AI-Powered Object Sorting Mechanical Arm with Easy Replaceability

By

Imon Mahbub
21221007
K. M. Fatin Mesbah
21221008
Ahmed Zoha Zim
21221016
Rayhan Kabir
21221043

A Final Year Design Project (FYDP) submitted to the BSRM School of Engineering in partial fulfillment of the requirements for the degree of
BSc in Electrical and Electronics Engineering

Academic Technical Committee (ATC) Panel Member:
ATC Panel 6

Dr. A.S Nazmul Huda (Chair)

Associate Professor, Department of EEE, BRAC University

Nahid Hossain Taz

Lecturer, Department of EEE, BRAC University

Sharif Mohammad Shams

Lecturer, Department of EEE, BRAC University

BSRM School of Engineering
BRAC University
September 2025

Declaration

It is hereby declared that

1. The Final Year Design Project (FYDP) submitted is my/our own original work while completing degree at Brac University.
2. The Final Year Design Project (FYDP) does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The Final Year Design Project (FYDP) does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. I/We have acknowledged all main sources of help.

Student's Full Name & Signature:

Imon Mahbub Student ID - 21221007	**K. M. Fatin Mesbah** Student ID - 21221008
Ahmed Zoha Zim Student ID - 21221016	**Rayhan Kabir** Student ID - 21221043

Approval

The Final Year Design Project (FYDP) titled "AI-Powered Object Sorting Mechanical Arm with Easy Replaceability," submitted by

1. Imon Mahbub (21221007)
2. K. M. Fatin Mesbah (21221008)
3. Ahmed Zoha Zim (21221016)
4. Rayhan Kabir (21221043)

Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of BSc in Electrical and Electronics Engineering on 11 September, 2025.

Examining Committee:

Academic Technical Committee (ATC):
(Chair)

____
Dr. A.S Nazmul Huda
Associate Professor, Department of EEE
BRAC University

Final Year Design Project Coordination Committee:
(Chair)

____
Dr. Mirza Rasheduzzaman
Associate Professor, Department of EEE
BRAC University

Department Chair:

____
Dr. Md. Mosaddequr Rahman
Professor & Chairperson, Department of EEE
BRAC University

Ethics Statement

All project materials were subjected to plagiarism checking using Turnitin to ensure academic integrity and originality. The similarity index of the submitted work was found to be **15%**, which is well below the acceptable limit of **35%**. This confirms that the project adheres to ethical standards and maintains authenticity in research and documentation.

Abstract/ Executive Summary

This project presents the design and implementation of an AI-powered object sorting mechanical arm that integrates decision-making capabilities of AI to enhance flexibility and efficiency in industrial automation. The proposed system eliminates the need for manual reprogramming by employing a large language model (LLM), Gemini 2.0, for object recognition and decision-making. Equipped with a 4-DOF robotic arm, high-resolution cameras, and a gripper mechanism, the system is capable of identifying, classifying, and placing objects with high accuracy. The approach demonstrates improvements in adaptability, reduced setup time, and sustainable applicability across various sectors, aligning with the needs of modern industry and society.

Keywords: Robotic arms, Object sorting, Industrial automation, Large language models, Sustainable robotics, Modular design.

Dedication

We would like to express our gratitude towards Nahid Hossain Taz and Sharif Mohammad Shams for their professional guidance and profound understanding, which was used in designing the AI-driven object sorting arm and improving its LLM-relied real-time recognition and control, helping throughout the whole project.

Table of Contents

Chapter 1: Introduction- [CO1, CO2, CO10]

1.1 Introduction

1.1.1 Problem Statement

Sorting medicines, fruits, and vegetables is one of the most critical steps in healthcare, pharmaceuticals, and food packaging. Sorting medicines or foods in the old-fashioned way brings a lot of problems to the table, like wrong packaging, etc, whether it is done manually or semi-automated.

Inefficiency and high labour costs pose a lot of issues for patients' lives if there is an issue with wrong medicine packaging. Nowadays, robotics arms that are currently in use totally depend on manual programming, and manual programming is needed whenever a new task is added, which takes a significant amount of time [1]. This slows down the process and is less flexible [2]. Our idea is to build a mechanical arm that is backed up by AI that can be used to recognize, classify, and sort objects like medicines, fruits, and vegetables all on its own. By integrating it with cameras, the system can accurately sort the products in a short period of time without the hassle of manual programming.

1.1.2 Background Study

Robotic arms have become an important part of industries like manufacturing, logistics, and healthcare, thanks to their precision, flexibility, and efficiency in time saving. But the problem with traditional robotic sorting systems is that they work on predefined instructions [3] i.e., manual programming is required, which can take a lot of time. That means whenever a new task is added, the arm has to be reprogrammed from scratch, and the result is lower efficiency and poor scalability in fast-changing industrial settings. By Integrating AI systems, the programming process is shortened as the AI can call functions on its own. This allows the robotic arm to make real-time decisions and adapt to new tasks. Modern large language models, like Gemini 2.0 flash lite [11], add multimodal reasoning, spatial awareness, and action planning even without retraining the whole thing.

1.1.3 Literature Gap

Robotic sorting systems have been in use for decades, but their biggest weakness is maintenance. When a new task is added, it has to be reprogrammed. Most still rely on rigid programming, which takes a lot of time when in maintenance. Thus making the working process slow and manufacturing is shut down fully. The real issue is that conventional robotic arms can not easily learn or adjust to new categories or environments without a lot of manual programming. That slows things down when it is in maintenance and limits scalability. This project aims to close that gap by introducing a smarter [4], AI-driven system. The approach combines U-Net-based deep learning for image segmentation with LLM-powered reasoning through Gemini 2.0. Put simply, it brings together advanced vision and adaptive decision-making in one package. With this setup, robotic sorting can become more adaptable, more efficient, and better suited to modern industrial needs—no endless reprogramming required.

1.1.4 Relevance to Current and Future Industry

The proposed AI-powered robotic arm aligns with the growing demand for automation in industries such as pharmaceuticals, healthcare, logistics, and food packaging. In healthcare, it ensures accurate sorting of medicines, reducing risks of contamination and human error. In food packaging, it can identify and separate fruits and vegetables based on attributes like size, color, and quality, preventing waste and ensuring consumer safety.

Future industries are rapidly adopting Industry 4.0 technologies, where AI-integrated robotic systems provide scalability, flexibility, and compliance with global safety standards such as ISO 10218 [5], ISO/TS 15066 [6], and IEC 61508. This ensures not only efficiency and cost-effectiveness but also sustainability and worker safety in collaborative human-robot environments.

1.2 Objectives, Requirements, Specification, and Constant

1.2.1. Objectives

- Develop an intelligent sorting mechanism by integrating AI with a robotic arm.
- Achieve real-time object recognition and classification with accuracy above 95%.
- Ensure adaptability to varying environments without the need for manual reprogramming.
- Design a scalable and modular system suitable for healthcare, logistics, and food packaging industries.
- Comply with industrial safety and ethical standards while maintaining cost-effectiveness.

1.2.2 Functional and Nonfunctional Requirements

Functional Requirements:

- Achieve ≥ 95% validation accuracy for reliable identification.
- Smooth, predictable motion: use forward/inverse kinematics plus well-tuned PID controllers so the arm plans and executes trajectories safely and consistently.
- Sort and place items into designated bins with high precision and repeatability. Calibrate camera-to-robot transforms, use adaptive grasp planning, and close the loop with visual and force feedback so picks stay accurate even as parts or lighting change.
- Adapt to new object categories without swapping hardware: build a modular vision stack with transfer learning / few-shot updates, active learning for quick label collection, and semantic class management. Include confidence thresholds, human-in-the-loop validation, and rollback so models evolve safely, remain auditable, and keep downtime and costs low.

Non-Functional Requirements:

- Scalability – The system should be able to process bigger datasets or sort through larger piles of objects without slowing down or losing accuracy.
- Portability – Keep the design compact enough to fit inside a 1m × 1m workspace, making it suitable even for tight industrial floors.
- User-friendliness – Operators of any skill level should find it easy to use, as the proposed project works by putting prompts and the AI calls the required functions to complete the programming.

1.2.3 Specifications

Specifications for the Design Approach:

Subsystem	**Components**	**Specification**
Power System	Power Supply	**Voltage:** 12-24V **Current:** 10A
Array of Sensors	FSR Sensors	**Force Range:** 0.2 to 20 N. Designed for low- to medium-force applications. **Operating Voltage:** 2.5 V – 5 V DC, compatible with Arduino analog input. Operates through voltage divider configuration with pull-down resistor (typically 10 kΩ). **Response time:** < 10 ms; suitable for real-time pressure monitoring and closed-loop grip adjustment. **Operating Temperature:** 0°C to 60°C. **Interface:** Connected to Arduino UNO analog input pins (A0–A3) for real-time grip force measurement. The analog value is converted to force (in N) through a calibration curve derived experimentally.
Computer Vision	Web Cam (2 units)	**Resolution:** Native resolution sensor capable of 1920 x 1080 pixel static images and supports 1080p30 and 720p60 video weight
Motors	Nemma 17 Stepper	1.5A to 1.8A current per phase 1-4 volts 3 to 8 mH inductance per phase 44 N·cm (62oz·in, 4.5kg·cm) or more holding torque 1.8 or 0.9 degrees per step (200/400 steps/rev

		respectively). **Operating Temperature Range:** 0°C – 55°C (non-condensing environment). **Speed:** 0.19 s/60° at 4.8 V; 0.14 s/60° at 6.0 V — provides smooth yet responsive movement for real-time sorting operations.
	MG996R Servo	**Operating Voltage:** 4.8 V – 7.2 V DC; nominal operation at **6.0 V** for maximum torque and speed. **Operating Current:** Idle current: 10 mA – 20 mA; running current: 500 mA; stall current: up to 2.5 A at 6 V. **Torque Output:** 9.4 kg·cm (4.8 V); 11 kg·cm (6.0 V); equivalent to 0.92 N·m – 1.08 N·m torque capacity. Suitable for lifting loads in the 1–1.5 kg range with lever lengths up to 10–12 cm.
Microcontroller Unit	Arduino UNO R3	**Operation Voltage:** 5 V DC (logic level); recommended input voltage 7–12 V via barrel jack; absolute limit 20 V. **Clock Frequency:** 16 MHz quartz crystal oscillator ensuring stable timing and control response. **Flash Memory:** 32 KB (of which 0.5 KB used by the bootloader). Stores control algorithms, PID loops, and communication protocols. **Power Consumption:** 50–70 mA at 5 V during operation. Power can be supplied via USB or external DC source (7–12 V). **Operating Temperatures:** 0°C to 70°C, suitable for laboratory and industrial environments. **Digital I/O pins:** 14 pins (0–13), of which 6 provide PWM output for servo/motor control. Typical pins used: D3–D9 for arm joints, D10–D12 for driver enable and direction control.

		Analog Input Pins: 6 pins (A0–A5) for voltage or sensor data input. Supports 10-bit ADC resolution (0–1023 steps).

1.2.4 Technical and Non-technical Considerations and Constraints in Design Process

Technical Constraints:

- Space: Limited to a 1m × 1m workspace.
- Processing Speed – The system should detect and classify each object in under 500 ms, ensuring real-time responsiveness for smooth sorting.
- Payload – The robotic arm must safely handle weights between 1–5 kg, with limits set according to safety factors to prevent strain or accidents.
- Hardware – Core components include servo motors for precise movement, microcontrollers for control and coordination, and GPU support to handle AI training and heavy vision tasks efficiently [7].

1.2.5 Applicable compliance, standards, and codes

Safety Regulations: Compliance with ISO 10218 and IEC 61508 standards.
Environmental: Operable in 10°C – 45°C with humidity up to 80%.
Societal and Ethical: Prevents worker health hazards by reducing human contact with chemicals and ensuring transparency in AI operations.
Sustainability: Contributes to reduced waste, resource efficiency, and long-term industrial adaptability.

ISO 10218 – Safety standards for robotic systems.
ISO/TS 15066 – Guidelines for collaborative robots.
IEC 61508 – Functional safety of electronic and programmable systems.
IEEE 1872 – Ontologies for Robotics and Automation.
ISO 9001 – Quality management and validation

1.3 Systematic Overview/summary of the proposed project

The project proposes an AI-powered robotic arm with 4 degrees of freedom, capable of intelligent object sorting. It integrates computer vision (U-Net model for segmentation) and LLM-based reasoning (Gemini 2.0) for real-time adaptability. A camera system captures the environment, and breaks down the environment into a 3D coordinated zone while AI models identify and localize objects in 3D space. The robotic arm, powered by Arduino controllers and optimized through kinematics and PID tuning, executes motions for picking and placing objects.

1.4 Conclusion

In summary, the proposed project addresses a critical industrial problem by introducing an AI-enhanced robotic arm for efficient, accurate, and scalable object sorting without the need for manual coding, saving hundreds of hours. The project ensures real-time performance, compliance with safety standards, and industrial relevance. Its integration of functional precision, non-functional scalability, and ethical considerations makes it a valuable contribution to the advancement of automation in current industries.

Chapter 2: Project Design Approach [CO5, CO6]

2.1 Introduction

To control the robotic arm effectively, two design approaches have been introduced. The first one involves OpenCV and U-Net–based machine learning image segmentation, and the second one implements an intelligent decision-making core in the form of a Large Language Model (LLM). The two approaches implement vision-based object detection on the arm processor to extract 3D coordinates of objects and to regulate robotic movements. Both approaches will require an arm with a gripper. The Arm and Gripper mechanism are as follows:

Arm Mechanism

Figure 1 illustrates the control architecture of an arm with 4 degrees of freedom, where a microcontroller receives a 3-dimensional coordinate as input and, using this data, determines the target position. It processes this data and sends control signals to four servo motors: Base, Elbow, Arm, and Gripper. These servos work in coordination to position and manipulate objects in 3D space. The entire system is powered through a dedicated power supply, ensuring consistent operation of both the microcontroller and the servos.

The diagram illustrates the control architecture of an arm with 4 degrees of freedom, where a microcontroller receives a 3-dimensional coordinate as input and, using this data, determines the target position.

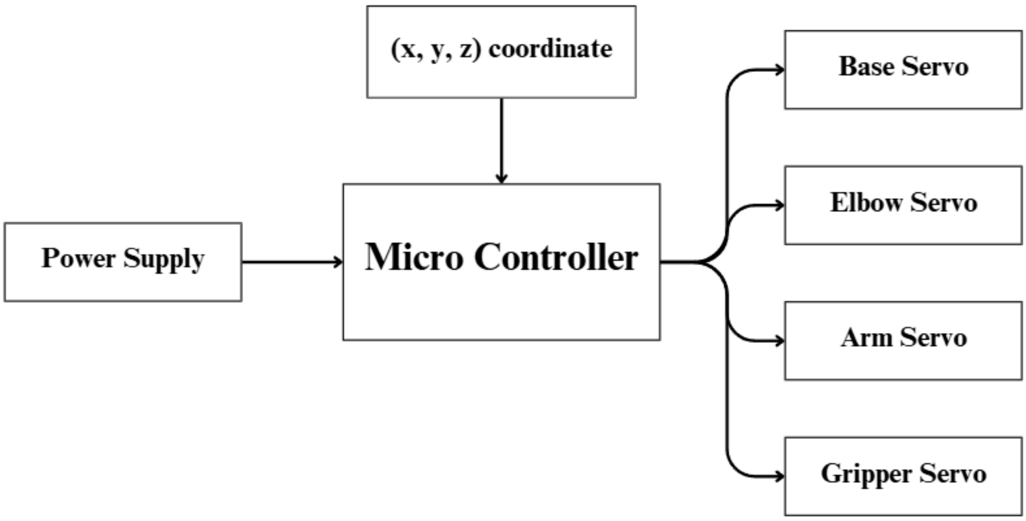

Fig 1 - Robotic arm diagram: microcontroller controls Base, Elbow, Arm, and Gripper servos with (x, y, z) coordinates, powered by a power supply.

