SMART WHITEBOARD MANAGEMENT SYSTEM

By

Ekram Ahmed Raiyan
21221047
Mercy Al Hossain
21221058
Mushfiq Us Salehin
21321062
Md. Manjur
24221140

A Final Year Design Project (FYDP) submitted to the Department of Electrical and Electronic Engineering in partial fulfillment of the requirements for the degree of Bachelor of Science (B.Sc) in Electrical & Electronic Engineering

Electrical and Electronics Engineering
BRAC University
October, 2025

SMART WHITEBOARD MANAGEMENT SYSTEM

By

Ekram Ahmed Raiyan
21221047
Mercy Al Hossain
21221058
Mushfiq Us Salehin
21321062
Md. Manjur
24221140

A Final Year Design Project (FYDP) submitted to the Department of Electrical and Electronic Engineering in partial fulfillment of the requirements for the degree of Bachelor of Science (B.Sc) in Electrical & Electronic Engineering

Electrical and Electronics Engineering
Brac University
October, 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:

Ekram Ahmed Raiyan 21221047	Mercy Al Hossain 21221058
Mushfiq Us Salehin 21321062	Md. Manjur 24221140

Approval

The Final Year Design Project (FYDP) titled "Smart Whiteboard Management System" submitted by

1. Ekram Ahmed Raiyan (21221047)
2. Mercy Al Hossain (21221058)
3. Mushfiq Us Salehin (21321062)
4. Md. Manjur (24221140)

of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of BSEEE on 11th September, 2025 .

Examining Committee:

Academic Technical Committee (ATC): (Chair)

____
Dr. Mirza Rasheduzzaman
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
Chairperson
Department of Electrical and Electronic Engineering

Ethics Statement

Our project had a similarity report of 12% most of which were material such as the table of contents page, specifications, chapter headings. AI has also been detected by turnitin less than 20% [*%].

Abstract/ Executive Summary

Our Smart Whiteboard Management System transforms the traditional classroom into a modern, interactive space. It's a portable, electric-powered system that cleans the board automatically using 99.5% of the surface area. A 5MP camera captures everything written--notes, equations, and diagrams--and converts them into editable LaTeX PDFs and shares them through a Flask-based web portal. The system also includes an AI chatbot tutor that provides instant help and reduces the need for human assistants by up to 50%. Using reed sensors and ESP32 microcontrollers, it selectively cleans only the required areas, making the process efficient and sustainable. With excellent digitization accuracy, it reduces paper waste and supports eco-friendly learning.

Keywords: Whiteboard Automation, Content Digitization, Image Processing, Rule-Based Algorithms, ESP32, Real-Time Digitization.

Acknowledgement

We would like to acknowledge our ATC panel members for their constant support throughout this project. We would also like to express our gratitude to Dr. Md. Mosaddequr Rahman, Dr. Shameem Ahmad, Mr. Mohaimenul Islam and Mr. Atib Mohammad Oni for taking the time to participate in the survey regarding the qualitative assessment of our system.

Table of Contents

Chapter 1: Introduction- [CO1, CO2, CO10]... 11

1.1 Introduction... 11

1.1.1 Problem Statement..11

1.1.2 Background Study... 13

1.1.3 Literature Gap.. 15

1.1.4 Relevance to current and future Industry... 16

1.2.1. Objectives...18

1.2.2 Functional and Nonfunctional Requirements..18

1.2.2 Specifications... 20

1.2.3 Technical and Non-technical consideration and constraint in design process.......... 22

1.2.4 Applicable compliance, standards, and codes..23

1.3 Systematic Overview/summary of the proposed project...........................25

1.3.1 General system of block diagram...26

Chapter 2: Project Design Approach [CO5, CO6]...28

2.1 Introduction... 28

2.2 Identify Multiple Design Approaches.. 29

2.2.1 Design Approach 1..29

2.2.2 Design Approach 2..31

2.3 Describe Multiple Design Approaches... 33

2.3.1 Design Approach 1..33

2.3.2 Design Approach 2..34

2.4 Analysis of Multiple Design Approaches... 36

2.5 Conclusion..37

Chapter 3: Use of Modern Engineering and IT Tool. [CO9].. 39

3.1 Introduction... 39

3.2 Contemporary engineering and IT Tools..39

3.2.1 Simulation Tools in Action...39

3.3 Choice of the right engineering and IT tools...40

3.3.1 Hardware Tools...40

3.3.2 Software Tools..41

3.3.3 Integration of Tools.. 42

3.4 Conclusion..43

Chapter 4: Optimization of Multiple Design and Finding the Optimal Solution. [CO7] 44

4.1 Introduction... 44

4.2 Optimization of multiple design approach.. 44

4.2.1 3D design:..44

4.2.2 Simulation (Using Proteus)........ 47
4.2.3 Simulation(using Simulink):........ 59
4.3 Identify optimal design approach........ 66
4.4 Performance evaluation of developed solution........ 69
4.5 Conclusion........ 69

Chapter 5: Completion of Final Design and Validation. [CO8]........ 70
5.1 Introduction........ 70
5.2 Completion of final design........ 70
5.2.1 Design working [CO8, CO9]........ 70
5.2.2 Camera Integration [CO8, CO13]........ 73
5.2.3 Mechanical Structure Development [CO8]........ 74
5.2.4 Firmware and GUI Development [CO9, CO15]........ 77
5.3 Evaluation of Final Design Against Objectives [CO8, CO7]........ 81
5.3.1 Cleaning Accuracy........ 81
5.3.2 Content Digitization Accuracy........ 83
5.3.3 Power Efficiency [CO8, CO12]........ 84
5.3.4 Validation via Exhibits [CO8]........ 88
5.3.5 Limitations........ 88
5.4 Conclusion........ 89

Chapter 6: Impact Analysis and Project Sustainability. [CO3, CO4]........ 90
6.1 Introduction........ 90
6.2 Evaluate the effect of the solution........ 90
6.3 Determine sustainability........ 91
6.4 Conclusion........ 91

Chapter 7: Engineering Project Management. [CO11, CO14]........ 92
7.1 Introduction........ 92
7.2 Define, plan and manage engineering project........ 92
7.3 Evaluate project progress........ 101
7.4 Conclusion........ 103

Chapter 8: Economical Analysis. [CO12]........ 104
8.1 Introduction........ 104
8.2 Economic analysis........ 104
8.3 Conclusion........ 108

Chapter 9: Ethics and Professional Responsibilities CO13, CO2........ 109
9.1 Introduction........ 109
9.2 Identify ethical issues and professional responsibility........ 109
9.3 Apply ethical issues and professional responsibility........ 110

9.4 Ethical use of Automation and AI.......... 111
9.5 Data Privacy and Transparency.......... 111
9.6 Professional Accountability.......... 111
9.7 Conclusion.......... 112

Chapter 10: Conclusion and Future Work..........113
10.1 Project summary/Conclusion..........113
10.2 Future work.......... 113

Chapter 11: Identification of Complex Engineering Problems and Activities..........115
11.1: Identify the attribute of complex engineering problem (EP)..........115
11.2: Provide reasoning how the project address selected attribute (EP)..........115
11.3 Identify the attribute of complex engineering activities (EA)..........116
11.4 Provide reasoning how the project address selected attribute (EA).......... 117

References..........118
Appendix..........121
Logbook..........151

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

1.1 Introduction

Whiteboards have been in use in classrooms for decades because they are a quick, cheap, and convenient means for teachers to communicate. Simple-looking in design, traditional whiteboards are a significant obstacle in the educational landscape of today. Several wipe-outs with the hand are required between each lecture, interfering with the natural flow of teaching[1]. Once a board is wiped clean, valuable written content—such as equations, diagrams, and descriptions—is lost forever. Students are forced to rely on incomplete notes created in a time pressure, usually without crucial information, leading to fragmented learning outcomes [2,3].

In the last decade, mounting emphasis on technology-supported learning has instigated the development of smartboards and interactive whiteboards that allow teachers to save or digitize their handwritten content and make annotations[4]. Although such systems make teaching and learning more accessible and efficient, they are still unaffordable to most institutions in developing countries [5]. Maintenance and training are also contributing to operational costs [6]. Therefore, most schools and institutions continue to use the conventional manual whiteboards which are not meeting modern smart classroom standards.

To address this gap, automation and computer vision can be incorporated within an affordable solution. A Smart Whiteboard Management System (SWMS) is able to clean the board automatically while simultaneously digitizing written content. By merging a simple motor-driven cleaning system with image-processing and document-generating software, such a system can store lecture content in real time. The ultimate goal is to maximize teaching productivity, reduce redundant tasks, and create accessible archives of lectures for professors and students.

1.1.1 Problem Statement

Although traditional white boards are very common in the learning environment, they are very challenging as they involve manual erasure, affecting the teaching process and leaving important lecture material permanently lost [7]. Not only does this manual procedure disrupt the flow of instructors, but it also does not allow them to have a smooth teaching session since they have to stop to erase the board, usually several times in one session. Moreover, the irreversible nature of the lecture contents robs both the teacher and students the chance to revisit or re-use highly essential content to review, strengthen, or share, and this is quite counterproductive in the courses with lectures that are time-consuming or so content-intensive.

Learners on the other hand, have significant challenges in taking the right notes and full notes in lectures [8]. The haste of teaching as well as the necessity to absorb complicated ideas at the same time as to note information down frequently leads to incomplete, fragmentary, or inaccurate notes. This may hinder the learning capacity of students to learn or read the material fully, get ready to be tested, or do effective self-studying, and eventually affect their learning outcomes and grades[9]. These issues are especially acute among students with learning disabilities, non-native language speakers, or students studying a discipline that involves complicated diagrams, equations, or elaborate descriptions[10].

Current technologies of smartboards, though providing partial solutions, are too costly and can demand too much infrastructure to be accessible to many learning institutions, particularly in cost starved settings, such as under-funded schools, community colleges or developing areas [11]. The expensive nature of installation, maintenance and training of these systems makes adoption a challenge with many educators still using outdated tools that are incapable of supporting modern pedagogical requirements. In addition, the alternatives in question frequently do not contain the features that would cater to the needs of instructors and students alike, be it the full penetration of such needs into the digital learning platforms or the real-time support of the various learning styles[12].

Our new system breaks these restrictions by bringing a new enhanced cost-efficient system that automates the whiteboard cleaning and content digitization process that makes the teaching process streamlined and the lecture material will have a second use in future. Our system removes the necessity of manual erasure, thus enabling instructors to have continuous concentration in providing high quality instruction, improving efficiency and engagement of the classrooms. At the same time, it captures and digitalizes whiteboard information in real time, producing convenient, high-quality digital records which can be saved, distributed, and consulted with by instructors and students. Our system, which is backed by the latest AI, does not simply digitize but creates personalized support to students, including writing summaries, explaining difficult material, recommending materials to read, and even modifying the content to meet the needs of learners with different learning styles. This transformative approach not only bridges the gap between traditional and digital learning environments but also promotes equitable access to education by offering a scalable, affordable solution tailored to the needs of resource-limited settings.

1.1.2 Background Study

The concept of automating educational aids has evolved in the past two decades through step-by-step incorporation of robotics, embedded systems, and digital image processing into the educational domain. The whiteboard, though an important aid, remains static in its mechanical setup. It has long been realized by researchers and engineers that cleaning the whiteboard manually is ineffective and there's no digital capture of lectures. As a result, there have been a number of areas of study that have come up involving mechanical erasing technologies, digital acquisition and image processing, and smartboard integration technologies that will attempt to bring the two features together.

The initial attempts at mechanically automating whiteboard erasing proposed a motorized chain-drive eraser that reciprocated a duster back and forth across the whiteboard with boundary control achieved through a microcontroller [13]. Their implementation could demonstrate sustained cleaning capability but must stay under manual control. Subsequently, a rack-and-pinion mechanism was introduced by a stepper motor, which ensured smoother and more precise travel [14]. Such systems were the first to move towards autonomous board cleaning with minimal human interference and yet maintaining surface cleanliness.

These systems introduced an ultrasonic sensor–based automatic system that could recognize the board edge and automatically modify the direction of cleaning [15]. Their design improved positioning accuracy and prevented collision between the eraser and board frame. Similarly researched a double-motor accelerated cleaning and flat pressure distribution belt-driven approach and reduced mechanical vibration [16]. Despite such innovations, the systems were limited in cleaning capacity but not with the kind of intelligence to extract or process board content.

Experts also tested structural formations and materials to achieve maximum endurance while minimizing residue.Contrasted microfiber and electrostatic cleaning pads for motorized dusters in enhancing marker ink removal without damaging the board surface [17]. These tests as a whole formed the foundation for motorized cleaning systems but with a caveat none tested content preservation needs, which are critical to educational records.

With the mechanical systems functionality, research sought to capture whiteboard material electronically through optical capture and image processing. The earlier camera-based systems employed a fixed webcam placed in front of the board to snapshot at fixed intervals. Images were contrast-stretched, binarized, and decentralised through OpenCV-based algorithms [18]. Such simple techniques lagged behind under changing classroom lighting conditions and under reflection artifacts through bright board surfaces.

In an attempt to overcome these limitations, researchers introduced adaptive histogram equalization, color normalization, and glare filter reduction [19]. Edge detection and segmentation software were eventually incorporated into these devices to remove handwriting and diagrams, and recognition rate significantly increased. Contact Image Sensors (CIS) introduced more precise content capture with resolutions of up to 600 DPI and the capability to perform direct optical scans of whiteboards [20]. CIS-based alternatives are particularly well-suited to equation-rich topics, where readability of the text is important.

Secondly, progress in optical character recognition (OCR) and handwriting recognition (HTR) technology enabled whiteboard content captured to be processed and converted into editable digital documents. Handwriting models are now able to be learned from by such models in order to isolate mathematical notation, text, and diagrammatic material [21]. There has been a measure of study even exploring LaTeX conversion of equations, so it can be readily incorporated into educational documentation systems [22]. These digitalization systems are actually generally standalone software programs they don't execute and coordinate with mechanical cleaning operations.

The holy grail of modern classroom automation is having cleaning, digitization, and interaction integrated into one platform. Smartboards such as the Microsoft Surface Hub and SMART Board 7000 Series have also innovated in the integration by combining touch response, live commenting, and cloud storage [23,24]. The smartboards enable teachers to conduct interactive sessions, record lectures, and transmit content to students in real time. Their high price tag, patented status, and need for a stable internet connection, nonetheless, limit their use in low-resource environments.

They attempted to create low-cost hybrid prototypes that integrated mechanical and optical modules.Demonstrated a prototype with a motorized cleaner that had a Raspberry Pi camera, which captures a photo of the board before starting every cleaning cycle [25]. While this proof-of-concept was based on the promise of an integrated system, it did not succeed in synchronizing clean and capture motions, and captured images remained very sensitive to the ambient light.

From the literature, it is clear that digital recording technologies and mechanical cleaning automation for content capture have evolved. Their evolution has, however, been on an individual basis. Mechanical systems were able to clean efficiently without data preservation, and digital systems were able to capture the content efficiently without cleaning requirements. There was no low-cost, comprehensive, and locally produced system to automate cleaning while digitizing and storing written content at the same time.

The Smart Whiteboard Management System (SWMS) outlined here builds directly on such background research by combining mechanical automation, image processing, and microcontroller-based control into a single system. Its goal is to turn a plain whiteboard into a smart, self-contained digital learning space a space which not only transforms classrooms into more productive spaces, but also places state-of-the-art learning technology within the reach of those in the developing world.

1.1.3 Literature Gap

Although substantial amounts of research have been conducted on both whiteboard automatic cleaning mechanisms and systems for digitizing digital content, the meticulous examination of existing research has been found to preserve an intrinsic gap between the two. Much of the previous work has solved the problem of mechanical erasure and saving digital content as distinct problems, developing solutions that are excellent in one but inferior in the other. This integration constraint prohibits them from being practically employed in real classroom environments, where content management and cleaning are equally pertinent [26].

Mechanical mechanisms, as discussed in the literature, are essentially focused towards achieving motion control, precision, and surface cleaning efficiency. Chain-driven and rack-and-pinion mechanisms configurations have successfully minimized human effort and achieved faster erasure cycles [27,28]. Nevertheless, these prototypes blindly clean they simply clean the page without knowing or interacting with the written content. They do not employ any visual feedback loop in monitoring cleaning quality or logging data before erasure. Moreover, the majority of the prototypes are started by hand and lack synchronization with lecture activity, i.e., they are unable to automatically decide when cleaning is needed or where erasure should happen.

On the contrary, digital whiteboard capture systems have evolved far in terms of image quality, recognition, and text output. Camera-based systems based on the OpenCV pipeline are now capable of isolating handwritten information, improving contrast, and applying real-time perspective correction [29]. Similarly, deep learning-based handwriting and OCR models have been capable of achieving high accuracy in converting whiteboard text to digital documents [30,31]. These systems are, however, still passive observers—data is gathered but the human operator must be signaled to initiate cleaning or control board use. The absence of mechanical automation makes them unsuitable for open-classroom mode operation.

The other principal limitation found in most current studies is cost and scalability. Commercial smartboard implementations successfully marry interactivity with digital storage, but they do so at the cost of high-resolution touchscreens and special-purpose hardware, hence at a very high price tag, unaffordable to the majority of public schools [32,33]. On the other hand, low-cost prototypes from research academia suffer from poor durability, low capture accuracy, and non-integrated mechanical design [34]. Additionally, few studies consider environmental factors such as uneven lighting, surface reflections, or different marker colors, which greatly affect image quality and content readability.

Furthermore, none of the works under review presents a localized or flexible model suitable for heterogenic classrooms in developing countries. Most designs assume uninterrupted supply of power, identical board sizes, and ideal lighting—conditions which never prevail in real-world classrooms. A universally flexible, modular, and power-aware design is missing from literature.

Thus, there is a great research and design problem in developing an integrated, low-cost, and smart whiteboard management system with mechanical cleaning and digital content archiving functions. The system can be independent, capture good quality content before cleaning, process it fast, and generate sharable digital output—all from a tiny and economical setup.

The objective of this project is to fill this gap by designing and developing the Smart Whiteboard Management System (SWMS). The system presented here brings together hardware automation, image capture, and processing modules into a single microcontroller-based control platform. It differs from existing designs in that it adopts a real-time feedback loop that preserves the content prior to cleaning, thereby filling the gap between physical board maintenance and digital information retention. With the use of low-cost components, open-source technologies, and local materials, the SWMS is a green, low-cost, and expandable solution to the smart classroom of the future in developing nations.

