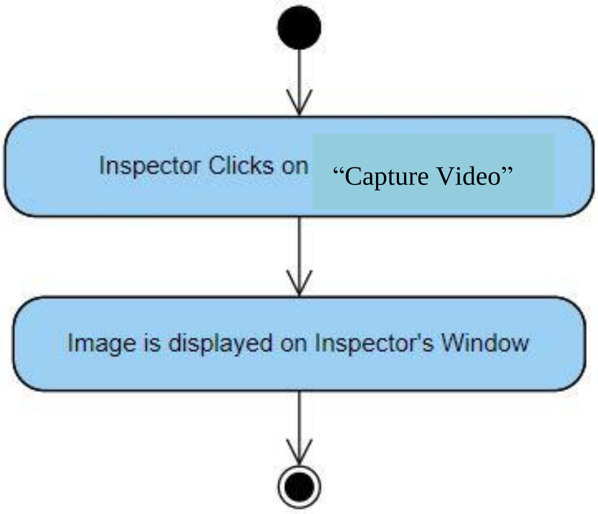
At change password Form user will simply enter the old password, new password and confirm password. If new password and confirm password are same and old password match the user password, then it will change the password and redirect it on to login page.



Figure# 7: Activity Change Password

3.2.4 Capture Video:

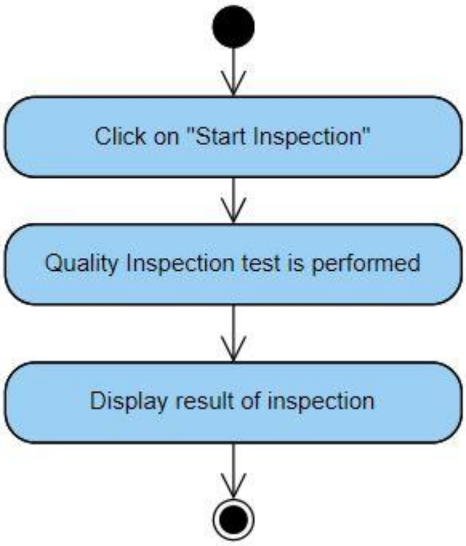
User will simply click on the capture video button and the capturing screen window will be shown on the window screen.



Figure# 29: Activity Capture Video

3.2.5 Start Inspection:

User will enter the start inspection button after clicking the button it will apply the inspections algo’s on video frames and show the result on the screen.


Click on "Start Inspection"
Quality Inspection test is performed
Display result of inspection

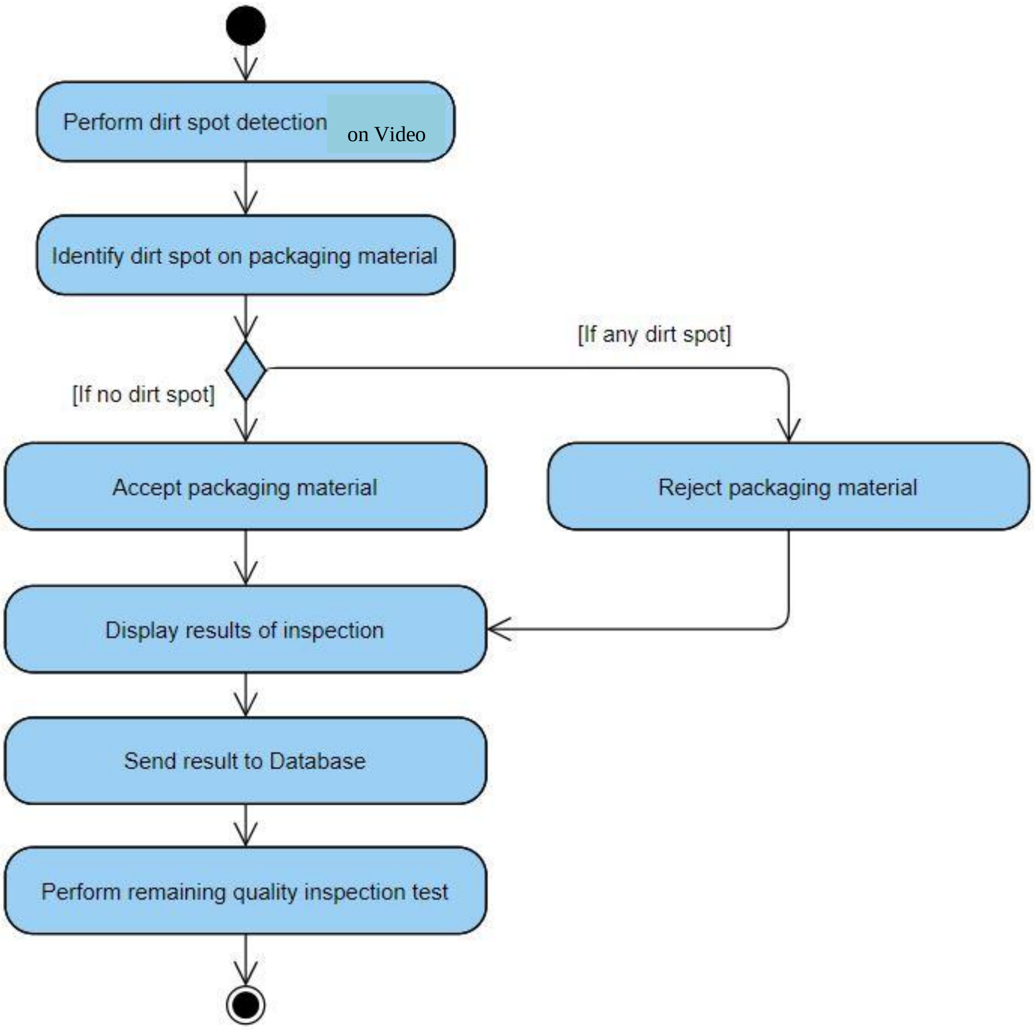
Figure# 30: Activity Start Inspection

3.2.6 Dirt spot Inspection:

A frame from the video is extract using different technique and then that frame will be send to the algorithm if there is a dirt spot on the frame then just simple reject the frame.

Else accept the packaging material and display the result of inspection.

Send this frame id and result into database and then take the next frame from video.

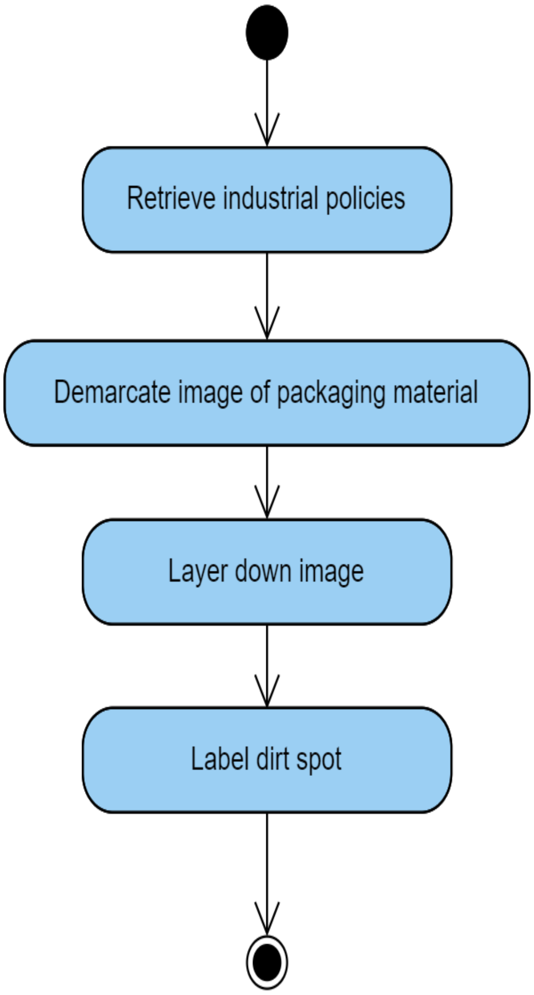

Perform dirt spot detection
on Video
Identify dirt spot on packaging material
[If any dirt spot]
[If no dirt spot]
Accept packaging material
Reject packaging material
Display results of inspection
Send result to Database
Perform remaining quality inspection test

Figure# 31: Activity Dirt Spot Inspection

3.2.7 Semantic segmentation:

Semantic segmentation is the technique to divide the frame into the segments according to its frame.

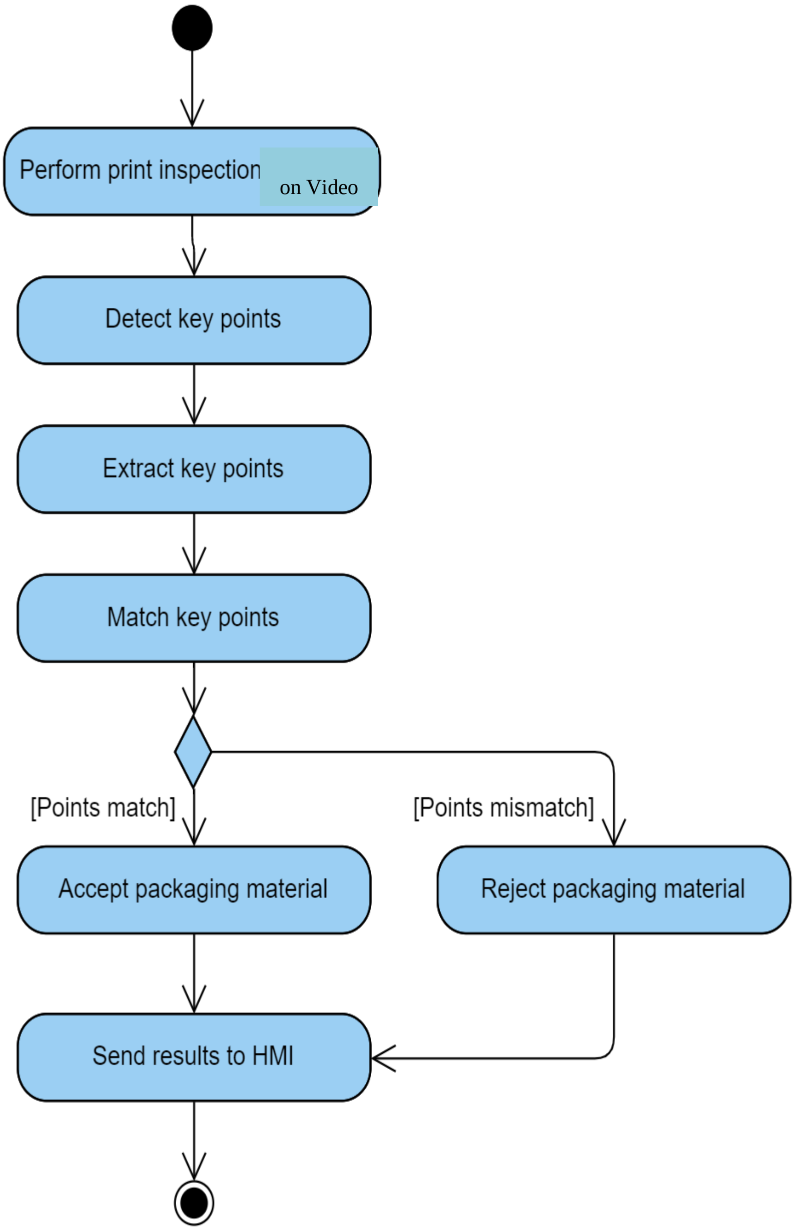
So we just simple apply the dirt spot inspection on the packet's after briefly observing the industrial policies.


Retrieve industrial policies
Demarcate image of packaging material
Layer down image
Label dirt spot

Figure# 32: Activity Semantic Segmentation

3.2.8 Print inspection:

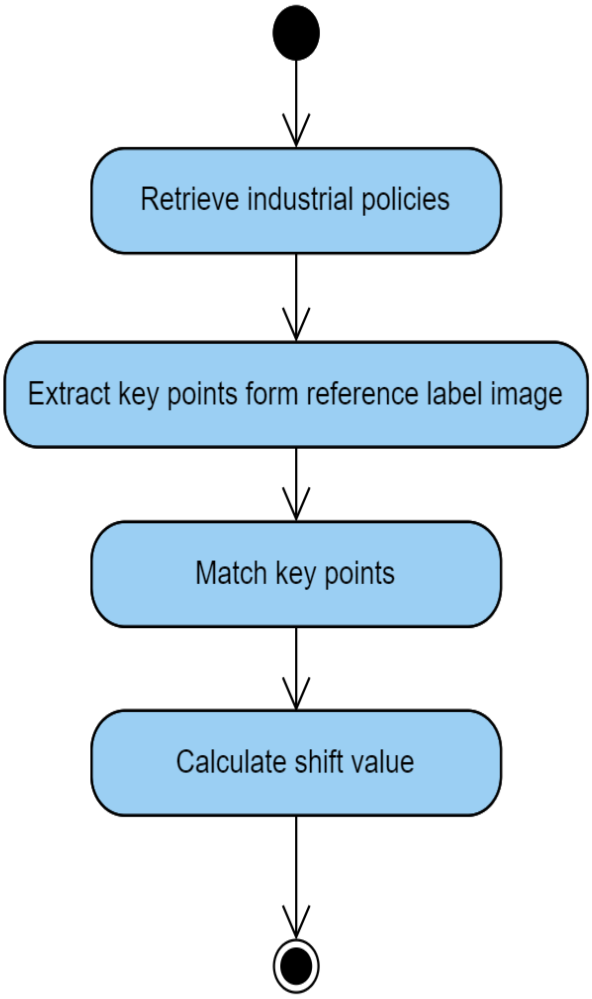
In print inspection we will extract the frame from the video then we detect the key points from the frames, then we extract that key points and apply the inspection if packet match then we accept the packet else reject the packet.



Figure# 33: Activity Print Inspection

3.2.9 Key point matching:

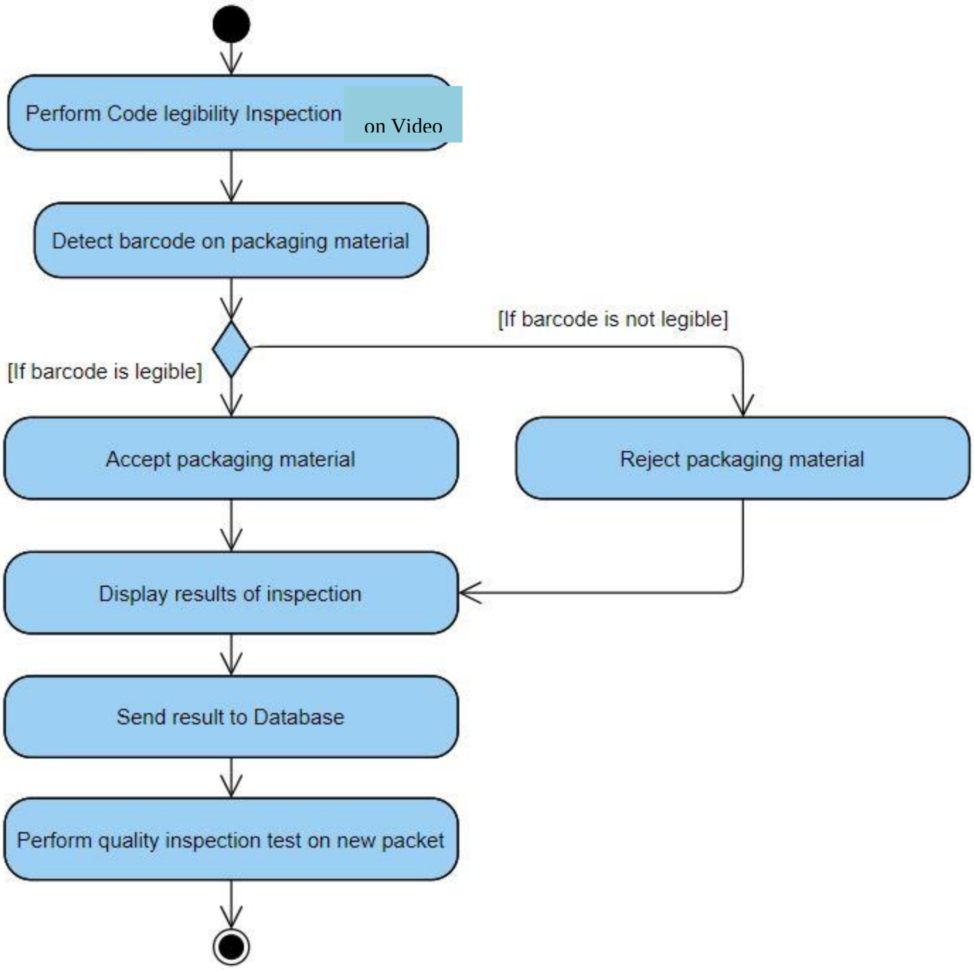
In key point matching we extract the key points from the label images and then match that images with correct image and then find the shift value from it.