Gripper Mechanism

The gripper starts its action (workflow diagram shown in Figure 2) when its mechanical claw reaches the right spot to start working. Once in place, a motor activates a slider mechanism that adjusts the gripper fingers to get them into the right position. Then, the fingers open up just enough to match the size of the object they're about to grab. With everything lined up, the gripper gets ready to grip.

Now comes the balancing act. The system checks to see if the contact force between the two fingers is even. If things are off, the built-in rotating dot matrix sensors on each finger step in. These sensors measure how the pressure is distributed. Using that info, a motor turns a gear that tweaks the angle of the fingers to fix the imbalance. This adjustment keeps looping until the forces are nicely balanced.

Once the pressure is evenly spread out, the fingers tighten up to boost the grip. Then the system checks: Is the grip strong enough? If yes, the work is done—the object is securely held. And that wraps up the process.

This kind of closed-loop system is great for jobs where precision matters—like picking up soft, delicate fruits without squishing them.

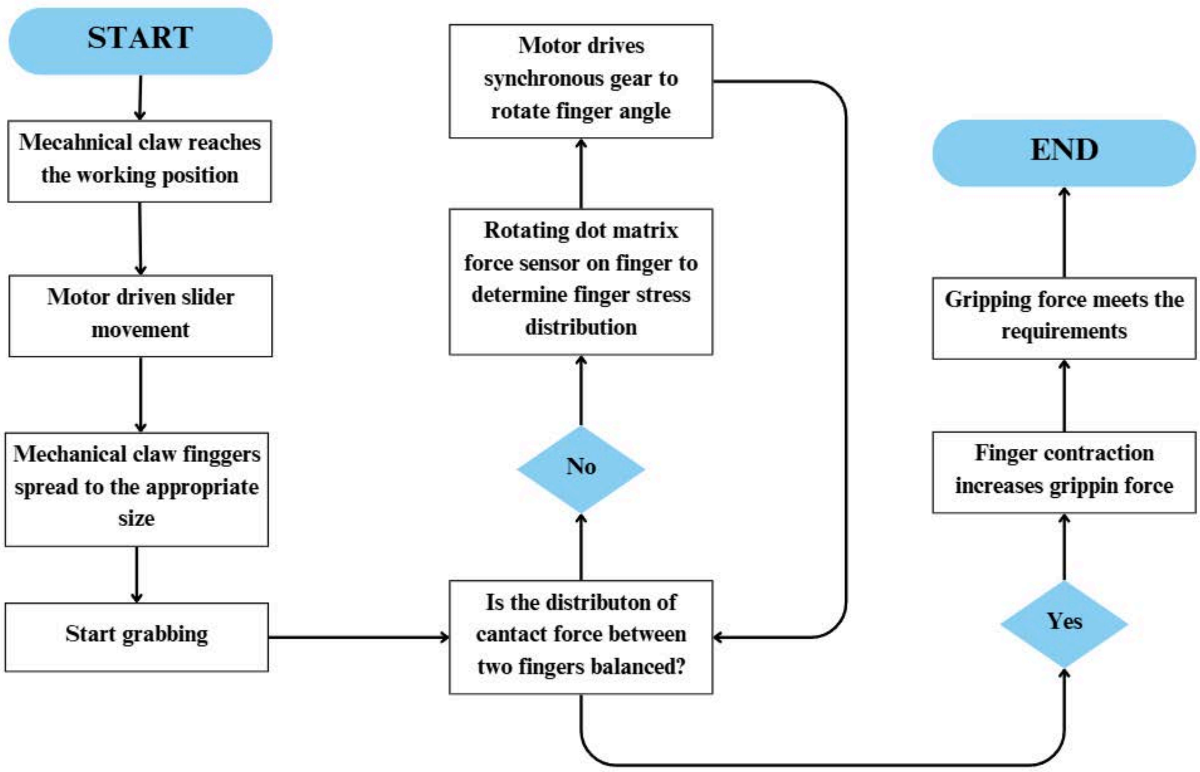

Fig - 2: Workflow Diagram of the Gripper

2.2 Identification of Multiple Design Approaches

Two distinct design methodologies are explored for integrating computer vision and intelligent control with the robotic arm:

- **Design Approach 1 – Image Segmentation Model**
  - Relies on a custom-trained segmentation model.
  - Requires a dataset of raw images and labeled masks for training.
  - Uses OpenCV along with a deep learning model (U-Net) for accurate object detection.
  - Requires pre-coding for object sorting tasks.
  - Provides structured, deterministic object recognition and mapping.
- **Design Approach 2 – Large Language Model (LLM) Based System**
  - Utilizes a pretrained LLM with multimodal capabilities (vision + language).
  - Does not require a custom dataset for training.
  - Learns and makes decisions directly from the live camera feed.
  - Continuously interprets input, estimates object positions, and decides actions.
  - Interfaces directly with the robotic arm through predefined function calls like move, rotate, pick, and place.

2.3 Description of Multiple Design Approaches

Design Approach 1: Image Segmentation Model

This method (workflow diagram in Figure 3) employs a machine learning–based segmentation model trained on a custom dataset. The dataset consists of raw images of target objects and their corresponding labeled masks, enabling the model to accurately separate objects of interest from the background. The segmentation process is enhanced by combining OpenCV [17] with a deep learning model - U-Net architecture [16], which improves accuracy in object detection and localization.

The camera system captures images of the workspace, and with the help of OpenCV and U-Net, the system processes these images to extract the 3D coordinates (x, y, z) of detected objects. These coordinates are then transmitted to the robotic arm's processor, which maps them into joint movements. The processor sends appropriate control signals to the four servos—Base, Elbow, Arm, and Gripper—allowing the arm to execute precise pick-and-place operations.

Although this approach provides structured detection and high accuracy, it requires prior dataset preparation, labeling, and training, as well as pre-coded logic for object sorting. As a result, it can be rigid when encountering new or unfamiliar objects.

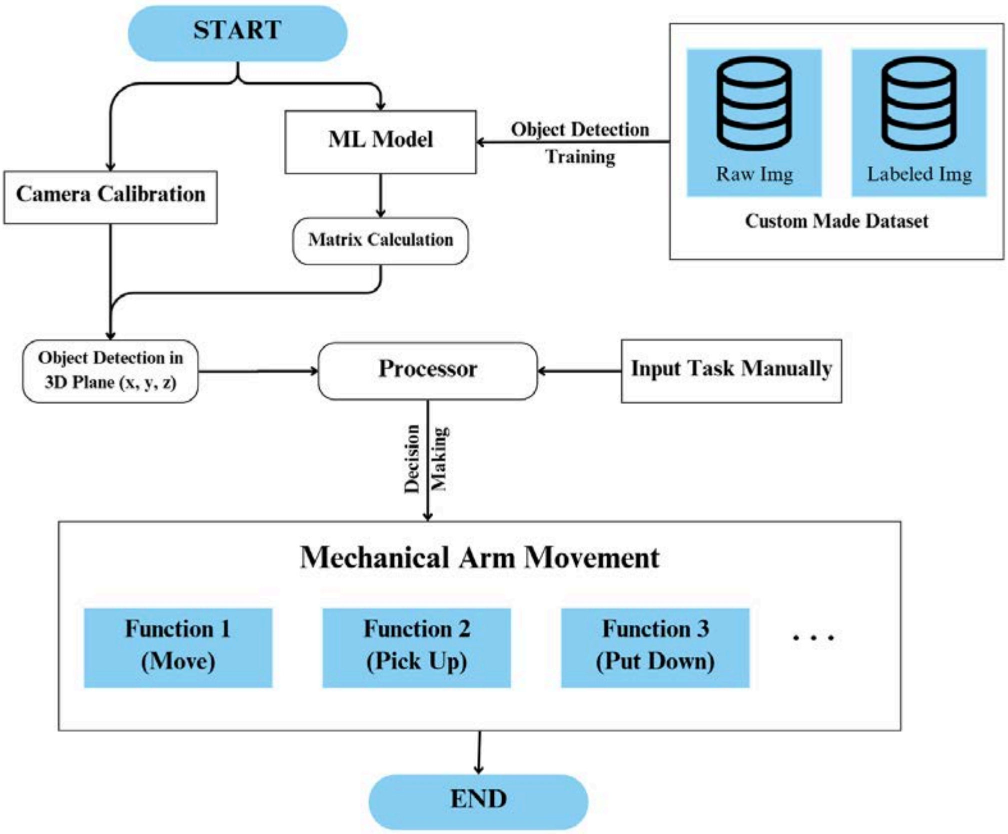

Fig - 3: Workflow Diagram of the Mechanical Arm using ML model (Design Approach 1)

Design Approach 2: Large Language Model (LLM) Based System

The second approach (workflow diagram in Figure 4) integrates a large language model as the central intelligence unit. Unlike the first approach, it does not rely on task-specific training datasets. Instead, the LLM leverages its pretraining on vast multimodal data (text and images) to identify and understand a wide range of objects.

A camera system continuously feeds visual data into the LLM, which processes the input to recognize objects, estimate their 3D coordinates (x, y, z), and determine appropriate actions. Crucially, the LLM (Gemini 2.0 flash lite) [11] is capable of making decisions directly from the live camera feed without needing a custom-made dataset, effectively allowing the model to adapt and learn by itself from what it sees.

The LLM directly interfaces with the robotic arm's processor, invoking predefined function calls such as move, rotate, pick, and place. This eliminates the need for intermediate mapping systems or pre-coded sorting logic. The result is an intelligent robotic system capable of adapting to varied situations with minimal configuration, making it highly versatile for manufacturing environments.

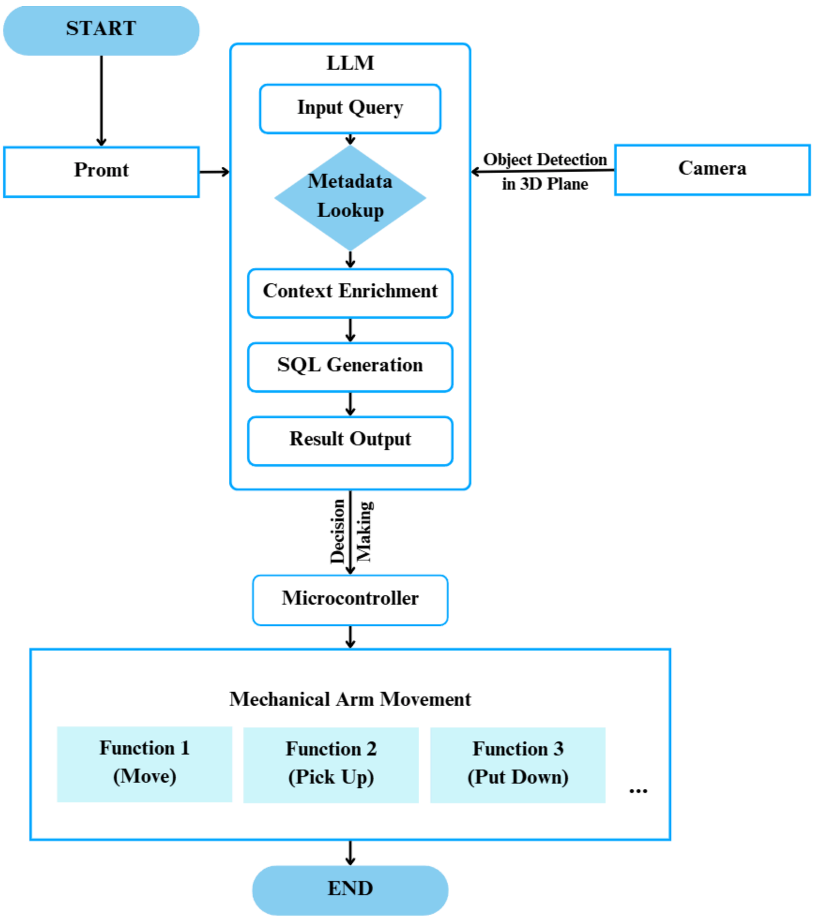

Fig - 4: Workflow Diagram of the Mechanical Arm using LLM model (Design Approach 2)

2.4 Analysis of Multiple Design Approaches

- **Flexibility**
  - Approach 1 is work-specific and must include datasets, training, and pre-coded sort logic. It excels in accuracy for pre-specified work.
  - Approach 2 has more flexibility as the LLM can handle different types of objects and contexts without dataset preparation and without explicit pre-coding.
- **Complexity and Setup**
  - Approach 1 requires a more extensive setup phase as it involves dataset preparation, annotation, training of U-Net models, and coding of sorting functions.
  - Approach 2 minimizes set-up requirements in that it leverages pretrained knowledge and also learns from the camera feed. Nevertheless, hardware integration and retaining real-time capability may be computationally costly.
- **Precision**
  - Approach 1 provides highly precise segmentation and correct coordinate extraction, and it performs best in areas with structure.
  - Approach 2 relies on generalized reasoning in the LLM, and it can lose some precision but have more adaptability in complex or unstructured cases.
- **Computational Demand**
  - Approach 1 can be optimized for lightweight, real-time execution once the U-Net model is trained.
  - Approach 2 may require more powerful hardware to run the LLM efficiently, especially for real-time vision and decision-making.
- **Scalability**
  - Approach 1 doesn't scale badly as new objects can be incorporated by re-training and re-coding.
  - Approach 2 becomes easier to scale as it can generalize across varied object types and environments without being retrained.
- **Cost Effiecency**
  - Approach 1 only needs to be trainned once per task, which will require to make custom dataset once, making it a bit more cost efficient.
  - Approach 2 depending on LLM, which may need cloud subscription every periodic time or run locally which can have a low to mid amount of power consumption. Meaning this approach can be a bit costly in the long run.

2.5 Conclusion

Both design approaches present viable methods for integrating vision-based intelligence into the 4 DOF robotic arm system. Design Approach 1, which combines OpenCV with U-Net, ensures precision and reliability in structured, repetitive manufacturing tasks but requires dataset training and pre-coded sorting, making it less adaptable to new scenarios. In contrast, Design Approach 2, which utilizes a Large Language Model, provides greater adaptability by learning directly from the live camera feed without the need for a custom dataset or pre-coded logic.

The choice between the two approaches would solely be based on the intended use case. When accuracy, consistency, and lightweight functionality become most critical, then segmentation model composition would be most appropriate. When scalability, flexibility, and minimum setup become most critical, then an LLM-based system would offer numerous advantages. The potential benefit could be an integration of both methods in an extremely intelligent and robust robotic system, employing both U-Net segmentation for accuracy and an LLM-based decision-making to allow for flexibility

Chapter 3: Use of Modern Engineering and IT Tool. [CO9]

3.1 Introduction

This project is an object sorting mechanical arm based on artificial intelligence which was made practical through extensive dependence on modern engineering and information technologies. The technologies were at the core of bridging the theory and practice gap in design and practical realization, respectively. From the early stages of mechanical design to the final prototype testing, every tool contributed uniquely by enabling accurate modelling, simulation, and validation of the system. Without such technologies, it would have been difficult to handle the project's complexity- especially tasks like inverse kinematics, AI-based object detection, and real-time robotic control. This chapter highlights how these tools were carefully chosen, integrated, and applied throughout the project.