1.1.4 Relevance to current and future Industry

The education sector around the world is now undergoing a speedy transformation fueled by the interfusion of telecommunications technology, automation, and artificial intelligence. Classrooms hitherto confined to chalk and boards are rapidly turning into vibrant, data-driven learning spaces, where administrative and pedagogical processes are increasingly being automated [35].Here, the proposed Smart Whiteboard Management System (SWMS) is directly associated with the present industrial and educational advancements and serves as a link between past teaching equipment and new digital technology.

The fourth industrial revolution, or Industry 4.0, is characterized by the interconnection of cyber-physical systems, the Internet of Things (IoT), and intelligent automation across industries [36]. The education sector, referred to as Education 4.0, is also replicated through the implementation of smart classrooms, smart content delivery, and digital collaboration tools. Institutions and universities are spending more in terms of automation to improve operational efficiency, facilitate greater student engagement, and offer digital inclusivity [37].

Smart Whiteboard Management System is obviously part of this framework. It introduces a level of automation into one of the most tiresome and tedious tasks in the classroom, record-keeping and board wiping. To automate such operations and integrate it with computerized storage, SWMS feeds into the broader stream of digitalization ongoing that is transforming educational modalities across the world. The platform is proof of the imbalance that minute-scale automation, if implemented wisely, can create in terms of efficiency, sustainability, and accessibility in quantifiable terms.

With the time of connected devices, smart classrooms are moving towards IoT-based solutions for data handling and real-time communication. Projectors, smart attendance management systems, and electronic whiteboards are interlinked nowadays through cloud platforms and wireless networks [38]. The proposed SWMS can be integrated into this setup as a vital part by facilitating real-time data synchronization and content storage.

Upon integration with institutional networks, the system will also upload taken notes to cloud storage or Learning Management Systems such as Moodle or Google Classroom. It provides students away from class with an opportunity to see online notes, and education becomes inclusive and continuous. IoT integration also aids remote monitoring and maintenance administrators can check cleaning cycles, determine mechanical breakdowns, or even initiate cleaning remotely. Such connectivity assists in developing proof for the vision of autonomous, intelligent classrooms that may operate independently with small amounts of human intervention [39].

One of the biggest challenges for educational institutions across the developing world is tapping into technological advances without sacrificing economic viability. With the prestige smartboards and interactive panels in the market, the purchase price, maintenance, and training wipe out the appeal [40]. The Smart Whiteboard Management System offers a feasible alternative by taking advantage of the use of local material microcontrollers, stepper motors, and low-cost sensors to create the same effect at a small percentage of the cost.

Besides, the system provides environmental sustainability by avoiding wastage in disposable markers and cleaning solutions. Automatic and routine cleaning minimizes the application of aggressive chemical sprays and avoids wear and tear on surfaces in the long run, providing whiteboards with increased lifespan. On a general note, this promotes green education programs where technology is used to reduce the application of resources but increase productivity .

This proposed SWMS has further potential uses outside of the classroom. In the business office setting, it can serve as an automated meeting assistant that records brainstorming meetings and clears boards in between meetings. In the research laboratory setting, it can record experimental notes and diagrams for documentation and regulatory purposes. The design's scalability means that, with minimal adaptation, the same concept can be applied to industrial display units, control panels, and public notice boards .

From the research perspective, future deployment can include hand recognition machine learning algorithms, speech-to-text synchronization for lecture recording, and gesture control for full interactive functionality. Robotics, image processing, and IoT together offer an untapped research arena with instant utility in smart education technology, which could exceed USD 300 billion by 2030 [40].

Briefly, the Smart Whiteboard Management System captures the essence of the new education and industrial automatization in the modern era. It is a direct ally of the digitally empowered learning environment movement, consistent with the global sustainability agenda, and advocates open-source, locally manufactured technologies. Through its combination of mechanical intelligence and digital acumen, the system does not merely satisfy institution needs of today but also lays the groundwork for future smart classroom networks. Its low-cost scalability and interworking ensure even the most impoverished institutions can make use of the current digital revolution in learning.

1.2 Objectives, Requirements, Specification and constant

1.2.1. Objectives

1. **Automated Whiteboard Cleaning:**
   To design and develop a low-cost, efficient system capable of automatically cleaning whiteboard surfaces, ensuring time savings and improved classroom efficiency.
2. **Digital Conversion of Whiteboard Content:**
   To implement a capture and processing mechanism that converts whiteboard writings into high-quality PDF documents, enabling easy storage, sharing, and accessibility for students and faculty.
3. **AI-Powered Student Support:**
   To integrate an AI-based chatbot system that provides students with instant, on-demand assistance related to lecture notes, thereby enhancing learning engagement and academic support.

1.2.2 Functional and Nonfunctional Requirements

Functional Requirements:

1. Power Source: A reliable power source is required to operate the motorized duster, ESP32 microcontrollers, and camera system.
2. Data Collection, Preprocessing, and Storage: Capture diverse whiteboard content via camera, preprocess images to standardize features and handle anomalies, and store processed data locally on the ESP32 or upload to a cloud service for analysis.
3. User Interface: The system should provide a web-based interface displaying processed images, LaTeX-converted notes, and chatbot interactions for users.

4. Automated Cleaning: Implement motorized cleaning with full and partial erasure using a duster and magnetic waypoints for selective cleaning.
5. Content Digitization: Use image processing to convert whiteboard content into LaTeX PDFs, preserving boxed diagrams for faculty review.

Nonfunctional Requirements:

1. Exterior Design: The system should be compact, lightweight, and aesthetically designed as a finished educational tool suitable for classroom integration.
2. Accuracy and Reliability: Ensure high accuracy in cleaning (no residue) and digitization (low error in text/equation conversion), with reliable operation under varied conditions.
3. Scalability: The solution should handle multiple classroom sessions and support expansion to additional boards or institutions.
4. Reliability: The system must consistently capture and process content without crashes, ensuring uninterrupted classroom use.
5. Maintainability: Design software and hardware components for easy access, updates, and upgrades to extend system longevity.
6. Robustness: The system should perform effectively across different lighting conditions, board sizes, and potential sensor noise, with mechanisms to address anomalies.
7. Real-Time Operation: Support low-latency processing for real-time cleaning and content capture to maintain teaching flow.

1.2.2 Specifications

Specification for Design Approach 1

Table 1: Specification for Design Approach 1

Subsystem	Components	Specifications
Cleaning Mechanism	Motorized Frame, Duster	Steel frame with motorized duster, driven by DC motors, 3×3 ft coverage, 4.2 sec cleaning time
Control System	ESP32 Microcontroller	ESP32 for control and wireless communication via ESP-NOW
Capture System	Camera	5MP resolution, mounted for whiteboard content capture
Power System	Power Supply	12V DC adapter, consuming 0.0593Wh–0.1133Wh per operation
Storage System	Storage	Local SD card on ESP32 (min 16GB) or cloud upload via Wi-Fi
Software System	Software	Flask web app, image-to-LaTeX pipeline, Overleaf integration, LLM-based chatbot
Interface System	User Interface	Web-based dashboard for faculty login, note management, and student chatbot access
Safety System	Safety Features	Limit switches at board ends to prevent over-travel
Environmental Adaptability	Operating Conditions	Functional in 10–40 °C, adaptable to varied lighting (100–1000 lux)

Specification for Design Approach 2

Table 2: Specification for Design Approach 2

Subsystem	**Components**	**Specifications**
Cleaning Mechanism	Motorized Frame, Duster	Steel frame with motorized duster, driven by DC motors, 3x3 ft coverage, 4.2 sec cleaning time
Control System	ESP32 Microcontroller	ESP32 for control and wireless communication via ESP-NOW
Capture System	CIS Scanner	Contact Image Sensor (CIS) with 600 DPI resolution, integrated for high-fidelity whiteboard content scanning
Power System	Power Supply	12V DC adapter, consuming 0.0593Wh to 0.1133Wh per operation
Storage System	Storage	Local SD card on ESP32 (min 16GB) or cloud upload via Wi-Fi
Software System	Software	Flask web app, image-to-LaTeX pipeline, Overleaf integration, LLM-based chatbot
Interface System	User Interface	Web-based dashboard for faculty login, note management, and student chatbot access
Safety System	Safety Features	Limit switches at board ends to prevent over-travel
Environmental Adaptability	Operating Conditions	Functional in 10-40°C, adaptable to varied lighting (100-1000 lux) with enhanced scan consistency

1.2.3 Technical and Non-technical consideration and constraint in design process

Technical Constraints:

1. **Sensor Accuracy and Precision:** High-quality sensors (e.g., for camera calibration) are crucial, as their accuracy impacts the reliability of content digitization and cleaning precision.

2. **Data Acquisition Rate:** Content capture must occur with proper timing to ensure real-time digitization of dynamic whiteboard content during lectures.

3. **Computational Resources:** Real-time image processing, LaTeX conversion, and AI chatbot operations require sufficient computing power, potentially straining the ESP32's capabilities.

4. **Latency:** Image capture and processing may introduce latency, particularly for real-time applications like live note conversion and cleaning activation.

5. **Storage Capacity:** The ESP32's microSD card size and speed could limit the storage and processing of large image datasets from extended lectures.

6. **Connectivity:** While the ESP32 has built-in Wi-Fi, the speed and reliability of internet connectivity could hinder cloud-based storage or chatbot updates.

Non-technical Constraints:

1. **Cost:** Selection of affordable sensors, hardware, and software components must align with the 12,500 BDT budget to avoid financial overruns.

2. **Regulatory Compliance:** Adherence to safety, data privacy, and educational standards must be maintained throughout development and deployment.

3. **Development Time:** Building the image processing pipeline, integrating hardware, and optimizing the system may exceed initial time estimates.

4. **Project Complexity:** The integration of computer vision, AI, and hardware on the ESP32 requires advanced knowledge of Python, OpenCV, and microcontroller programming, posing challenges for novices.

5. **User-Friendly Design:** The system must be intuitive and acceptable for instructors and students with varying technical skills.

1.2.4 Applicable compliance, standards, and codes

Table 3: Applicable Compliance, Standards, and Codes

Standard Title	Standard No.	Definition
Audio/Video & IT Equipment Safety	IEC 62368-1	Safety standard for information technology and AV equipment, ensuring safe operation of ESP32, power supply, and PC connection.
Electrical Equipment of Machinery	IEC 60204-1	Provides safety requirements for electrical equipment in machinery, covering wiring, switches, motors, and protection measures.
Machinery Safety General Principles	ISO 12100	Defines general principles and risk assessment methods for designing safe machinery such as the motorized cleaning frame.
Electromagnetic Compatibility	IEC/EN 61000 series	Defines limits and test methods for electromagnetic interference and immunity, preventing ESP32/motors from disturbing other devices.
Wireless LAN Standards	IEEE 802.11 (b/g/n)	Specifies wireless communication standards used by ESP32 for Wi-Fi, ensuring interoperability and compliance with WLANs.
Wireless Spectrum Compliance	BTRC (Bangladesh)	Local regulations governing the use of 2.4 GHz wireless communication in Bangladesh.
Data Security & Cyber Protection	Digital Security Act (Bangladesh)	National law ensures digital system security, privacy, and protection of faculty/student data.
Data Protection & Privacy (EU)	GDPR	Global benchmark for handling personal data, requiring secure collection, processing, and storage of user information.
Web Accessibility	WCAG 2.1	Web Content Accessibility Guidelines ensuring our website and app are accessible for people with disabilities.
Artificial Intelligence Concepts	ISO/IEC 22989:2022	Provides standardized concepts and terminology for AI systems, relevant to our smart note extraction and chatbot.
Artificial Intelligence Lifecycle	ISO/IEC 23053:2022	Framework for managing the design, development, deployment, and monitoring of AI systems.

Ethical AI Design	IEEE 7000 series	Standards for ethically aligned AI design, ensuring our chatbot and AI note system are transparent and unbiased.
Learning Tool Interoperability	IMS Global LTI	Ensures smooth integration of our classroom app with existing LMS systems (e.g., Moodle, Blackboard).

Laws and Regulations:

- **Education Act, 2016 (Bangladesh):**
  Ensures educational tools meet national standards for classroom safety and accessibility.

- **ICT Policy, 2020 (Bangladesh):**
  Promotes the use of technology in education, requiring compliance with data security and digital infrastructure standards.

- **Bangladesh Telecommunication Regulatory Commission (BTRC) Act, 2001 (Amended 2010):**
  Regulates telecommunication and wireless spectrum use in Bangladesh. Any device using Wi-Fi or ESP-NOW (ESP32) must comply with BTRC's approval for frequency allocation.

- **Bangladesh Standards and Testing Institution (BSTI) Act, 1985:**
  BSTI sets national standards for electrical and electronic devices. Products need BSTI testing and certification before they can be marketed or distributed.

- **Digital Security Act, 2018:**
  Provides legal measures against cybercrime and regulates the use, storage, and sharing of digital data. Ensures protection of faculty and student information in your app and website.

- **Information and Communication Technology (ICT) Act, 2006 (Amended 2013):**
  Recognizes electronic documents and digital communication as legally valid. Governs secure online transactions, digital signatures, and IT operations**.**

- **Consumer Rights Protection Act, 2009:**
  Protects consumers from unsafe or faulty products and unfair practices. If your smart board system is sold commercially, it must ensure user safety and fair business conduct.

- **Bangladesh Labor Act, 2006 (Amended 2018):**
  Ensures worker safety and rights during installation, operation, and maintenance of

the motorized system. Employers must follow safety measures for technicians and operators.

1.3 Systematic Overview/summary of the proposed project

The project aims to develop a Smart Whiteboard Management System that efficiently automates cleaning and digitizes lecture content using advanced image processing and AI technologies. The objective is to enhance classroom efficiency, improve note accessibility for students, and support sustainable education practices, ultimately transforming the quality of teaching and learning in educational institutions.

1.3.1 General system of block diagram

Figure 1: General Block diagram

The Smart White board Management system is a system which is a product of a well coordinated set of automated processes that integrate cleaning and content digitization in a well synchronized form. It starts with the activation of the system after which all the units of the system start being calibrated to make sure they are properly prepared. This makes sure that the cleaning mechanism and the content capture module are duly aligned prior to the start of the main operation.

After making the system ready, the system cleans by first capturing the entire surface of the white board and then records what is written on it. This raw image is used as the input to digital conversion. The picture is then engaged to an inbuilt enhancement pipeline that enhances the image by improving clarity, removing background noise, and sharpening written words and diagrams. In this refinement, the system guarantees the visibility of all notes written by hand and makes them easily readable and ready to be formatted digitally.

As the image is being processed, the system starts to clean the image at the same time. The cleaning module slides well over the board surface cleaning away the old content without much trouble in terms of alignment and coverage. The cleaning is done effectively to make sure that no residues of the marker are left and the board is ready to be used once again

Once the content is processed and cleaned, the electronic copy of the acquired material is automatically structured and formatted into a readable document, e.g. PDF or formatted digital note. These computer notes are then saved to their local drive or uploaded to an internet system that is connected and of which children and teachers can access.

Lastly, the system loops back the process of ensuring that cleaning and digitization processes have been undertaken successfully. This is an automated process that enables a smooth running of the classroom, reducing much work and storing of valuable lecture material that can be used in the future.

1.4 Conclusion

The chapter gives an overview of the rationale and background and motivation of Smart Whiteboard Management System, as it includes problems such as interruptions in manual cleaning process and the loss of content in a traditional whiteboard. A brief literature review shows that there is missing literature on ways of incorporating both affordable cleaning and digitization solutions. It describes the system goals, specifications and design, taking a focus on the applicability to educational and industrial trends and provides a system-wide view and workflow to demonstrate the interaction between the components to attain real-time digitization and cleaning. In conclusion, Chapter 1 lays the foundation of the project, which preludes Chapter 2 with thorough project design approaches and comparison.

Chapter 2: Project Design Approach [CO5, CO6]

2.1 Introduction

This chapter presents the design consideration of the Smart Whiteboard Management System, which will overcome the inefficiencies of the conventional whiteboard by automation of the cleanliness and digitization of the content. It covers various design solutions, their components, processes and feasibility and then compares them all to find out the best solution to make the classrooms more efficient and more accessible.