Retrieve industrial policies
Extract key points form reference label image
Match key points
Calculate shift value

Figure# 34: Activity Key point matching

3.2.10 Code legibility inspection:

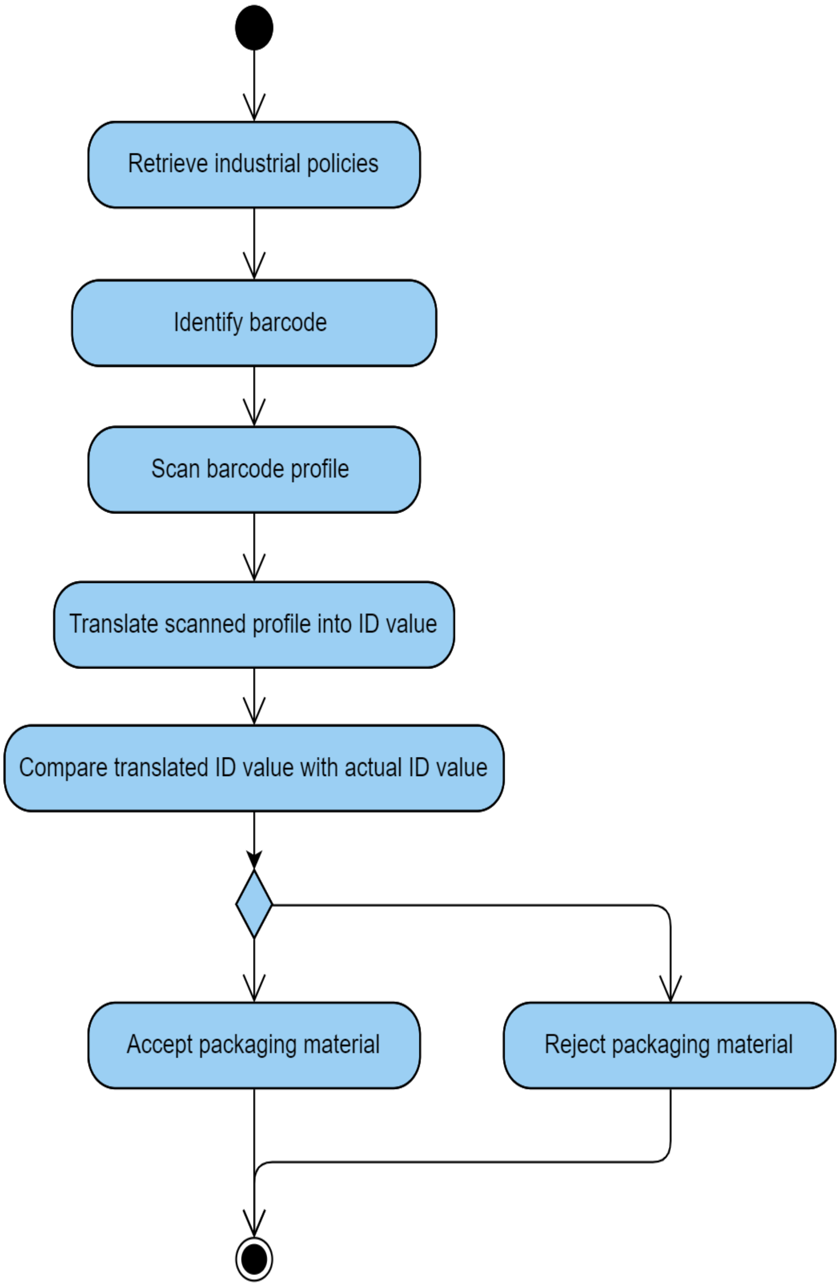
Convert the video into frames then we apply the barcode detection and inspection algo on it if it is eligible then Accept the packet and pass the result to both screen and database.


Perform Code legibility Inspection
on Video
Detect barcode on packaging material
[If barcode is not legible]
[If barcode is legible]
Accept packaging material
Reject packaging material
Display results of inspection
Send result to Database
Perform quality inspection test on new packet

Figure# 35: Activity Code legibility

3.2.11 Object detection:

In object detection we first identify the barcode and then scan the barcode profile after extracting the specific key point we convert it into ID value then we match the extract id value with real value if match then accept the packet else reject the packet.

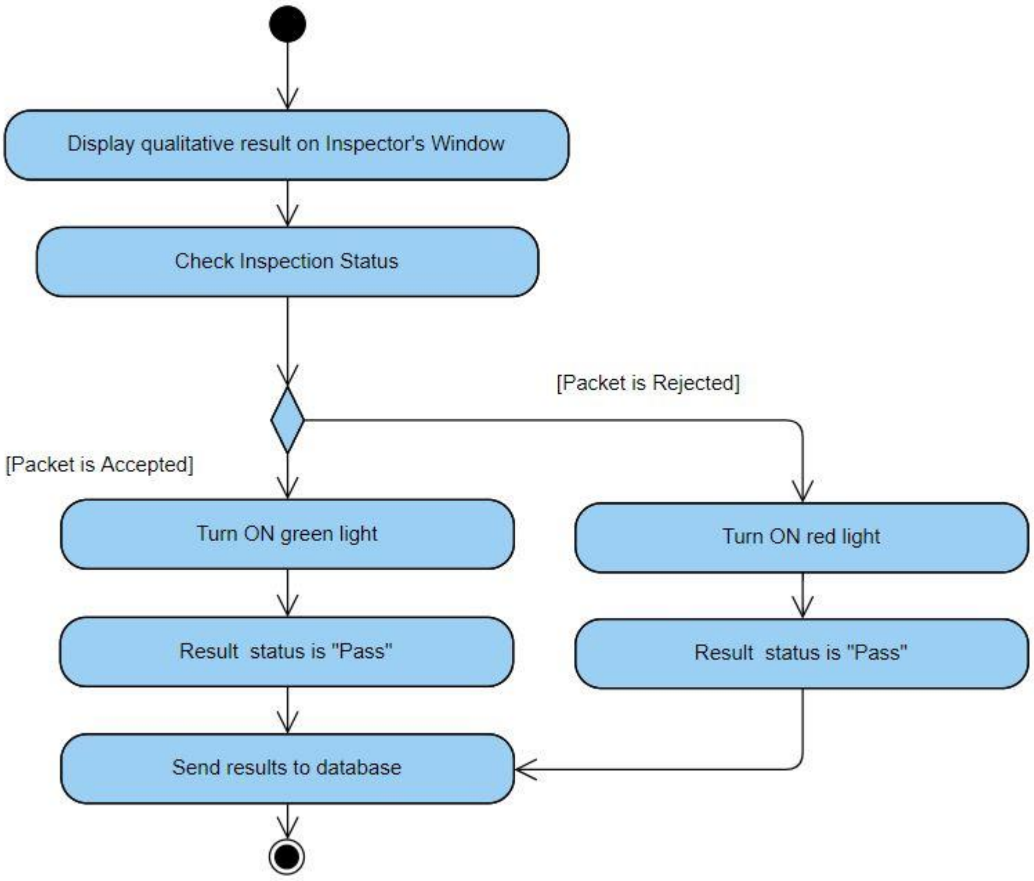

Retrieve industrial policies
Identify barcode
Scan barcode profile
Translate scanned profile into ID value
Compare translated ID value with actual ID value
Accept packaging material
Reject packaging material

Figure# 36: Activity Object Detection

3.2.12 Show results:

The user will see the inspection result on desktop window.

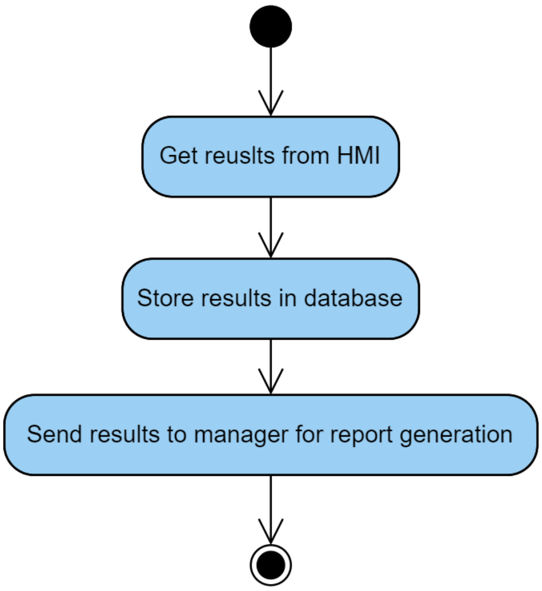
If after inspection the packet is pass the computer will on green light else red light.



Figure# 37: Activity Show Results

3.2.13 Store in database:

The system will send the results at the backend server. So that manager can easily check the reports.


Get reuslts from HMI
Store results in database
Send results to manager for report generation

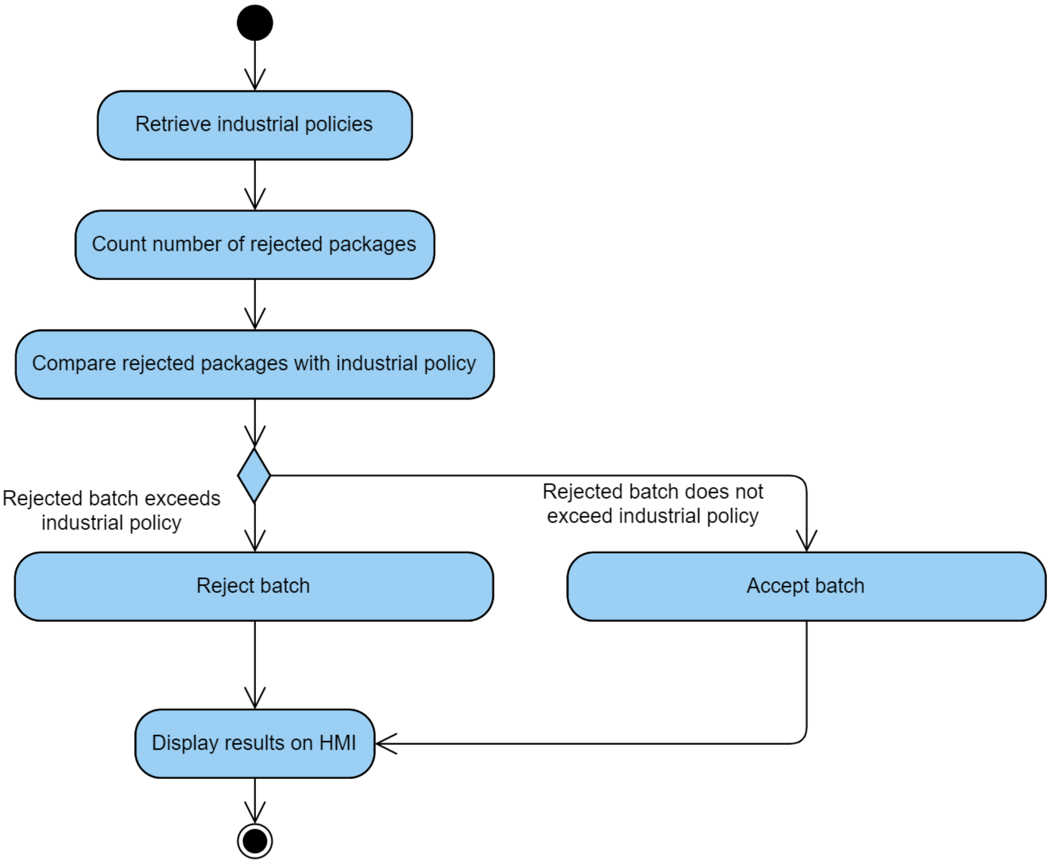
Figure# 38: Activity Store in database

3.2.14 Decision of batch status:

So, the real task is to accept or reject the whole batch base on their packets printing.

So, we will examine all the packets of batch then just simple counter the accept packet and reject packets.

If accept packet counter is greater than other than accept the batch else reject.


Retrieve industrial policies
Count number of rejected packages
Compare rejected packages with industrial policy
Rejected batch exceeds industrial policy
Rejected batch does not exceed industrial policy
Reject batch
Accept batch
Display results on HMI

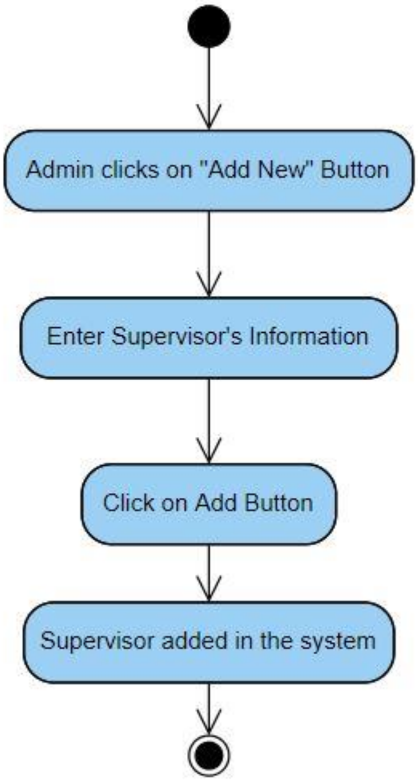
Figure# 39: Activity Decision of Batch Status

3.2.15 Add Supervisor:

Click the add supervisor button to add new supervisor.

Then type his information and click add button.

The supervisor will be added to the system.


Admin clicks on "Add New" Button
Enter Supervisor's Information
Click on Add Button
Supervisor added in the system

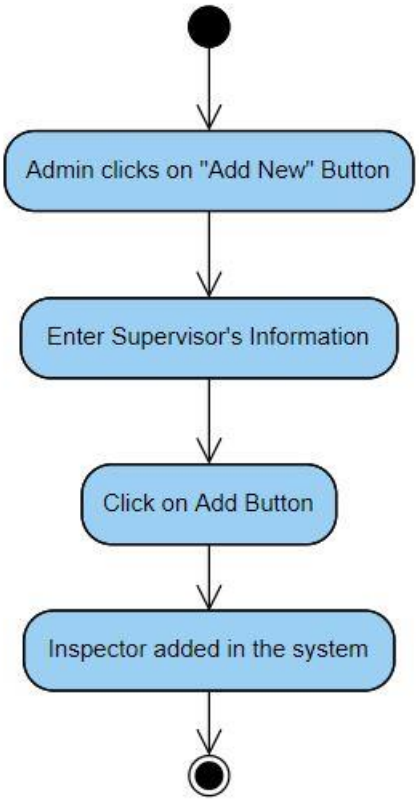
Figure# 40: Activity Add Supervisor

3.2.16 Add Inspector:

Click the add inspector button to add new inspector.

Then type his information and click add button.

The inspector will be added to the system.

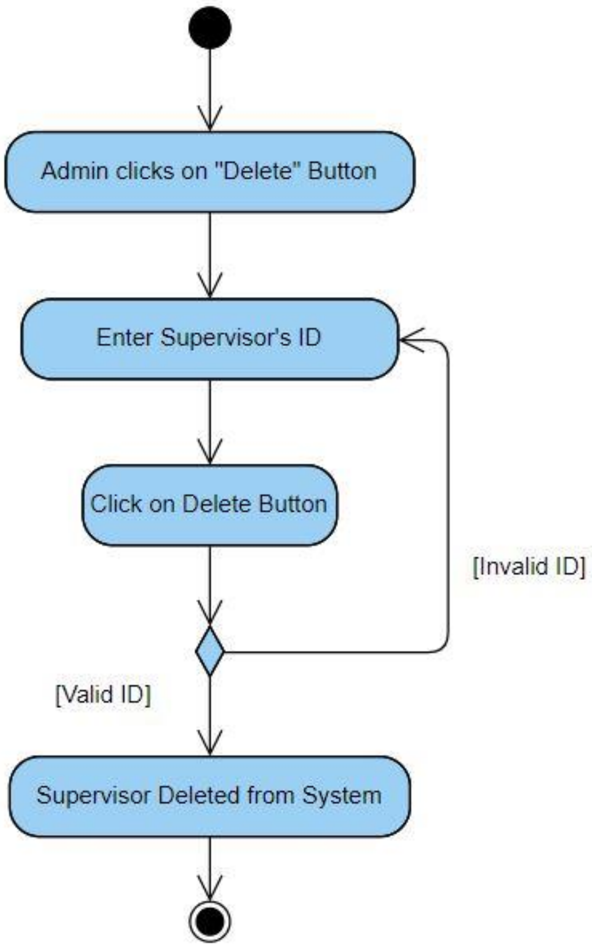

Admin clicks on "Add New" Button
Enter Supervisor's Information
Click on Add Button
Inspector added in the system

Figure# 41: Activity Add Inspector

3.2.17 Delete Supervisor:

Admin click the delete button and then enter the supervisor id.

If user exist, then delete it.

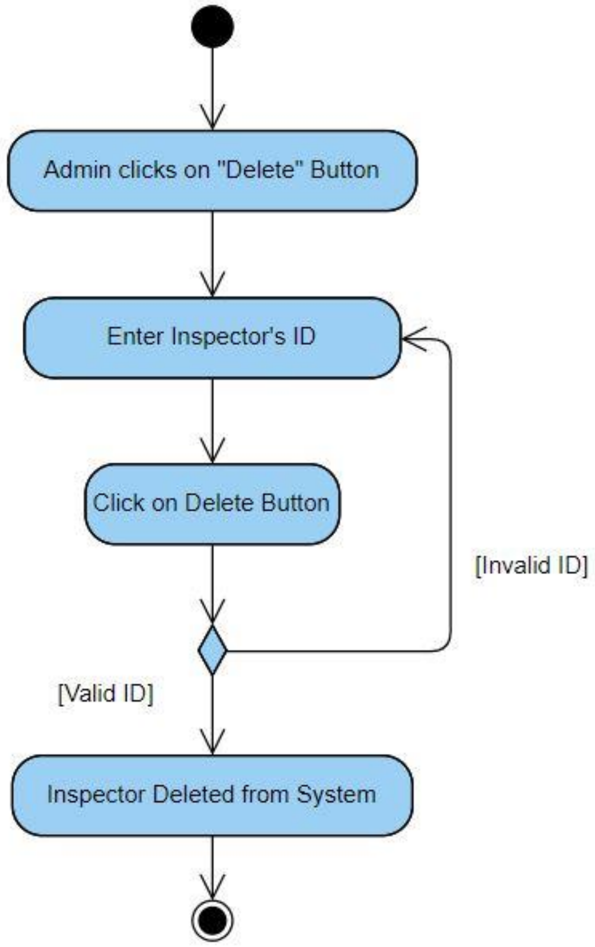


Figure# 42: Activity Delete Supervisor

3.2.18 Delete Inspector:

Admin click the delete button and then enter the inspector id.

If user exist, then delete it.

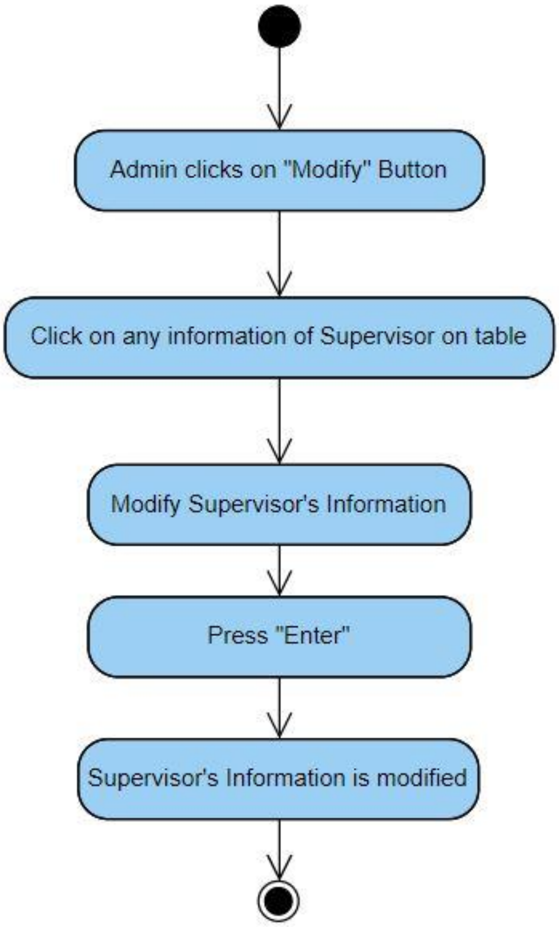


Figure# 43: Activity Delete Inspector

3.2.19 Modify Supervisor’s Information:

Admin click on edit button and select the supervisor from the table whom information he want to edit.