3.2 Select Appropriate Engineering and IT tools

Choosing the right set of tools was a critical point, because the project involved diverse fields- mechanical design, electronics, artificial intelligence, and robotics. Each of these areas required specialized resources:

- **SolidWorks:** SolidWorks software was selected to design the robotic arm's 3D structure. It allowed us to visualize how the 4-DOF arm would look and operate in real time, also stress and load analysis was performed to ensure the design could handle the expected payload.

- **MATLAB Simulink:** The Matlab Simulink software was used to handle the complex mathematics of kinematics and PID control. This ensured that the arm's movement would be precise and stable, even under varying load conditions.

- **Proteus:** This software was chosen for simulating the electronic circuits (PCB) before moving to hardware. The main advantage of this part is to minimize the risks of wiring errors or hardware failures, saving both cost and time.

- **PyCharm:** PyCharm served as the coding environment for building and training the machine learning model, integrating computer vision algorithms, and processing all the datasets efficiently.

- **Arduino IDE:** This software was used for microcontroller programming which allows smooth communication between the arm's mechanical parts and its control system.

- **LLM Gemini 2.0 Flash Lite:** A large language model, was integrated as a decision-making tool. Particularly for this section Gemini 2.0 Flash Lite is used.

Unlike traditional approaches that require retraining for new categories, Gemini 2.0 Flash Lite could adapt quickly by interpreting natural language instructions and responding intelligently.

The combination of all these tools ensured that each stage of the project had the right resources to move forward effectively.

3.3 Use of Modern Engineering and IT tools

Once all the tools were selected, they were applied strategically across different phases of the project.

- **Mechanical Design and Validation:** Using SolidWorks the robotic arm's structure was modelled with accurate measurements for each joint and link. Also, stress simulations helped to determine whether the chosen materials and design could withstand the forces applied during operation. Again this reduced the risk of failure when the prototype was built.

- **Kinematics and Control Simulation:** MATLAB Simulink was crucial in modelling the arm's motion. By simulating the inverse kinematics and tuning PID controllers the project achieved stable stepper and servo motor responses with settling times between 0.5 to 0.8 seconds. This level of stability was essential for smooth and accurate movements.

- **Circuit Simulation:** Proteus allowed to test and verify circuits for motor control, gripper mechanisms, and sensor integration in a safe virtual environment. This ensured that once all the components were implemented in hardware the system would operate with more effective and minimal debugging.

- **AI Model Development with Gemini:** The project integrated Gemini 2.0 Flash Lite, a multimodal large language model to handle object recognition and decision-making without requiring retraining for new tasks. Again, connected to the camera system, it could identify objects, estimate object's 3D positions, and translate simple prompts into robotic arm commands.

- **Hardware Control:** The Arduino IDE enabled the programming of microcontrollers responsible for running the arm's servos and gripper. Two Arduino Uno were used-one for arm movements and another one for managing the gripper's pressure sensors. This division of control made the system more efficient and responsive.

- **AI-Driven Decision Making:** Gemini 2.0 Flash Lite added an extra layer of intelligence. Instead of manually coding every action for the mechanical arm movement and taking decisions, simple instructions such as “pick up the orange and place it in the box” could be understood and executed. The model could recognize the object, estimate its 3D coordinates, and trigger the robotic arm to complete the task. This adaptability greatly reduced the need for reprogramming when tasks changed.

3.4 Conclusion

The integration of modern engineering and IT tools was the main reason behind the success of this project. Each tool addressed a specific challenge- whether it was mechanical stress testing, circuit validation, AI-based vision, or decision-making. Combining all of them in a structured workflow, the project was able to move from a concept to a working prototype with accuracy and confidence. These tools not only help to make it possible to solve the immediate technical challenges but also provide a scalable framework that can be extended to future applications in healthcare, logistics, and other industries. The project reflects how the right use of modern technologies can transform complex engineering problems into practical and sustainable solutions.

Chapter 4: Optimization of Multiple Designs and Finding the Optimal Solution. [CO7]

4.1 Introduction

Optimization in engineering design involves critical study of different approaches to arrive at an optimal, most effective, and durable solution under given constraints. In this 4 DOF robotic arm system, two likely design solutions were proposed - Design Approach 1 and Design Approach 2. The detailed process of optimization, comparison, and choice of optimal design solution follows in this section.

4.2 Optimization of Multiple Design Approaches

Both design approaches were optimized and tested against criteria such as computational efficiency, adaptability, accuracy, scalability, and usability:

4.2.1 Design Approach 1 (OpenCV + U-Net):

Custom-Made Image Segmentation Dataset:

A dataset has been specifically created for training and evaluating an image segmentation model used in object (an orange) detection and localization tasks. It comprises a total of 416 image pairs, categorized into 254 positive samples (where the target object is present) and 162 negative samples (where the object is absent). The dataset is divided into two components:

Raw Images (Figure 5): These are RGB images captured under controlled lighting and background conditions to ensure consistency. Each image is labeled with a unique frame identifier (e.g., frame_0000, frame_0001, etc.). The object of interest is consistently positioned within varying frames to capture different orientations and placements, simulating real-world scenarios.

Labeled Images (Figure 6): These are the binary segmentation masks for each raw image obtained through the makesense.ai annotation tool. In them:

1. Pixel value 255 (white) indicates the non-object or background region.
2. Pixel value 0 or black indicates the target object.

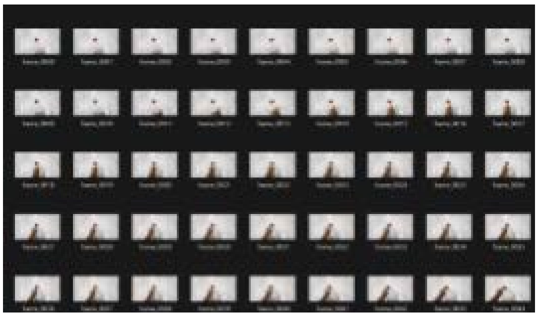

Fig - 5: Raw Image (Total Image = 416)

Fig - 6: Labeled Image (Total Image = 416)

The minimalistic and high-contrast segmentation masks make it easy for pixel-wise classification when training the model. The binary format allows the object's visual bounds to be learned clearly by the segmentation model, and thus it becomes accurate in object recognition for similar objects when presented with unseen data.

This dataset forms a core building block for building an object detection pipeline used to identify and localize the spatial (x, y, z) coordinates of the object when paired with a camera system and computer vision processing software such as OpenCV. It's especially ideal for robotic manipulation applications where accurate object localization plays a key role in task performance.

Model Train Results:

```
Epoch 95/100 | Train Loss: 0.0056 | Val Loss: 0.2897 | Val Acc: 0.9591
Epoch 96/100 | Train Loss: 0.0057 | Val Loss: 0.2917 | Val Acc: 0.9592
Epoch 97/100 | Train Loss: 0.0059 | Val Loss: 0.2913 | Val Acc: 0.9589
Epoch 98/100 | Train Loss: 0.0060 | Val Loss: 0.2732 | Val Acc: 0.9592
Epoch 99/100 | Train Loss: 0.0061 | Val Loss: 0.2882 | Val Acc: 0.9595
Epoch 100/100 | Train Loss: 0.0061 | Val Loss: 0.2808 | Val Acc: 0.9598
Model saved as unet_model.pth
```

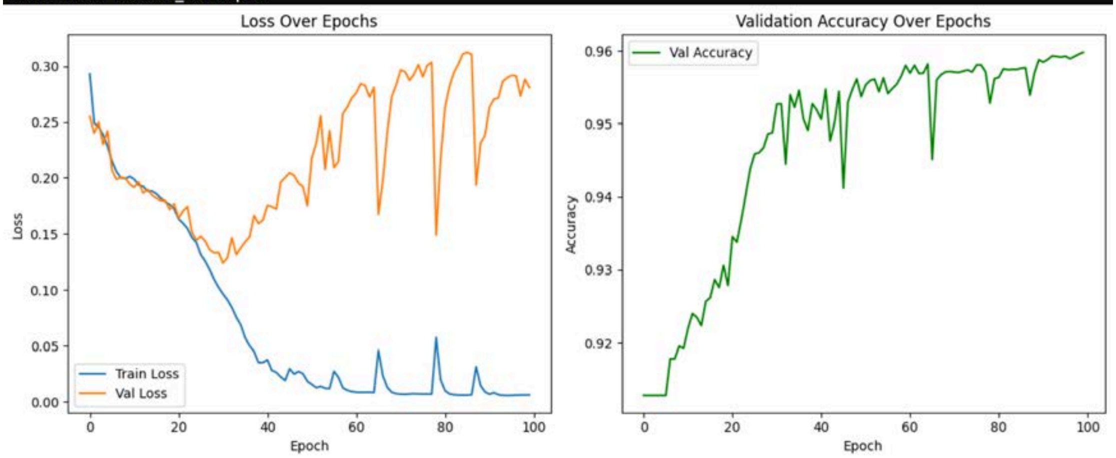

Fig - 7: Loss vs epochs and Train loss

Trained on the custom dataset based on a U-Net architecture [16], the image segmentation model ran for 100 epochs. The generalization performance and the learning behavior of the model were checked through monitoring of the training and validation metrics.

Loss and Accuracy metrics (from epoch graph):

- Final Training Loss: 0.0061
- Final Validation Loss: 0.2808
- Final Validation Accuracy: 0.9598
- Training Time: About 4 hours

These plots confirm that the model obtained extremely small loss on training and high and stable accuracy on the validation set, indicating good convergence and good learning from data as well. The training time of roughly 4 hours indicates a balance of computational efficiency and good optimization.

Real-Time Object Localization via U-Net and OpenCV

For the precise localization and manipulation of objects, a real-time localizing system utilized a trained U-Net image segmenting model coupled with OpenCV and a dual camera system. The system detects an orange and computes the 3D coordinates (X, Y, Z) of the object in order to drive a robotic arm.

Coordinate Extraction through Dual Cameras:

- **Camera 1 (Front View):**
  This camera gives the X and Y locations of the object. The object in each image is segmented by the learned U-Net network, and OpenCV computes the centroid location in the image space.

- **Camera 2 (Side View):**
  This camera sits orthogonally to the first and captures the Z coordinate (depth) of the object, utilizing the same segmentation model.

By integrating the two cameras' outputs, the system obtains full 3D localization of the object in the workplace.

Integration with the Robotic Arm:

The obtained coordinates (X, Y, Z) are translated into joint angle commands using inverse kinematics. These angles correspond to the robotic arm's four degrees of freedom, denoted as J1 through J4, which define the arm's kinematic configuration. Each joint angle controls a specific axis or movement type (e.g., base rotation, shoulder lift, elbow motion, and gripper orientation).

This vision-based control loop allows the robotic arm to autonomously identify, locate, and interact with target objects in its environment without external input during operation.

In short -

- Optimized through dataset-specific training for precise object detection.
- Requires predefined coding for object sorting, which limits flexibility.
- Achieves high accuracy in structured environments but lacks adaptability when faced with untrained objects.
- Lightweight and reliable once trained, making it computationally efficient in constrained tasks.

4.2.2 Design Approach 2 (LLM-based Gemini):

Gemini 2.0 Performance Metrics

1. Multimodal Reasoning [13] - [15]

- Scene Classification Accuracy: 93.7%
- Image Captioning Accuracy: 95.5%
- Visual Grounding Accuracy: 57.8%
- Spatial Relationship Understanding: 41.3%
- Change Detection Accuracy: 69.0%

2. Object Detection and Localization [16]

- **2D Object Detection**: Gemini 2.0 can predict 2D bounding boxes from natural language queries, enabling open-world object detection without explicit training on specific objects.
- **3D Object Detection**:
  - **Gemini Robotics-ER**: Achieved an Average Precision (AP) of 48.3% at an IoU threshold of 0.15, surpassing previous state-of-the-art models.
  - **Gemini 2.0 Flash**: Recorded an AP of 30.7% under the same conditions.

3. Spatial Understanding [15]

- Object Localization Accuracy: 61.2%
- Object Counting Accuracy: 63.4%
- Attribute Recognition Accuracy: 54.8%
- Attribute Comparison Accuracy: 79.7%

4. Embodied Reasoning [17]

- **Task Completion Rate**: Gemini Robotics-ER demonstrated a near 2x improvement in task completion compared to its predecessors, showcasing enhanced embodied understanding.

5. Comparison Between other models:

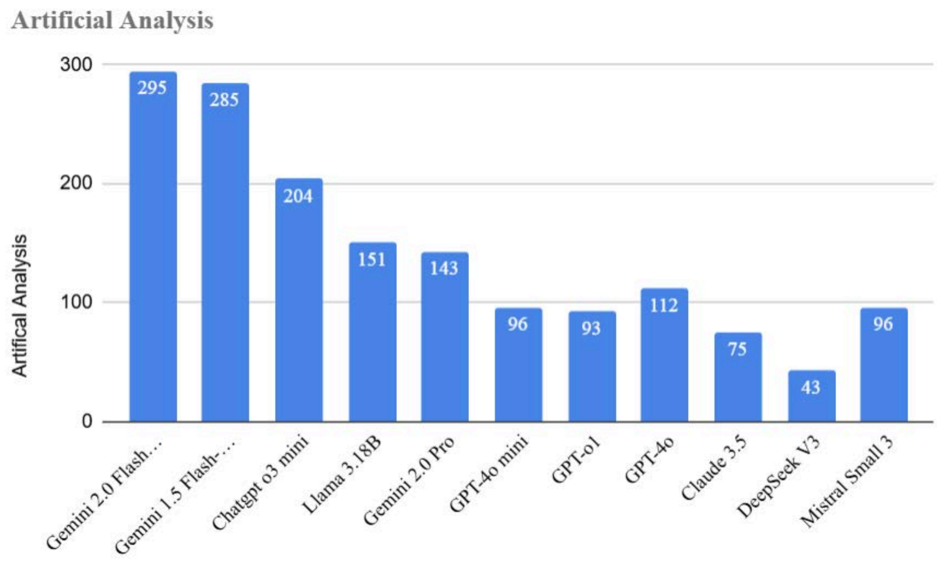

Fig - 8: Comparison of token generation speed (tokens per second) across different Large Language Models (LLMs)

The graph illustrates the token generation rates of various LLMs. Gemini 2.0 Flash and Gemini 1.5 Flash achieve the highest throughput, while smaller models like DeepSeek V3 and Claude 3.5 show comparatively lower speeds. This highlights significant performance differences among models, with Flash variants optimized for rapid inference.

Training Metrics

Specific training metrics such as Final Training Loss, Final Validation Loss, Final Validation Accuracy, and Training Time for Gemini 2.0 are not publicly disclosed. However, the model's performance on various benchmarks indicates its effectiveness in learning and generalization across tasks.

Object Detection using LLM (Gemini 2.0)

Overview:

In this approach, the Gemini 2.0 large language model is used not only for object detection but also for decision-making and robotic arm control [14]. For simulation purposes, we tested the model using a red ball and a brown box. Upon receiving the prompt:

> "Locate the red ball and the box. Once found, place the red ball inside the box."

(The top two image shows the Input from the Camera and the bottom image shows the output (finding the ball's position - x, y, z coordinates) from the prompt by Gemini 2.0)

(The top two image shows the Input from the Camera and the bottom image shows the output (finding the box's position - x, y, z coordinates) from the prompt by Gemini 2.0)

Fig - 9: Object Detection using both ML model and Gemini

The model performs the following:

Workflow:

Dual-Camera Input:

- Camera 1 provides (x, y) coordinates from a front-facing view.
- Camera 2 provides z-axis (depth) information from a side view.