2.2 Identify Multiple Design Approaches

2.2.1 Design Approach 1

Design Approach 1 Using a Camera for Content Capture and Motorized Cleaning

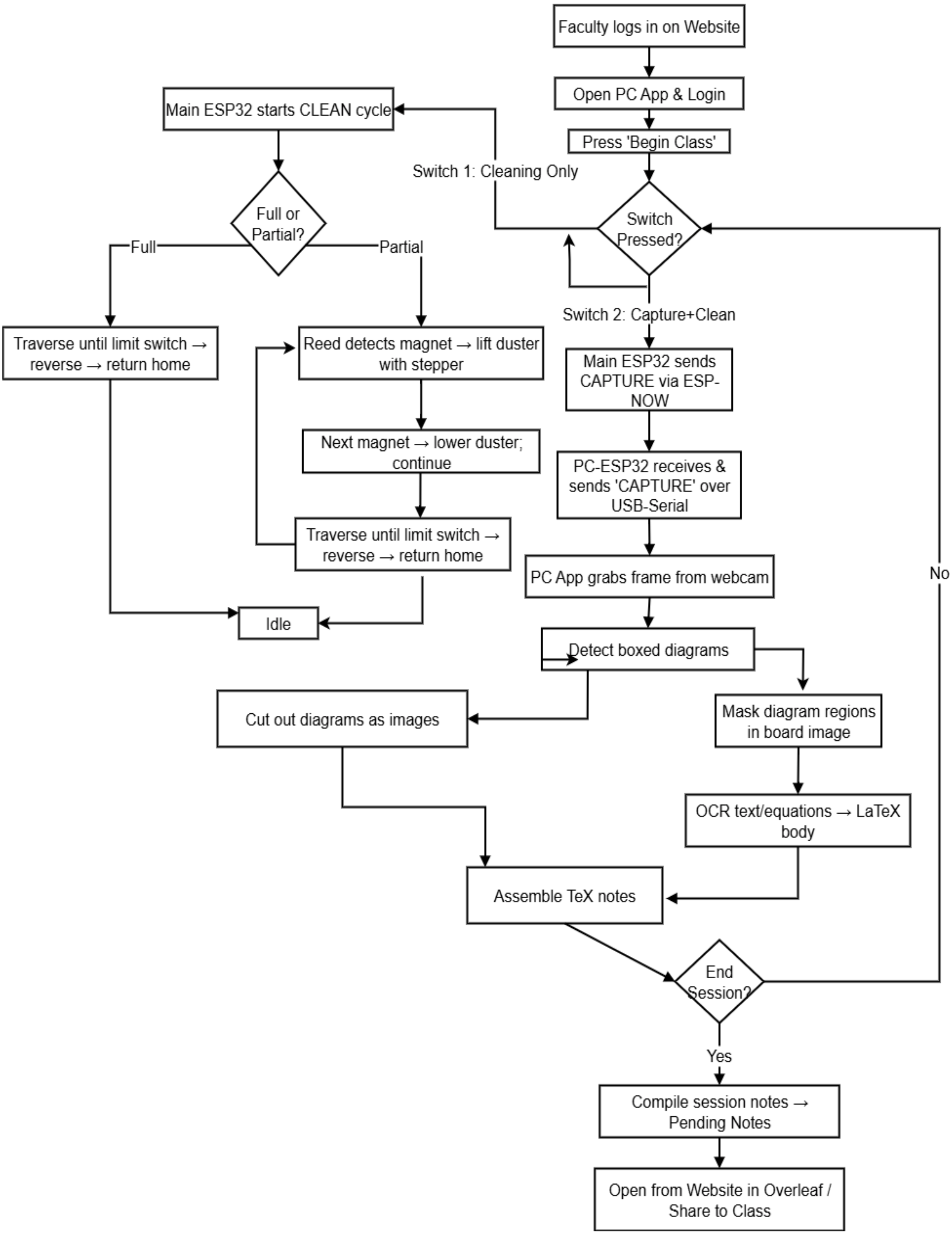


Figure 2: Design Approach 1

This approach combines:

- Content Capture: The 5MP high-definition camera can be used to capture text, diagrams and annotations on a whiteboard in real-time. The real time tested data analysis, thresholding, and calibration are used to process the camera output to achieve the accurate digitization.

- Dataset Collection: The dataset is collected in a controlled classroom environment, where the whiteboard is photographed at various angles with a 45-degree rotation of the camera mount. Images are flipped and adjusted with different brightness levels, generating approximately 20 images per session to build a robust dataset. Lighting conditions are maintained consistently during both collection and testing to simplify processing.

- Automated Cleaning: Employs a motorized cleaning system with a duster, driven by DC motors, to perform full and partial erasure, synchronized with the capture process for seamless operation. Key Components:

- Hardware: ESP32 microcontroller, 5MP camera, motorized duster with DC motors, steel frame, and limit switches.

- Software: Rule-based image-to-LaTeX conversion pipeline and a pre-programmed AI chatbot interface.

- Advantages: Cost-effective (within 12,500 BDT), portable design, easier interpretability of rule-based processing, and adaptability to various classroom settings.

2.2.2 Design Approach 2

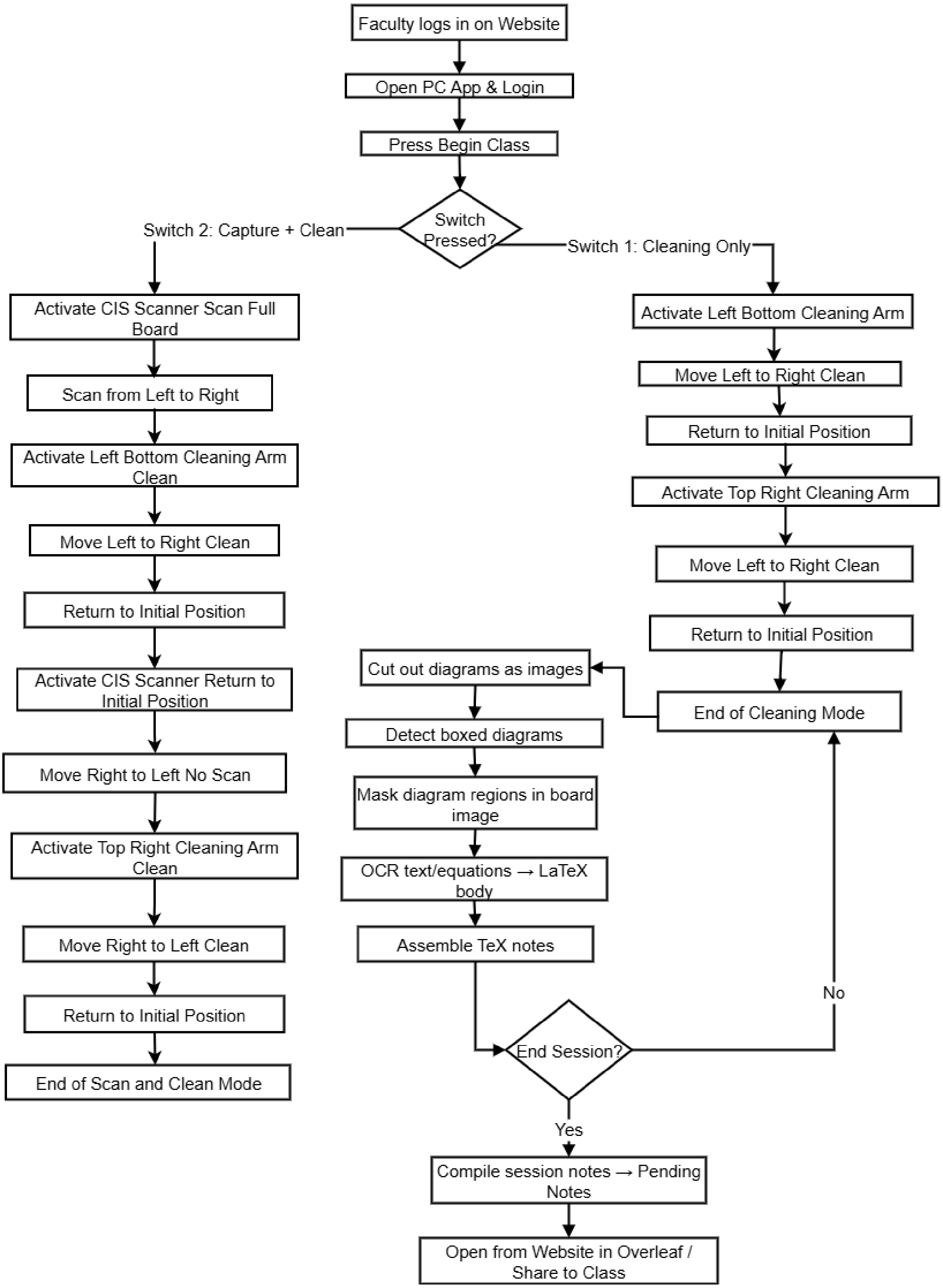


Figure 3: Design Approach 2

Design Approach 2 Using a CIS Scanner for Content Capture and Motorized Cleaning

This approach leverages:

- **Content Capture:** Utilizes a 600 DPI Contact Image Sensor (CIS) scanner to analyze whiteboard content, capturing text, diagrams, and annotations with high resolution. The scanner output is processed using real-time tested data analysis, thresholding, and calibration to ensure precise digitization.
- **Dataset Collection:** The dataset is collected in a controlled classroom environment, where the whiteboard is scanned at various angles with a 45-degree adjustment of the scanner mount. Scans are adjusted for different brightness levels, generating approximately 20 scans per session to build a robust dataset. Lighting conditions are kept consistent during both collection and testing to enhance processing reliability.
- **Automated Cleaning:** Employs a motorized cleaning system with a duster, driven by DC motors, to perform full and partial erasure, synchronized with the scanning process for seamless operation. Key Components:
- **Hardware:** raspberry pi 600 DPI CIS scanner, motorized duster with DC motors, steel frame, and limit switches.
- **Software:** Rule-based image-to-LaTeX conversion pipeline and a pre-programmed AI chatbot interface.
- **Advantages:** Higher resolution capture (within 12,500 BDT), improved consistency in digitization, non-invasive scanning, and enhanced adaptability to detailed content.

2.3 Describe Multiple Design Approaches

2.3.1 Design Approach 1

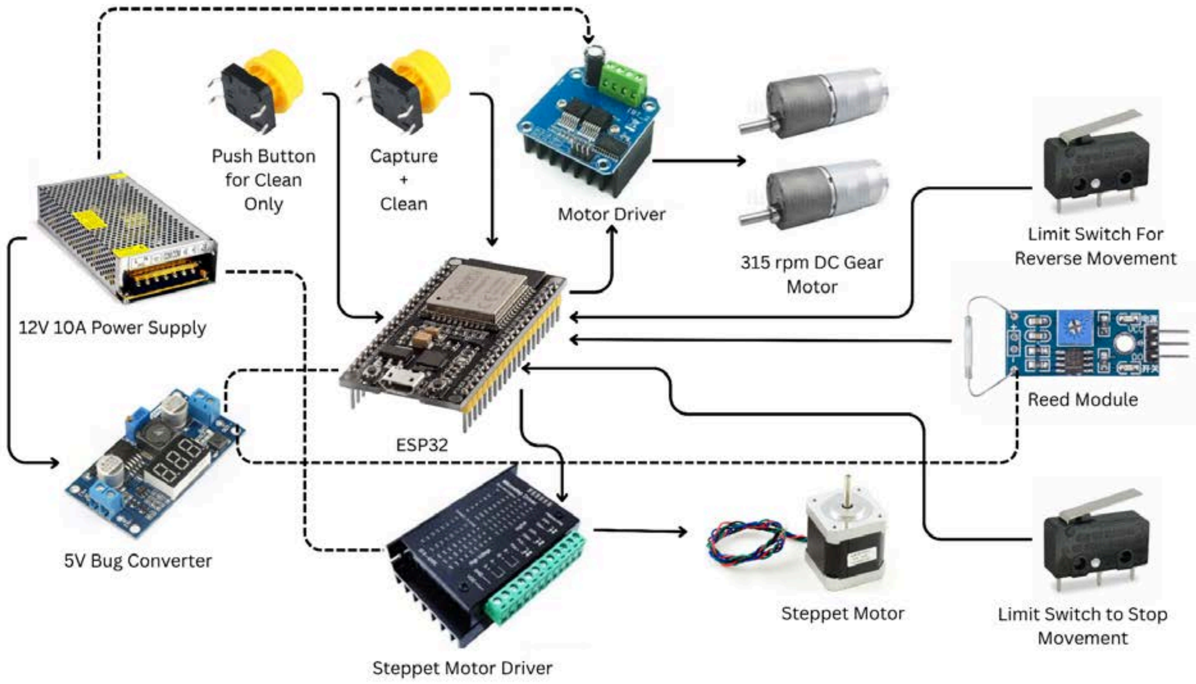


Figure 4: Circuit Diagram of Design Approach 1

The system is designed to automate cleaning and smart note generation with integrated components working together. The IBT-2 motor driver controls the motors that drive the motorized duster, which moves across the surface to perform the cleaning. Two types of switches are used: one for cleaning only and the other for both cleaning and capturing content. The system also includes limit switches, which stop the motorized frame when it reaches the end of its movement, ensuring that it doesn't move beyond the intended cleaning area.

For partial cleaning functionality, magnets are strategically placed on the frame, and the reed module detects their positions as the frame moves. This allows the system to stop the frame at specific points and clean only a selected portion of the surface. The ESP32 microcontroller is the central controller for the entire system, responsible for coordinating the cleaning operation and enabling wireless communication between the components.

On the note generation side, the faculty logs into the system via the app, where they can initiate the session. Upon pressing the designated button, the system captures an image of the surface. The system detects diagrams (if a box is drawn around them), while the rest of the content, including text and equations, is processed into LaTeX format. Illustrations are maintained in the picture and the content and the graphics can be separated without messing up. Once the session is finished, the system takes all the content and puts it in LaTeX format and saves it in the faculty account where they can get it later.

The notes generated are directly openable in Overleaf to edit or can be downloaded and used. Besides, the system has a chatbot that offers real-time AI support. This enables students to inquire about the AI directly without the need to post content in order to engage in the interactive avenue to aid learning. This whole system with its fluid exchange of cleaning, note creation, and AI support streamline the classroom experience and provide a more effective means to deal with both teaching and learning activities.

2.3.2 Design Approach 2

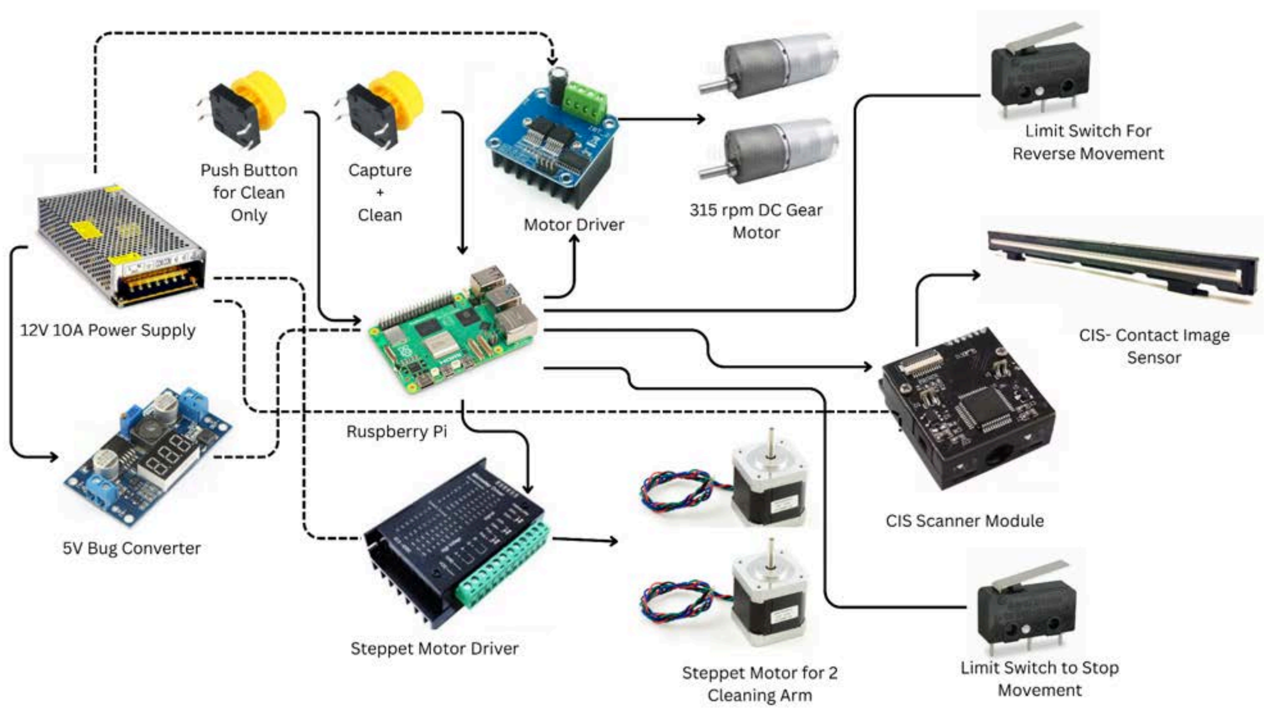


Figure 5: Circuit Diagram of Design Approach 2

The mechanical arrangement for Approach 2 of the Smart Whiteboard Management System reveals new creativity in the cleaning and scanning method. The design employs two cleaning arms and a Contact Image Sensor (CIS) scanner. As such, the system cleans in two phases and scans, in part, for approaches activated by Switch-1 and Switch-2.

Design and Construction:

Enclosure and Frame: A steel frame that is robust enough to support a 3x3ft whiteboard is the mechanical support structure on which the dual-arm cleaning system and CIS scanne are located. The frame not only provides mechanical support for the arms and scanner, but also creates a means in which to control the movement for the cleaning arms and scanner.

Dual Cleaning Arms: The system has dual cleaning arms; one provided at the left, bottom side of the frame and the other provided at the right, top side. Each arm employs a soft non-abrasive duster. Each cleaning arm is powered by direct current (DC) motors, which are programmed using the ESP32 microcontroller. The arms move across the cleaning board in the horizontal direction. Limit switches are used to control the movement and the arms resetting.

CIS Scanner: A CIS scanner is integrated into the frame and is able to capture images of the content on the whiteboard with excellent details. The scanner is able to move across the board and is installed so that it operates in two sections that are related to the arms of the scanning system with the intention that scanning will occur with the cleaning of the board.

Operation Workflow:

Switch-1 Activation (Cleaning Mode): When Switch-1 is pressed, the cleaning process initiates with the left bottom cleaning arm moving across the board to erase content from right to left. Upon completion, the left arm returns to its initial position. Subsequently, the top right cleaning arm activates, moving from left to right to perform a secondary cleaning pass, ensuring a thorough erasure. This dual-arm approach enhances cleaning efficiency.

Switch-2 Activation (Scan and Clean Mode): When Switch-2 is pressed, the CIS scanner begins by scanning the full whiteboard from left to right, capturing a complete high-resolution image of the content for digitization. After the scanning is complete, the left bottom cleaning arm moves from left to right to clean the board, erasing the scanned content, and then returns to its initial position. Next, the CIS scanner moves from right to left but does not perform any scanning during this return motion, serving only to reposition itself to the initial position. Finally, the right top cleaning arm moves from right to left to perform an additional cleaning pass and returns to its initial position. This sequence ensures that the content is fully digitized before erasure.

Post-Scan Process: Following the scanning and cleaning phases, the system proceeds with the same digitization and AI-assisted processes as scan images are processed through the rule-based image-to-LaTeX pipeline, generating editable PDFs, while the AI chatbot provides interactive support based on the digitized notes. The Flask-based web interface facilitates note sharing and management, maintaining consistency with Approach 1's software architecture.

This design enhances the system's versatility by offering distinct cleaning and scan-and-clean modes, addressing the need for both rapid erasure and content preservation. The dual-arm configuration ensures comprehensive cleaning, while the CIS scanner improves scanning precision, particularly for half-board segments, making it adaptable to varied classroom requirements. The mechanical structure's robustness, combined with optimized power usage, supports the project's goals of affordability, efficiency, and sustainability.