Then enter the new information of supervisor and then click enter.

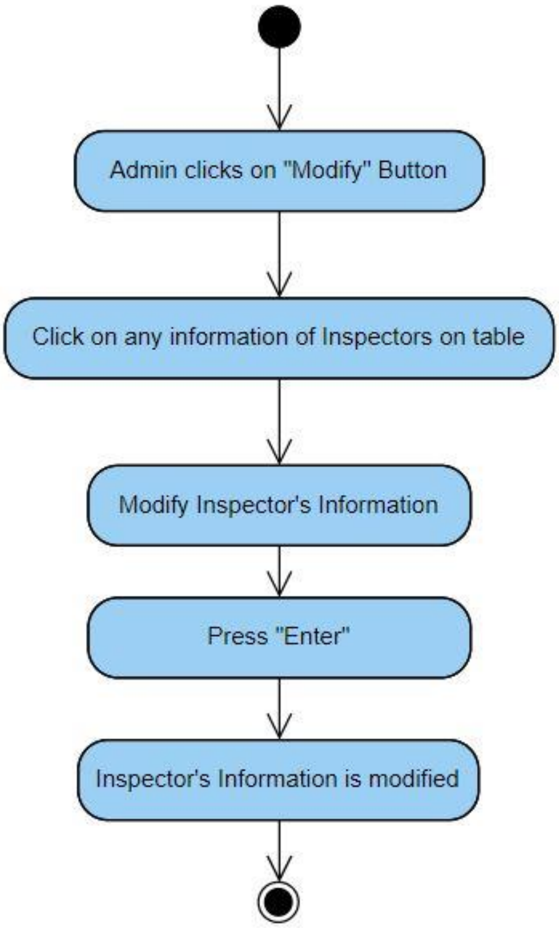

Admin clicks on "Modify" Button
Click on any information of Supervisor on table
Modify Supervisor's Information
Press "Enter"
Supervisor's Information is modified

Figure# 44: Activity Modify Supervisor's Information

3.2.20 Modify Inspector's Information:

Admin click on edit button and select the inspector from the table whom information he want to edit.

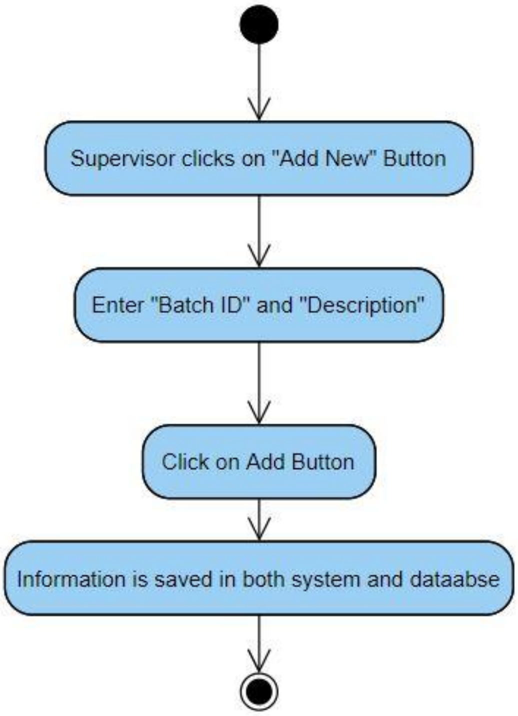
Then enter the new information of inspector and then click enter.


Admin clicks on "Modify" Button
Click on any information of Inspectors on table
Modify Inspector's Information
Press "Enter"
Inspector's Information is modified

Figure# 45: Activity Modify Inspector’s Information

3.2.21 Batch Entry:

Supervisor click on the New Batch Entry button then click Batch ID and Description then click on Add button so new data will be added to database.

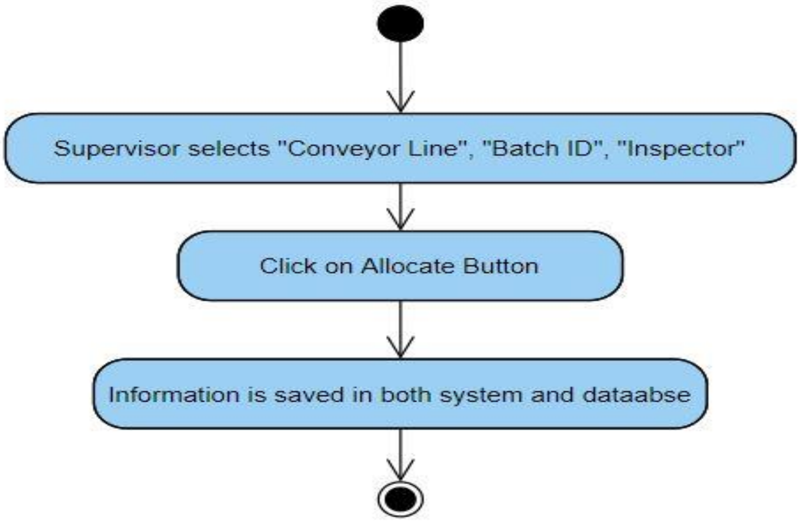

Supervisor clicks on "Add New" Button
Enter "Batch ID" and "Description"
Click on Add Button
Information is saved in both system and dataabse

Figure# 46: Activity Batch Entry

3.2.22 Job Allocation:

Supervisor will open the Job allocation page and just simple add the information and click allocate button.

Information will be receive to others.


Supervisor selects "Conveyor Line", "Batch ID", "Inspector"
Click on Allocate Button
Information is saved in both system and dataabse

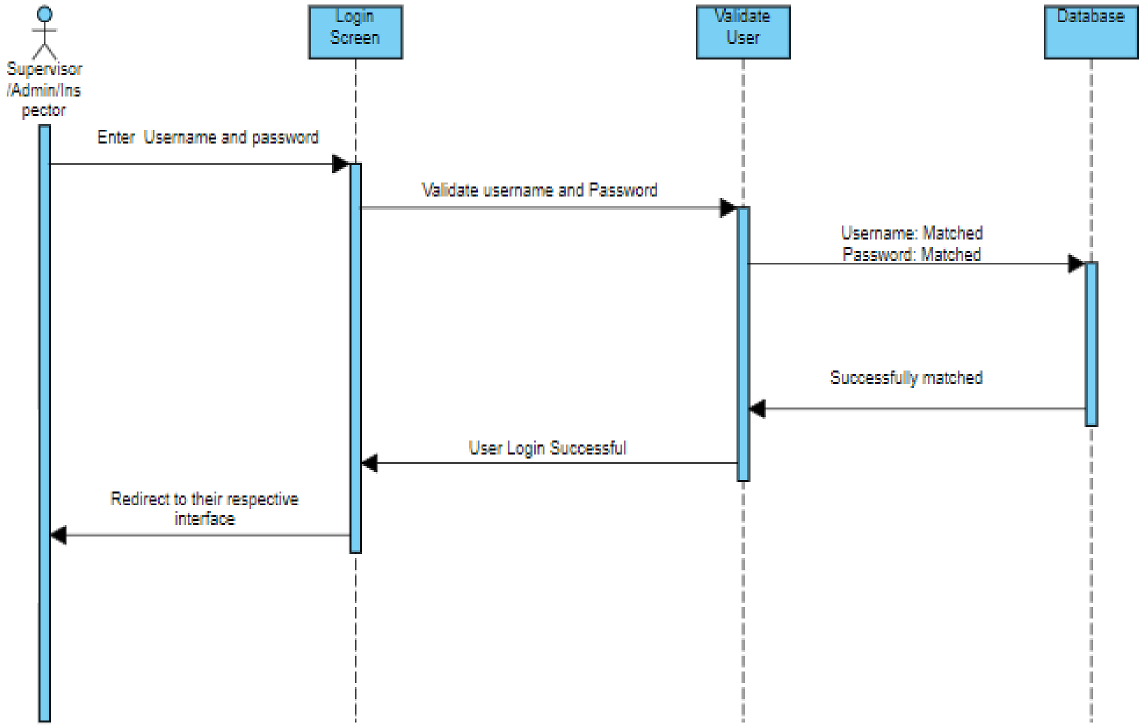
Figure# 47: Activity Job Allocation

3.3 Sequence Diagrams

3.3.1 Login:

User will provide email and password on login screen. Entered data will be validated then authenticated from database and redirected to main screen if authenticated.

If data validated correctly

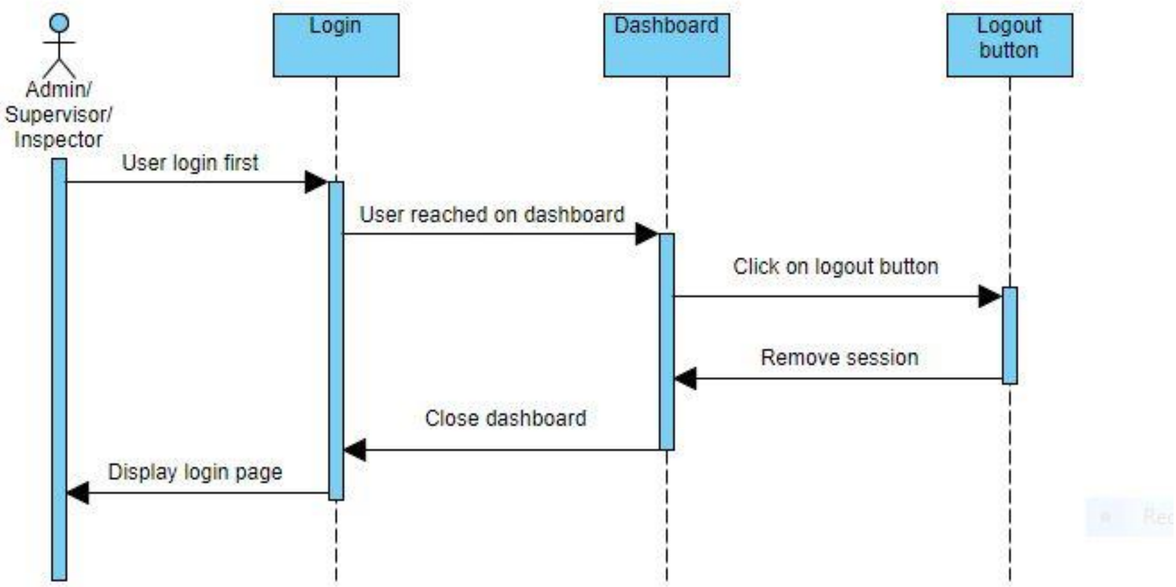


Supervisor /Admin/Inspector
Login Screen
Validate User
Database
Enter Username and password
Validate username and Password
Username: Matched
Password: Matched
Successfully matched
User Login Successful
Redirect to their respective interface

Figure# 8: Sequence Login

3.3.2 Logout

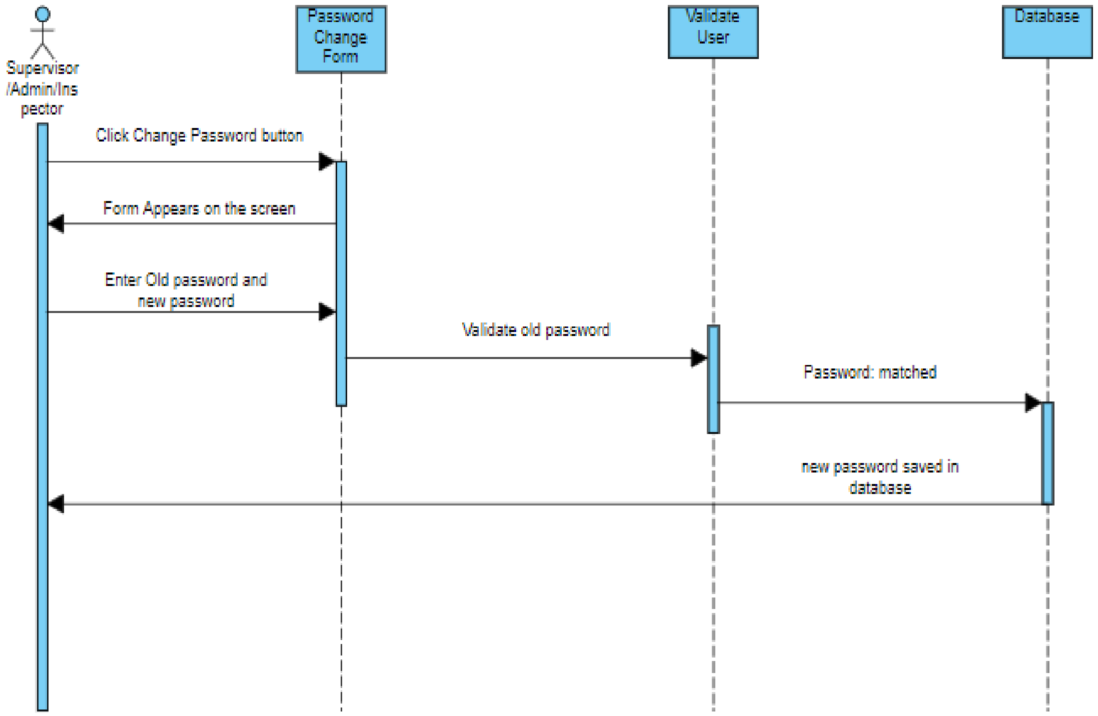
If user is logged in, then he/she can log out by clicking logout button.



Figure# 9: Sequence Logout

3.3.3 Change Password

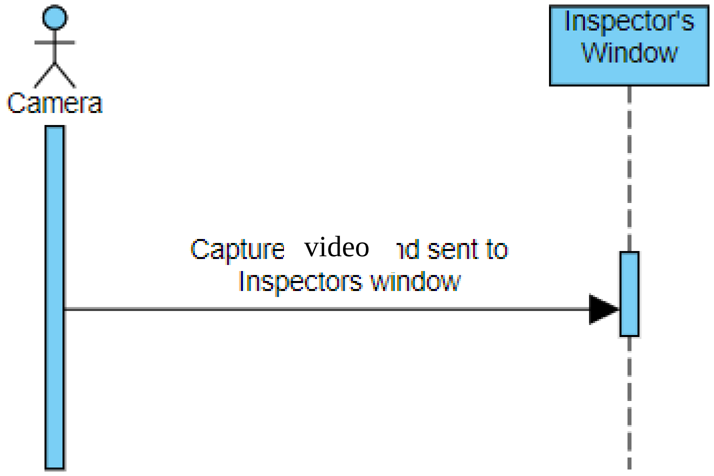
User can change his password by entering his new and old password if old passwords are validated then password will change.


Supervisor /Admin/Ins pector
Password Change Form
Validate User
Database
Click Change Password button
Form Appears on the screen
Enter Old password and new password
Validate old password
Password: matched
new password saved in database

Figure# 10: Sequence Change Password

3.3.4 Capture Video

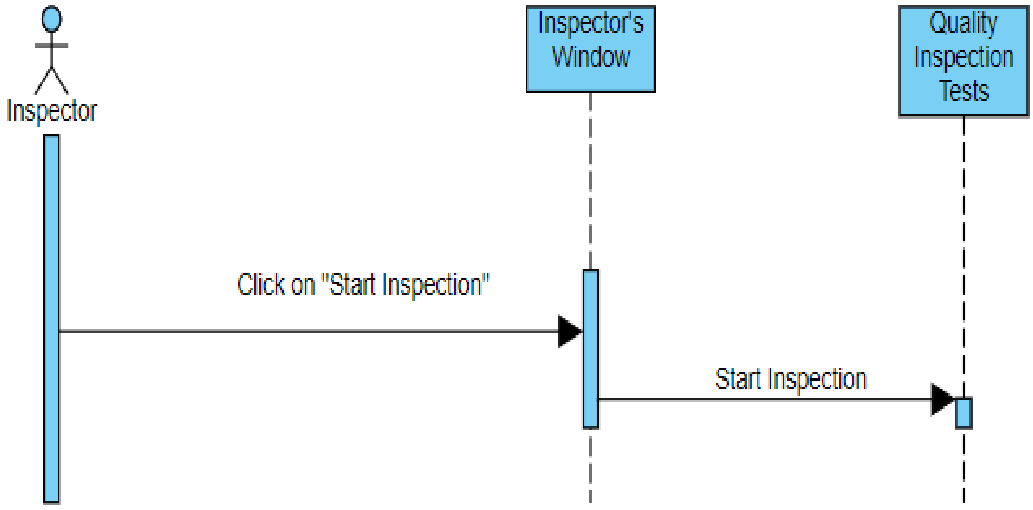
Camera will capture video from packaging line and send it to inspector.



Figure# 52: Sequence Capture Video

3.3.5 Start Inspection

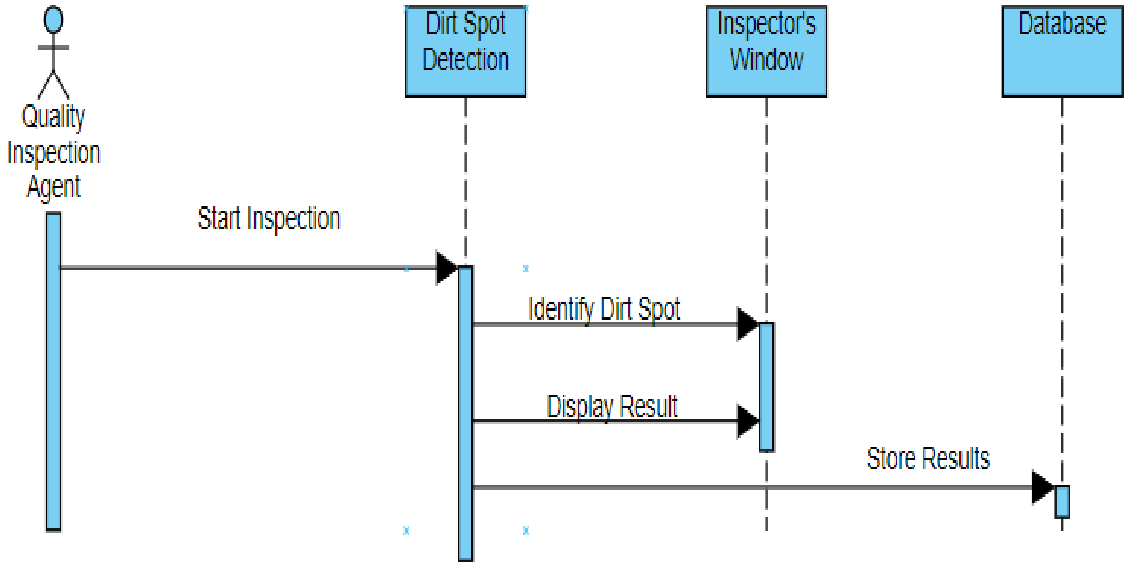
Quality inspection tests can be performed by inspector from its interface.