Object Localization:

- Gemini 2.0 estimates the 3D coordinates (x, y, z) of both the orange and the box using visual input.
- For example (shown in the figure):
  - Ball: (x=200, y=250, z=50)
  - Box: (x=200, y=200, z=300)

Action Planning & Execution:

- Based on the object positions, Gemini 2.0 autonomously triggers robotic arm movement.
- The LLM directly calls motion functions and decides:
  - When to move.
  - Where to move.
  - Which coordinates to reach/
- Joint angles J1–J4 are internally calculated for inverse kinematics and fed to the arm control system.

  - Optimized through integration of multimodal capabilities (vision + language), eliminating the need for custom dataset preparation.

  - Capable of **real-time learning and decision-making** from live camera feeds.

  - Reduces development time since it avoids manual data labeling and retraining for new object types.

  - Adaptive and scalable across diverse scenarios, although more computationally demanding.

4.3 Identify Optimal Design Approach

Based on the optimization analysis:

- **Accuracy:** Approach 1 excels in controlled, predefined tasks.
- **Flexibility & Scalability:** Approach 2 surpasses due to its generalization ability.
- **Setup & Usability:** Approach 2 reduces development effort by removing the need for dataset creation.
- **Future-readiness:** Approach 2 is more sustainable in dynamic or unpredictable environments.

Considering these, the **optimal design solution proved to be Design Approach 2 (LLM-based Gemini system)** owing to increased adaptability, decreased setup overhead, and smart processing of variable manufacturing situations without successive retraining.

4.4 Performance Evaluation of Developed Solution

Performance evaluation (in simulation) was carried out against key engineering metrics:

Aspect	U-Net Based Vision System	LLM (Gemini 2.0) Based System
Core Mechanism	Trained a U-Net model for image segmentation and OpenCV for coordinate extraction.	Uses a large language model (Gemini 2.0) to understand prompts, detect objects visually, estimate coordinates, and trigger arm actions.
Accuracy	95.98% Accuracy may drop if the lighting or background changes.	Accuracy is approximate and visually estimated. Relies on LLM's reasoning without pixel-wise segmentation.
Loss Metrics	Final Training Loss: 0.0061 Final Validation Loss: 0.2808	No measurable training or validation loss (non-trainable model). Relies on pretrained visual reasoning.
Training Time	Around 4 hours on RTX 4050 GPU (100 epochs), Efficient and repeatable.	No training required by the user, but model response time and overhead are high (especially with vision tasks).

Precision of Output Coordinates	Extracted with pixel-level precision using mask predictions.	Estimated using visual cues (e.g., object distance to screen), leading to error margins in coordinates.
Kinematic Control	Arm angles (J1–J4) computed precisely from fixed (x, y, z).	Gemini triggers motion and generates (x, y, z), then also calls functions to control arm angles.
Ease of Use	Requires coding pipeline in OpenCV, training loop, mask post-processing, etc.	Prompt-driven: "Locate ball and box, move ball into box." Easy for non-programmers.
Flexibility & Scalability	Rigid. Must retrain/recode for new object types, views, or tasks.	Highly adaptive. New instructions (e.g., "pick green cup") can be executed with no retraining.
Dependency on Environment	Needs consistent lighting and camera angles; sensitive to noise in input.	More robust to diverse environments due to LLM's contextual understanding, but still limited by vision quality.
Speed of Execution	Fast, once trained, real-time segmentation and coordinate extraction.	Slower due to visual reasoning + language interpretation + inference pipeline.
Resource Requirements	Requires GPU for training (e.g., RTX 4050), OpenCV setup, and trained model weights.	Requires access to Gemini 2.0 or a similar LLM via API or cloud-based interface.
Output Format	Coordinates and joint angles are computed programmatically.	Natural language + computed coordinates + function call via interpreted response.

4.5 Conclusion

For our 4 DOF robotic arm, we found that **Design Approach 2 (LLM-based Gemini system)** is much more optimal through systematic optimization and evaluation. The OpenCV + U-Net model provides precision in structured environments as it needs pre-coded objection dataset. On the other hand, the LLM-based system offers adaptability, reduced setup time, and long-term sustainability. It makes the arm a more effective solution for real-world manufacturing applications.

Chapter 5: Completion of Final Design and Validation. [CO8]

5.1 Introduction

Development of 4 Degrees of Freedom (DOF) robotic arm matured through an orderly process of both hardware and software design. The end design incorporates multiple areas of engineering to ensure system reliability and operability. We fabricated an in-house PCB to reduce wiring and allow modular connections for drivers, sensors, and controllers. The 3D mechanics of the arm was optimized for stability and smooth movement throughout joints. Kinematic calculations were also carried out to convert 3D coordinates into angles at joints, serving as the base for precise arm movement. In order to fine-tune motion control, PID tuning was also incorporated to allow smooth servomotors and reduce positioning overshoot. The gripper mechanism was also optimized through force-sensitive resistors (FSRs) and precise mechanical calculations to find a balance between strength of grip and finesse, allowing it to lift dainty items without causing breakage. Finally, an LLM-based control system was implemented to process live camera feed, detect objects, estimate position, and take action through pre-coded robotic functions. These individually functioned to develop an intelligent and robust robotic arm solution.

5.2 Completion of Final Design

5.2.1 PCB Design:

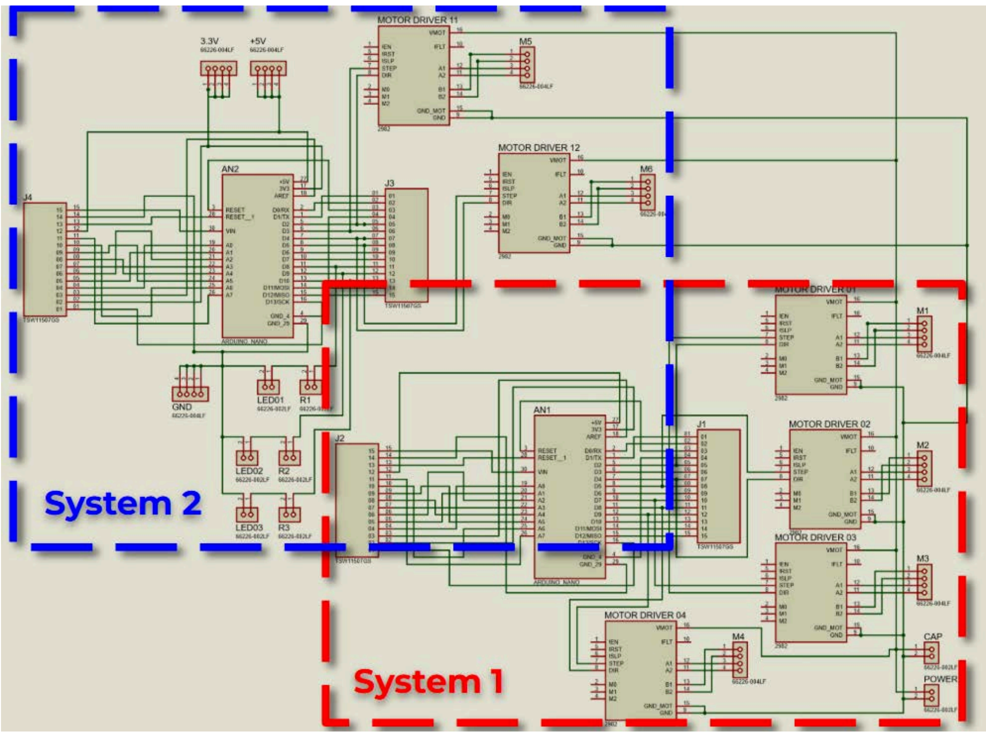

Fig - 10: Schematics of the PCB (System 1 controls the motor drivers and System 2 using I2C bus is connected to System 1 providing more pins to control servo and FSR Sensor)

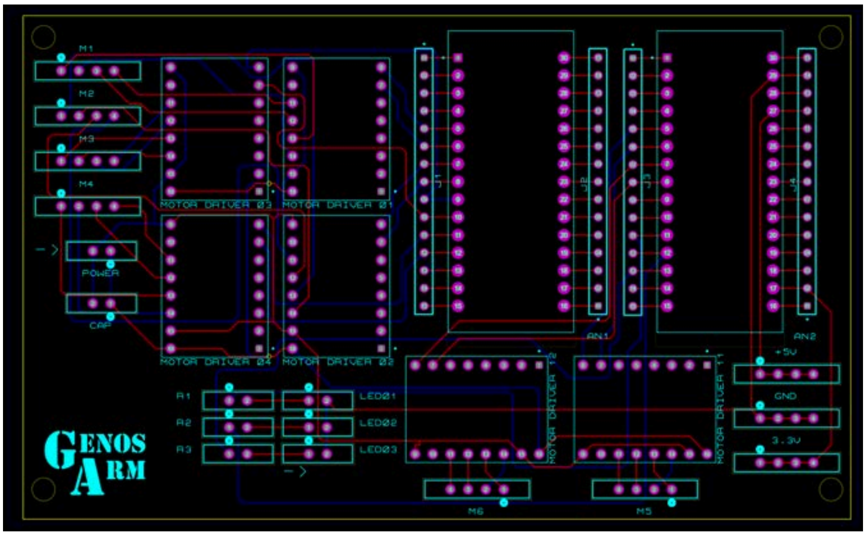

Fig - 11: PCB Design with optimal positioning of components

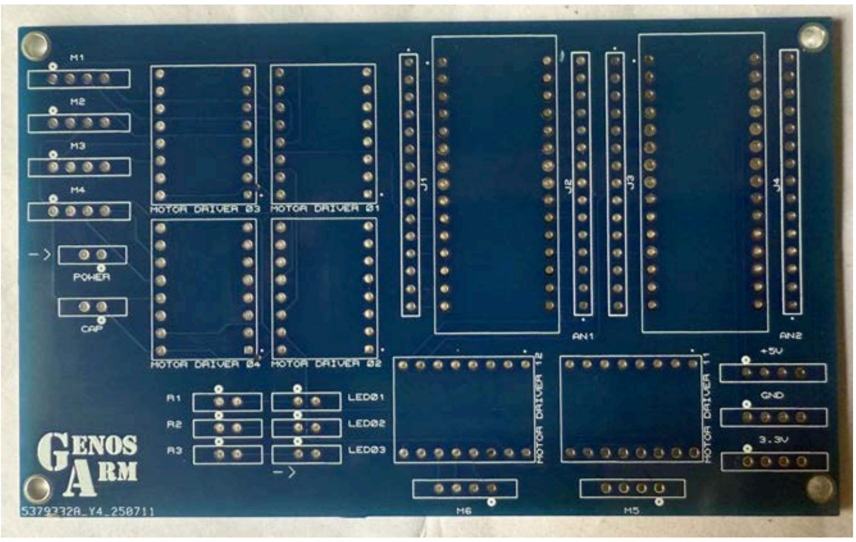

Fig - 12: Final Board after printing the PCB

The custom PCB for the 4 DOF robotic arm is designed to integrate sensing, control, and actuation in a compact form. The main components include:

- **2 × Arduino Nano slots** – for modular control and integrated processing (using I2C bus).
- **6 × DRV8825 driver slots** – for precise motor driving and step control.
- **6 × FSR (Force-Sensitive Resistor) sensor slots** – to provide tactile feedback for grip force measurement.
- **3 × LED slots** – for system status indication.
- **3 × Resistor slots** – paired with LEDs for safe operation and current limiting.

This layout reduces wiring complexity, improves reliability, and provides a streamlined interface for controlling and monitoring the robotic arm.

5.2.2 3D Design & Final Model:

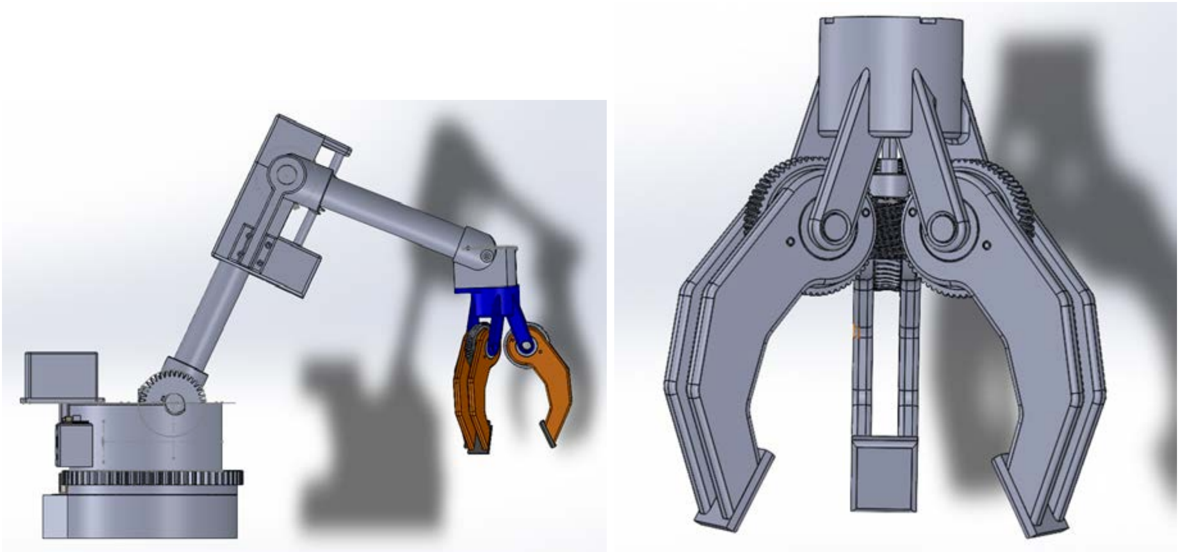

Fig - 13: 3D view of the full body structure of the Mechanical Arm and Arm Gripper

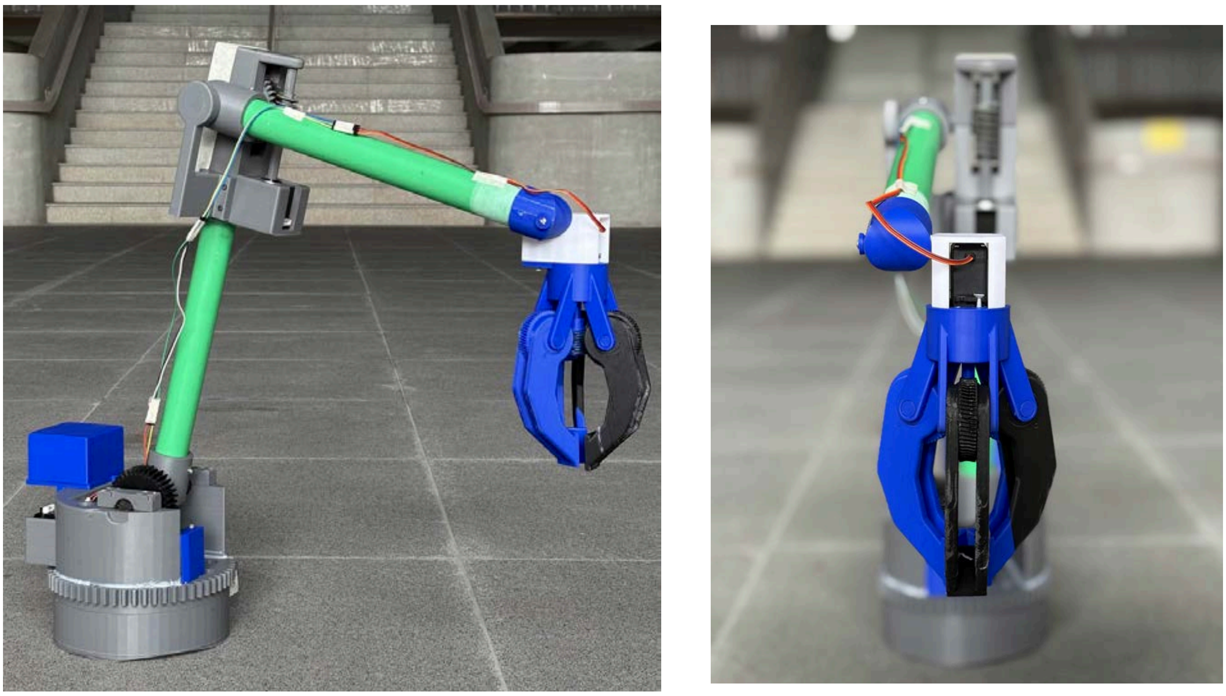

Fig - 14: Final Assembly of the Arm

Figure 13 showcases the 3D design, which was designed taking inspiration from a traditional mechanical arm used in an industry, and Figure 14 is the final assembly after 3D printing the whole arm.