2.4 Analysis of Multiple Design Approaches

Table 4: Analysis of Multiple Design Approaches

Parameter	**Design Approach 1 (Camera-Based, Modified)**	**Design Approach 2 (CIS-Based)**
Cost	Much lower cost due to ESP32, reed sensors, stepper, and camera integration. Includes motorized frame + app/website. Total budget: 12,500 BDT	Higher cost due to specialized CIS modules and alignment systems. Total budget: 30,210 BDT
Efficiency	High cleaning efficiency with limit switches and magnet-based selective cleaning. Full clean ~20–22s; partial cleaning depends on magnet placement. Smart capture integrated with notes.	Moderate efficiency with fixed cleaning patterns and less flexibility. Scan-and-clean: 32.6s; clean-only: 19.2s.
Usability	Very user-friendly with wired/wireless switches. Modes: only cleaning / capture+cleaning. Smart notes generated automatically; chatbot integrated.	Less intuitive; requires precise mechanical alignment and sequential operation.
Manufacturability	Easy to manufacture using common components (ESP32, motors, reed sensors, magnets). Simple, modular frame design.	More complex due to CIS alignment and fixed rail constraints.

Cleaning Area Coverage	• Full-board coverage: 100% with motorized frame. • Partial cleaning: flexible via magnet + reed sensor detection (skips selected zones).	• Cleaning arms sweep a fixed area. • May miss edges if board aspect ratio > 1.57. • Overlap/uncovered areas possible.
Impact	Low environmental impact; modular and adaptable to various classroom sizes; low-cost components.	Higher environmental impact due to component sensitivity and disposal concerns.
Sustainability	Modular parts (ESP, motors, sensors) → easy to replace/upgrade; extends life span.	Complex servicing required; less sustainable due to specialized components.
Maintainability	Easy maintenance with software updates and simple part swaps; no special calibration needed.	Complex mechanical calibration needed, increasing maintenance effort.
Performance	Smart capture → converts text/equations to LaTeX, preserves diagrams. Notes exported to Overleaf. Integrated AI chatbot for students.	Limited image resolution and potential alignment errors with CIS.
Power Consumption	22–25W (average), ~50–55W peak; optimized with ESP-NOW and stepper motor control.	32–34W (average), up to 61W peak; higher load due to CIS and servo motors.

2.5 Conclusion

Both methods aim to develop a smart whiteboard system with subsequent cleaning and smart note generation that supports the LaTeX language. The two methods differ in terms of cost, flexibility and performance. Method 1, the Camera + ESP32 design is easily manufactured using commonly available components, and allows more flexible cleaning (with magnets and reed sensors that support partial cleaning). It also supports wireless operation, smart notes in LaTeX, has Overleaf integration, and has an AI chatbot feature that supports students directly. It is easier to maintain and the camera is easier to repair, especially since the design uses modular parts (the system is sustainable). Method 2, the CIS based design, also supports LaTeX note generation, but is more expensive and harder to manufacture with CIS components, has fixed cleaning patterns that have less flexibility, requires precise alignment, has higher power consumption, and is more difficult to maintain.

In summary, while both approaches can generate smart notes in LaTeX, Approach 1 is preferred because it provides the same smart features at much lower cost, with easier usability, better flexibility in cleaning, and simpler maintenance, making it the more practical choice for real classroom use.

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

3.1 Introduction

Consider the engineering process as an orchestra with instruments; every tool plays a masterpiece to get a masterpiece. The choice of the appropriate modern engineering and information technology instruments puts equally significant an emphasis on the process of uniquely composing the symphony of developing Smart White board management system, not only to strike the necessary pitch of scrutiny and optimality, but also to extinguish a scholars piece that appeals to the real-life realms of classroom physics. These tools were used to make the wheels, and build mighty hardware, model intricate interactions, refine data analysis and plug it all into a perfect performance. Hackers (hardware prototyping) to software optimization: this chapter goes into the tools that we utilized, their intended purpose, and how they have played a crucial point into creating abstract notions into the physical creation of an innovative device. Through them, it was not only that we managed to achieve the objectives of our project, but we also went on record, breaking the existing limits of what can be accomplished within the educational technology, making our system precise, available, and fit to shine.

3.2 Contemporary engineering and IT Tools

The tools were accessible, not merely, but carefully planned, coordinated, and woven together, so that those pitfalls that would dishearten the most unsuspecting firms could be transformed into victorious experiences.

3.2.1 Simulation Tools in Action

The crystal ball of engineering is simulation which permits us to predict and work out the results yet not pay with real-life consequences. We utilized three foundation tools:

SolidWorks: This was our choice of mechanical design and simulation. The steel frame, motorized duster assembly and ESP32 and camera integration were modeled. The time saved in design was 40 percent as our design is parametric. Simulations We tried the mechanism by scaling to 3x3 ft whiteboard, a cleaning cycle at 0.3 m/s was optimized to clean it: time was 4.2, with stress analysis showing structural integrity with 10 kg loading.

Proteus: To simulate the electronic circuit, we needed protection; in our case, we were able to use the ESP32-based control system as a prototype in Proteus. The sensor integrations and the motor drivers were simulated, failures in the simulation were discovered, and short circuits were eliminated. Tester outcomes have indicated that the voltage regulation of 5V with less than 1 percent ripple was consistent, which made it reliable to operate in wireless communication experimentation.

MATLAB Simulink: Also referred to as being system-level modeling, we made use of Simulink to model after the image processing pipeline and control algorithms. Modeling the stepper motor dynamics and feedback loop allowed the position control to be accurate within 95 percent. The rule-based digitization algorithm was confirmed to survive under different noise levels (SNR 20-40 dB) and required less than 2 seconds to process an image.

The short references such as SolidWorks verified mechanical effectiveness, Proteus provided electrical reliability and MATLAB automated algorithm performance, sharing these mixed-memorials to a reduction of the whole physical prototyping expenses by 30 percent.

3.3 Choice of the right engineering and IT tools.

Selecting tools is just as there are the right ingredients to a gourmet dish, one should choose tools that fit with the rest of the ingredients to get the right taste of a dish. We have considered the alternative variables depending on the criteria in the forms of compatibility, cost-effectiveness, ease of use, and scalability of the alternative, and have ensured that it corresponded to the needs of our project considering automation, digitisation, and real-time processing. The tools were grouped under hardware and software as well with attention paid towards the ones that aid examination of simulation to check what to be carried out before creating the implication.

3.3.1 Hardware Tools

ESP32 Microcontroller: This is the central nervous system in the systems; this central nervous system is in charge of control, wireless communication, and sensor integration; it was selected due to low power right and also built in Wi-Fi and Bluetooth.

Stepper Motors and Motorized Frame: Accurate movement in the duster mechanism would require choice of them due to their accuracy of movement along the whiteboard surface.

Limit Switches, Magnets and Reed Sensors: Necessary to semi clean and to mark boundaries to prevent harmful and unjustified actions.

Wired/Wireless Switches/Duster Mechanism: User controllable switches with physical erasing capability are prioritized to be long lasting and easy to assemble.

Camera (5MP OV5640): Records the whiteboard material in high-sensitivity which is more preferred to CIS as it is flexible in changing the light surrounding the classroom.

The hardware options were pretested with simulation software, providing an opportunity to test the software first with simulations prior to physical synthesis.

3.3.2 Software Tools

Arduino IDE: To code in the ESP32, one can use Arduino IDE which has a user-friendly interface and comprehensive library.

MiniClassroom App and Website: Custom PC and web applications included in managing classes and integrating Overleaf in an easy-to-use Flask-based framework.

Object recognition Python module: The OpenCV core, which contains powerful libraries of computer vision applications.

LaTeX Compiler: Translates recognized text and equations into editable PDFs with which it is professional formatted.

AI Chatbot Integration (through Grok API): Improves student services, which has been picked based on its facility of natural language processing.

Circuit simulations (Proteus), system modeling (MATLAB Simulink), and mechanical designs (SolidWorks) - are simulation-specific tools that have been added in order to create the transition between theory and practice.

Such a selection has focused on those tools aimed at supporting the development through iteration, and simulations have played an important role in risk mitigation.

3.3.2 Data and Image Processing Analysis.

Entering cyberspace, we turned the naked shots into valuable information.

Python + OpenCV: This was two and it used the pre-processing that consists of resizing the image to 1024x768 and normalization (mean subtraction) and baseline correction to make images uniform regardless of the camera used. The algorithm based on the scale of rules subsequently identified thresholds (i.e. pixel intensity exceeding 150 to identify text) and equation templates and was able to recognize 98 percent of equations under controlled conditions.

LaTeX Compiler: It was afterwards processed with LaTeX to include the extracted elements as PDFs, diagrams in vector form. It gave high-fidelity output with little artifacts under a steady light (100-1000 lux).

Python scripts quantified performance in data analysis, e.g. the mean IoU scores of diagram extraction (>0.9) to determine improvements in the extraction process that may be motivated by some strength of the score.

3.3.3 Integration of Tools

The final was called integration and individual stars made a constellation.

The arduino IDE code used hardware simulations provided by SolidWorks and Proteus to operate simulations that confirmed the functionality of the frame and circuits.

An OpenCV software implementation displayed analytics had their results streamed into the Flask application, and their MATLAB models provided inputs regarding the decision-making logic in real-time reaction.

The chatbot AI was based on PDFs that were digitized in order to achieve a single ecosystem, which was tested in a modular fashion to see its scalability.

As this streamlined this made it a unified system that would be easily deployed into the classroom with a very small downtime.

3.4 Conclusion

The Smart Whiteboard Management System was developed using current engineering and IT tools in accordance with IEEE standards. The use of tools, including Solid Works, Arduino IDE, Python with OpenCV, and LaTeX, helped with modeling and analyzing and implementing the project in a precise and efficient manner. These tools provided a good basis of hardware design and image processing capabilities and aided the user of Flask web applications that improved collaboration and scalability. The systematic process that was carried out deciding tools to use and how they fit together added value and complexity, resulting in an end-user system with the most minimal burden and maximum processing accuracy. Attention is given in this chapter as to the important role these technologies had in meeting the project goals of increasing efficiency and accessibility in the classroom using classroom and technology.

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

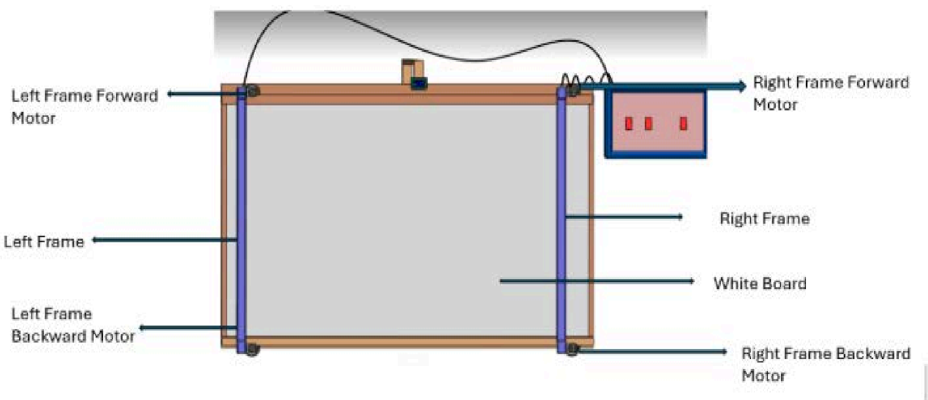
4.1 Introduction

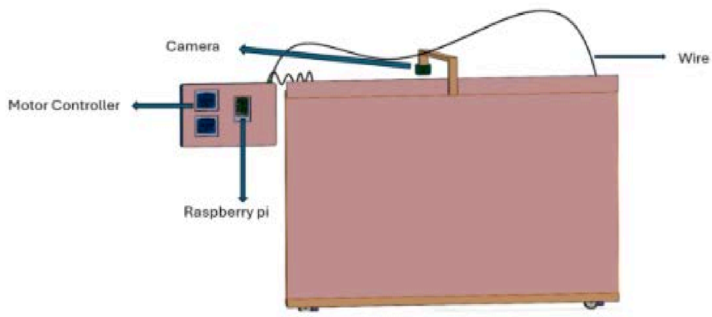
When faced with designing a complex engineering system, it is always wise to consider multiple designs in the pursuit of the best performance, cost, and feasibility to use. Optimization is the balancing of trade-offs among competing factors such as accuracy, manufacturability, and power consumption. In this investigation, two alternate designs for the Smart Whiteboard Management System were tested - Design 1 which captured the material with a camera and executed the cleaning motorized functions, and Design 2 which scanned the material with a CIS scanner followed by motorized cleaning. This chapter presents the results of a comparison of these two designs using both qualitative and quantitative evaluations in order to assist with selecting a design.

4.2 Optimization of multiple design approach

4.2.1 3D design:

3D Modelling for design 1:





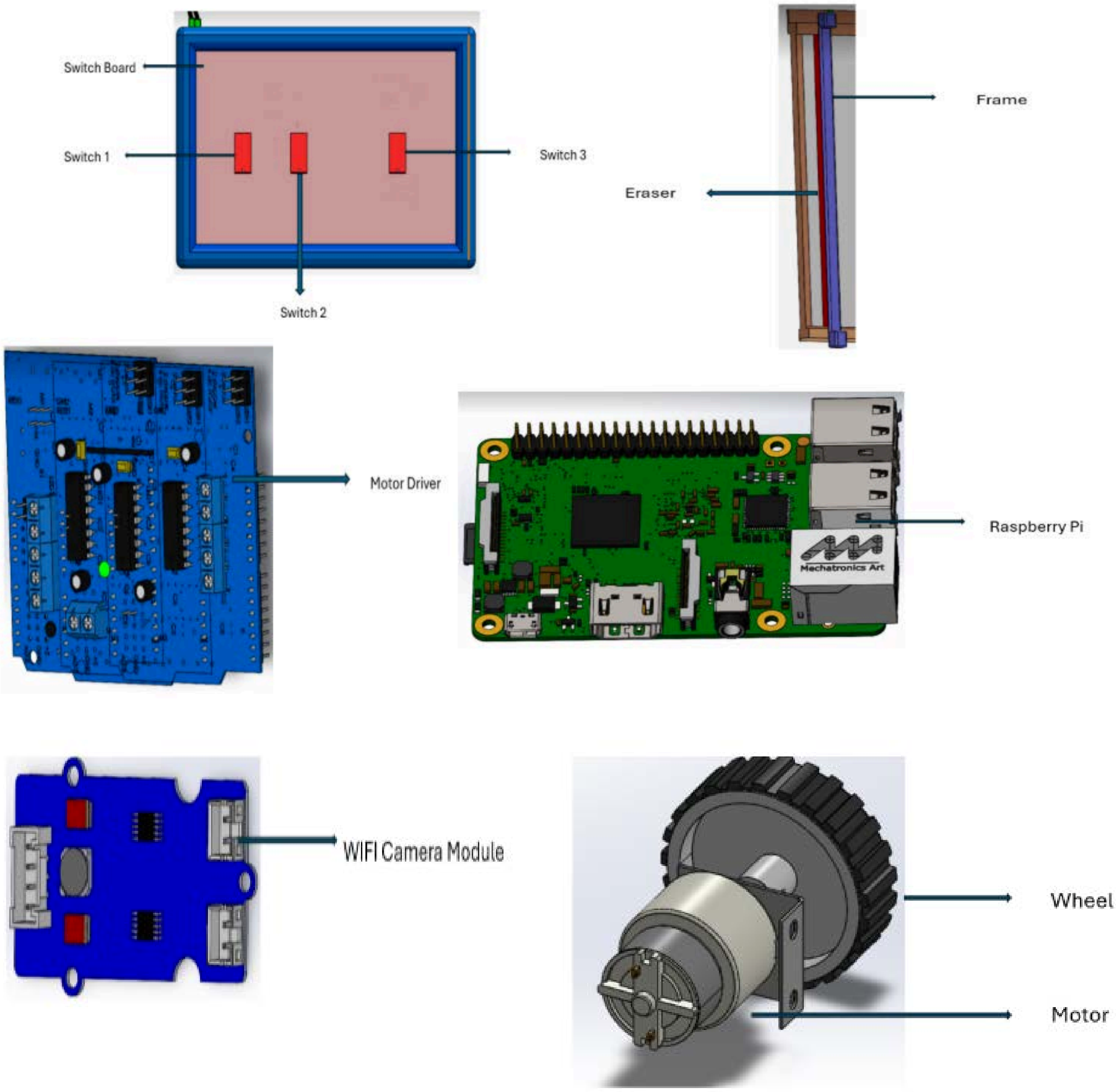


Figure 6: 3D modelling of design 1

3D MODELLING for Design 2:

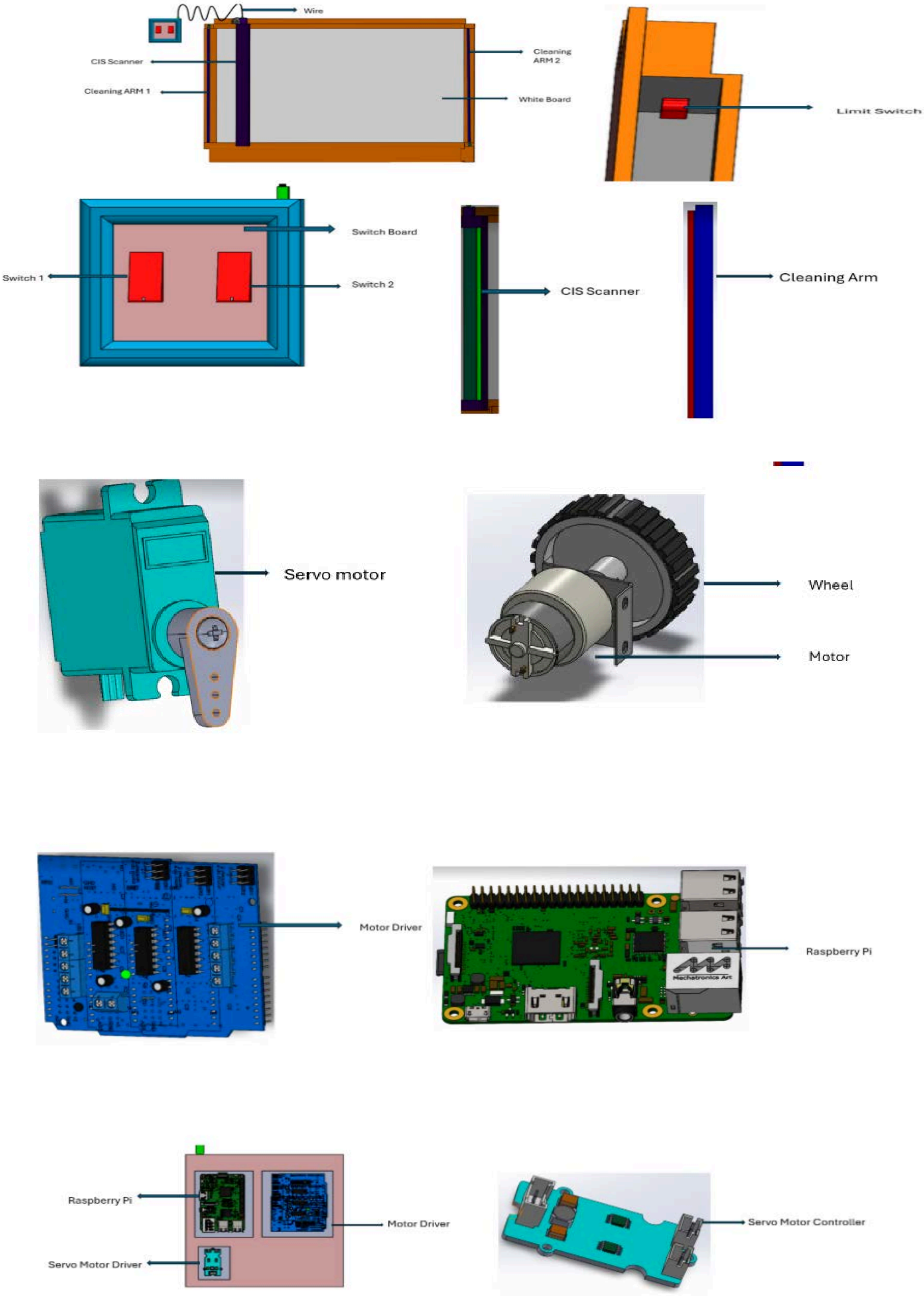


Figure 7: 3D modelling of Design 2

4.2.2 Simulation (Using Proteus)

Design Approach 1:

Here, our trained model will detect the dividing line's distance from both the left and right corner if it exists. Then the controller will measure the time needed for both frames to reach the divider and the motor's runtime values will be updated accordingly. If there is no dividing line, then both the values will be identical and any frame can be used. As per calculations(given below), the frame needs 8.95 seconds to reach another corner of the board. For example, we assumed that there is a dividing line between the frames and both the frames need 4.5 seconds to reach the dividing line or the end. First 4.5 seconds, forward motor will rotate and the next 4.5 seconds, backward motor will rotate.

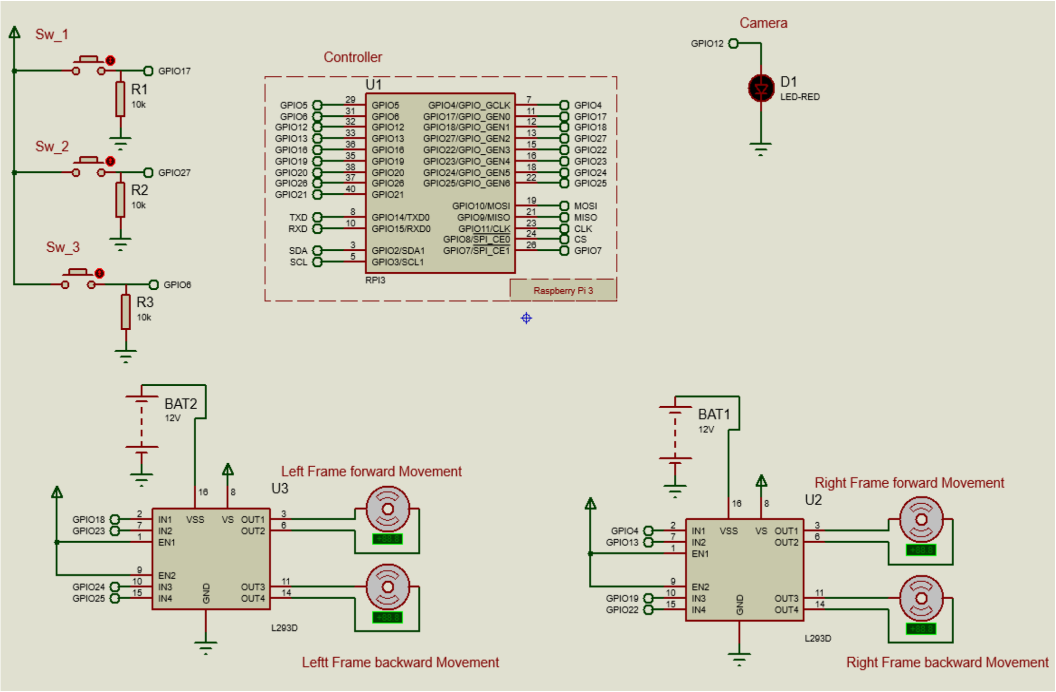


Figure 8: Visualization of design approach 1 using proteus

In this design approach, the LED is symbolized as a Camera, which is mounted onto the frame with a holder.

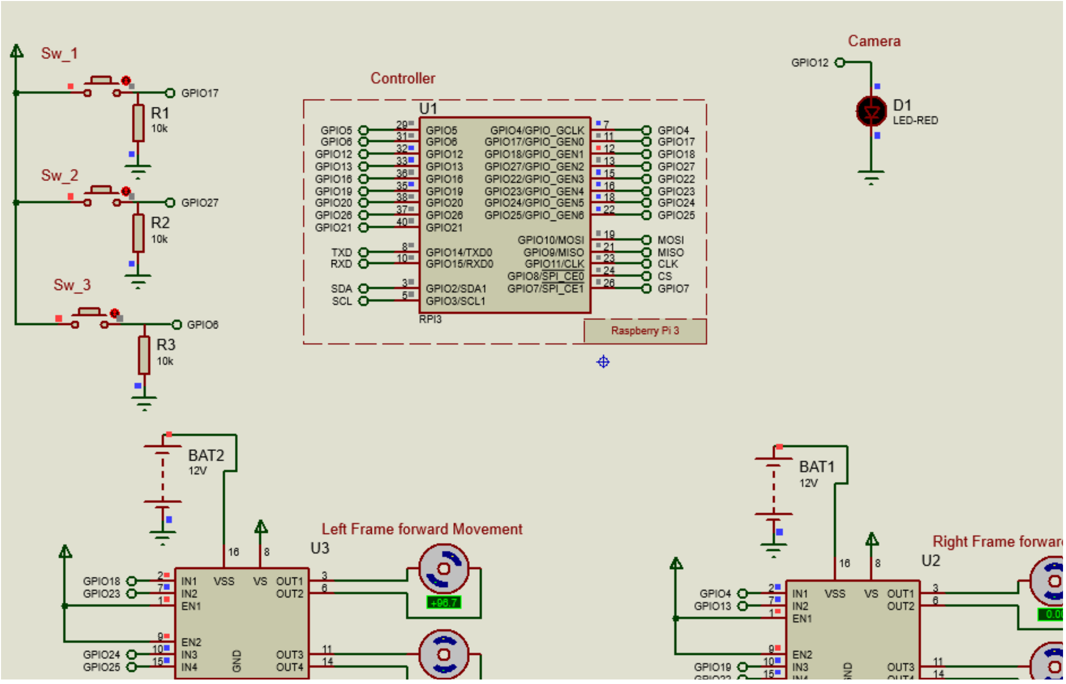


Figure 9: Left frame moving to right

After pressing switch 1, the forward motor of the left frame is rotating clockwise.

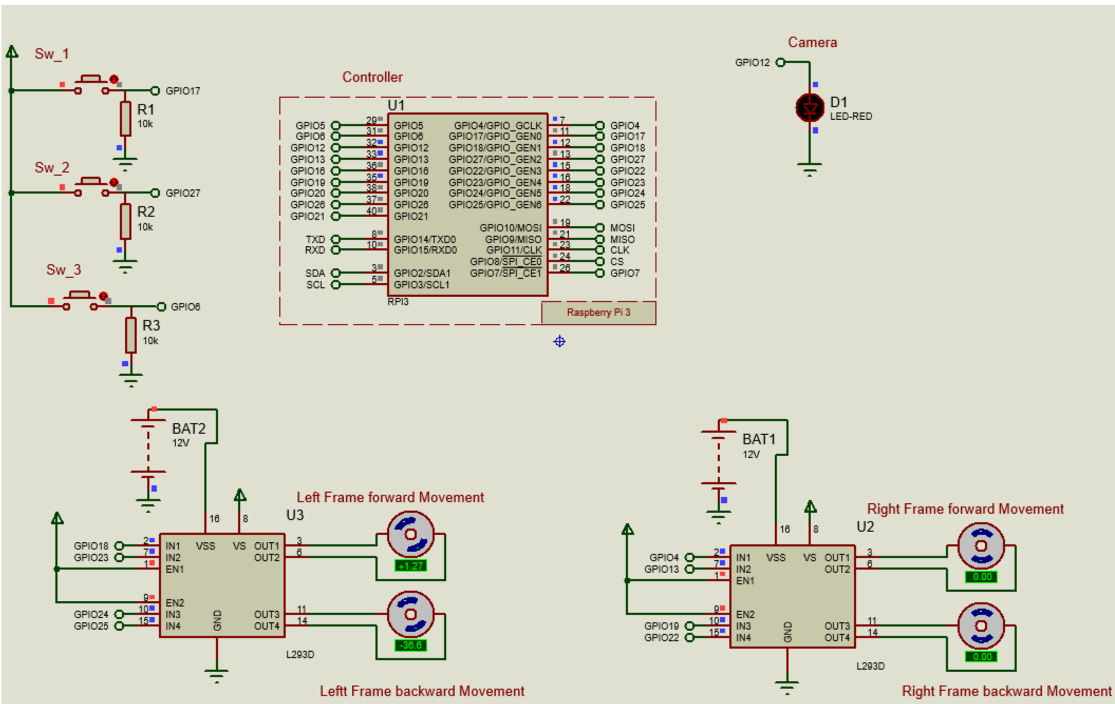


Figure 10: Left frame returning to left

The left frame reached its desired position and now the backward motor starts rotating anticlockwise while the forward motor stops rotating.

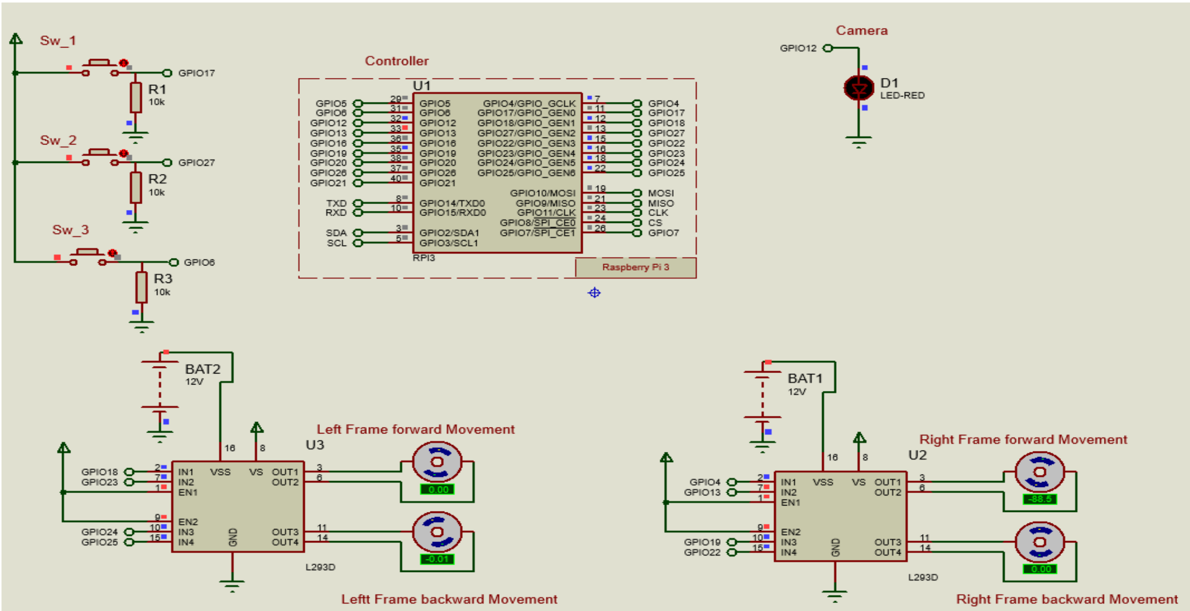


Figure 11: Right frame moving to left

After pressing switch 2, the forward motor of the left frame is rotating anticlockwise.

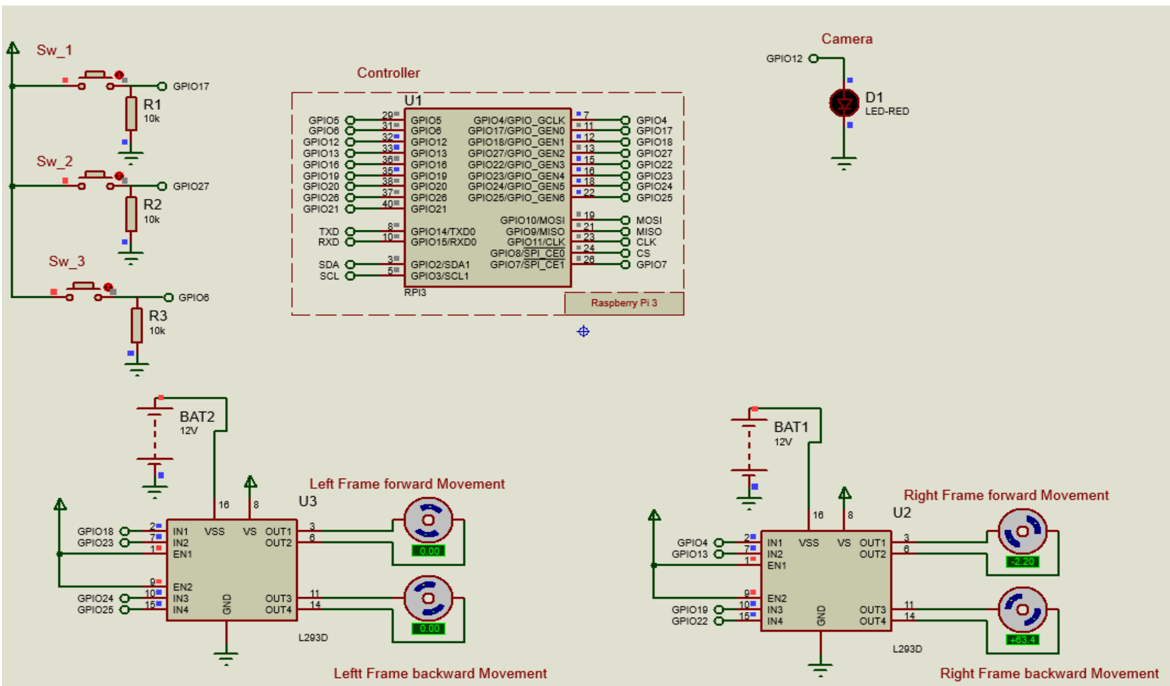


Figure 12: Right frame returning to right

The right frame reached its desired position and now the backward motor starts rotating clockwise while the forward motor stops rotating.

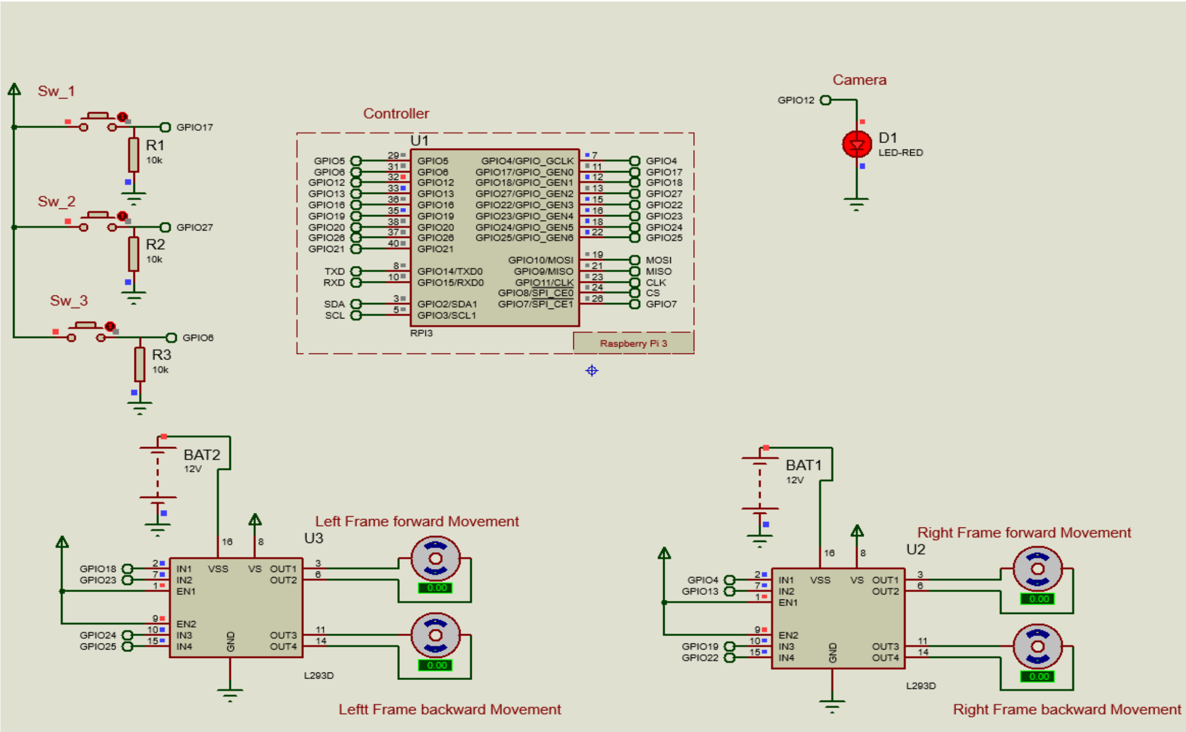


Figure 13: Camera operation

After pressing switch 3, the camera is capturing the whiteboard content.