Figure# 53: Sequence Start Inspection

3.3.6 Dirt Spot Detection

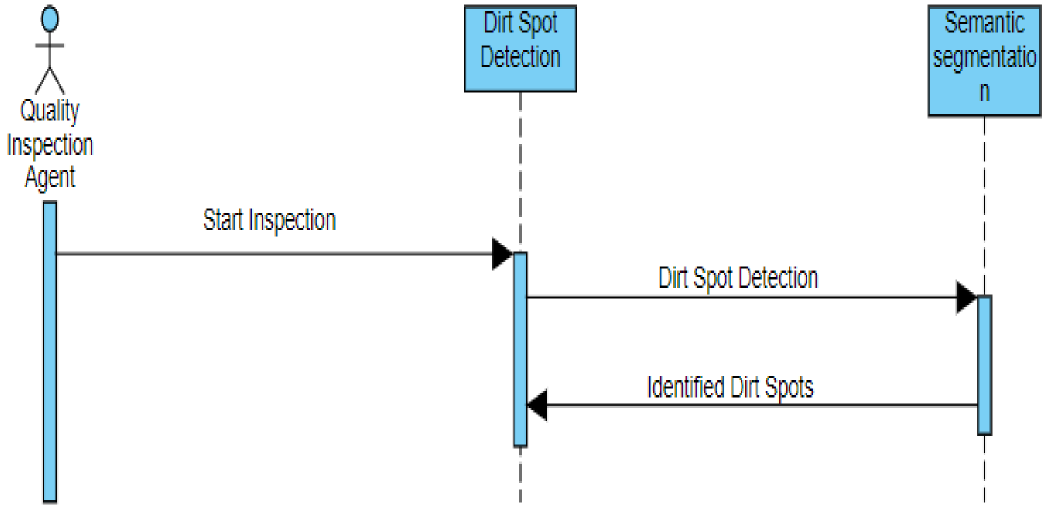
In inspection dirt spot test will be performed on packets. identified spots will be displayed on inspector screen and result will be send to database for later use.



Figure# 54: Sequence Dirt Spot detection

3.3.7 Semantic Segmentation

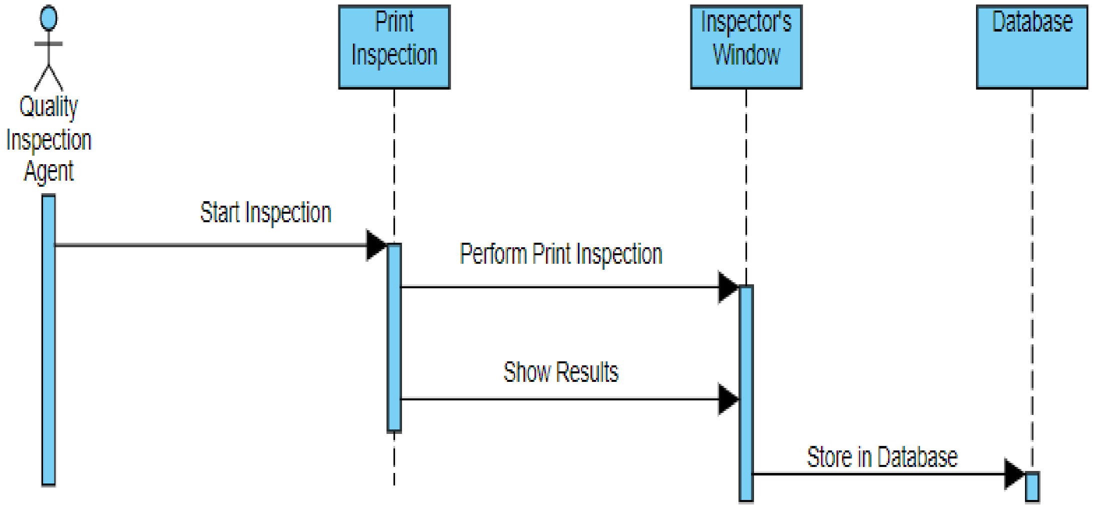
Semantic segmentation will be performed on packets.



Figure# 55: Sequence Semantic Segmentation

3.3.8 Print Template Analysis

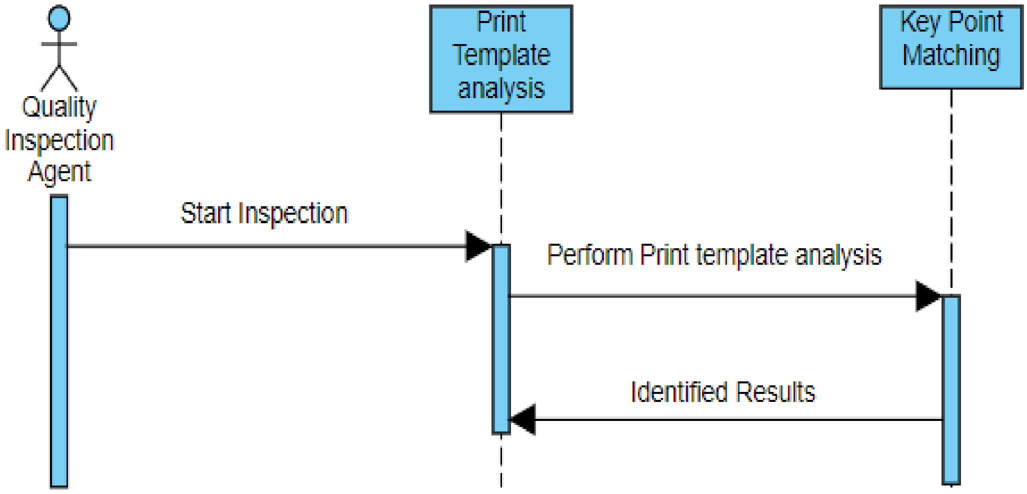
This test will be performed by in inspector to check the quality of printing of colours and result will be sent to database.



Figur#e 56: Sequence Print inspection

3.3.9 Key Point Matching

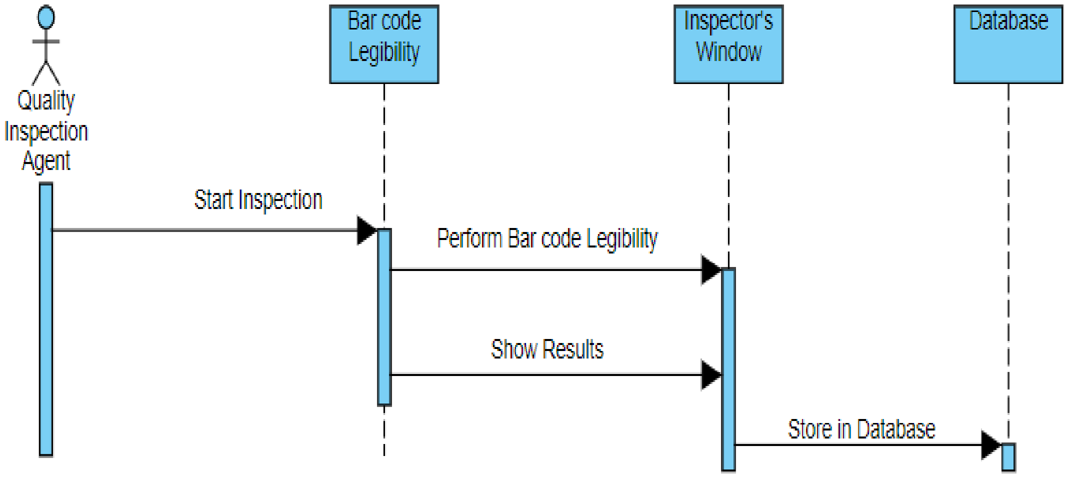
Key point matching will be performed on packets to ensure the packaging quality.



Figure# 57: Sequence Key Point Matching

3.3.10 Code Legibility

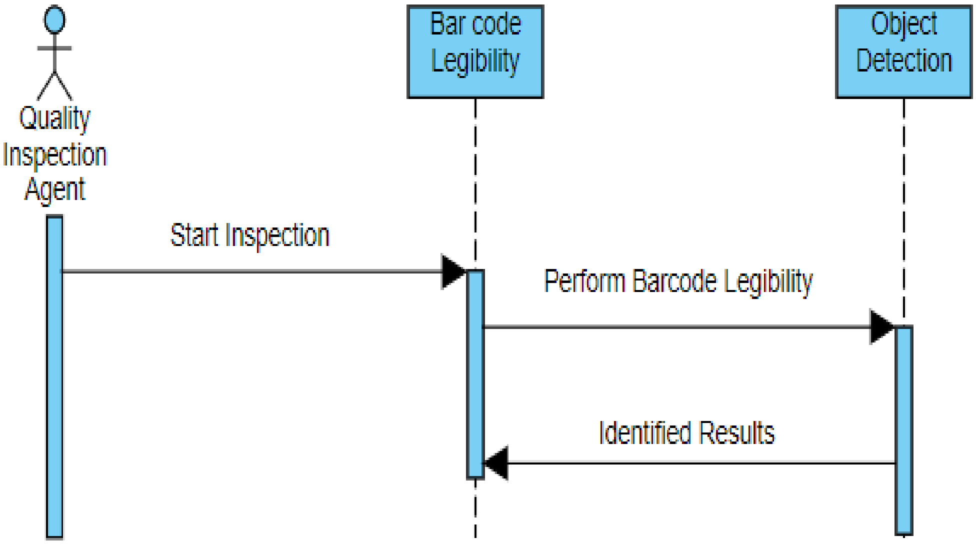
Bar code legibility test will be performed by the inspector to check that packet barcode is readable or not. If there is error, then t will be sent to database.



Figure# 58: Sequence Barcode Legibility

3.3.11 Object Detection

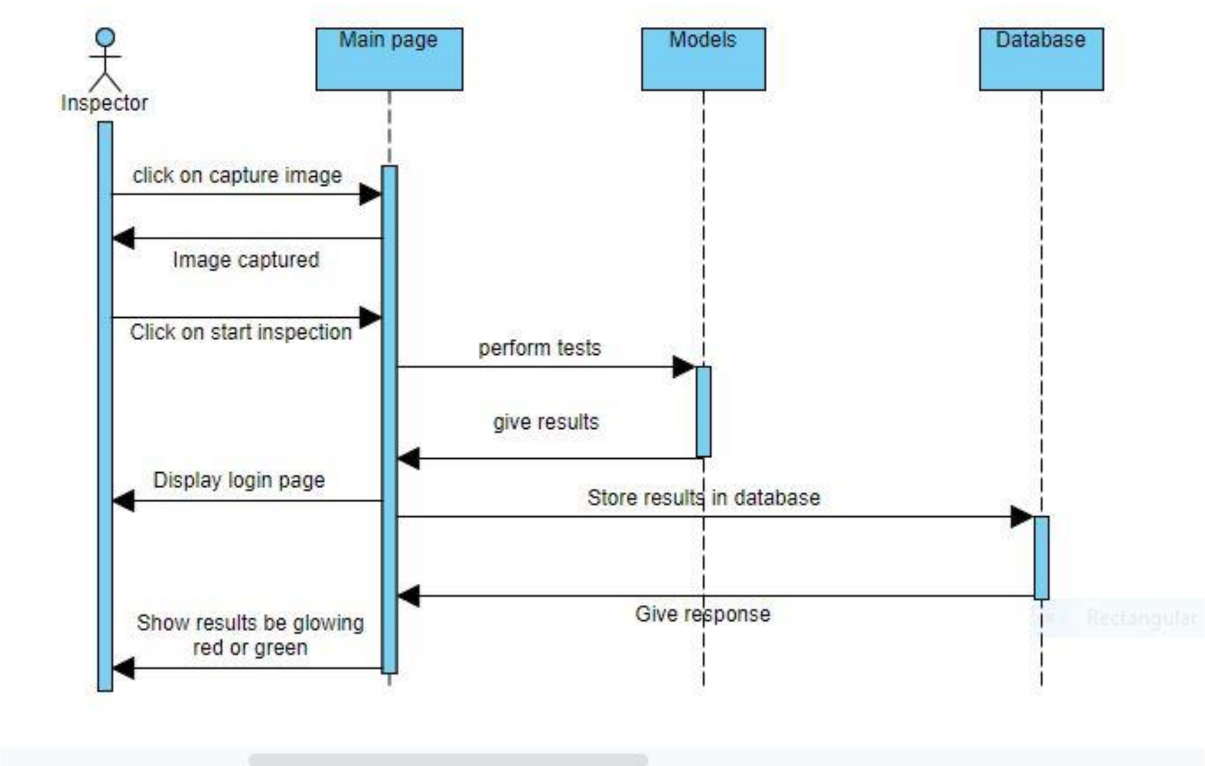
On starting project packets are identified



Figure# 59: Sequence Object Detection

3.3.12 Show Results

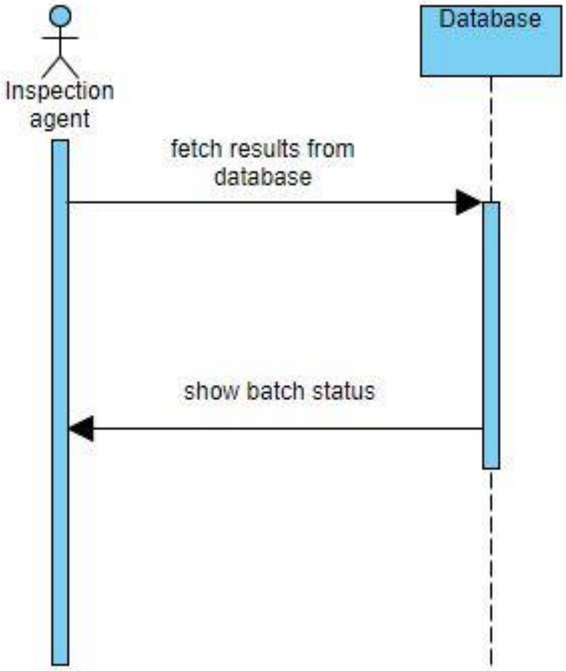
Results can be viewed by inspectors performed by them.



Figure# 60: Sequence Show results

3.3.13 Decision of Batch Status

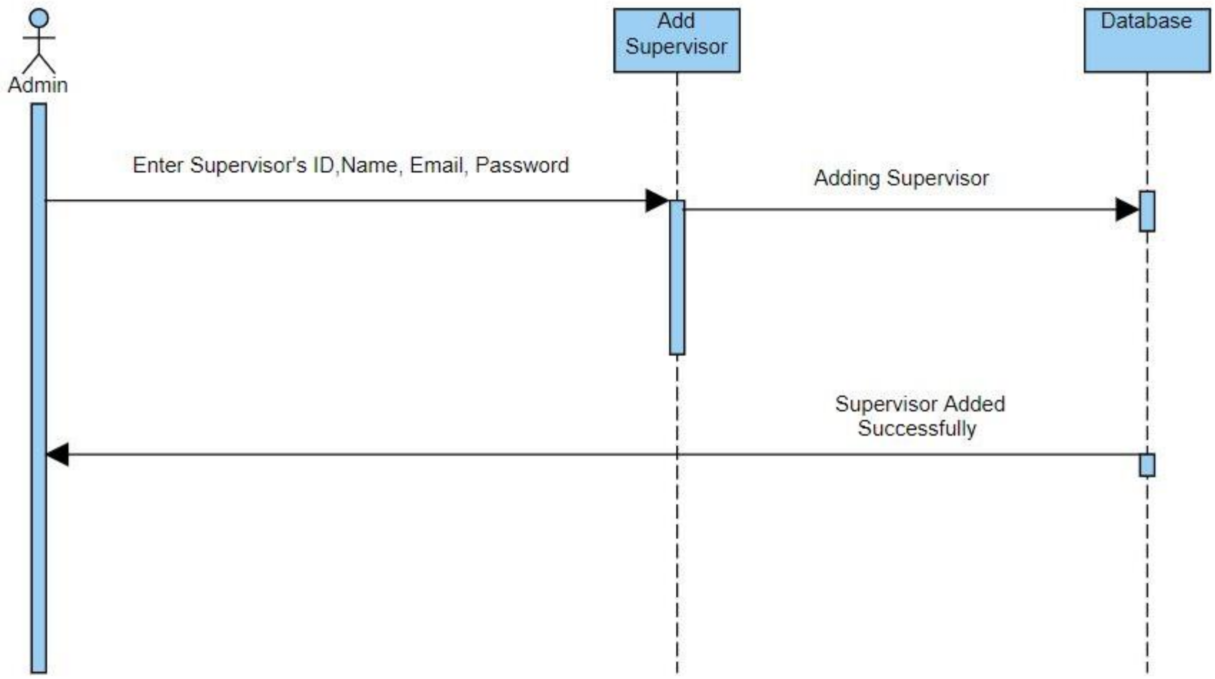
Overall batch status can be seen on screen



Figure# 61: Sequence Decision of batch status

3.3.14 Add Supervisor:

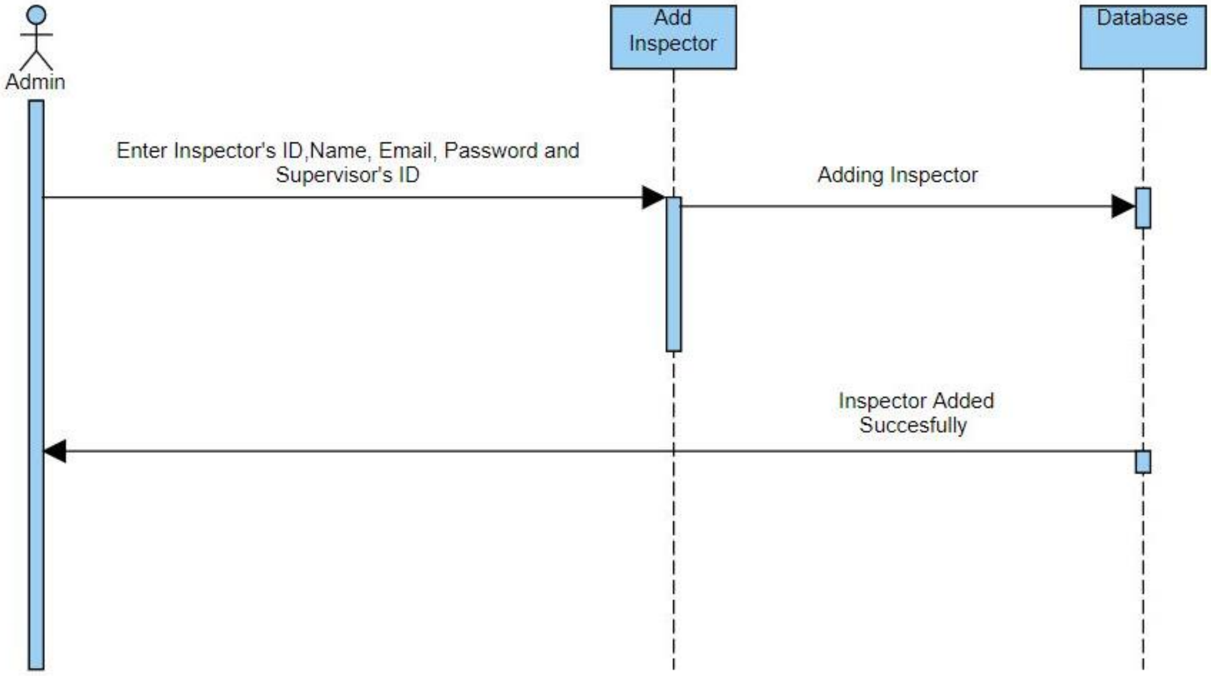
Admin can add supervisor in database so he can perform inspections.