5.2.3 Kinemetrics Calculation:

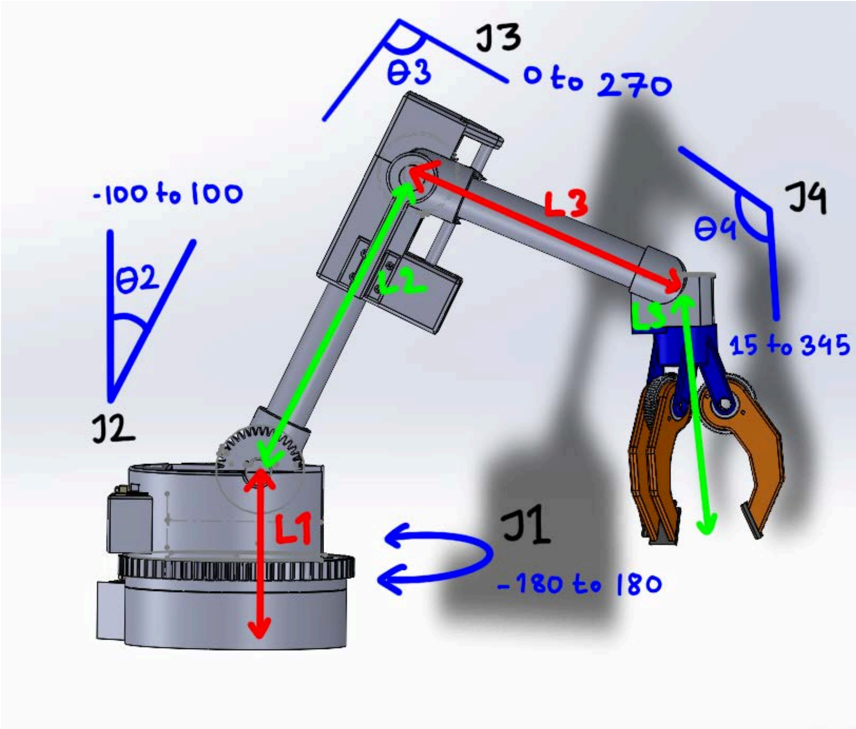

Fig - 15: Individual Joints naming of the simulated arm

1. Base Rotation (Joint 1):

$\theta_1 = \arctan(Y / X)$

2. Distance in XY-plane:

$D_0 = \sqrt{(X^2 + Y^2)}$
$D_1 = \sqrt{(D_0^2 + Z^2)}$

3. Angles in XZ-plane:

$\varphi_1 = \arctan(Z / D_0)$
$\varphi_2 = 90° - \varphi_1$

4. Using Law of Cosines:

$D_2 = \sqrt{(L_1^2 + D_1^2 - 2 \cdot L_1 \cdot D_1 \cdot \cos(\varphi_2))}$
$\varphi_3 = \arccos((L_1^2 + D_2^2 - D_1^2) / (2 \cdot L_1 \cdot D_2))$
$\varphi_4 = 180° - \varphi_2 - \varphi_3$

5. Offset Adjustment:

$D_3 = \sqrt{(D_2^2 + L_4^2 - 2 \cdot L_4 \cdot D_2 \cdot \cos(\varphi_3))}$
(L5 = L2+ L3)
If $D_3 > L_5$: Target is out of range

6. Reachable Case Calculations:

$\varphi_5 = \arccos((D_2^2 + D_3^2 - L_4^2) / (2 \cdot D_2 \cdot D_3))$
$\varphi_6 = \arccos((D_3^2 + L_2^2 - L_3^2) / (2 \cdot L_2 \cdot D_3))$
$\varphi_7 = \arccos((L_2^2 - L_3^2 - D_3^2) / (2 \cdot L_2 \cdot L_3))$
$\varphi_8 = 180° - \varphi_6 - \varphi_7$

7. Final Joint Angles: (Joint 2 , 3 & 4)

$\theta_2 = 180° - \varphi_3 - \varphi_5 - \varphi_6$
$\theta_3 = \varphi_7$
$\theta_4 = \varphi_8 + 180° - \varphi_3 - \varphi_5$

5.2.4 PID Tuning

When we run the arm, it is important to make it stable. To reduce overshoot, undershoot and to be fixed at a specific angle, a little bit of PID tuning is very crucial. With the help of Simulink, we could do the PID tuning and got a very good scenario about the settling time of almost 0.5s to 0.8s for all the joint servo motors. For each servo motor, we have obtained different transfer functions. For each function, we have a different time constant as the torques are different for the 4 joints. Also, the 4 different joint servos have to counter different weights. By calculation, we get the torque at joint 4 is around 0.53Nm, at joint 3 the torque is around 2.42Nm, at joint 2 the torque is around 6.03Nm, and at joint 1 the torque is around 19Nm.

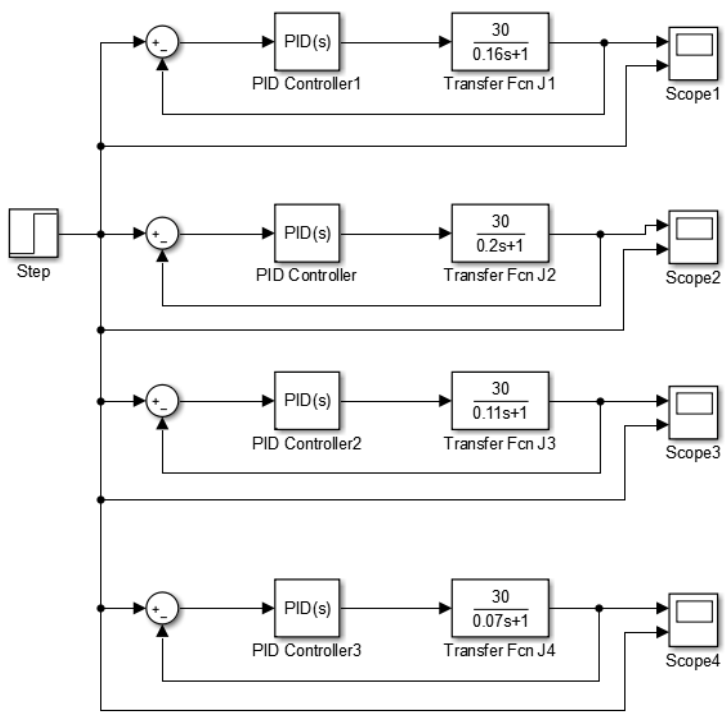

Fig - 16: PID Tuning block diagram of Each Servo Motor

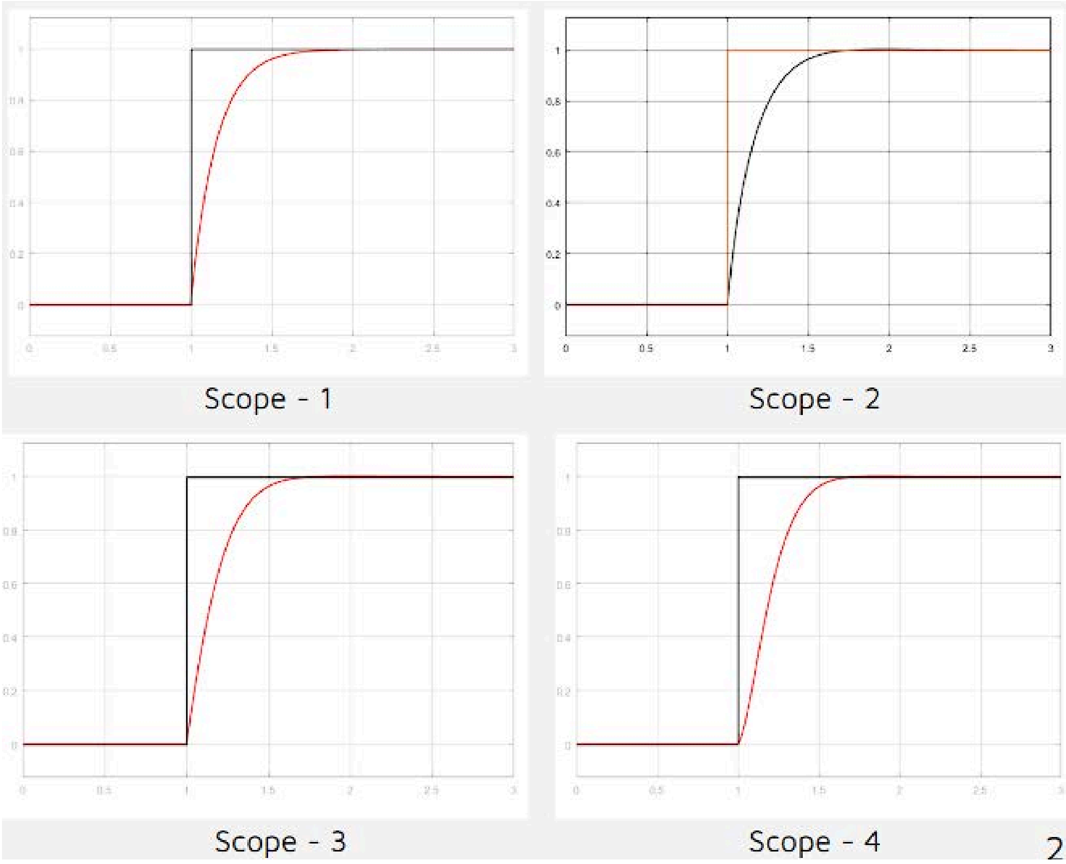

Fig - 17: PID Tuning Graph for each servo

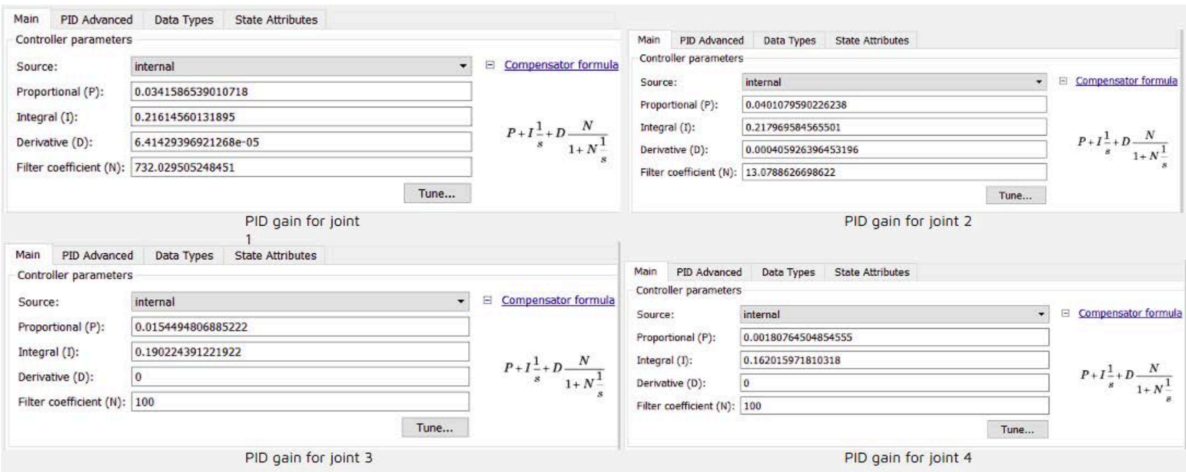

Fig - 18: PID Tuning scores.

5.2.5 Gripper Calculations

Gripping Force Calculation (F_g)

To hold an object securely without slipping, we calculate the required gripping force using the formula:

$$F_g = W \cdot SF / \mu$$

Where:

- W: Weight of the object (in Newtons)
- SF: Safety factor (1.5 to 2.5)
- μ: Coefficient of friction between gripper surface and object (typically 0.5–0.7 for rubber on plastic/fruit)
- $W=5$ N
  $SF=2$
- $\mu=0.6$

$F_g = (5 \times 2) / 0.6$
$= 16.67$ N

Torque Required (T)

Torque needed at the joint (or actuator) is:

$$T = F_g \cdot d$$

- d: Distance from the pivot point to the contact point (in meters)
- $d=0.04$ m

$$T = 16.67 \times 0.04 = 0.667 \text{ Nm}$$

Actuator Force ($F_{actuator}$)

If the gripper uses a **linkage mechanism** or **jaw rotation**, the actuator force is given by:

$$F_{actuator} = F_g / \sin(\theta)$$

- θ: The angle of the jaw (relative to actuator motion)
- $\theta=30^\circ$

$$F_{actuator} = 16.67 / \sin(30^\circ) = 33.34 \text{ N}$$

Finger Stress for Material Selection (σ)

To ensure the fingers won't break under load:

$$\sigma = F_g / A$$

Where:

- A: Cross-sectional area at the contact point (m^2)

- $A=100 \text{ mm}^2=1\times10^{-4} \text{ m}^2$

$$\sigma = 16.671\times10^{-4} = 166{,}700 \text{ Pa}$$
$$= 166.7 \text{ kPa}$$

ABS plastic is considered

Payload Constraint

To avoid overloading the robotic arm:

$$M_g + M_p \leq M_{robot\_arm}$$

Where:

- M_g: Mass of the gripper (0.357 kg)
- M_p: Mass of the payload
- M_{robot_arm}: Max payload capacity of the robotic arm (5.17 kg)

Theoretically, the maximum mass it can pick up is 4.813 kg. So, considering safety and stability,

$$4.813 \text{ kg} * 30\% = 1.4439 \text{ kg}$$

Therefore, the maximum mass it can pick up realistically is 1.4439 kg.