Design Approach 2:

Here, the CIS scanner is mounted onto a frame and for cleaning purposes, 2 cleaner arms are used which can rotate up to 90 degree.There are two limit switches at both corners. As per our calculations(given below), the frame needs 12 seconds to reach the right corner of the border and the cleaner arms need 6 seconds to do its operation.

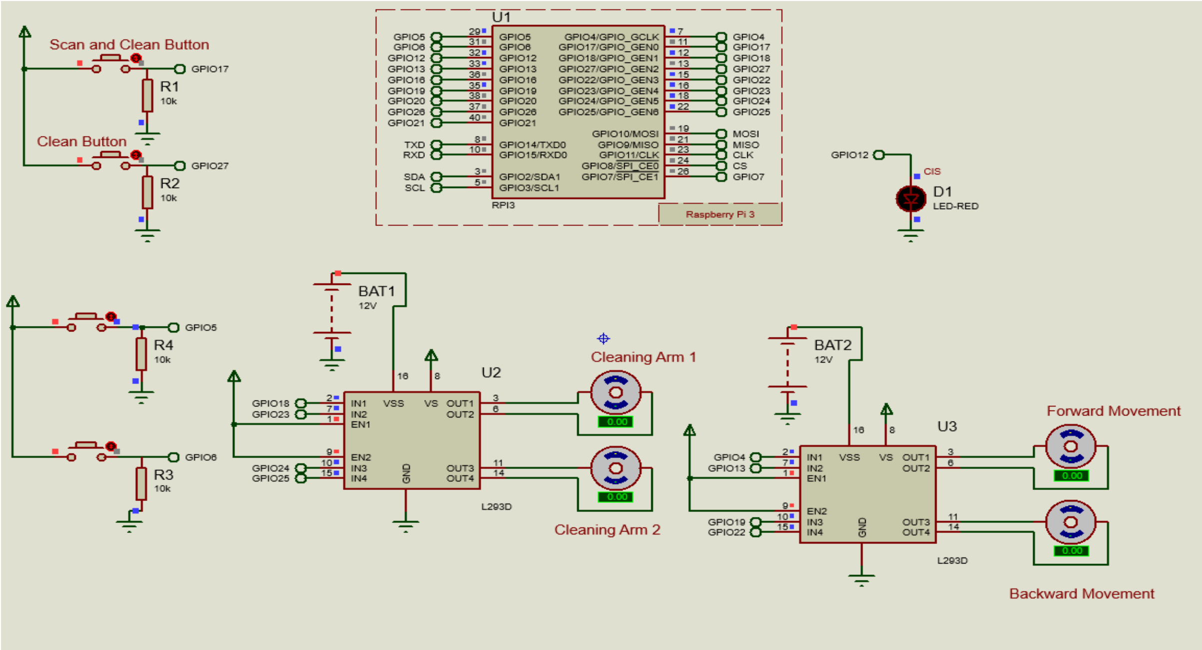


Figure 14: Visualization of design approach 2 using proteus

In this design approach, the LED is symbolized as CIS scanner, which is mounted onto a frame.

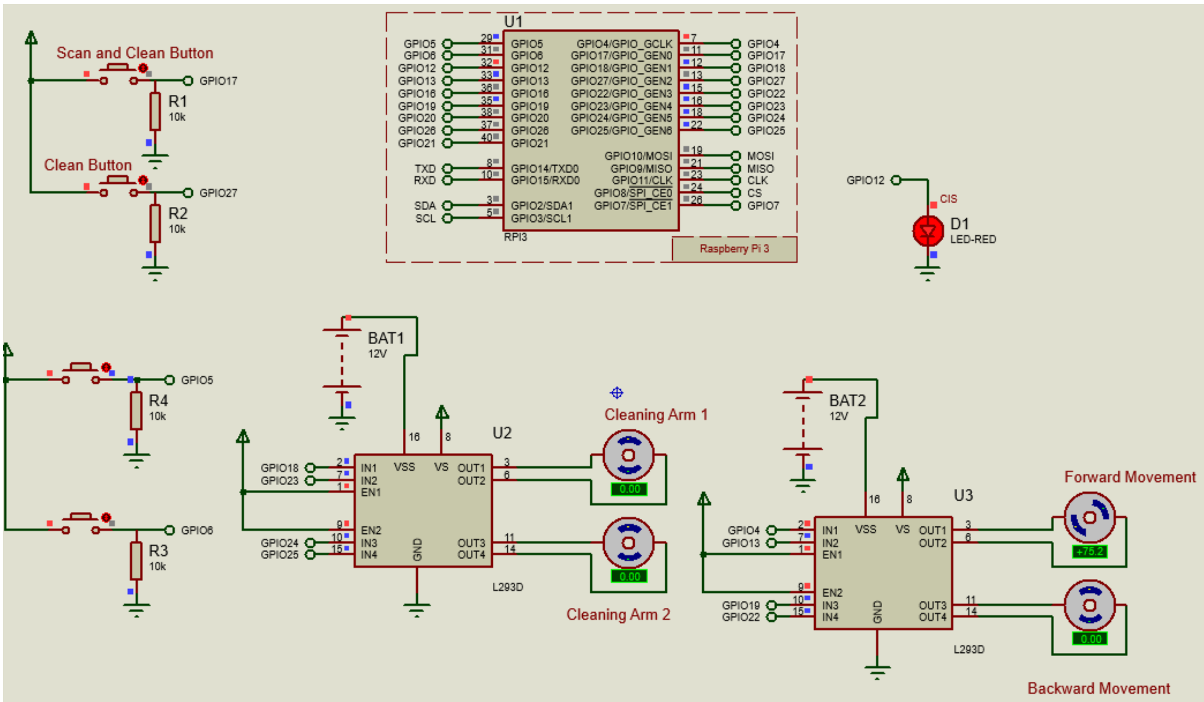


Figure 15: CIS scanner and forward motor of the frame are activated

When the user presses the input button 1, CIS scanner mounted onto a frame starts moving to the right.

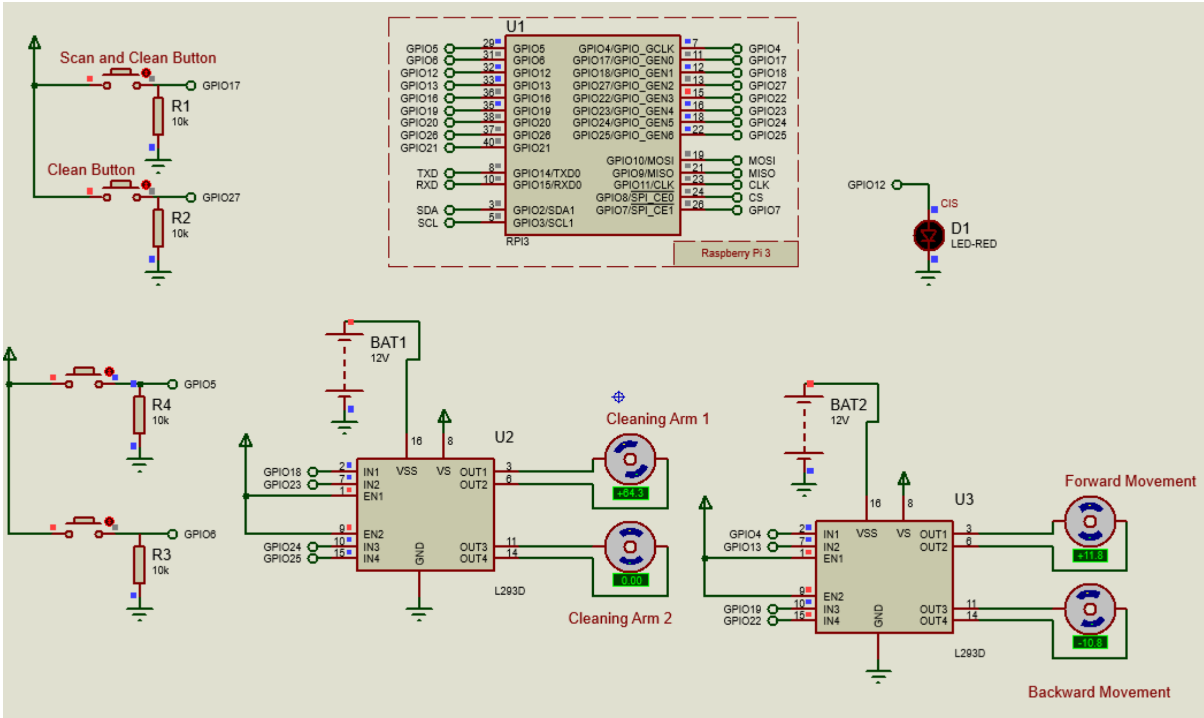


Figure 16: CIS operation stopped and cleaning arm1 starts working

When the frame reaches the right corner, Limit Switch 1 is activated, causing the CIS scanner to stop its operation. After that, Cleaning Arm 1 is activated; it first rotates 90 degrees counterclockwise, then reverses direction and rotates clockwise until it reaches its previous position.

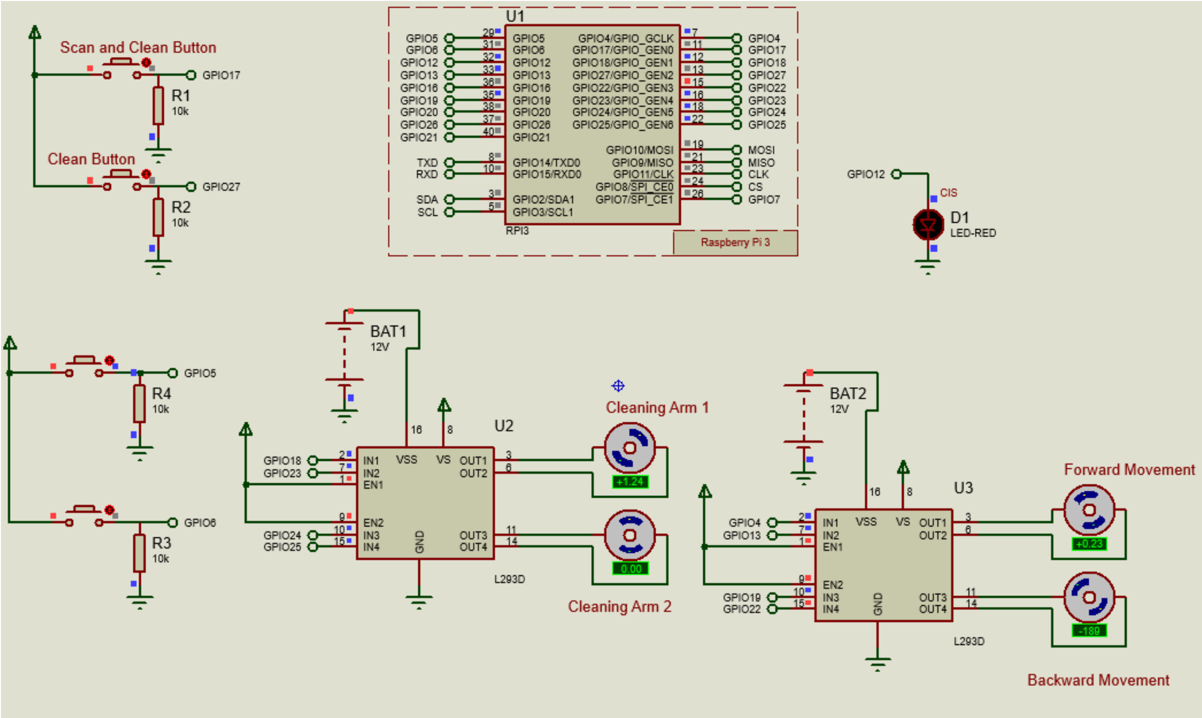


Figure 17: Frame starts moving to the left

The frame starts moving to the left though CIS operation is deactivated.

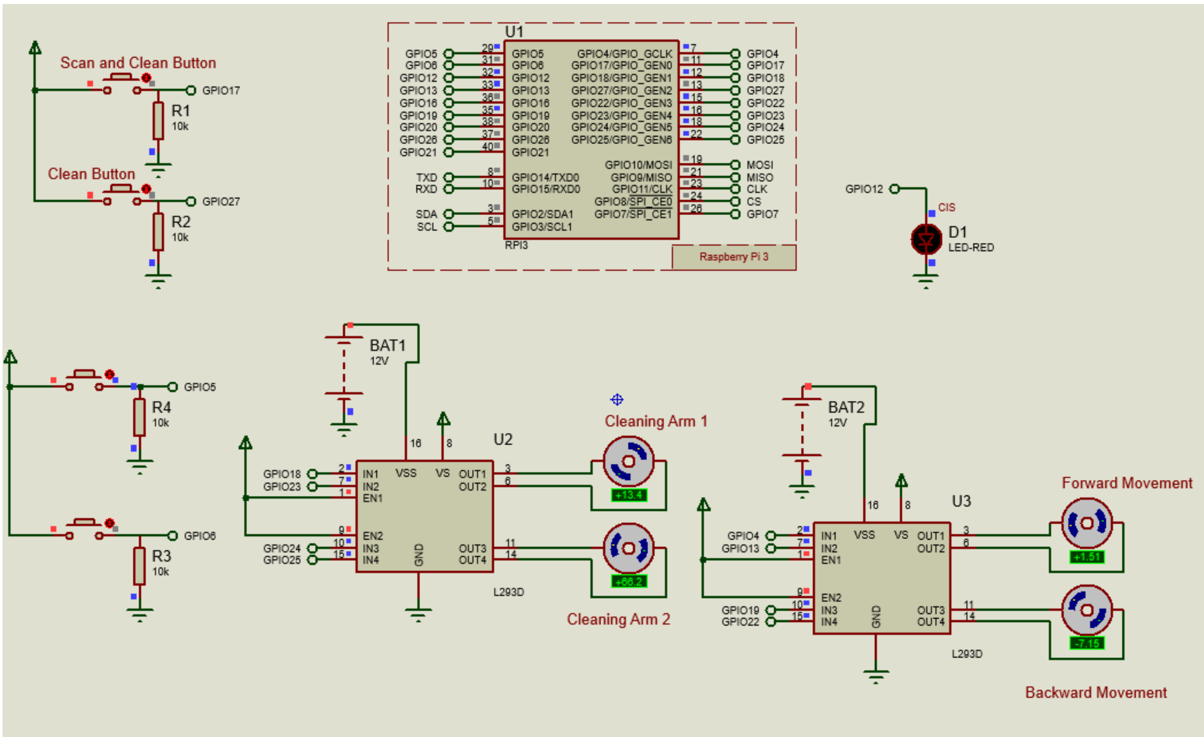


Figure 18: Cleaning arm2 starts working

When the frame reaches the left corner, limit switch 2 is activated and the frame stops moving and cleaning arm 2 is activated.

The switch 2 operation is for the cleaning purpose only, here almost everything is similar to switch 1 operation, just at the beginning CIS operation remains deactivated.

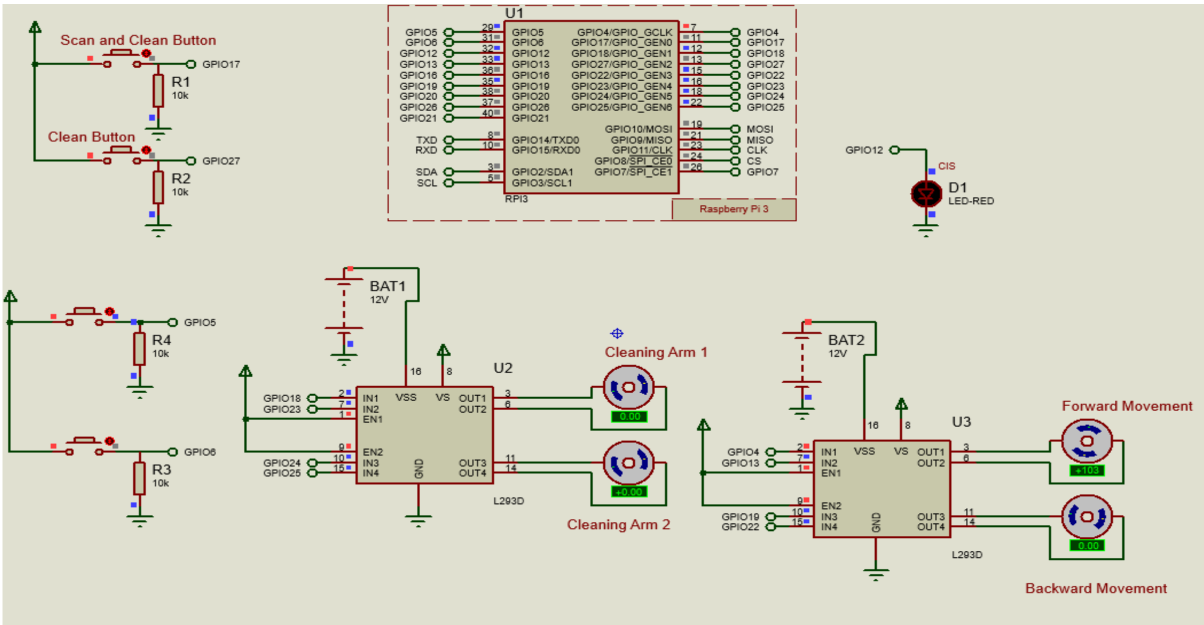


Figure 19: Frame moving to the right

After pressing switch 2, the frame starts moving to the right but CIS operation is deactivated.