Admin
Add Supervisor
Database
Enter Supervisor's ID,Name, Email, Password
Adding Supervisor
Supervisor Added Successfully

Figure# 62: Sequence Add supervisor

3.3.15 Add Inspector:

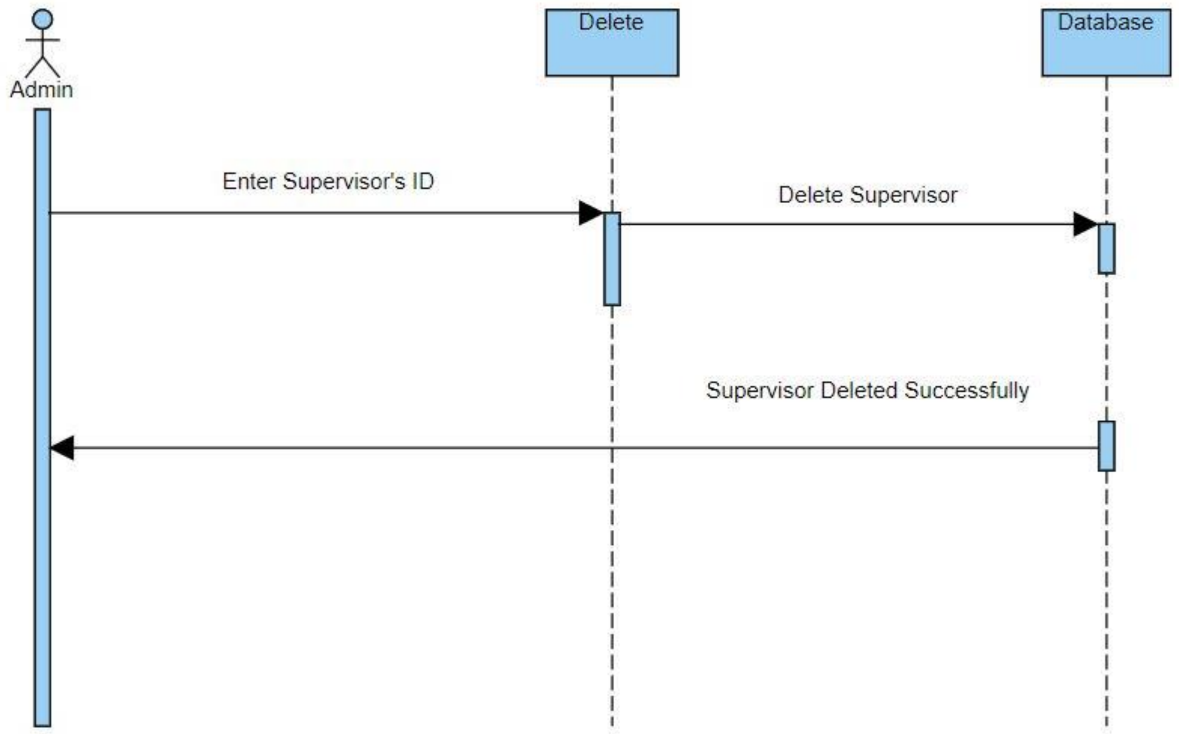
Admin can add inspectors in database by entering credentials


Admin
Add Inspector
Database
Enter Inspector's ID,Name, Email, Password and Supervisor's ID
Adding Inspector
Inspector Added Succesfully

Figure# 63: Sequence Add Inspector

3.3.16 Delete Supervisor:

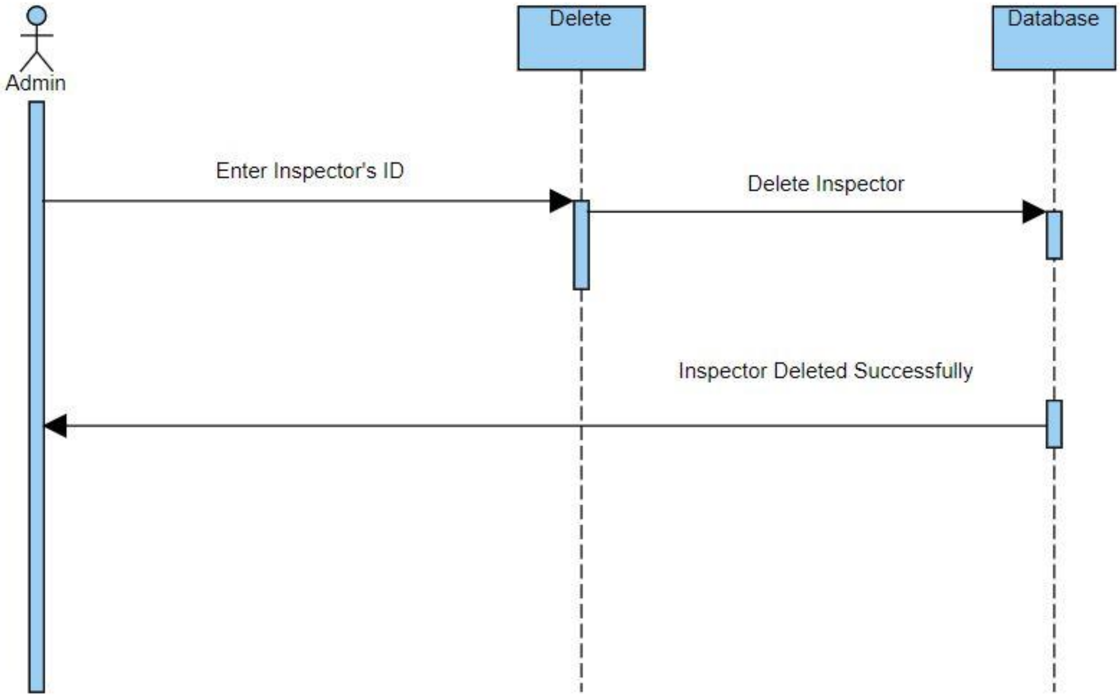
Admin can delete a supervisor by using his interface.



Figure# 64: Sequence Delete supervisor

3.3.17 Delete Inspector:

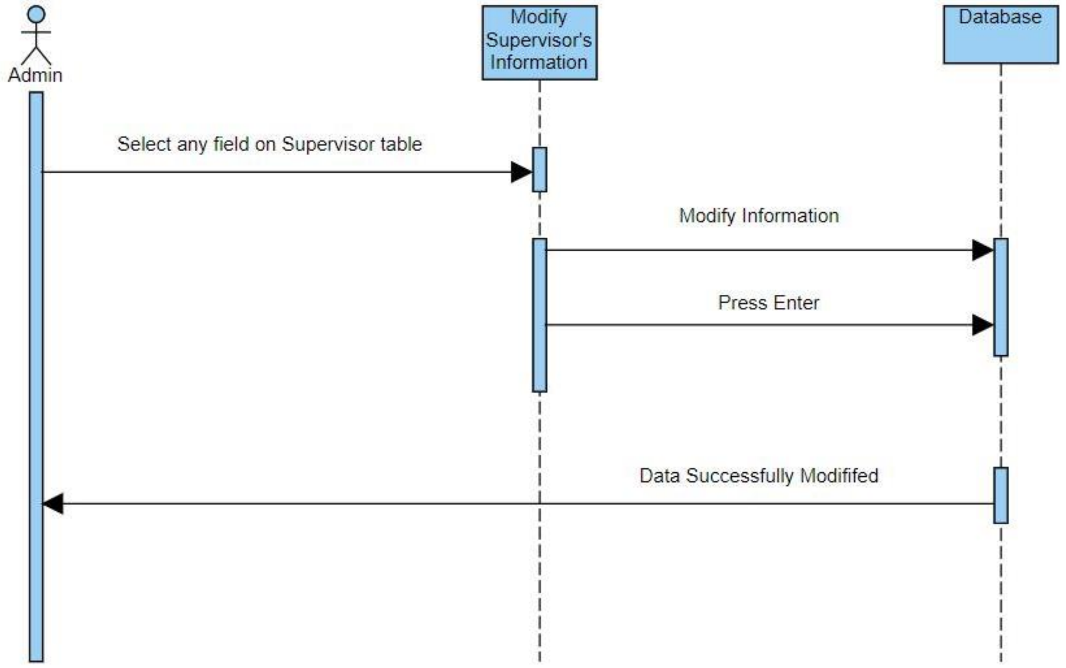
Admin can delete inspector from its interface.



Figure# 65: Sequence Delete Inspector

3.3.18 Modify Supervisor’s Information:

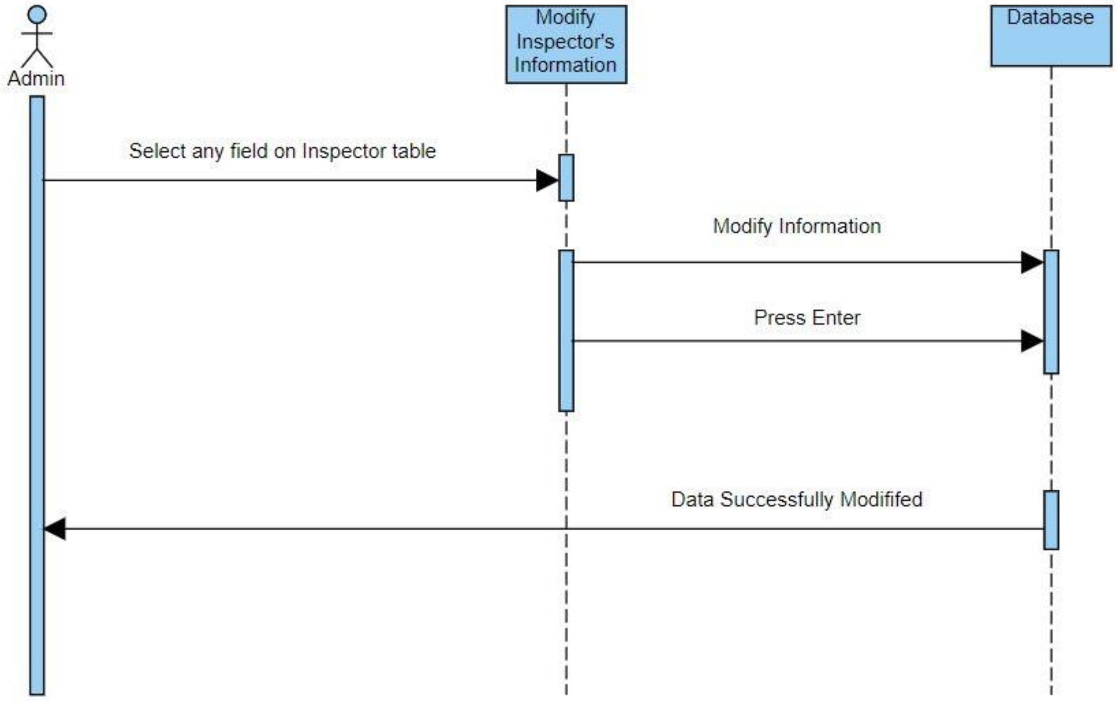
Admin can modify supervisor’s information from his interface.


Admin
Modify Supervisor's Information
Database
Select any field on Supervisor table
Modify Information
Press Enter
Data Successfully Modififed

Figure# 66: Sequence Modify Supervisor Information

3.3.19 Modify Inspector's Information:

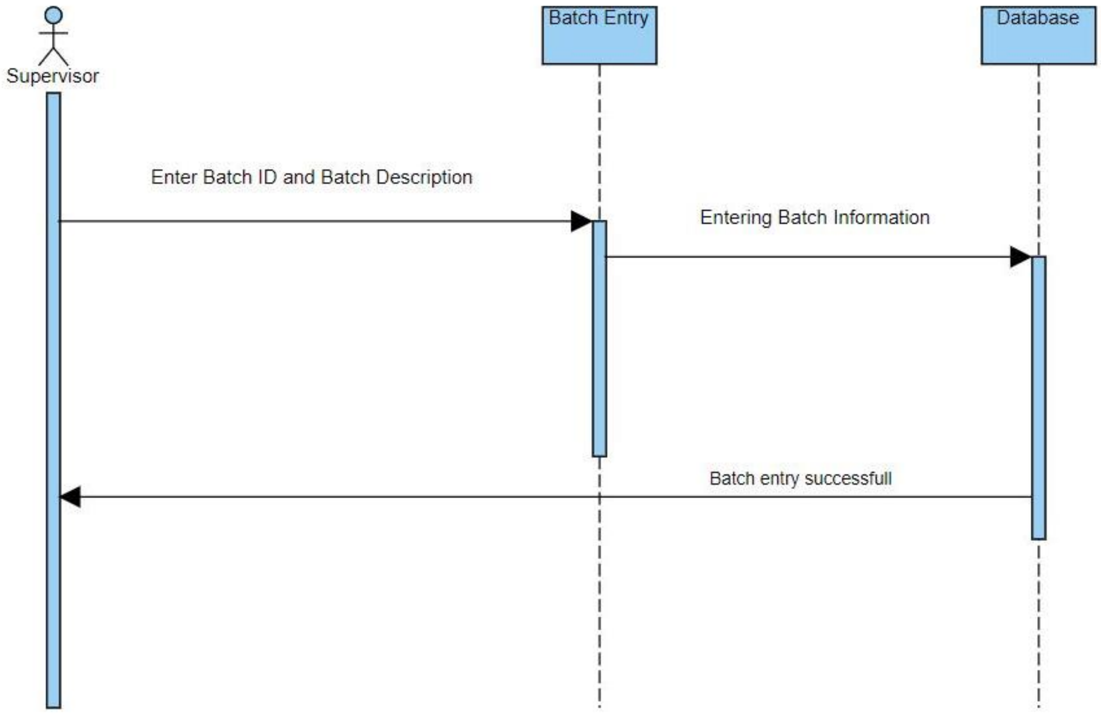
Admin can modify any inspector information from his interface.


Admin
Modify Inspector's Information
Database
Select any field on Inspector table
Modify Information
Press Enter
Data Successfully Modififed

Figure# 67: Sequence Modify Inspector's Information

3.3.20 Batch Entry:

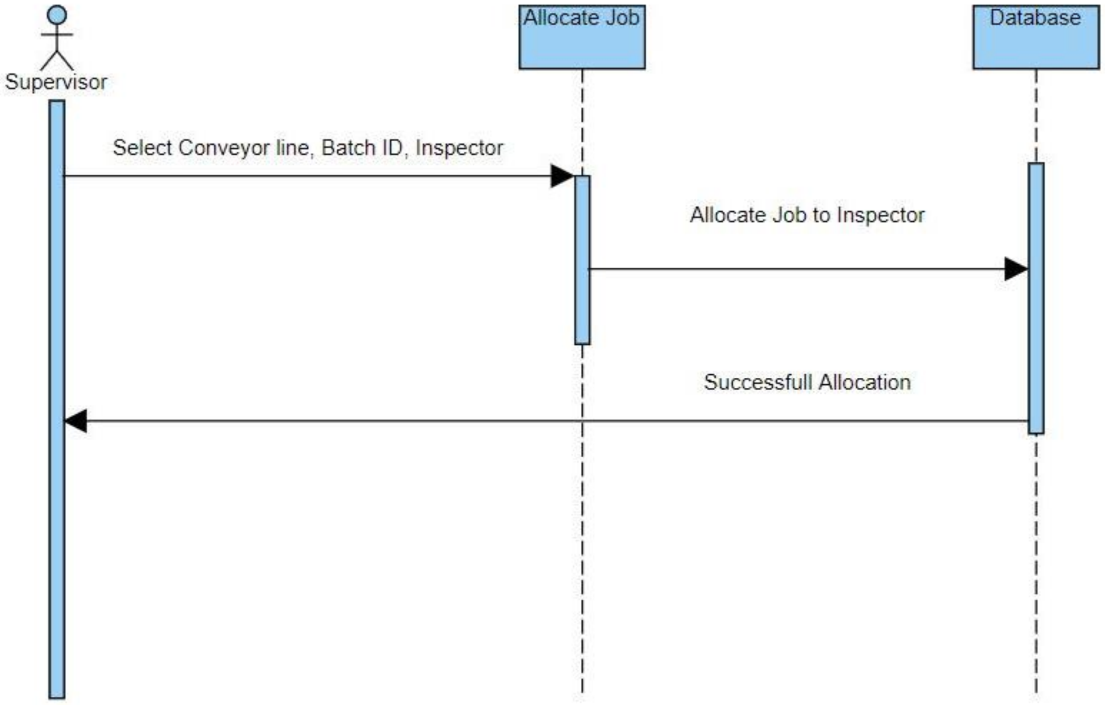
Supervisor can add a new batch from its interface.


Supervisor
Batch Entry
Database
Enter Batch ID and Batch Description
Entering Batch Information
Batch entry successfull

Figure# 68: Sequence Batch entry

3.3.21 Job Allocation:

Supervisor can allocate a job like specific packaging line inspection to a specific inspector.