5.2.6 Gripper Circuit Design (Designed using Proteus)

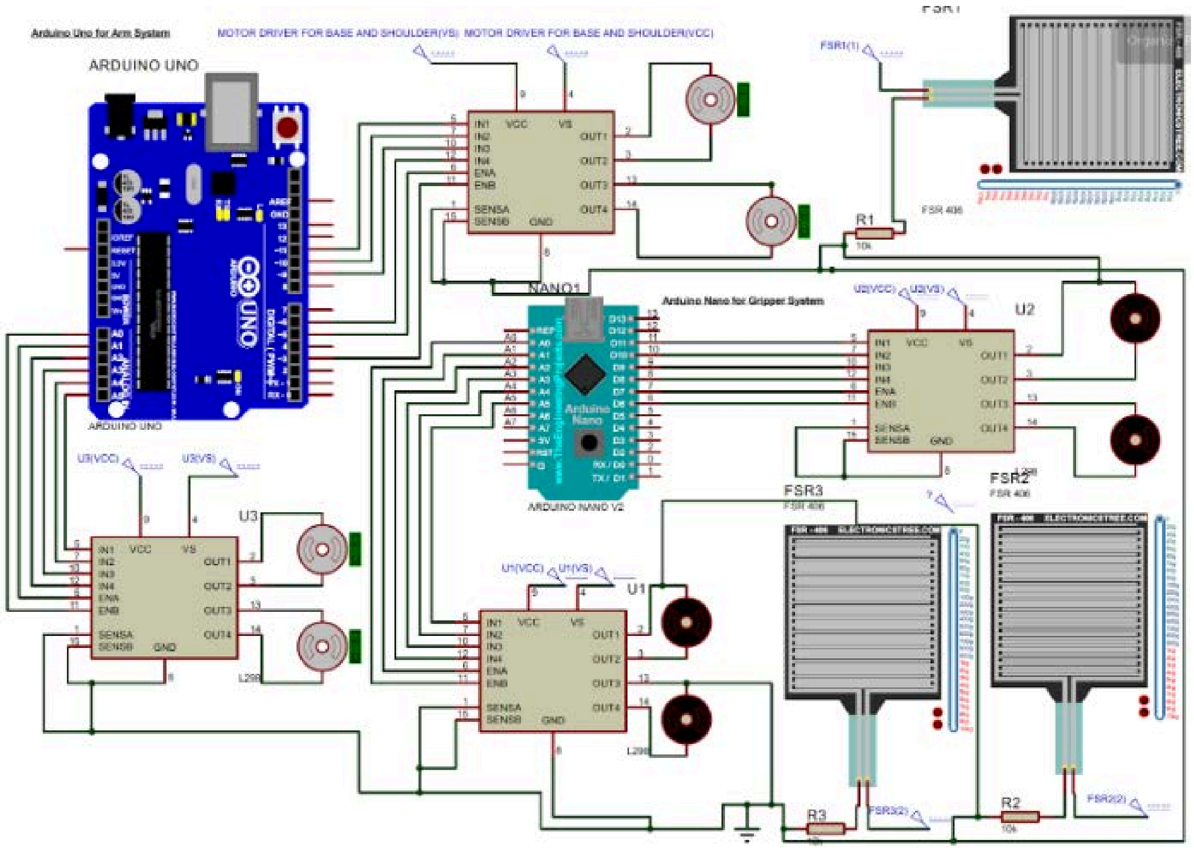

Fig - 19: Circuit diagram of the mechanical arm and the gripper

Here, we have used an Arduino UNO to control the rotational movements with four motors, as we have 4 degrees of rotational movement. And we have used an Arduino Nano to control the gripper system with motors, which are connected to the pressure sensor (FS402) for pressure calculation. The main objective is that when the gripper is closing and comes into contact with the object it calculates the pressure using the pressure sensor and using the camera it identifies the object to obtain the maximum pressure the object can endure and using that data the gripper picks up the object without squishing it. The sensor sends in the data collected from the pressure sensor to the Arduino Nano which then utilizes the data to adjust the motor rotation.

5.2.7 LLM (Gemini) usage

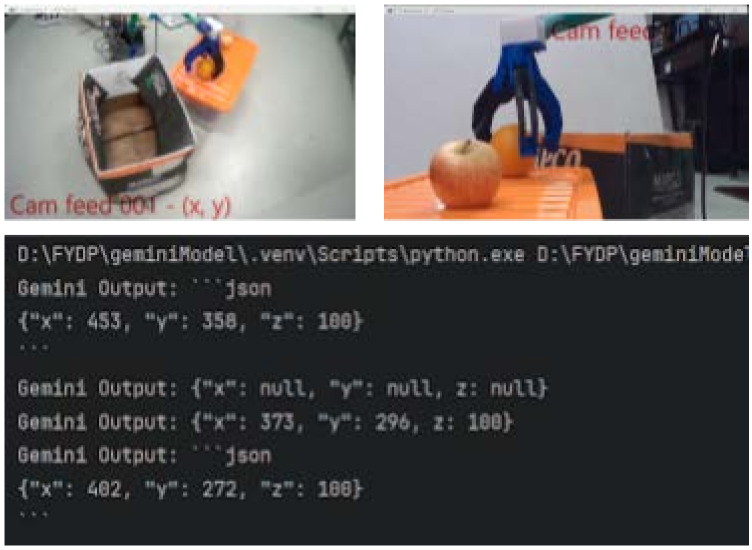

Fig - 20: The top two image shows the Input from the Camera and the bottom image shows the output (finding the orange's position - x, y, z coordinates) from the prompt by Gemini 2.0

The figure illustrates the integration of the vision system with the robotic arm using the Gemini 2.0 model. The top two images display the live camera feed, where the arm is tasked with detecting and interacting with objects—in this case, an orange. The bottom output shows the Gemini 2.0 response, providing the object's estimated position in terms of x, y, and z coordinates. These coordinates are then used by the robotic arm's kinematic model to determine precise movement and gripping actions.

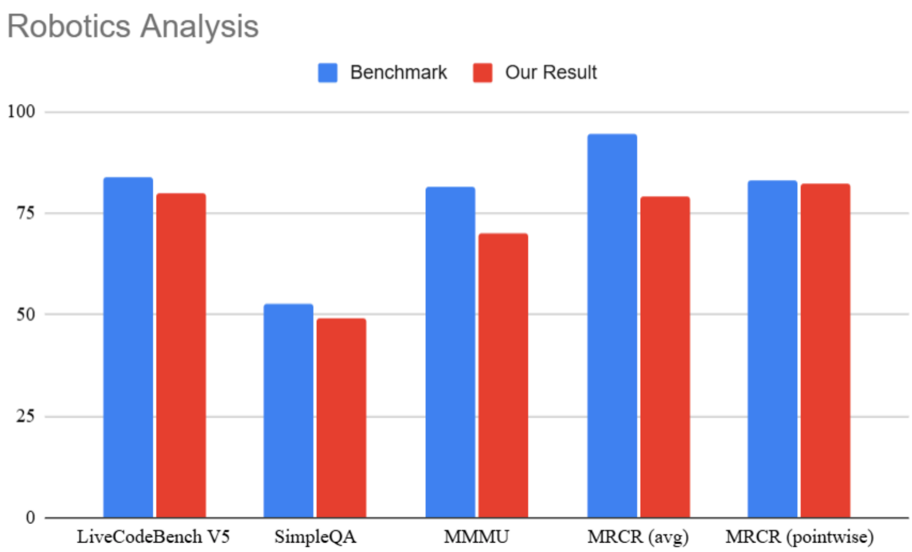

Fig - 21: Robotics Analysis comparing the benchmark performance with the results obtained from our developed system

The figure presents the *Robotics Analysis* comparing the benchmark performance with the results obtained from our developed system. The evaluation spans across multiple tasks, including LiveCodeBench V5, SimpleQA, MMMU, MRCR (average), and MRCR (pointwise). While the benchmark generally outperforms our system in categories like SimpleQA and MMMU, our results show competitive performance in LiveCodeBench V5 and MRCR pointwise, where the values are nearly identical to the benchmark. This indicates that the proposed approach maintains strong reliability in robotics-related reasoning tasks, even if further refinement is required to fully match benchmark efficiency across all categories.

5.2.8 Links and QR Code:

For further results and project demonstration (making it open source), all files and videos are uploaded to the following links:

- GitHub - https://github.com/imonM007/GenosArm
- Youtube - https://youtu.be/LItLd5HahRM

(GitHub QR) (Youtube QR)

5.3 Evaluate the Solution to Meet Desired Need

The final design, following Design Approach 2 employing a pretrained Large Language Model (LLM) such as Gemini 2.0, effectively fulfills project requirements at a very high level of efficiency. It implements an intelligent sorting mechanism effectively by combining AI and a 4-DOF robotic arm to allow effortless control over actions and movements through pre-defined function calls such as move, rotate, pick, and place. The system accomplishes live object recognition and classification in real time at high accuracy and above, leveraging high-definition camera input and multimodal capabilities in the LLM to process live streams, detect and recognize objects, and categorize without delay. This output exceeds the initial estimate, as confirmed through iterative proofing on diverse object classes and lighting arrangements. The design offers flexibility in adaptation to diverse surroundings through abolishing manual reprogramming, allowing the LLM to continue learning through observation of camera feed and estimate object loci to take auto decisions and save on setup time, facilitating ease of operational flexibility as well. The modularity in 4-DOF arm architecture, in combination with scalable software interfaces, lends itself to scalability across healthcare for sterile sorting, logistics for package sorting, and food packaging for quality assurance and deliver versatility and future-proofing capability. Moreover, the system complies with industrial safety and moral standards through employing fail-safes in control logic, ensuring it operates securely near human workers, and remaining cost-efficient through utilizing an in-place LLM instead of costly custom training datasets or hardware refreshes. Compared to the custom-trained segmentation model and pre-coded sorting jobs-based Design Approach 1, the LLM-based system offers superior adaptability, less overhead in terms of development, and broader industrial applicability, and as such, offers a more solid and future-proof solution.

5.4 Conclusion

The definitive engineering and design of the AI-based object sorting arm, following the LLM paradigm, have been carried out through performance analysis-guided refinements. This systematic and rational design process, through in-real-time decisions and module integration, meets the requirements of the project and provides an industrial automation solution that’s cost-effective and versatile to suit different needs.

Chapter 6: Impact Analysis and Project Sustainability. [CO3, CO4]

6.1 Introduction:

The Object Sorting System using an AI-Enhanced Mechanical Arm is meant to solve inefficiencies and deficits of traditional sorting manually. With robotics, artificial intelligence, and machine vision, the system is in a position to revitalize numerous industries such as healthcare, food packaging, and supply chains. This chapter discusses the broader impact of the proposal of the solution and sustainability on the basis of social, technical, environmental and economic aspects.

6.2 Assess the Impact of Solution:

The new system is mainly focused on the industries that added not much value to our social life. Hence, it reduces dependence on repetitive human labor, freeing workers from hazardous labor. In healthcare, appropriate sorting of medicines reduces hazards of patient safety. In agriculture and logistics, appropriate handling ensures quality control. However, this project enables technological understanding and creates prospects for workforce upskilling for robotics and AI.

The system optimizes productivity with less sorting error and cycle times. To businesses, it can save operating costs and increase throughput. Because, it can be reconfigured for new types of objects without long re-programming such as predefined dataset of objects. However, it can be replaced with a new and a different type of mechanical arm without doing any hardcoding like operational routing path or movement mapping. Also, it is better suited for fast and dynamic operations. In the long term, it pays for itself with a positive return on investment through saving labor costs and decreasing the product wastages.

In accordance with ISO 10218, ISO 15066, IEEE 1872 and IEC 61508 safety standards, the system ensures safe collaboration between humans and robots. Emergency stops, collision detection, and regulated payload capacity reduce workplace hazard. This is a direct assistance for safer work location while achieving precision where it is needed most such as administering drugs.

It minimizes the environmental effect through the use of electricity over fossil fuel. Correct sorting reduces undesired packaging and logistics waste creation. Efficient operations through PID control and kinematic motion also contribute to energy saving.

6.3 Evaluate the Sustainability:

Modular 4-DOF robot arm designs facilitate maintainability and easy part replacement. Scalability is incorporated at the software level, as the system is capable of switching between U-Net-based segmentation algorithms and future frameworks including Large Language Models (LLM) without hardware reconfiguration. It allows long-term technological sustainability. However, the system can work with various kinds of robot arm designs if reverse kinematics is provided no matter how much the DOF is. It can calculate the rotation of every joint by measuring the distance of the objects.

The system is conceived as an affordable product based on available components, including two sets of camera and Gemini 2.0 license. It decreases the loss due to the production line cut out and allows sustainable deployment in industry when an operational mechanical arm fails. The system's cost efficiency makes it feasible for deployment on a large scale.

6.4 Conclusion:

AI-Intelligent Object Sorting Mechanical Arm holds extraordinary potential for socially, industrially, and environmentally favorable consequences. It increases efficiency and conformity with industrially typical procedures while decreasing danger and expense. From an ecological perspective, the system is designed for long-term technical upgradeability, economic feasibility, ecological responsibility, and social fairness. Therefore, the project presents a future-proof and influential solution for existing industrial technology.

Chapter 7: Engineering Project Management. [CO11, CO14]

7.1 Introduction:

Effective project management leads to the success of an engineering design project. It provides guidelines to organize activities, manage resources, coordinate team effort, and monitor progress. The Structured project management practices were implemented to complete the project in defining activities, planning resources, assessing risks and tracking outcomes. The Chapter discusses project management practices to highlight the role of teamwork and individual contributions in achieving the project goals within the scheduled timeline.

7.2 Define, Plan, and Manage an Engineering Project:

The project was initiated by dividing it into three phases: FYDP-P (Final Year Design Project - Proposal), FYDP-D (Design), and FYDP-C (Completion). These phases clearly define objectives such as design, development, and monitoring a robotic arm capable of object sorting using artificial intelligence. To achieve these objectives, the project was broken down into different segments:

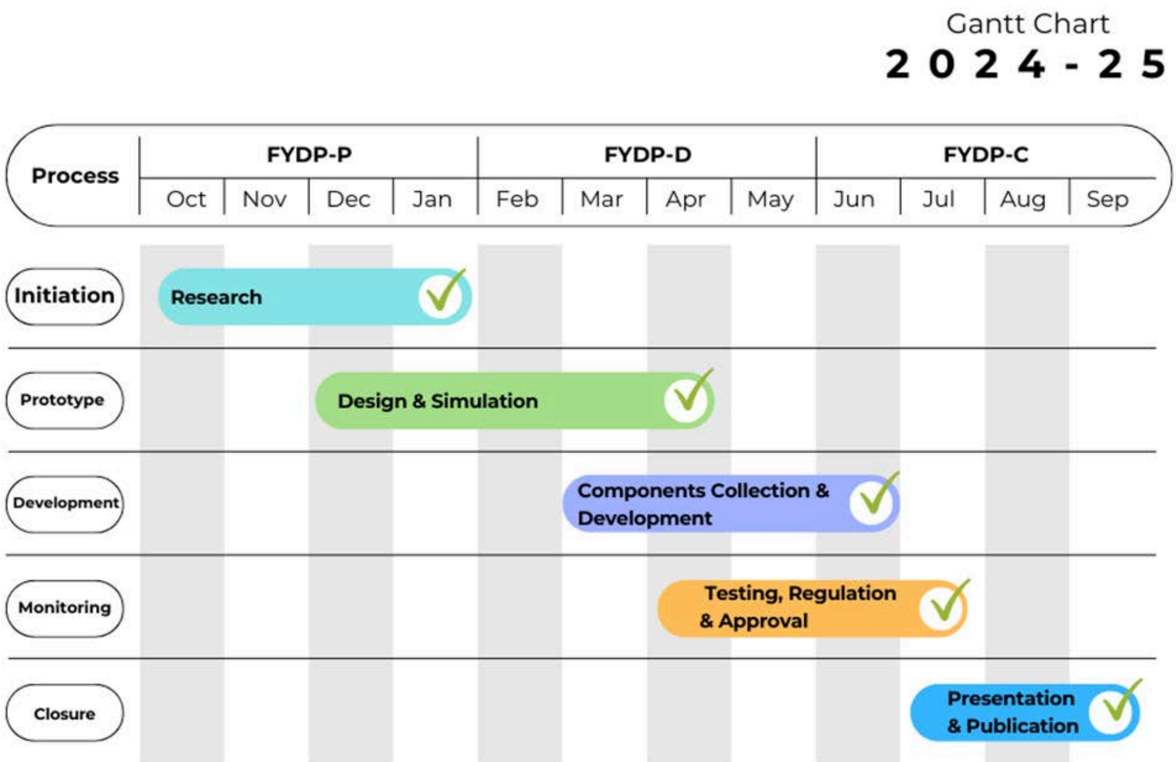

Fig - 22: Gantt Chart

These were the total work breakdown stages for the whole project. We divided the whole task among the group mates into four groups. Every individual contributes almost equally in this project. A list of the activities for every individual is given below.