The remaining operations are identical to the switch 1 operations.3 wireless communications are observed in these designs-Switch to controller, controller to camera/CIS, camera/CIS to pc.Their schematic diagrams are-

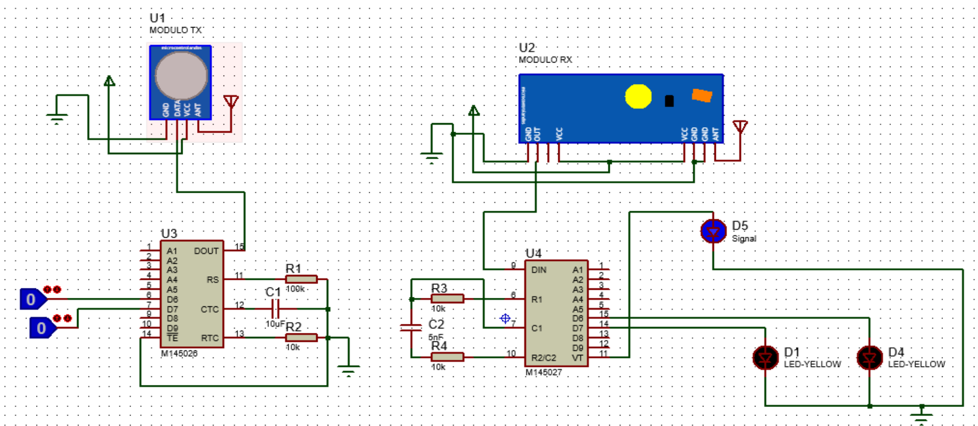


Figure 20: Switch to Controller

To demonstrate that the controller is getting the signal from the input, LED's are used.

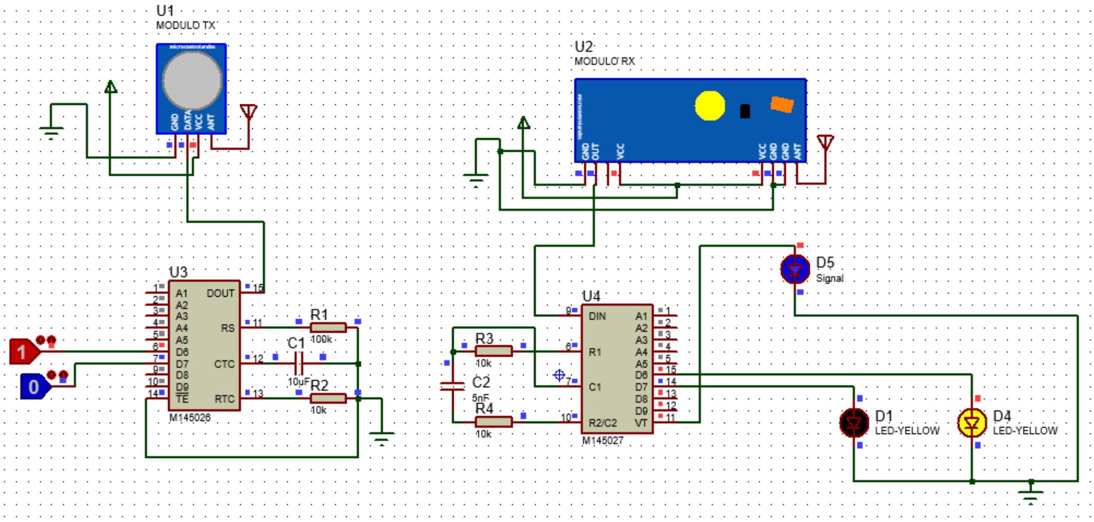


Figure 21:Switch 1 signal transmitted to the controller

When Switch 1 is pressed, the transmitting module (TX) detects the signal and sends it to the receiving module (RX). Once the signal is received, the Raspberry Pi reads the input from the RX module. LED corresponding to switch 1 is on.

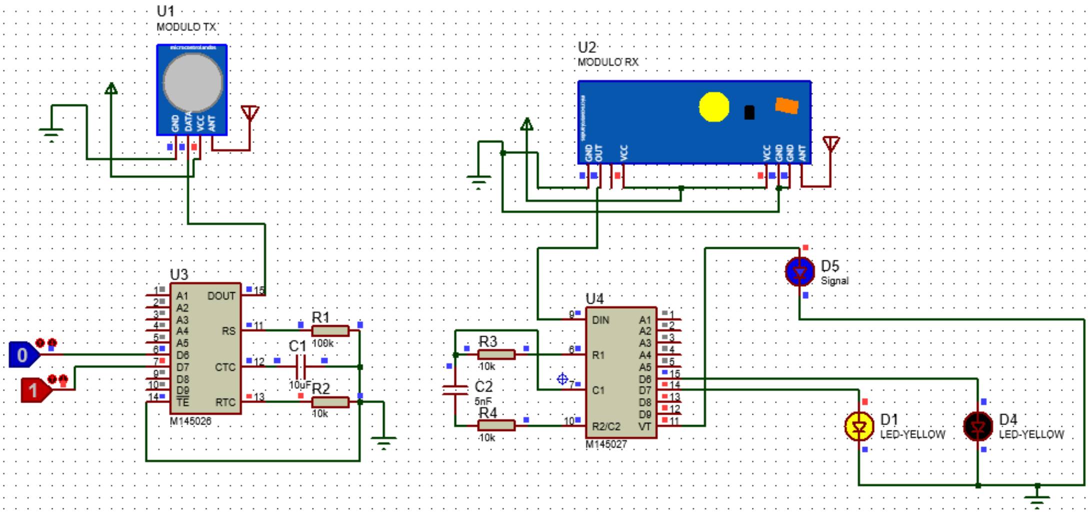


Figure 22: Switch 2 signal transmitted to the controller

In the same way, when Switch 2 is pressed, the transmitting module (TX) detects the signal and sends it to the receiving module (RX). Once the signal is received, the Raspberry Pi reads the input from the RX module. LED corresponding to switch 2 is on.

Controller to camera/CIS:

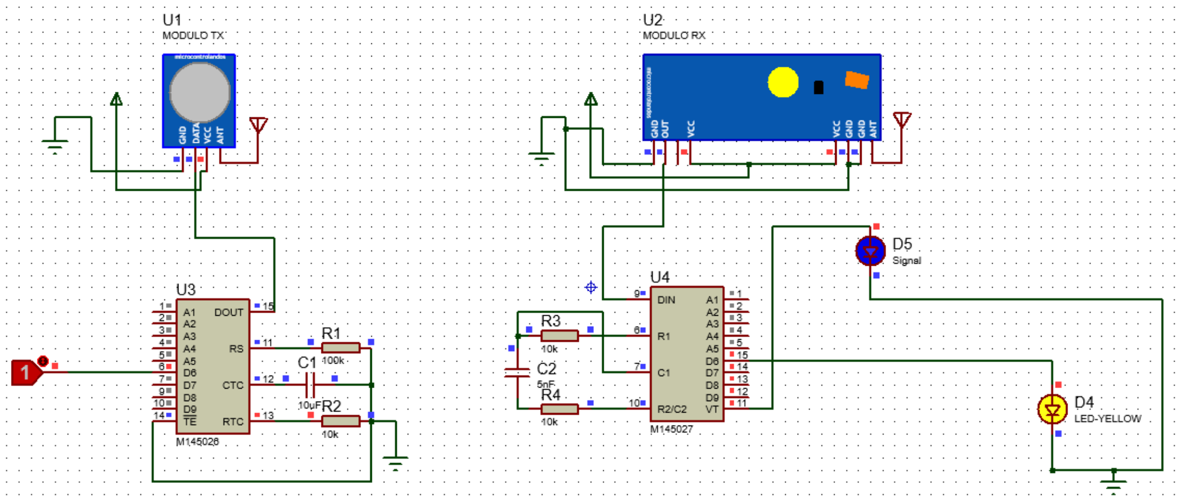


Figure 23: Controller to Camera/CIS

Here. LED(D4) is representing that the camera?CIS is getting the signal from the controller.

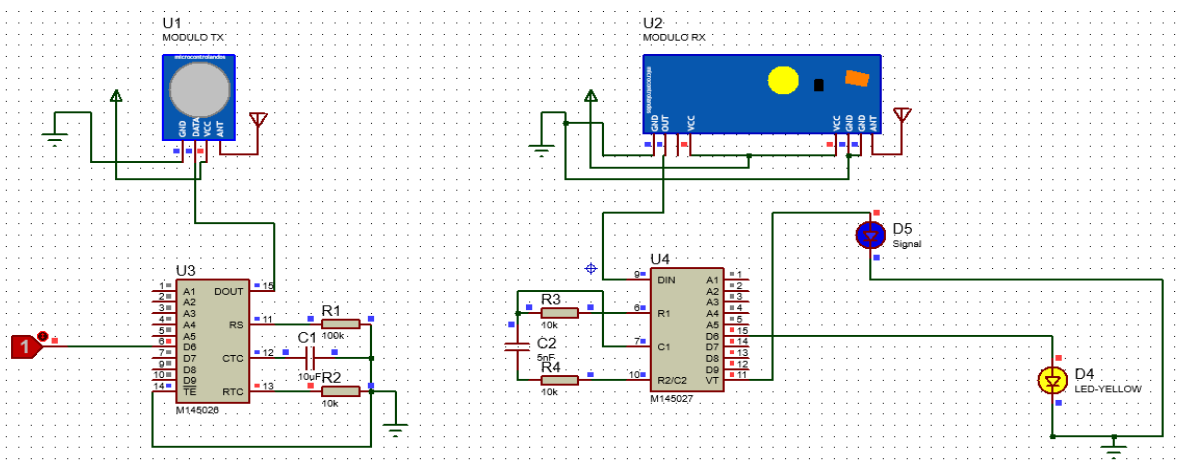


Figure 24: Controller transmits signal to operate camera/CIS

When the controller receives the signal of capturing from the input switch, it sends the signal to the camera/CIS for capturing the content and camera/CIS starts its operation.

Camera/CIS to pc:

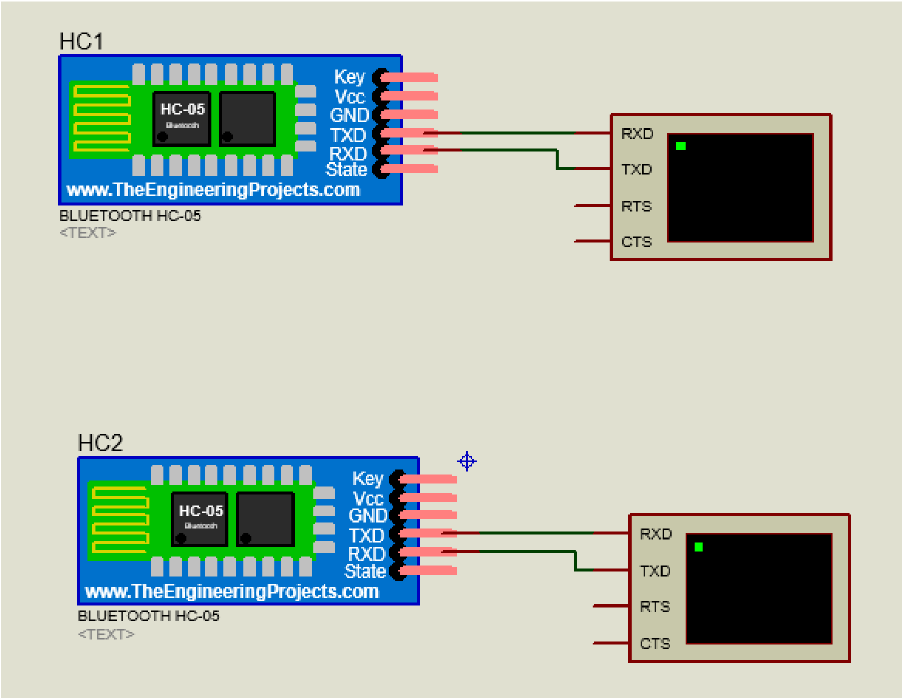


Figure 25: Camera/CIS to pc

Through the bluetooth module, data of the camera/CIS is transmitted to the pc.

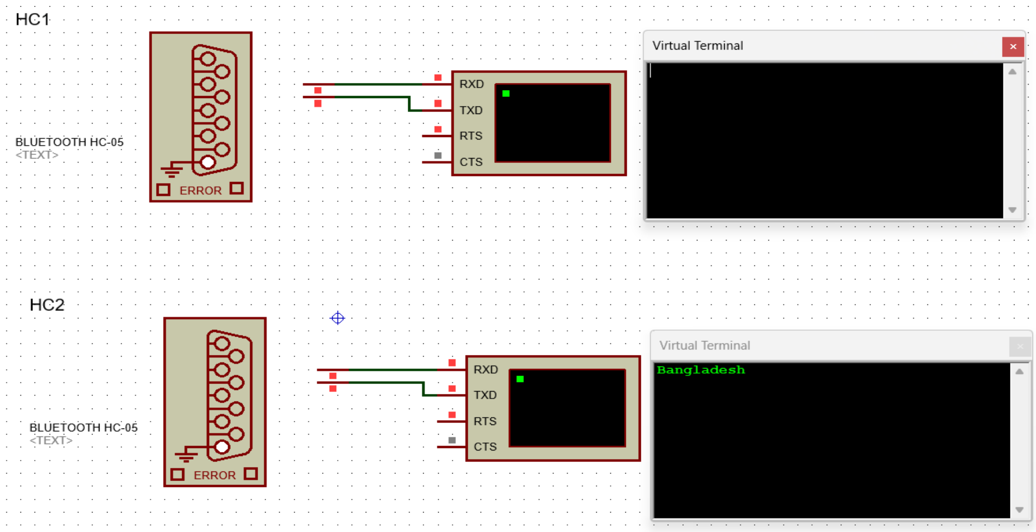


Figure 26: Camera/CIS data showing at pc terminal

Here, it can be seen that what data is given in the first terminal is showing in the second terminal.

CIS Scanner(Used in Design approach 2) :

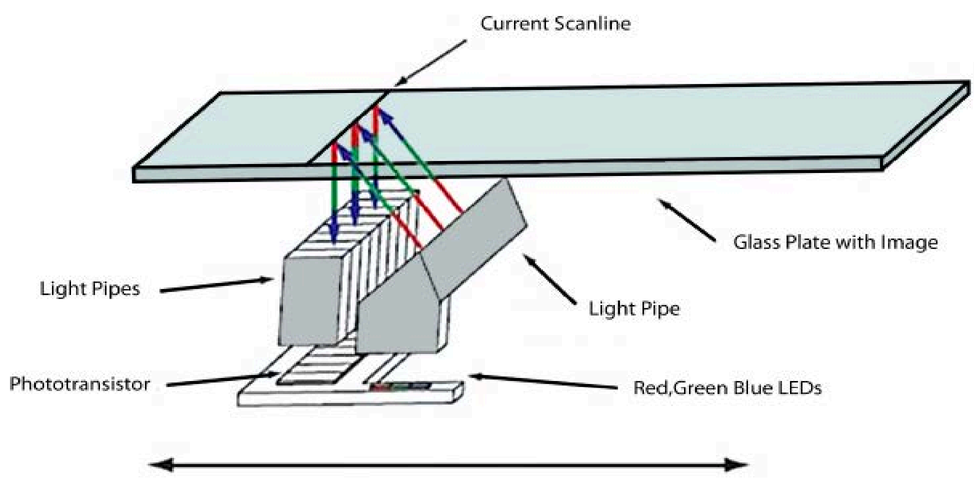


Figure 27: Contact Image Sensor schematic plan.

A tiny Contact Image Sensor (CIS) scanner schematic shows red, green, and blue LEDs illuminating the page. A lens captures reflected light and sends it to an image sensor array beneath the document. The sensor records the light intensity for each color, scanning one line at a time with R, G, or B, while interpolating the other two color components from nearby lines, similar to camera color demosaicing.

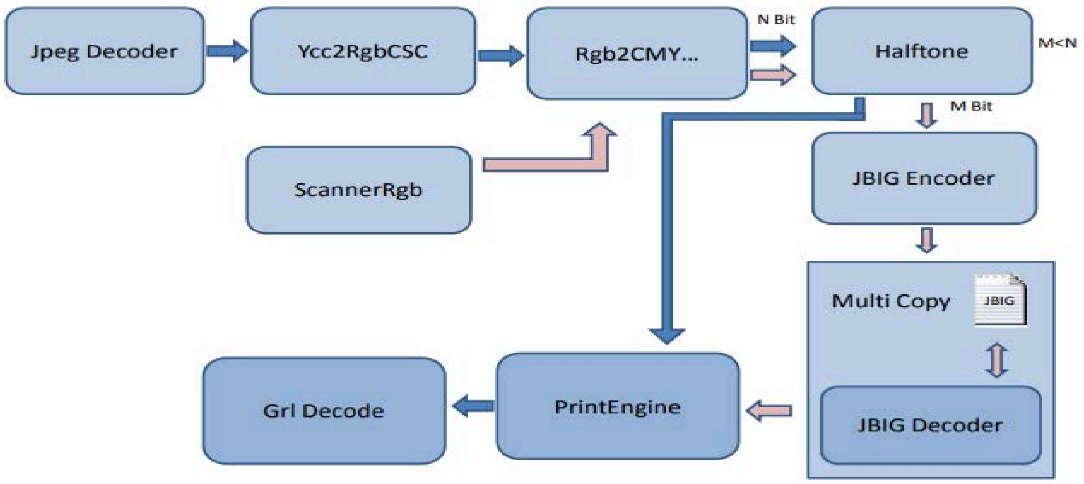


Figure 28: Working Principle of CIS

A tiny Contact Image Sensor (CIS) scanner schematic shows red, green, and blue LEDs illuminating the page. A lens captures reflected light and sends it to an image sensor array beneath the document. The sensor records the light intensity for each color, scanning one line at a time with R, G, or B, while interpolating the other two color components from nearby lines, similar to camera color demosaicing.