Supervisor
Allocate Job
Database
Select Conveyor line, Batch ID, Inspector
Allocate Job to Inspector
Successfull Allocation

Figure# 69: Sequence Job Allocation

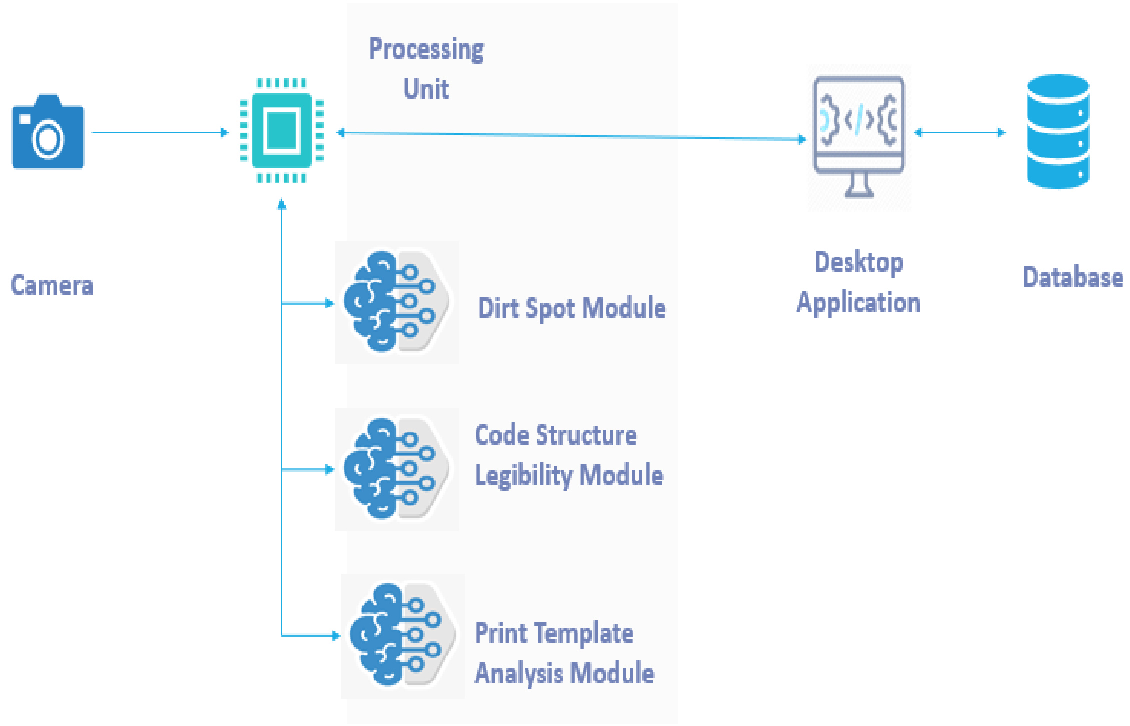
3.4 Software architecture

System Architecture is illustrated in the figure 2 given below. Our system shall have a camera to capture the video of the packaging material and a processing unit/computer or laptop to perform the quality inspection of the packaging material. The Inspector will place the product under the camera. The camera will take the pictures of the packaging material. Then the processing unit will perform various quality inspections on the packaging material. Three quality inspections will be performed on the captured video of the packaging material. At the first, the dirt spot inspection will be performed by the processing unit. In this inspection, the dirt spot on the packaging material will be identify using semantic segmentation technique. If there exist any dirt spot packaging material the package shall be rejected.

After dirt spot inspection, print inspection will be performed by the processing unit. In this inspection, the printing errors will be identify using key point matching. The inspection check, whether the front panel print is on the back panel or vice versa. The inspection extracts the key points of the captured video and match these key points from the sampled video. If the front panel print of the packaging material is on the back panel or vice versa then the product will be rejected.

At the end, the system shall perform code legibility inspection using object detection. In this inspection, the system shall identify the bar code on the packaging material. After identifying the bar code, the system inspects the legibility of the bar code and check whether the code is legible or not according to the defined policies. If the code is legible then the product will be accepted otherwise it will be rejected. The result of the quality inspection performed on the packaging material will be stored in the database and displayed on the HMI.

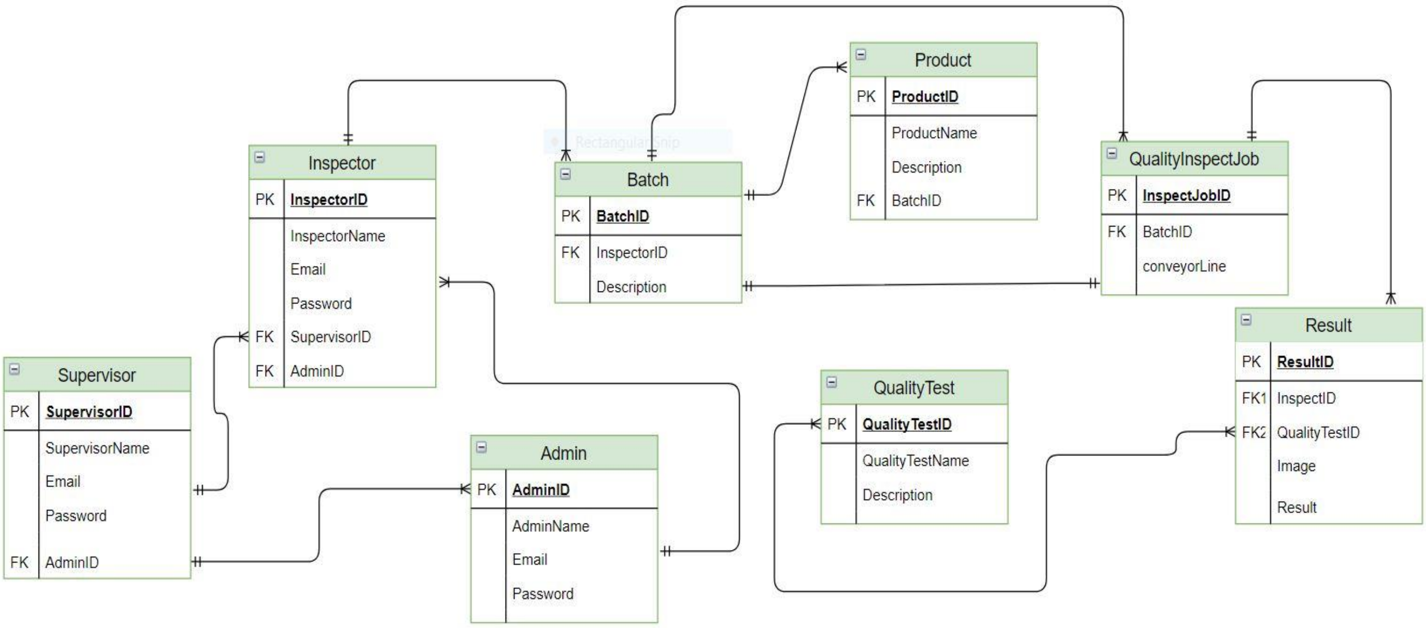
All the processing units will be connected to the server by the network. The results of all the quality inspection performed will be stored in the database. The manager can generate the results from the database and can view the amount of product that fail the inspection. The manager can also view the results in statistical form. The manager can find the reasons of failure and can rectify the issues. The database stores the results for three months.



Figure# 11: Software Architecture

3.5 Database Diagram

The database is normalized to 2NF level. We have tables of Inspector, Product, Batch, Supervisor, SupervisorID, AdminID, InspectorID, Admin, QualityTest, QualityInspectJob and Result



Figure# 12: Database Diagram

Chapter 4: Solution Proposed

For our system we had selected three industrial problems faced during quality inspection of the packaging material. We divided these problems in three different modules. These modules are:

1) Code Structure Legibility module
2) Dirt Spot Detection module
3) Print inspection module

4.1 Module of Code Structure Legibility

This module checks the legibility of barcode. If the barcode is legible then the product is accepted otherwise rejected. The product selected for the barcode module is “Ariel”. The inspection is performed on the packet of Ariel. The first step towards the development of the module was the identification of the problem, we have dealt this problem as Object Detection problem. For Object Detection YOLOv3 Darknet is used for the training. The first step towards the creation of object detection model was the collection of data set. The Industries never provide their data that has issues to anyone, as it is confidential and can cause problems to their image in the market.

Thus, we decided to create the dataset by our own. We took around 100 images of Ariel packet of rupees 20. Then with the help of Photoshop we added distortion and noise on the barcode section of the video. Once we had 100 different types of damaged video of barcode, we performed data augmentation on these 100 images. We used Keras library for the data augmentation and using 5 parameters of data augmentation that include:

- Width shift
- Height shift
- Zoom
- Rotate
- Brightness

Using these parameters, we generated 1000 different images from 100 images, once our data set was complete it was time for YOLOv3 model. We divided our data set into 800 training images and 200 test images. After that, we performed video annotation on 800 test images, after annotation one .txt file and .jpg file was generated. We did our model training on Google Collaboratory. Once our YOLOv3 model was successfully trained and we had achieved satisfactory results we used OpenCV for the detections of accepted and rejected barcodes.

4.2 Module of Dirt Spot Detection Module

For the detection of dirt spot, we selected Ariel 20-rupee pack. The video was captured for the data set collection. This problem is being solved as semantic segmentation. We applied different types of dirt spots on these images having different shapes, and sizes. Once dataset was created model was created on Google Collaboratory. Once we received satisfactory results, we saved the model in our device and integrated it with our interface.

4.3 Module of Print Template Analysis

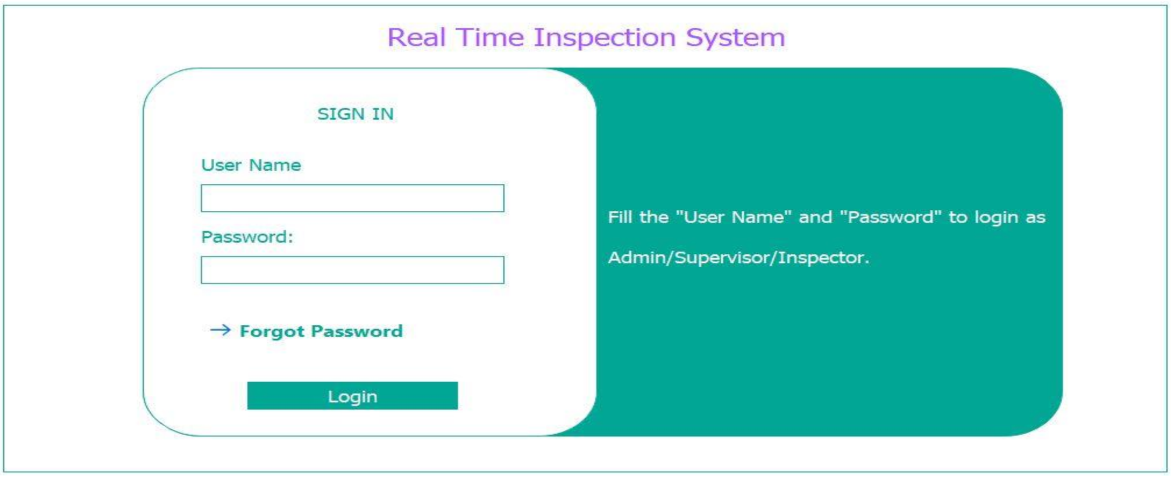
For the detection of dirt spot, we selected Packet from Lays. This problem is being solved as template matching. The packet was disoriented manually, and the video were captured for the data set collection. Once dataset was created model was created on Google Collaboratory. Once the model was created, we saved the model in our device and integrated it with our interface.

4.4 Our System Interfaces

Desktop App Interfaces:

We have developed our application in PYQT5, the interfaces are as under:

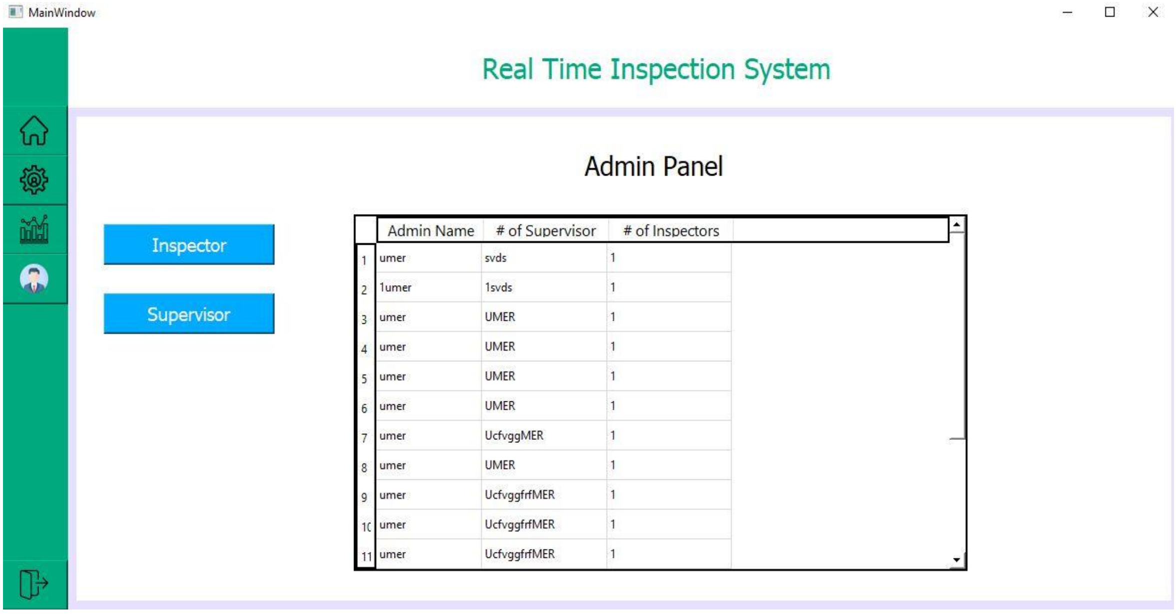
This is the SignIN Screen user have to give his credentials here



Fill the "User Name" and "Password" to login as Admin/Supervisor/Inspector.

Figure# 13: Sign in Interface

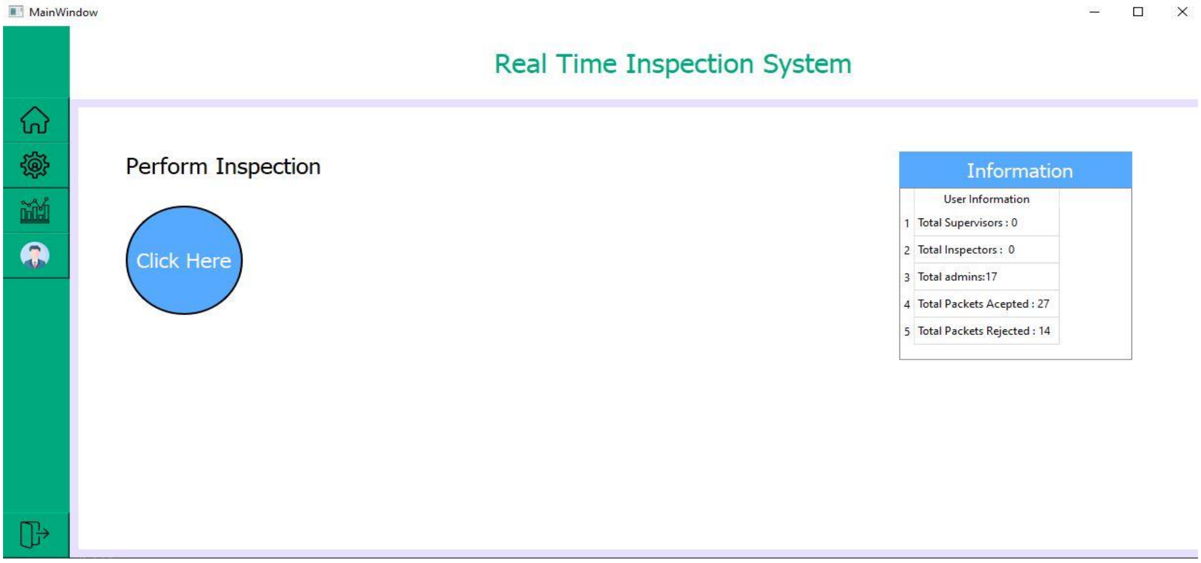
This is the Admin Panel which shows the users and their data.



	Admin Name	# of Supervisor	# of Inspectors
1	umer	svds	1
2	1umer	1svds	1
3	umer	UMER	1
4	umer	UMER	1
5	umer	UMER	1
6	umer	UMER	1
7	umer	UcfvggMER	1
8	umer	UMER	1
9	umer	UcfvggfrfMER	1
10	umer	UcfvggfrfMER	1
11	umer	UcfvggfrfMER	1

Figure# 14: Admin Panel Interface

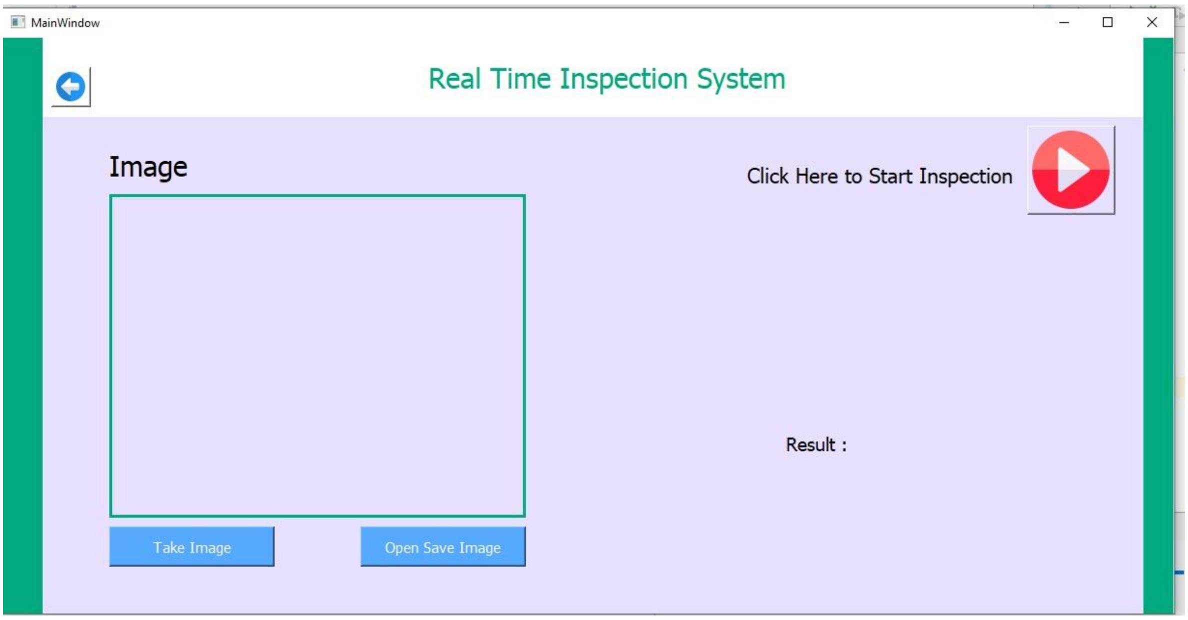
This is the Home interface from where we can see the inspection stats and can trigger the inspection.