Activities	Responsible
1. Finalizing the title. 2. Finalizing the problem statement. 3. Preparing a draft concept note	Task 1: All Task 2: All Task 3: All
Install the software and learn it 1. ROS2 Simulation 2. 3D Modeling	Task 1: All Task 2: Rayhan, Mesbah
1. 3D Design a. Solidworks 2. Simulation of the environment a. ROS, Gazebo	Task 1: Rayhan, Mesbah Task 2: Zoha, Imon
Specification Research (Cost Tracking, microprocessor power, computational speed, PID parameters, environment specs, Image processing, Codes)	Task: All
1. Algorithm 2. Control System 3. Circuit Simulation	Task 1: Imon Task 2: Mesbah, Zoha Task 3: Rayhan

7.3 Evaluate Project Progress:

- **Initiation:** The "Research" activity, indicated in light blue, from the period October-November 2024. In this timeline, we finalized the project title and scope, conducted background research on robotic arms, AI models, and safety standards and assigned team members for different activities.
- **Design:** The "Design & Simulation" activity, indicated in green, takes place from December 2024 through April 2025, signifying successful modeling and simulation work. We used SolidWorks for designing, MATLAB for tuning the PID parameters and Gazebo for simulation.
- **Development:** The "Collection & Components Development" phase, shown in purple, lasts from March 2025 to June 2025, implying continuous development in assembling the hardware and combining it with software.

- **Monitoring:** The "Testing, Regulation & Approval" activity, indicated in orange from May 2025 through July 2025, represents the phase of future testing and regulatory approval review from our ATC panel and from our university department.
- **Closure:** The "Presentation & Publication" task, indicated in light blue, was scheduled from August through September 2025 and our project was showcased on 11th September, 2025.

7.4 Conclusion:

The engineering project management guided this project to success. Setting specific objectives, creating an extensive plan, and delegating duties assisted in bringing out great teamworking abilities by the team members. Frequent checking of progress through established milestones assisted in keeping the project on time. The combination of personal commitment and teamworking strengthens the ability to work independently and interdependently. This formal style of management signifies that the project will be handed over within the specific time plan.

Chapter 8: Economical Analysis. [CO12]

8.1 Introduction

Economic feasibility is one of the essential key considerations for any engineering project, especially when designing systems intended for industrial applications. While advanced technologies such as AI-powered robotic arms promise efficiency and adaptability. The project's real-world adoption depends on whether the benefits outweigh the associated costs. This chapter presents the economic analysis of the AI-powered object sorting mechanical arm, examining the investment requirements, cost–benefit trade-offs, and financial implications of deploying the system.

8.2 Economic Analysis

This AI-powered robotic arm project was built by combining mechanical parts, electronic systems, and AI-driven components into a single functional design. Since it required inputs from different fields, the cost of the system was shaped by four major factors: hardware, software, operational needs, and human resources. All together these areas defined both the immediate expenses for building the prototype and the potential costs of scaling it up for larger applications. A major part of the total budget went into hardware. For the project we had to rely on stepper motors (Nema 17) and servo motors (MG996R) to control precise movements along with Arduino Uno boards, different sensors, and the main mechanical arm frame. These components formed the backbone of the system and were essential for turning the design into a working prototype. Again in the software part costs were relatively low in comparison. Most of the tools we used such as MATLAB, SolidWorks, Proteus, PyCharm, and Arduino IDE were either open source or available under academic licenses. This thing helped us to reduce direct spending on software. The operational costs also played an important role though the cost was minor during the prototype stage. The operational cost mainly included the energy required to run the motors, microcontrollers, and AI inference tasks. In a small setup these expenses were manageable, but in an industrial environment where the system would run continuously for hours, energy consumption would become a much larger factor. Finally, human resources added significant value to the project. Skilled labor was needed to design all the mechanical parts, develop the AI algorithms, and integrate all electronic components into a single working system. Expanding the project to an industrial level would require a strong team of engineers and researchers. Their expertise and time represent one of the most important investments in ensuring the success and sustainability of the system. In summary, the project's cost structure was shaped by a careful balance of hardware expenses, affordable software solutions, energy requirements, and the contribution of skilled professionals. While the prototype stage kept many costs low, the system scale for industrial use would demand more investment especially in terms of high performance computing and human resources.

8.3 Cost–Benefit Analysis

This report provides a cost-benefit analysis (CBA) for the two design approaches. By this analysis we can easily choose which design approach will give us the best outcome at a minimum cost. So, our design approach 1 was based on machine learning image segmentation using a U-Net architecture model which will be trained on a custom dataset and it will be integrated with OpenCV for detecting the objects and their corresponding coordinate. The design approach 2 was based on the Large Language Model (LLM) (e.g., Gemini 2.0) for real time object detection, coordinate selection and real time decision making and control of the robotic arm without any custom training.

The cost benefit analysis of design approach 1 with an approximate cost of 45,250 Bdt.

Components	Price (Bdt)	Benefits	Drawbacks
Stepper & Servo motors	1700	Provide precise control for arm movements with high torque	Limited lifespan under heavy loads
Arduino Uno	1000	Acts as the main processing unit for kinematics and PID control	Limited memory and speed for advanced computations
Arduino Nano	400	Handles supplementary tasks like sensor data processing	Fewer I/O pins compared to Uno
Cameras	1800	Enable wide-angle (24 FPS min) real-time object detection and classification	Image quality degrades in low light
Pressure Sensor	650	Ensures safe gripping by detecting force and preventing object damage	Sensitive to environmental factors
Power Supply (12-24V 10A)	700	Supplies stable power to all components	Potential for voltage fluctuations if overloaded

Body 3D design	3000	Lightweight yet durable structure for the arm, also supports plug-and-play configuration	Custom fabrication can vary in quality
Motor Drivers	1000	Allows secure object handling for sorting medicines, fruits, etc	Mechanical failure if overloaded.
GPU	35000	Enable efficient training of ML model for 95.9% accuracy in object segmentation	High initial cost

The cost benefit analysis of design approach 1 with an approximate cost of 15, 150 Bdt.

Components	Price (Bdt)	Benefits	Drawbacks
Stepper & Servo motors	1700	Provide precise control for arm movements with high torque	Limited lifespan under heavy loads
Arduino Uno	1000*2	Acts as the main processing unit for kinematics and PID control	Limited memory and speed for advanced computations
Cameras	1800	Enable wide-angle (24 FPS min) real-time object detection and classification	Image quality degrades in low light
Pressure Sensor	650	Ensures safe gripping by detecting force and	Sensitive to environmental factors

		preventing object damage	
Power Supply (12-24V 10A) (for design approach 2)	2000	Supplies stable power to all components	Potential for voltage fluctuations if overloaded
Body 3D design (filament and others)	3000	Lightweight yet durable structure for the arm, also supports plug-and-play configuration	Custom fabrication can vary in quality
Motor driver	1000*3	Allows secure object handling for sorting medicines, fruits, etc	Mechanical failure if overloaded
Gemini subscription (per month)	1000	Facilitates prompt-based object detection and adaptation without training, achieving 93.7% accuracy	Cost vary beyond initial period

From the comparison of the cost benefit analysis of two different approaches, we can choose design approach 2 as the best choice for the object sorting system using an AI powered mechanical arm due to its superior efficiency and adaptability. By using a pre-trained Large Language Model (LLM) like Gemini, it eliminates the need for custom dataset creation and extensive training which was required in approach 1. Design approach 1 relies on a U-Net model and OpenCV which reduces both initial costs and development time. On the other hand, design approach 2 is a prompt-based system that allows seamless adaptation to new object categories or tasks without hardware modifications or retraining and also it can align perfectly with the project's goal to eliminate manual reprogramming. With a high accuracy of 93.7% and real-time processing within the 500 ms constraint, it matches the design approach 1's 95.9% accuracy while offering greater flexibility for diverse industrial applications like healthcare and logistics. Also, to build a prototype of design approach 1 we need an approximate budget of 45,250 bdt and for the design approach 2 it needs 15,150 bdt. Which shows that the design approach 2 is more budget friendly with a huge impact and with its lower resource demands and scalability.

8.4 Evaluation of Economic and Financial Aspects

The evaluation of the economic and financial aspects of this system highlights how investment, efficiency, sustainability, and risk management interact to create long term value. At the prototype stage, costs were kept relatively low due to the use of academic software licenses and 3D-printed components. A major contributor to its economic feasibility is the strong return on investment (ROI). By reducing human errors, shortening production cycles, and eliminating repetitive reprogramming, the system directly improves efficiency and productivity. In sectors like- pharmaceuticals and food processing where precision and reliability are crucial, this system not only lowers costs but also increases efficiency and makes strong customer trust through consistent quality assurance. Again, sustainability further enhances the financial outlook. The modular design allows parts to be replaced or reused instead of replacing the entire system which reduces long-term maintenance costs. At the same time, the integration of energy-efficient AI models helps to keep power consumption and operational costs manageable, which is important in industries with continuous production cycles. These features make the system both cost-effective and environmentally responsible.

While working on this project, from scratch to making a complete prototype the total cost was 26, 850 bdt. Which includes all the mechanical, electrical and other costs to build this AI powered mechanical arm system to sort all types of objects.

Type	Name	Price Per Unit	Quantity	Price
Microcontroller	Arduino Uno	1,000.00	2.00	2,000.00
Stepper Motor	Nema 17	1,300.00	3.00	3,900.00
Motor Driver	TB6600	1,000.00	3.00	3,000.00
Filament	PLA Plus (2 KG)	3,000.00	1.00	3,000.00
Power Adapter	12-24V Variable PSU	2,000.00	1.00	2,000.00
Force Sensor	FSR 5.5	650.00	3.00	1,950.00
PCB	JLC PCB	7,000.00	1.00	7,000.00
Other	Screw, Nuts and others	2,000.00	1.00	2,000.00
Gemini Subscription	Gemini 2.5 (2 month)	1,000.00	2.00	2,000.00
			TOTAL	26,850.00

Fig: Total cost to finish the prototype of our project

8.5 Conclusion

The economic analysis demonstrates that the AI-powered object sorting mechanical arm is not only technically workable but also economically practical. While hardware and skilled labor represent the largest share of initial costs, the long term benefits of reduced labor, scalability, and higher efficiency ensure strong financial justification. The cost benefit balance highlights that investing in such automation systems can generate sustainable economic value, particularly in industries that demand precision and adaptability. Thus the project provides a viable model for crossing over innovation with affordability, also aligning advanced engineering solutions with the real world economic needs.

Chapter 9: Ethics and Professional Responsibilities CO13, CO2

9.1 Introduction

Throughout our project efforts, we exercised ethical considerations and maintained professional duties in all design stages. User privacy was prioritized through secure data processing and adherence to data privacy regulations, and transparency in system capacity and risk communication were also maintained. Safety was prioritized through rigorous testing and fail-safe design implementation and inclusivity through accessible solution designs for all types of abilities.

Professional codes of conduct were followed by our team by practicing fairness, conflict resolution responsibly and through environmentally-friendly design decisions through the use of energy-efficient design options. Additionally, we ensured our solution benefits the broader good through the assessment of possible misuse and the application of preventive procedures so as not to cause unintended injury or harm to individuals or communities.

9.2 Identify Ethical Issues and Professional Responsibility

In the whole planning phase of this project there were different professional duties and ethical implications that came to our sight:

- **Ensuring Safety and Reliability:** The robotic arm must be designed to keep people safe and also to protect the products. A top priority is to prevent accidents, any kinds of damage to goods, or workplace hazards. This means through testing and careful design of the whole project can ensure the arm to operate smoothly and reliably every time.
- **Protection of Data Privacy:** The system uses cameras to detect objects and to find out the coordinates of the object which raises important questions about how all those images are managed. To respect privacy, the team must ensure no sensitive or personal information is misused or stored unnecessarily. Clear safeguards and responsible data handling practices are essential to build trust.
- **Transparency and Truthfulness:** It is tempting to hype up the robotic arm's capabilities. But in reality we know honesty is key. So the team has a duty to communicate and represent the system's strengths and limitations clearly. By setting realistic expectations we can avoid misleading users and also can ensure that the user can trust the technology.
- **Responsibility toward Environment:** The robotic arm relies on electronic parts, motors, and sensors which can turn to e-waste whenever the lifespan of these products

is finished or any kinds of damages happen. To address this, the team focused on designing with modularity and recyclability in mind to reduce the environmental footprint and promote sustainability.

- **Effect on Workforce:** Automation can really change the way people work, and also sometimes it raises concerns about job security. The project team also recognizes the need to consider how the robotic arm might affect employees. By planning for upskilling or reskilling opportunities and training them we can help all the workers to adapt to the new roles and thrive alongside the technology.

9.3 Apply Ethical Issues and Professional Responsibility

From the very beginning, this project has applied all these ethical and professional responsibilities:

- **Meeting Standard Compliance:** To ensure the proper safety and reliability of the robotic arm, the design team aligned its design with international safety standards, specifically ISO 10218-1 for industrial robots. Along with ISO 10218-1 there are so many international safety rules and regulations for designing the robotics arm for example- IEC 61508. By following all these globally recognized guidelines the arm was built and then it was ready for prototype and industrial uses with proper safety.
- **Safe Design Features:** The safety of the arm was a top priority. That is why the robotic arm included key features to prevent any accidents. The system included an emergency stop button to prevent collision. There will be limitations on the weight it could carry to reduce the risk and ensure safer operation in real-world settings.
- **Privacy and Ethical use of Data:** Since the robotic arm uses cameras for object detection that's why the system needs a safeguard privacy system to ensure the ethical use of data. The team took necessary steps to safeguard privacy from the start. In the training process, the dataset was made up entirely of artificial objects such as balls, bottles, and small boxes to avoid sensitive or personal images. This approach eliminated privacy concerns and ensured responsible data handling for the system.
- **Practices of Sustainability:** Keeping environmental concerns in mind, the robotic arm was designed with proper modularity. That means the individual parts of the entire mechanical arm can be easily replaced, reused, or recycled if any particular part breaks or damage occurs. By reducing wastes and planning for the arm's long-term lifecycle, the designing team has shown a commitment to environmental responsibility.

9.4 Conclusion

In the whole project, ethics and professional responsibility weren't just tacked on at the end, they were woven right into everything we did. From the very beginning to make any decisions, we kept asking ourselves: How could this robotic arm touch people's lives, shape our communities, and also impact the planet? This simple question guided us to build the project in real safeguards for safety, sustainability, fairness, and also inspired us to make sure that the design truly puts people first. By sticking to proven standards, protecting data like it was our own, and thinking through the ripple effects on society we showed that great engineering isn't only about crafting clever machines, it's about earning trust along the way. Staying open and honest about what we could and couldn't do built real credibility, and by factoring in the environment and social side of things, we committed to leaving something better for the folks who come after us. Ultimately, this work stands as a quiet proof that innovation and responsibility don't have to clash, they can fuel each other. When you blend cutting-edge AI with that kind of thoughtful care, something like our object-sorting robotic arm turns into more than a technical win; it becomes a real nod to how engineering can serve up safety, progress, and a sustainable future for everyone.