	User Information
1	Total Supervisors : 0
2	Total Inspectors : 0
3	Total admins:17
4	Total Packets Acepted : 27
5	Total Packets Rejected : 14

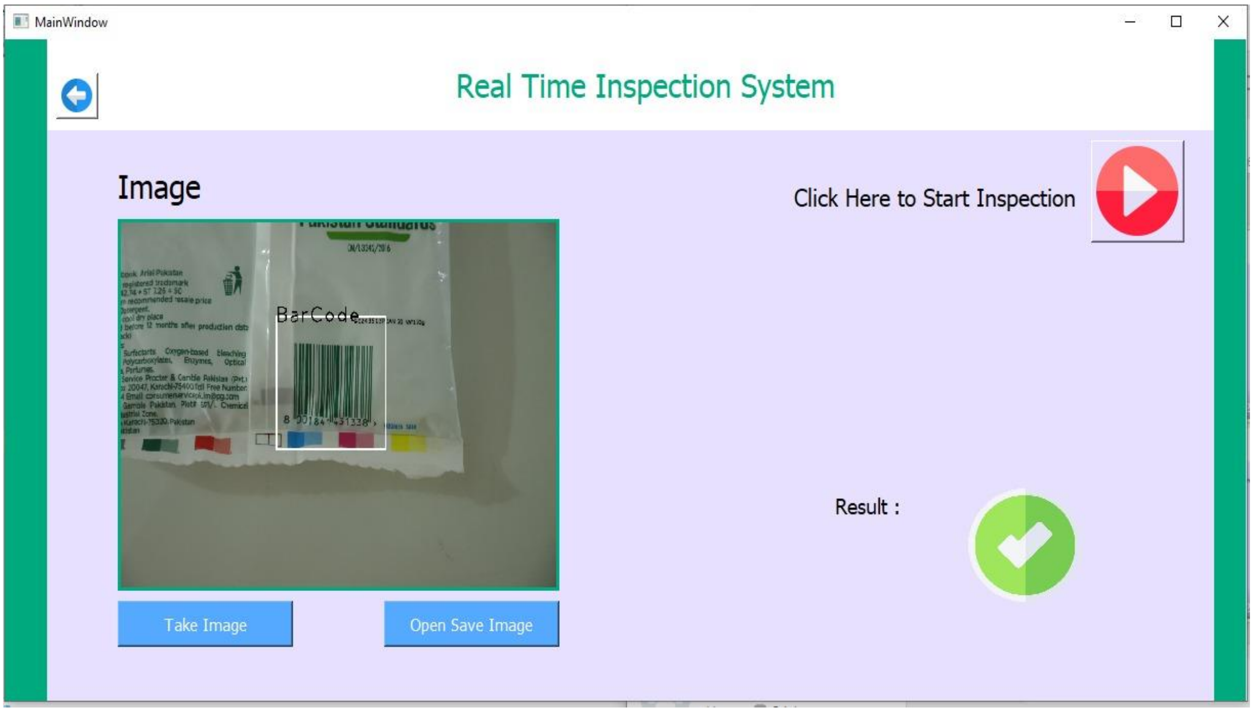
Figure# 75: Home Interface

This is the inspector main page from where inspector will trigger the inspection.



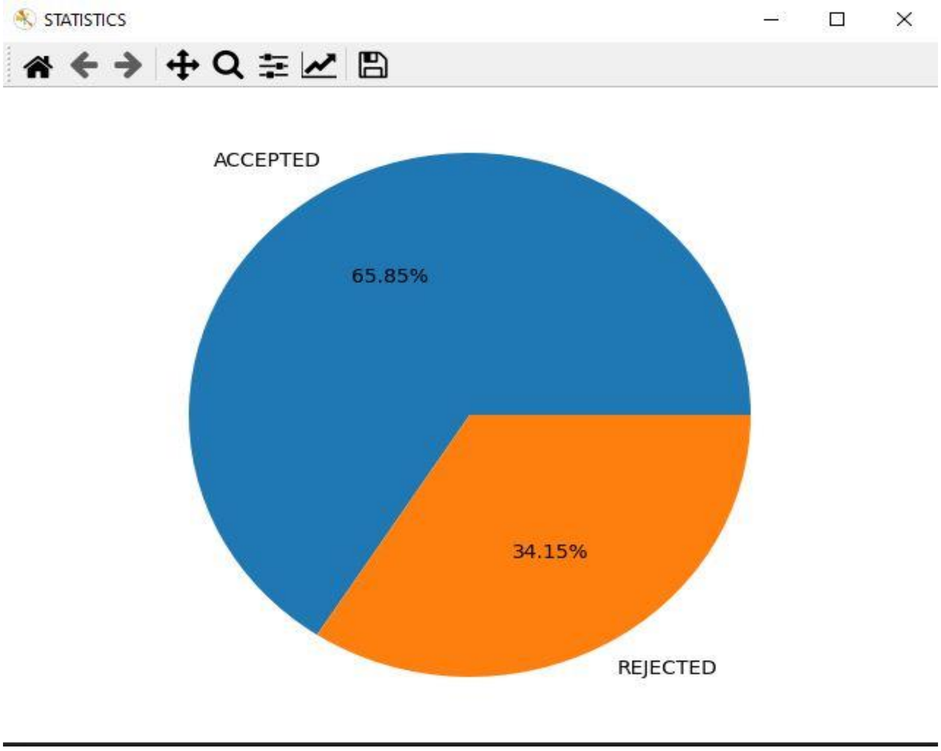
Figure# 76: Inspector Main Page Interface

This is the result screen where the passed and failed status will be shown.



Figure# 77: Result Interface

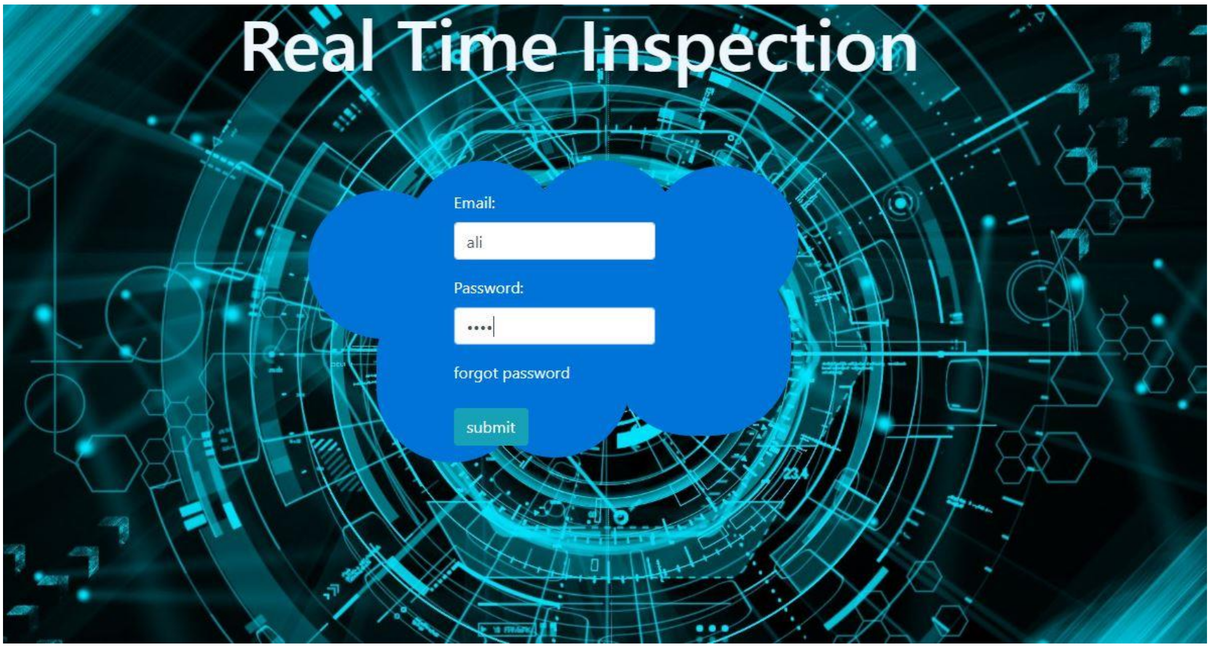
This is the statistics interface where we can see the passed and failed products stats.



Figure# 78: Stats Interface

Web App Interfaces:

This is our web app login screen.



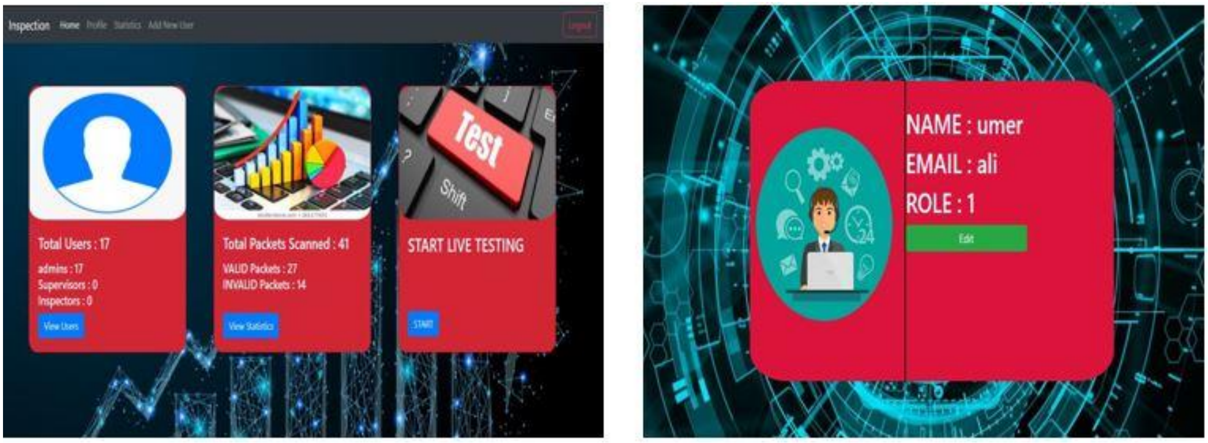
Real Time Inspection

Email: ali

Password: ••••

Figure# 79: Web Log in Interface

This is our web user profile and Home Interface.



Total Users : 17
admins : 17
Supervisors : 0
Inspectors : 0

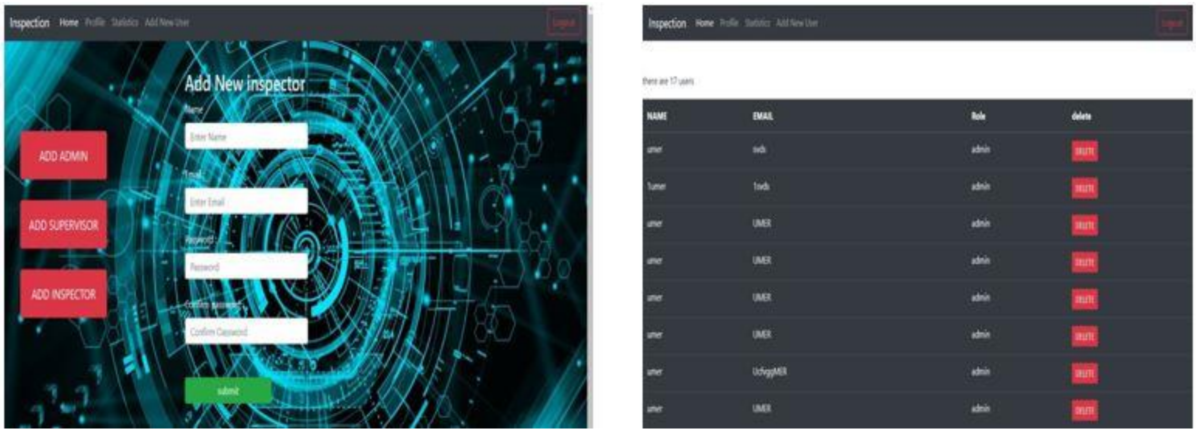
Total Packets Scanned : 41
VALID Packets : 27
INVALID Packets : 14

START LIVE TESTING

NAME : umer
EMAIL : ali
ROLE : 1

Figure# 80: Web Profile and Home Interface

This is our Web app Admin panel and data showing interface.

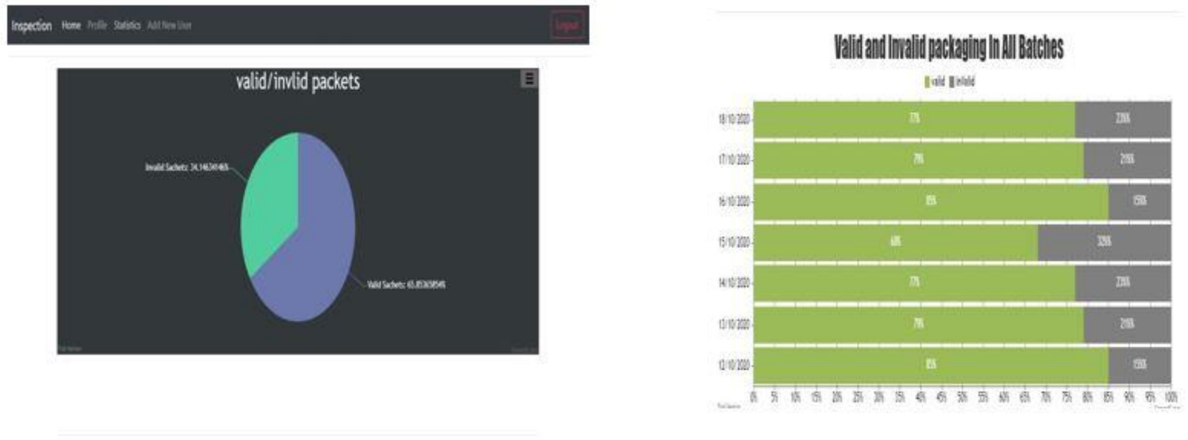


there are 17 users

NAME	EMAIL	Role	delete
umer	svds	admin	DELETE
1umer	1svds	admin	DELETE
umer	UMER	admin	DELETE
umer	UMER	admin	DELETE
umer	UMER	admin	DELETE
umer	UMER	admin	DELETE
umer	UcfvggMER	admin	DELETE
umer	UMER	admin	DELETE

Figure# 81: Adding Users and data showing Interface

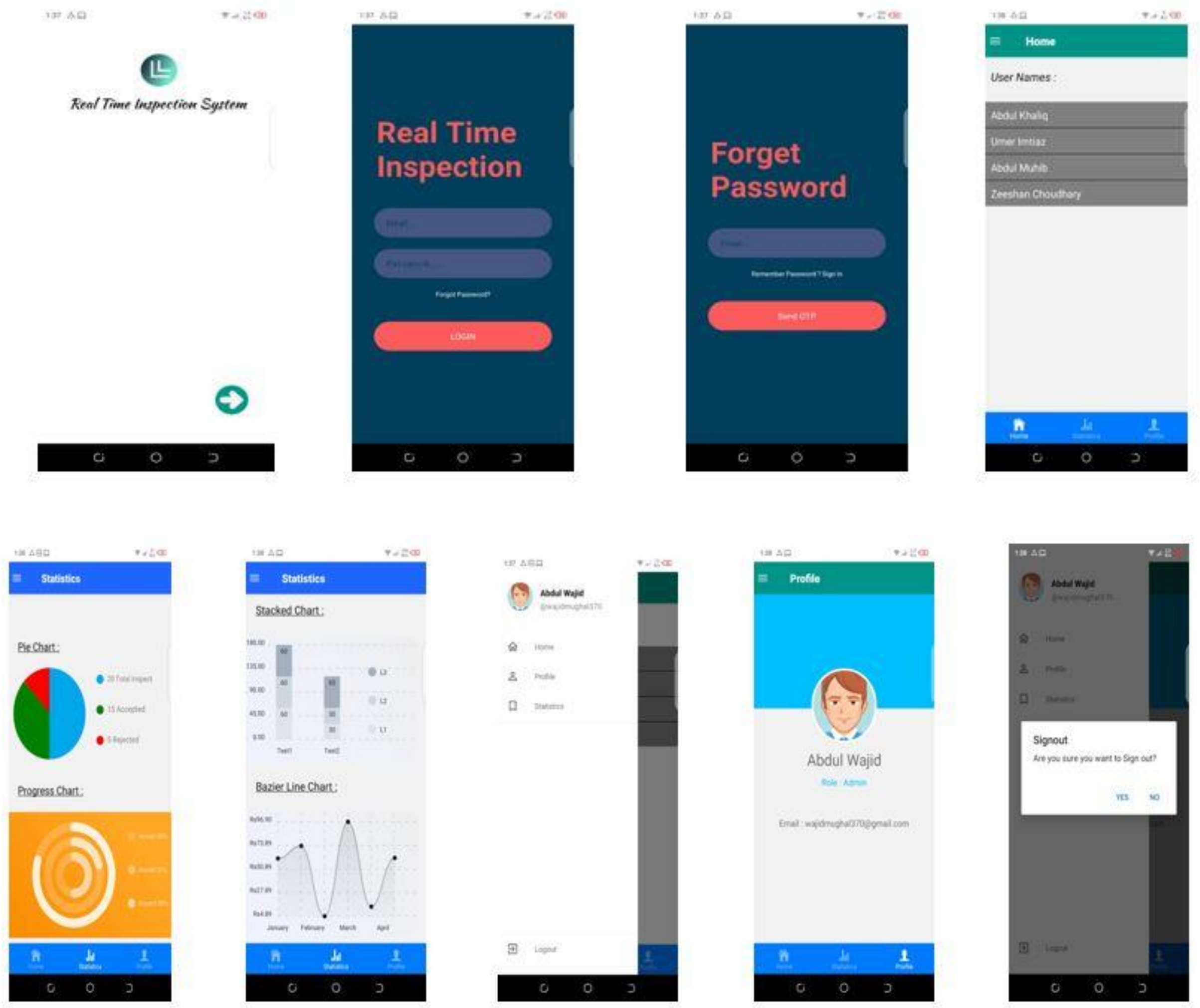
This is our web app stats interfaces which are showing the stats of inspected packages.



Figure# 82: Web Stats Interface

Mobile App Interfaces:

This is our mobile app login, forget password, user profile, stats, home and sign-out interfaces.



Figure# 83: Mobile App Interfaces

Chapter 5: Testing of System

5.1 Driven Test Cases

5.1.1 Login Test Case

No: 1	**Name:** Login
Description of System: Quality Inspection System	**Subsystem: None**

Design of Abdul Wajid	**Date of Design:** 9 November 2020
Done Execution: Umer Imtiaz	**Date of Execution:** 10 November 2020

Little Description: The test case is designed to verify if the system allows the user to login into the system.

Preconditions:

The credentials of Supervisor, Admin, and Inspector are stored in the database.

Step#	**Action performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The user enters correct username and password.	The user should be directed to their respective interface.	**passed**	
2	The user entered either wrong username or password.	The system gives error message.	**passed**	

Postconditions:

The user is redirected to their respective interface

Table# 24: Test case for Login

5.1.2 Logout Test Case

No: 2	**Name:** Logout
Description of System: Quality Inspection System	**Subsystem: None**
Design of Abdul Wajid	**Date of Design:** 9 November 2020
Done Execution: Umer Imtiaz	**Date of Execution:** 10 November 2020

Little Description: The test case is designed to verify if the system allows the user to logout from the system.

Preconditions:

The user is logged in into the system.

Step#	Action performed	System Response (Expected)	Passed/Failed	comment
1	The user clicks on logout button.	The user should be logged out of the system and should be directed on the login page.	**passed**	

Postconditions:

The user is redirected on the login page.

Table# 25: Test case for Logout

5.1.3 Password Changes Test Case

No: 3 **Name:** Password Change

Description of System: Quality Inspection System **Subsystem: None**

Design of Abdul Wajid **Date of Design:** 9 November 2020

Done Execution: Umer Imtiaz **Date of Execution:** 10 November 2020

Little Description: The test case is designed to verify if the system allows the user to change the existing password.

Preconditions:

The user is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The user clicks on Change Password button.	Form of password change appears.	**passed**	
2	The user enters the existing password.		**passed**	
3	The user enters new password and confirms the password.		**passed**	
4	The user clicks on Confirm change.	The system checks the existing password from database, if correct then change password.	**passed**	

Postconditions: Password is changed.

Table# 26: Test case for Change Password

5.1.4 Capturing Video Test Case

No: 4	**Name:** Capture Video
Description of System: Quality Inspection System	**Subsystem: None**
Design of Abdul Muhib	**Date of Design:** 9 November 2020
Done Execution: Umer Imtiaz	**Date of Execution:** 10 November 2020
Little Description: The test case is designed to verify if the system will capture the Video of the packaging material.	

Preconditions:

The packaging material is placed Infront of the camera.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The inspector clicks on button “Capture Video”.	The video should be captured.	**passed**	

Postconditions:

The video is captured for quality inspection.

Table# 27: Test case for Capture Video

5.1.5 Dirt spot detection Test Case

No: 5	**Name:** Dirt Spot Detection
Description of System: Quality Inspection System	**Subsystem: None**
Design of Abdul Muhib	**Date of Design:** 10 November 2020
Done Execution: Abdul Wajid	**Date of Execution:** 11 November 2020

Little Description: The test case is designed to verify if the system performs dirt spot inspection on the packaging material.

Preconditions:

The Video is captured.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**

1	The inspector has pressed button “Start Inspection”	The dirt spot result is displayed on Inspector’s interface. Shows green signal for accepted product and red signal for rejected product.	**passed**	

Postconditions:

Next inspection is performed.

Table# 28: Test case for dirt spot detection

5.1.6 Code Legibility Test Case

No: 6	**Name:** Code Legibility
Description of System: Quality Inspection System	**Subsystem: None**
Design of Abdul Muhib	**Date of Design:** 10 November 2020
Executed By: Abdul Wajid	**Date of Execution:** 11 November 2020
Little Description: The test case is designed to verify if the system performs code legibility inspection on the packaging material.	

Preconditions:

The video is captured.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The inspector has pressed button "Start Inspection"	The code legibility result is displayed on Inspector's interface. Shows green signal for accepted product and red signal for rejected product.	**passed**	

Postconditions:

Next inspection is performed.

Table# 29: Test case for Code Legibility

5.1.7 Adding Supervisor Test Case

No: 7	**Name:** Add Supervisor
Description of System: Quality Inspection System	**Subsystem: None**
Design of Umer Imtiaz	**Date of Design:** 10 November 2020
Done Execution: Abdul Wajid	**Date of Execution:** 11 November 2020

Little Description: The test case is designed to verify if the system allows the admin to add supervisor in the database.

Preconditions:

The admin is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The admin clicks on Supervisor button	Existing supervisor's list is displayed.	**passed**	
2	The admin clicks on Add new.	Form for adding new Supervisor appears	**passed**	
3	The admin enters SupervisorID, supervisor name, password, and email.		**passed**	
4	The admin press Add button	The supervisor record is entered into the database	**passed**	

Postconditions:

The entered supervisor is displayed on the table and the admin can view the details.

Table# 30: Test case for Adding Supervisor

5.1.8 Adding Inspector Test Case

No: 8	**Name:** Add Inspector
Description of System: Quality Inspection System	**Subsystem: None**
Design of Umer Imtiaz	**Date of Design:** 10 November 2020
Done Execution: Abdul Wajid	**Date of Execution:** 11 November 2020

Little Description: The test case is designed to verify if the system allows the admin to add inspectors in the database.

Preconditions:

The admin is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The admin clicks on Inspector button	Existing inspector's list is displayed.	**passed**	
2	The admin clicks on Add new.	Form for adding new Inspector appears	**passed**	
3	The admin enters Inspector ID, Inspector name, password, and email.		**passed**	
4	The admin press Add button	The inspector record is entered into the database	**passed**	

Postconditions:

The entered inspector is displayed on the table and the admin can view the details.

Table# 31: Test case for Adding Inspector

5.1.9 Deleting Supervisor Test Case

No: 9 **Name:** Delete Supervisor

Description of System: Quality Inspection System **Subsystem: None**

Design of Umer Imtiaz **Date of Design:** 13 November 2020

Done Execution: Abdul Muhib **Date of Execution:** 15 November 2020

Little Description: The test case is designed to verify if the system allows the admin to delete supervisor form database.

Preconditions:

The admin is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The admin clicks on Supervisor button	Existing supervisor’s list is displayed.	**passed**	
2	The admin clicks on Delete.	Form for deleting the existing Supervisor appears	**passed**	
3	The admin enters Supervisor ID.		**passed**	
4	The admin press Delete button	The supervisor record is deleted from the database	**passed**	

Postconditions:

The deleted supervisor is removed from the table.

Table# 32: Test case for Deleting Supervisor

5.1.10 Deleting Inspector Test Case

No: 10 **Name:** Delete Inspector

Description of System: Quality Inspection System **Subsystem: None**

Design of Umer Imtiaz **Date of Design:** 13 November 2020

Done Execution: Abdul Muhib **Date of Execution:** 15 November 2020

Little Description: The test case is designed to verify if the system allows the admin to delete inspector form database.

Preconditions:

The admin is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The admin clicks on Inspector button	Existing inspector's list is displayed.	**passed**	
2	The admin clicks on Delete.	Form for deleting the existing Inspector appears	**passed**	
3	The admin enters Inspector ID.		**passed**	
4	The admin press Delete button	The inspector record is deleted from the database	**passed**	

Postconditions: The deleted inspector is removed from the table.

Table# 32: Test case for Deleting Inspector

5.1.11 Modifying Supervisor Test Case

No: 11	**Name:** Modify Supervisor
Description of System: Quality Inspection System	**Subsystem: None**
Design of Umer Imtiaz	**Date of Design:** 13 November 2020
Done Execution: Abdul Muhib	**Date of Execution:** 15 November 2020
Little Description: The test case is designed to verify if the system allows the admin to modify any existing supervisor form database.	

Preconditions: The admin is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The admin clicks on Supervisor button	Existing supervisor's list is displayed.	**passed**	
2	The admin clicks on any record that he/she want to update and update	The record is updated in the database.	**passed**	

	that field in the table.			

Postconditions: The record of the supervisor is updated in the database and the table.

Table# 33: Test case for Modifying supervisor

5.1.12 Modifying Inspector Test Case

No: 12	**Name:** Modify Inspector
Description of System: Quality Inspection System	**Subsystem: None**
Design of Umer Imtiaz	**Date of Design:** 13 November 2020
Done Execution: Abdul Muhib	**Date of Execution:** 15 November 2020
Little Description: The test case is designed to verify if the system allows the admin to modify any existing inspector form database.	

Preconditions: The admin is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The admin clicks on Supervisor button	Existing inspector's list is displayed.	**passed**	

2	The admin clicks on any record that he/she want to update and update that field in the table.	The record is updated in the database.	**passed**	

Postconditions: The record of the inspector is updated in the database and the table.

Table# 34: Test case for Modifying Inspector

5.1.13 Batch Entry Test Case

No: 13	**Name:** Batch Entry
Description of System: Quality Inspection System	**Subsystem: None**
Design of Abdul Wajid	**Date of Design:** 14 November 2020
Done Execution: Abdul Muhib	**Date of Execution:** 16 November 2020
Little Description: The test case is designed to verify if the system allows the supervisor to enter batch into the database.	

Preconditions: The supervisor is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**
1	The supervisor clicks on Add Batch.	Form for entering batch appears.	**passed**	
2	The supervisor enters Batch ID		**passed**	

	and description of the batch.			
3	The supervisor clicks on Add button.	The batch is entered in the database.	**pass**	

Postconditions: The batch entered in database.

Table# 35: Test case for Batch Entry

5.1.14 Job allocation Test Case

No: 14	**Name:** Batch Allocation
Description of System: Quality Inspection System	**Subsystem: None**
Design of Abdul Wajid	**Date of Design:** 14 November 2020
Done Execution: Abdul Muhib	**Date of Execution:** 16 November 2020
Little Description: The test case is designed to verify if the system allows the supervisor to allocate batch to the inspector.	

Preconditions: The supervisor is logged in into the system.

Step#	**Action Performed**	**System Response** **(Expected)**	**Passed/Failed**	**comment**

1	The supervisor selects conveyor line, batch id and inspector		**passed**	
2	The supervisor enters clicks on Allocate button.	The job is allocated to inspector and saved in database.	**passed**	

Postconditions:

The job allocated to the inspector and batch allocated to the conveyor line.

Table# 32: Test case for Job Allocation

Chapter 5: Conclusion of the Report

6.1 Problems we faced and what we learned from them

Problems we faced and what we learned from them

- Distribution of group tasks were not properly defined and distributed to the group. The group members were later distributed and defined the tasks they were good at, which helped in increasing the task’s efficiency.
- There were many issues in report and HMI design, it was getting hard to identify and resolve the issues individually. So, we decided to identify the issues step-by-step together from start which helped us in identifying and resolving all the issues.
- Due to no proper backup and naming of documents, we lost many important documents due to one of the member’s window crash and improper naming. After these incidents we kept backup of each file with all members with proper naming.
- Creation of data set was very difficult, but we created the dataset on our own under the supervision of our supervisors.
- In Barcode Legibility module, YOLOv3 weights were used initially, but we did not receive satisfactory results, later we came to know that it was for large models with at least thousands of images. So, we worked on YOLOv3 tiny weights which is specifically designed for small models.

6.2 Project summary

We are offering the quality inspection of packaging material with the use of industrial 4.0. There will be three members of the system i.e. Admin, Supervisor, and Inspector, they will manage the reports, working of each module of the system. Three tests will be performed by the quality inspection agent, which includes Dirt spot identification, print inspection, code legibility on packaging material to check, show the batch status and improve the packaging material's quality.

6.3 Future work

In future one more module will be added in the system, which we are not adding in the system this time. The module will identify wrinkles on the top and bottom seal of the packaging material, and we shall improve the other three modules which are already in the system by increasing data set.

References

[1] Rossana, "tracesoftware," 12 Nov 2018. [Online]. Available: https://www.trace-software.com/blog/what-is-industry-4-0/.

[2] B. Marr, "Forbes," 2 sept 2018. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2018/09/02/what-is-industry-4-0-heres-a-super-easy-explanation-for-anyone/#4a8444959788.

[3] gempackindia. [Online]. Available: https://www.gempackindia.com/form-fill-seal-machines.html.

[4] A. Cholewa-Wójcik and A. Kawecka, "PACKAGING QUALITY ASSURANCE IN SUPPLY CHAIN," in *14th International Scientific Conference*, Osijek, Croati, 2014.

[5] S. Mamah, "A Method for Damage Detection in the Packaging Materials," March 2019.

[6] M. Bahaghighat, f. Abedini, A.-J. Molnar and M. , "Vision Inspection of Bottle Caps in Drink Factories Using Convolutional Neural Networks," in *Conference: IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP 2019)*, Romania, 2019.

[7] Smith-Spark, Katz and Wilcockson, "Optimal approaches to the quality control checking of product labels.," *International Journal of Industrial Ergonomics, 68,* pp. 118-124, 2018.

[8] Meliones, Baltas, S. P. Kammenos, Vardangalos and Kuleschow, “A distributed vision network for industrial packaging inspection,” in *In International Conference on High-Performance Computing and Networking,* 2006.

[9] Aguilar-Torres, Argüelles-Cruz, A. J, Yánez-Márquez and M. A., “A real time artificial vision implementation for quality inspection of industrial products,” in *In 2008 Electronics, Robotics and Automotive Mechanics Conference (CERMA'08)*, 2008.

[10] Godina, Radu, Matias and Joao, “Quality Control in the Context of Industry 4.0.,” *Industrial Engineering and Operations Management II,* pp. 177-187, 2019.

[11] “Cognex,” Cognex Solutions, [Online]. Available: https://www.cognex.com/industries/food-and-beverage.

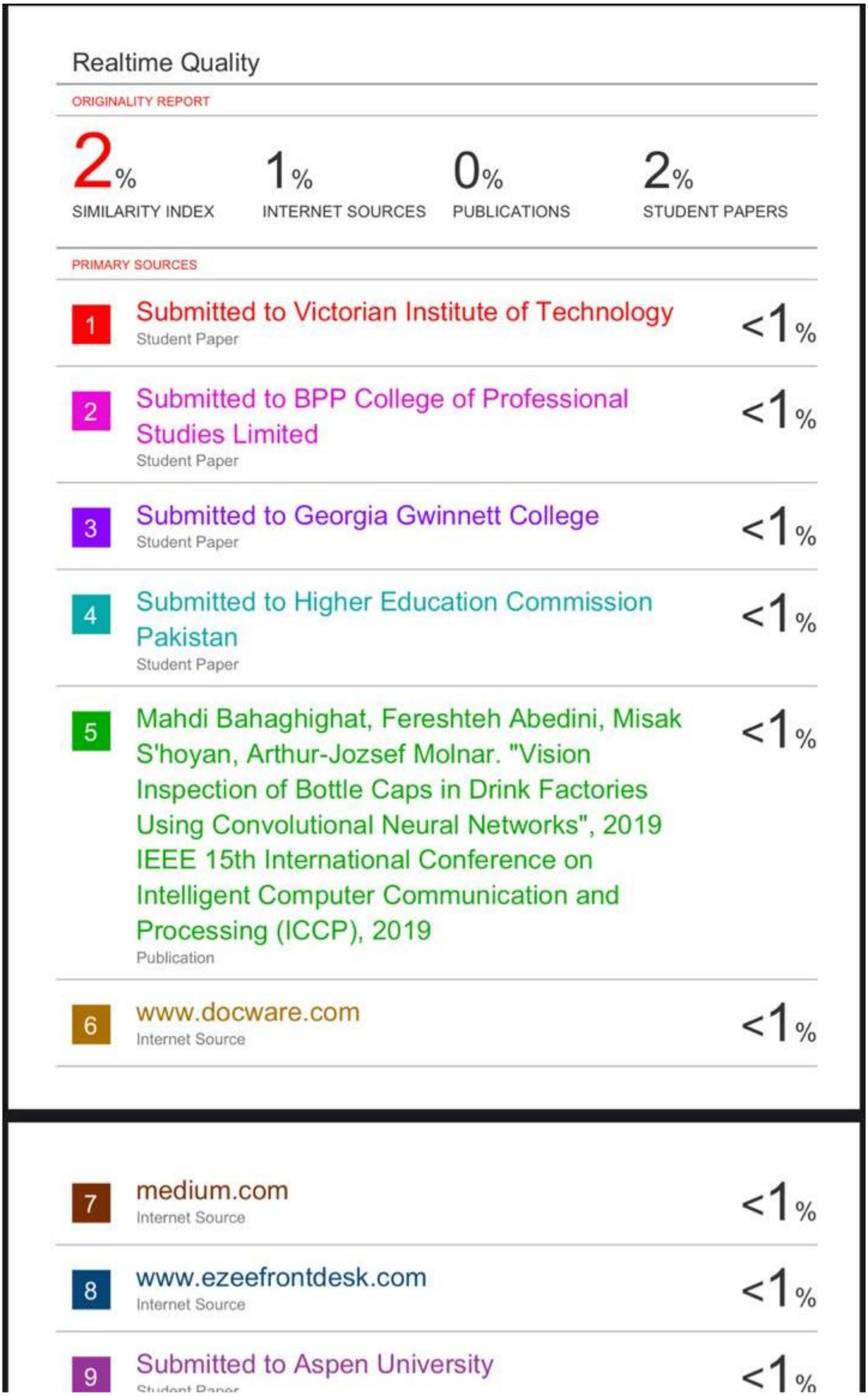
[12] “Futec,” [Online]. Available: https://www.futeceurope.com/printing-quality-control-and-process/.

[13] “EyeC "Your Vision for quality",” EyeC GmbH, [Online]. Available: https://www.eyec-inspection.com/.

[14] “IML packaging Inspector,” [Online]. Available: https://trivision.dk/iml-packaging-inspector/.

[15] “Bottle Vision System,” [Online]. Available: http://www.yuzenvision.com/products_list-6.html.

[16] S.-H. Huang and Ying-Cheng Pan, “Automated visual inspection in the semiconductor industry,” *Computers in Industry,* pp. 1-10, 2015.



Realtime Quality

ORIGINALITY REPORT

2%	1%	0%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Victorian Institute of Technology Student Paper	<1%
2	Submitted to BPP College of Professional Studies Limited Student Paper	<1%
3	Submitted to Georgia Gwinnett College Student Paper	<1%
4	Submitted to Higher Education Commission Pakistan Student Paper	<1%
5	Mahdi Bahaghighat, Fereshteh Abedini, Misak S'hoyan, Arthur-Jozsef Molnar. "Vision Inspection of Bottle Caps in Drink Factories Using Convolutional Neural Networks", 2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP), 2019 Publication	<1%
6	www.docware.com Internet Source	<1%
7	medium.com Internet Source	<1%
8	www.ezeefrontdesk.com Internet Source	<1%
9	Submitted to Aspen University Student Paper	<1%