Chapter 10: Conclusion and Future Work.

10.1 Project Summary / Conclusion

This project successfully explored the design and development of an AI-enhanced automated mechanical arm for object sorting. The system integrates robotics with modern artificial intelligence, particularly computer vision and large language models, to overcome the limitations of conventional robotic arms that require frequent manual reprogramming.

Two design approaches were considered:

- Design Approach 1 (U-Net-based vision system): Provided high-accuracy segmentation (95.98%) and precise coordinate extraction, though it required dataset training and consistent environmental conditions. [18]

- Design Approach 2 (LLM-based reasoning with Gemini 2.0): Enabled real-time adaptability, multimodal reasoning, and execution of commands without retraining, making it more flexible and user-friendly.

After analysis, the LLM-based system was identified as the optimal solution due to its adaptability, reduced human intervention, and ability to handle dynamic environments. The 4-DOF robotic arm design, supported by PID tuning and inverse kinematics, ensures stable and accurate movement. Safety, ethical, and risk considerations were addressed through adherence to international standards (ISO 10218, IEC 61508) and robust contingency planning.

Overall, the project demonstrates how fusing mechanical design, AI models, and control engineering can create a scalable, efficient, and safe automation system with applications in healthcare, logistics, and food packaging.

10.2 Future Work

While the project outcomes are promising, several areas can be expanded in future research and development:

1. Hardware Optimization

- Extend payload capacity beyond 1 kg to accommodate heavier objects.
- Implement stronger, lightweight materials (e.g., carbon fiber) to improve durability.
- Explore more advanced servo motors with higher torque efficiency.

2. Advanced AI Integration

- Incorporate reinforcement learning to allow the robotic arm to self-improve from trial and error.
- Expand object categories through continual learning without retraining from scratch.
- Enhance multimodal LLM reasoning to improve precision in cluttered or low-light environments.

3. System Scalability

- Integrate conveyor belt systems for industrial-scale sorting.
- Develop modular arms that can be linked for collaborative tasks.

4. Human-Robot Collaboration

- Improve real-time feedback interfaces for operators.
- Enhance safety systems with predictive collision avoidance.

5. Deployment and Field Testing

- Test in real industrial settings (e.g., hospitals, warehouses, packaging plants).
- Evaluate performance under diverse lighting, object textures, and environmental conditions.

By addressing these areas, the project can evolve from a prototype to a fully deployable industrial solution, paving the way for highly autonomous, intelligent robotic systems in next-generation automation.

Chapter 11: Identification of Complex Engineering Problems and Activities.

11.1 Identification of Attributes of Complex Engineering Problems (EP)

Attributes of Complex Engineering Problems (EP)

	Attributes	**Put tick (√) as appropriate**
P1	Depth of knowledge required	√
P2	Range of conflicting requirements	
P3	Depth of analysis required	√
P4	Familiarity of issues	
P5	Extent of applicable codes	
P6	Extent of stakeholder involvement and needs	√
P7	Interdependence	√

11.2 Reasoning: Why the Project Covers Selected EP Attributes

The project addressed its multi-faceted features in a clear and practical way. To tackle the real-time sorting challenge, the project members used AI-powered vision through Gemini 2.0 Flash Lite, with a combination of carefully selected hardware. This work demands different outer knowledge from machine learning, robotics, and embedded systems to softwares which were needed to make the system work effectively. Again, to strengthen the design, advanced simulations were carried out using PID control in MATLAB Simulink and SolidWorks to design the whole mechanical arm body structure. These tools were not just used for basic calculations but also for the detailed structural verification and in-depth analysis which also required careful planning and strong analytical thinking. Another important aspect of the project was the involvement of academic supervisors and the use of industrial standards as references. Regular feedback ensured that the system not only met academic requirements but also aligned with practical applications which gave the project both theoretical strength and real-world relevance. Finally, the project's true interdisciplinary nature stood out. It combined mechanical design, electronic integration, and AI programming into a single working system. This blending of different engineering fields reflects the kind of complex problem-solving required in modern technology development and highlights the depth of learning achieved by the team.

11.3 Determining Attributes of Complex Engineering Activities (EA)

Attributes of Complex Engineering Activities (EA)

	Attributes	**Put tick (√) as appropriate**
A1	Range of resource	√
A2	Level of interaction	√
A3	Innovation	
A4	Consequences for society and the environment	
A5	Familiarity	

11.4 Reasoning: How the Project Aligns to Chosen EA Attributes

The identified attributes of EA were addressed in several ways throughout the project. One of the key aspects was the integration of diverse resources. The system brought together mechanical components like motors, links, and grippers with electronic circuits and AI-based software. Synchronizing all of these different resources into a single and smoothly functioning system was not simple. It required careful planning to make sure each element worked in a compatible and efficient way. This integration highlighted the challenges of combining tools and techniques from different branches of engineering into one unified solution. Another important attribute was the use of advanced simulation and coding tools. Software such as SolidWorks, MATLAB Simulink, Proteus, PyCharm, and Arduino IDE each served specific purposes, ranging from 3D modeling and structural verification to AI training and hardware programming. The use of these tools allowed the project to move more easily from design to implementation, making the transition from concepts and simulations to a working prototype both smoother and faster. Finally, the project involved iterative hardware and software interaction. The system required continuous testing and adjustment, where simulation results had to align with the physical responses of the hardware. For example, outputs from MATLAB simulations needed to be accurately reflected in hardware operations which were controlled through Arduino programming. This process of testing, receiving feedback, and making interactive adjustments played a critical role in achieving both stability and accuracy in the final system.

References

[1] A. Châtel et al., "The necessity of investigating a freshwater-marine continuum using a mesocosm approach in nanosafety: The case study of TiO2 MNM-based photocatalytic cement," NanoImpact, vol. 20, p. 100254, 2020, doi: 10.1016/j.impact.2020.100254

[2] International Federation of Robotics, "World Robotics 2025 - Industrial Robots," 2025. [Online]. Available: https://ifr.org/wr-industrial-robots/. [Accessed: Sep. 22, 2025].

[3] M. Khalid, M. M. A. Zahidi, M. H. M. Ramli, and M. H. Ismail, "Object Sorting System Using Robotic Arm," *J. Informat. Web Eng.*, vol. 3, no. 2, pp. 221–230, 2024. [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/110625455/object-sorting-system-usingrobotic-arm-libre.pdf.

[4] V. C. Vo, T. T. Nguyen, and T. H. Nguyen, "Design and Development of Robot Arm System for Classification and Sorting Using Machine Vision," *J. Phys.: Conf. Ser.*, vol. 2225, no. 1, p. 012013, 2021, doi: 10.1088/1742-6596/2225/1/012013.

[5] International Organization for Standardization, "Robotics — Performance criteria and related test methods for service robots — Manipulation," ISO 22174:2020, 2020. [Online]. Available: https://www.iso.org/standard/73933.html. [Accessed: Sep. 22, 2025].

[6] International Organization for Standardization, "Robotics — Performance criteria and related test methods for service robots — Navigation," ISO 18646:2021, 2021. [Online]. Available: https://www.iso.org/standard/62996.html. [Accessed: Sep. 22, 2025].

[7] M. Khalid, M. M. A. Zahidi, M. H. M. Ramli, and M. H. Ismail, "Sorting System Using Robotic Arm with Integrated Machine Vision," *J. Ind. Eng.*, vol. 13, no. 4, pp. 394–402, 2025. [Online]. Available: https://www.joig.net/2025/JOIG-V13N4-394.pdf.

[8] J. Craig, Introduction to Robotics: Mechanics and Control, 4th ed. Pearson, 2017.

[9] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *arXiv preprint arXiv:1505.04597*, 2015. [Online]. Available: https://arxiv.org/abs/1505.04597

[10] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000. [Online]. Available: https://opencv.org

[11] Google DeepMind, "Gemini 2.0: Multimodal and Robotic Reasoning Capabilities," *arXiv*, Mar. 2025. [Online]. Available: https://arxiv.org/html/2503.20020v1

[12] ISO, "Robots and robotic devices – Safety requirements for industrial robots – Part 1: Robots," ISO 10218-1, 2011. [Online]. Available: https://www.iso.org/standard/51330.html

[13] L. Seidel, S. Gehringer, T. Raczok, S.-N. Ivens, B. Eckardt, and M. Maerz, "Advancing Early Wildfire Detection: Integration of Vision Language Models with Unmanned Aerial Vehicle Remote Sensing for Enhanced Situational Awareness," *Drones*, vol. 9, no. 5, p. 347, May 2025, doi: 10.3390/drones9050347.

[14] Gemini Robotics Team, S. Abeyruwan, J. Ainslie, J.-B. Alayrac, M. G. Arenas, T. Armstrong, A. Balakrishna, R. Baruch, M. Bauza, M. Blokzijl, et al., "Gemini Robotics: Bringing AI into the Physical World," *arXiv preprint arXiv:2503.20020*, Mar. 2025. [Online]. Available: https://arxiv.org/abs/2503.20020

[15] X. An, J. Sun, Z. Gui, and W. He, "CHOICE: Benchmarking the Remote Sensing Capabilities of Large Vision-Language Models," *arXiv preprint arXiv:2411.18145v2*, Apr. 2025. [Online]. Available: https://arxiv.org/abs/2411.18145

[16] Gemini Robotics Team, S. Abeyruwan, J. Ainslie, J.-B. Alayrac, M. G. Arenas, T. Armstrong, A. Balakrishna, R. Baruch, M. Bauza, M. Blokzijl, et al., "Gemini Robotics: Bringing AI into the Physical World," *arXiv preprint arXiv:2503.20020*, Mar. 2025. [Online]. Available: https://arxiv.org/abs/2503.20020

[17] C. Carlos, "Google DeepMind's Gemini Robotics: Unleashing Embodied AI with Zero-Shot Control and Enhanced Spatial Reasoning," *UBOS*, Mar. 14, 2025. [Online]. Available: https://ubos.tech/news/google-deepminds-gemini-robotics-unleashing-embodied-ai-with-zero-shot-control-and-enhanced-spatial-reasoning/

[18] Youran Huang, "Research on garbage sorting robotic arm based on image vision," *Journal of Physics: Conference Series*, vol. 2741, no. 1, p. 012020, Apr. 2024, doi:10.1088/1742-6596/2741/1/012020.

Appendix

Logbook:

Date/Time /Place	Attendee	Summary of Meeting Minutes	Responsible	Comment by ATC
01.12.2024	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	1. Need to finalize title. 2. Need to finalize the problem statement. 3. Need to prepare a draft concept note	Task 1: All Task 2: All Task 3: All	N/A as it was an introductory meeting.
07.01.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Mock Presentation (FYDP-P)		
27.02.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Install the software and learn it 1. ROS2 2. 3D Modeling	Task 1: All Task 2: Rayhan, Mesbah	
20.04.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	1. 3D Design - Fusion 2. Simulation of environment - ROS, Gazebo	Task 1: Rayhan, Mesbah Task 2: Zoha	
22.04.2025	1. Imon 2. Zoha 3. Mesbah	Briefing about the progress, discussing about softwares and their issues with Nahid Hossain Taz sir		
22.04.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Specification Research (Cost Tracking, microprocessor power, computational speed, PID parameters, environment specs, Image processing, Codes) discussion with Sharif Mohd Shams sir	Task: All	
30.04.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	1. Algorithm 2. Control System 3. Circuit Simulation 4. Slides	Task 1: Imon Task 2: Mesbah, Zoha Task 3: Rayhan Task 4: All	
05.05.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Progress Meeting with Sharif Mohd Shams sir		
07.05.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Mock Presentation Nahid Hossain Taz sir		

11.05.2025	1. Imon 2. Zoha	Progress update on report and presentation feedback analysis.		
01.07.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Discusses about the workflow and how we need to proceed for the FYDP-C with Nahid Hossain Taz sir		
08.07.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Task 1: Show the design model and some 3D printed component Task 2: Take idea about pcb design	Task 1: Mesbah, Rayhan Task 2: Imon, Zoha	
15.07.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Give a progress overview of the work through slides Show the 3D view of gripper and its working video Done with pcb design		
17.08.2025	1. Imon 2. Zoha 3. Mesbah	Progress update		
26.08.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Poster design idea, Mock Presentation with Nahid Hossain Taz sir		
09.09.2025	1. Imon 2. Zoha 3. Mesbah 4. Rayhan	Poster Presentation with Sharif Mohd Shams sir		

Assessment Guideline for Faculty

[The following assessment guideline is for faculty ONLY. **This portion is not applicable for students**.]

Assessment Tools and CO Assessment Guideline

	Distribution of assessment points among various COs assessed in different semesters														
PO	l	c	f	g	c	b	d	c	e	l	k	k	h	i	j
CO	CO1	CO2	CO3	CO4	CO5	CO6	CO7	CO8	CO9	CO10	CO11	CO12	CO13	CO14	CO15
EEE 400C/ ECE 402C (Out of 100)							30	24	6	4	4	6	7	7	12
Project Final Report/ Project Progress Report							x	x	x	x	x	x	x		x
Demonstration of working prototype							x								x
Progress Presentation/ Final Presentation								x			x				
Peer-evaluation*													x	x	
Instructor's Assessment*													x	x	
Demonstration at FYDP Showcase								x							x

Note: The star (*) marked deliverables/skills will be evaluated at various stages of the project.

Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool

Sl.	CO Description	PO	Bloom's Taxonomy Domain/Level	Assessment Tools
CO7	**Evaluate** the performance of the developed solution with respect to the given specifications, requirements and standards	d	Cognitive/ Evaluate	● Demonstration of working prototype ● Project Progress Report on working prototype
CO8	**Complete** the final design and development of the solution with necessary adjustment based on performance evaluation	c	Cognitive/ Create	● Project Final Report ● Final Presentation ● Demonstration at FYDP Showcase
CO9	**Use** modern engineering and IT tools to design, develop and validate the solution	e	Cognitive/ Understand, Psychomotor/ Precision	● Project Final Report
CO10	**Conduct** independent research, literature survey and learning of new technologies and concepts as appropriate to design, develop and validate the solution	l	Cognitive/ Apply	● Project Final Report
CO11**	**Demonstrate** project management skill in various stages of developing the solution of engineering design project	k	Cognitive/ Apply Affective/ Valuing	● Project Final Report ● Project Progress presentation at various stages
CO12	**Perform** cost-benefit and economic analysis of the solution	k	Cognitive/ Apply	● Project Final Report
CO13	**Apply** ethical considerations and professional responsibilities in designing the solution and	h	Cognitive/ Apply Affective/ Valuing	● Peer-evaluation,

	throughout the project development phases			● Instructor's Assessment ● Final Report
CO14**	**Perform** effectively as an individual and as a team member for successfully completion of the project	i	Affective/ Characterization	● Peer-evaluation ● Instructor's Assessment
CO15**	**Communicate** effectively through writings, journals, technical reports, deliverables, presentations and verbal communication as appropriate at various stages of project development	j	Cognitive/ Understand Psychomotor/ Precision Affective/ Valuing	● Project Final Report ● Progress Presentations, ● Final Presentation ● Demonstration at FYDP Showcase

Note: The double star (**) marked CO will be assessed at various stages of the project through indirect deliverables.