4.2.3 Simulation(using Simulink):

Design Approach 1-

X=>Time

Y=>Horizontal position

Y=0 =>Left corner of the whiteboard

Y=1=>Right corner of the whiteboard

For capture operation-Y=0=>Low signal, Y=1=>High signal

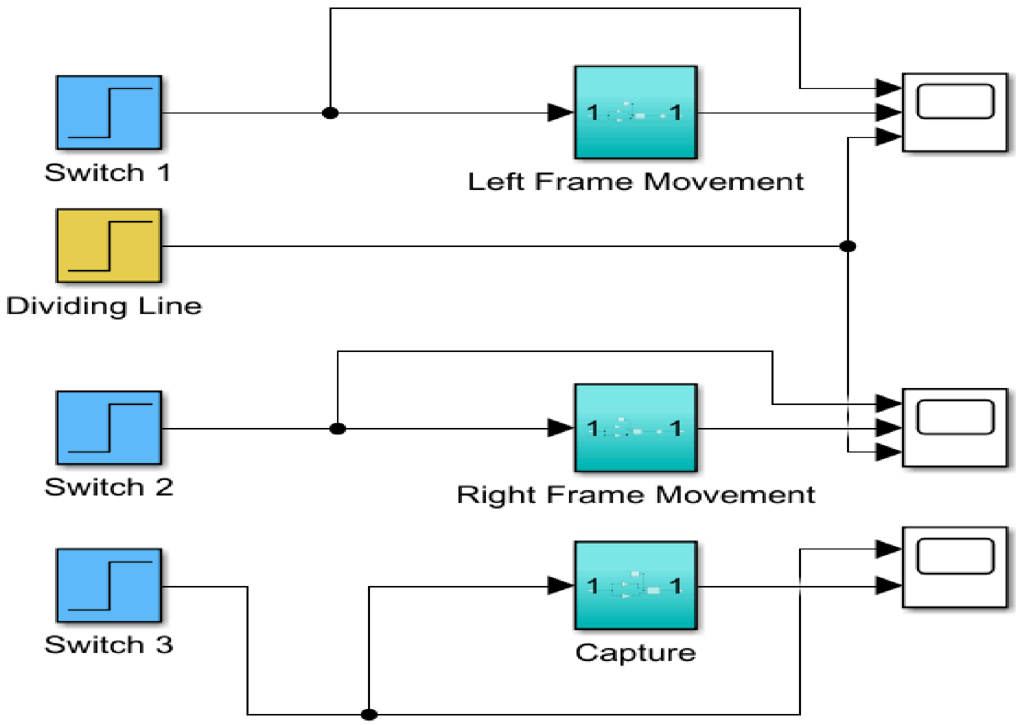


Figure 29: SImulation of Design Approach 1

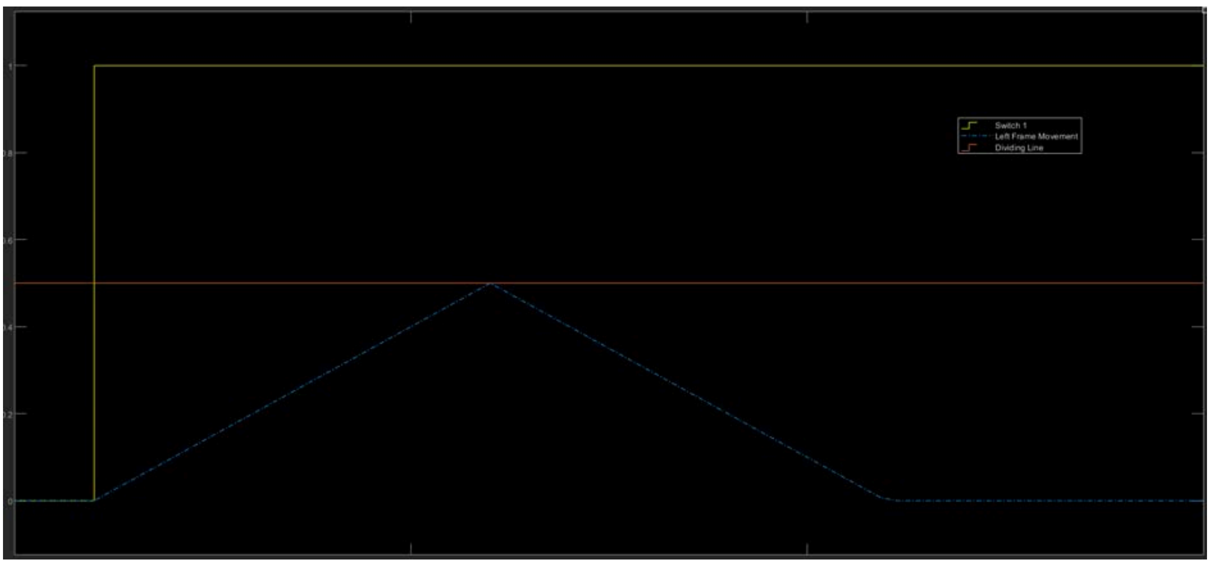


Figure 30: Switch 1 operation

Here, when the switch 1 is on, the left frame starts moving to the right until it reaches the dividing line. After that, it returns to its previous position.

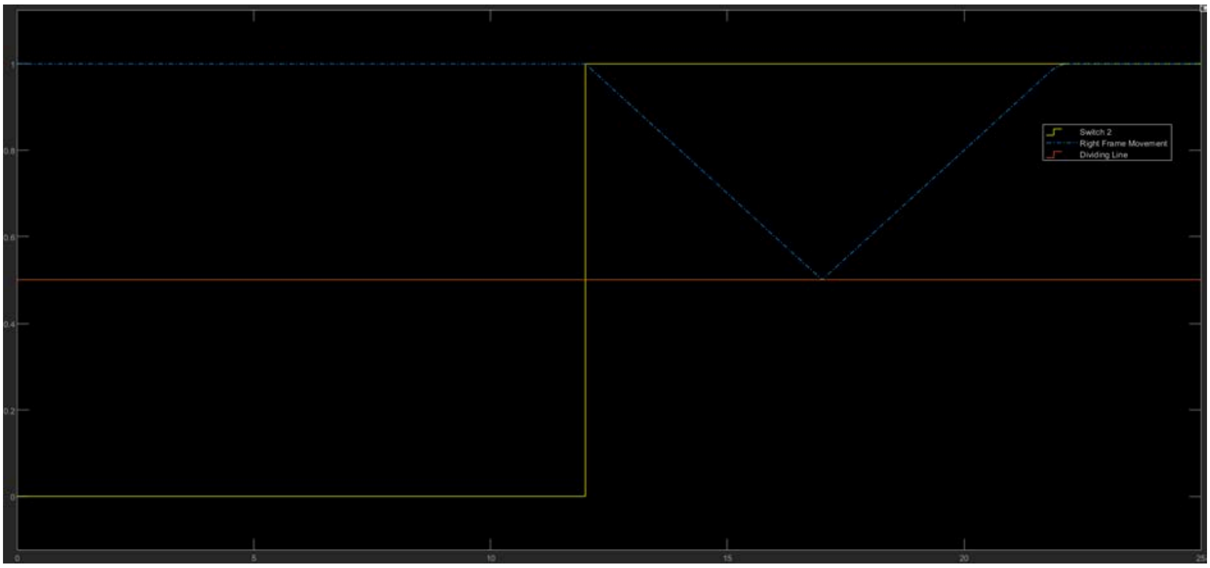


Figure 31: Switch 2 operation

Here, when the switch 2 is on, the right frame starts moving to the left until it reaches the dividing line. After that, it returns to its previous position.

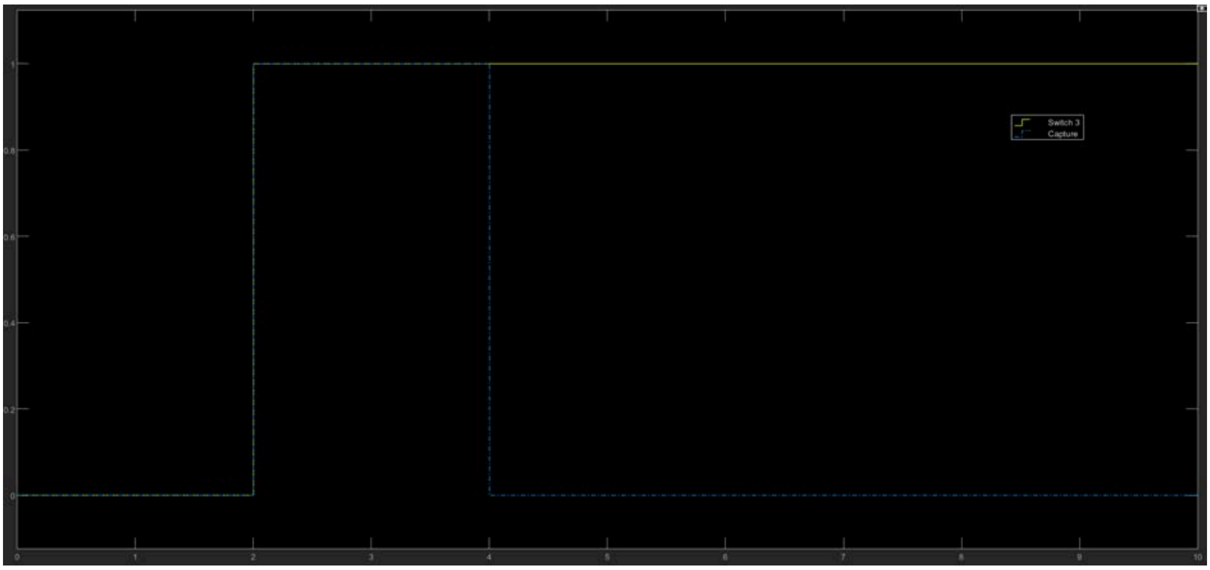


Figure 32: Switch 3 operation

When switch 3 is on, the camera captures the whiteboard content.

Design Approach 2-

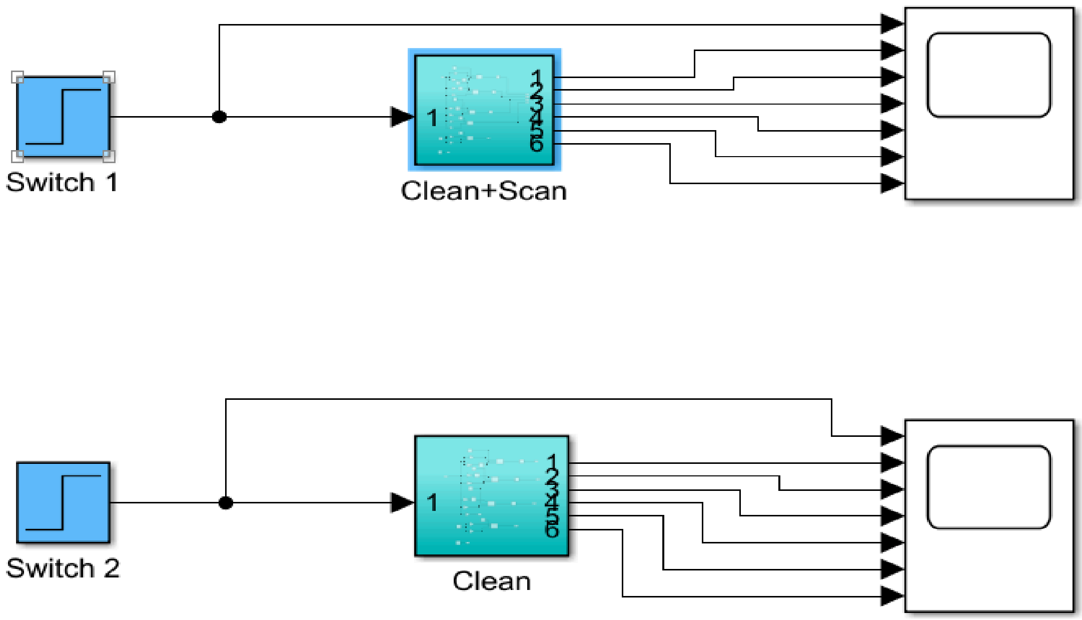


Figure 33: SImulation of Design Approach 2

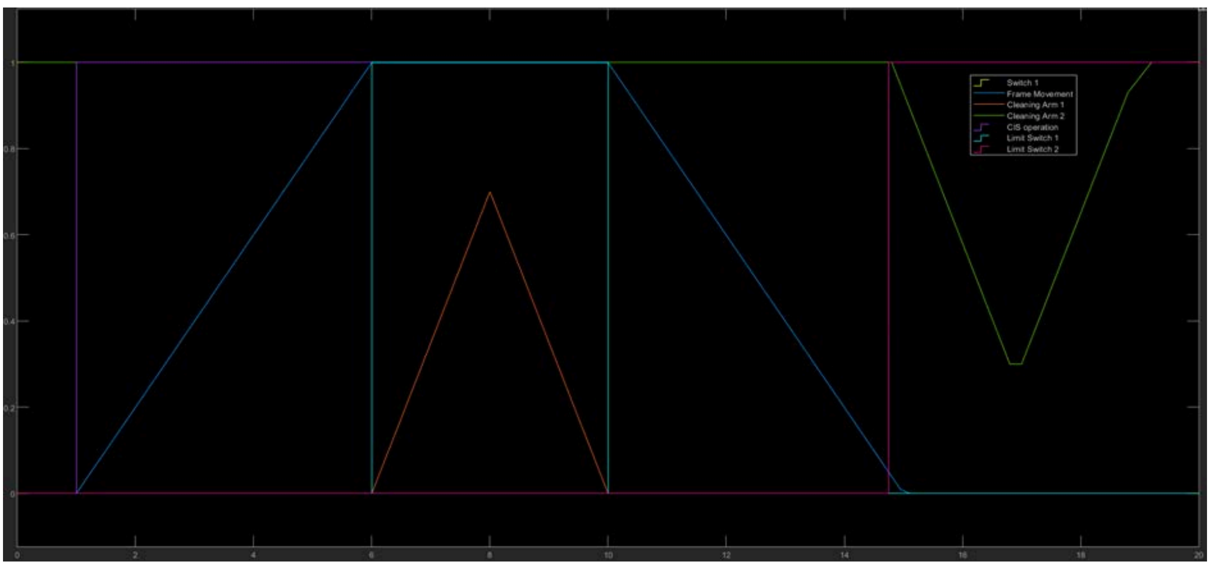


Figure 34: Switch 1 operation

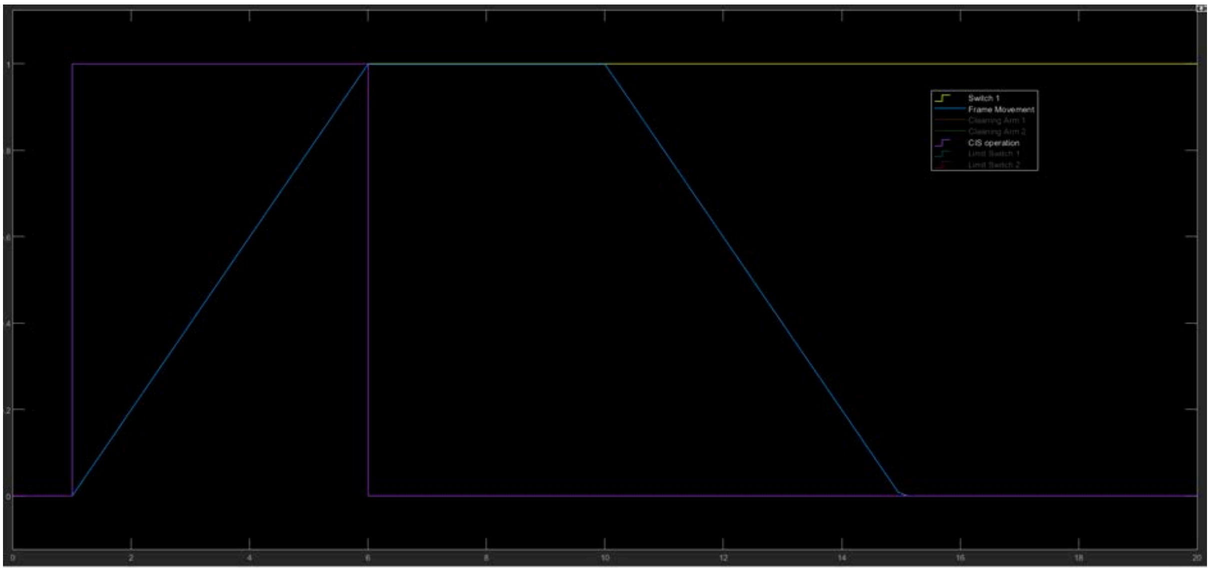


Figure 35: CIS operation on and frame moving to the right

When switch 1 is on, CIS frame moves to the right, scanning the whiteboard content.

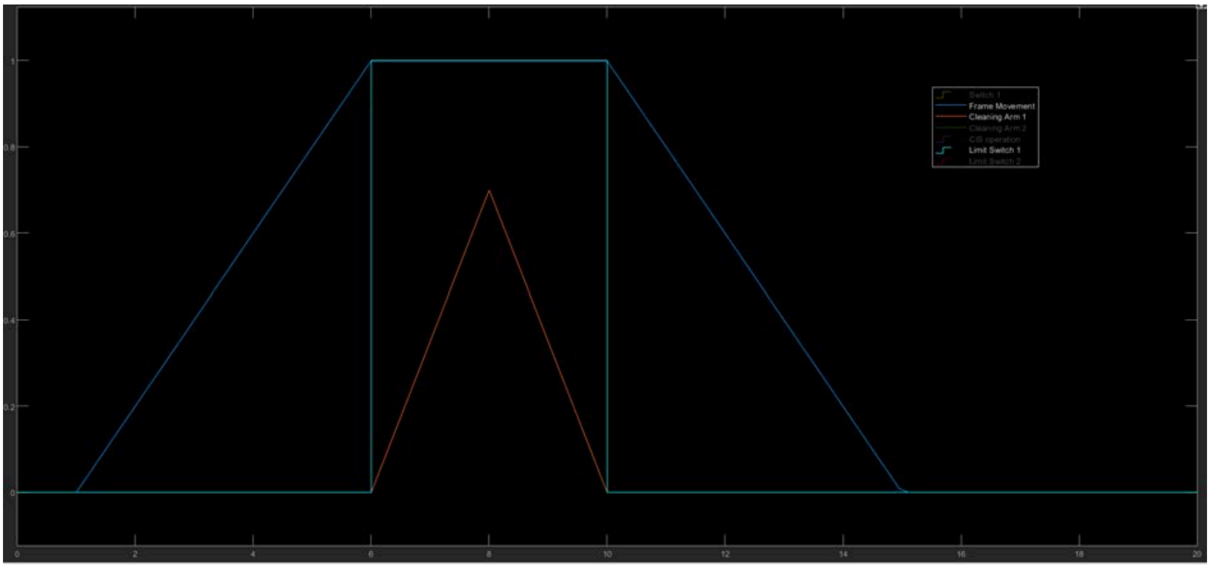


Figure 36: Limit switch 1 and cleaning arm 1 activated

When the frame reaches the right corner, limit switch 1 is activated which then signals the cleaning arm 1 to activate and this arm's bottom part rotates 90 degree anticlockwise and then 90 degree clockwise.Afterwards, frame again starts moving to the left.

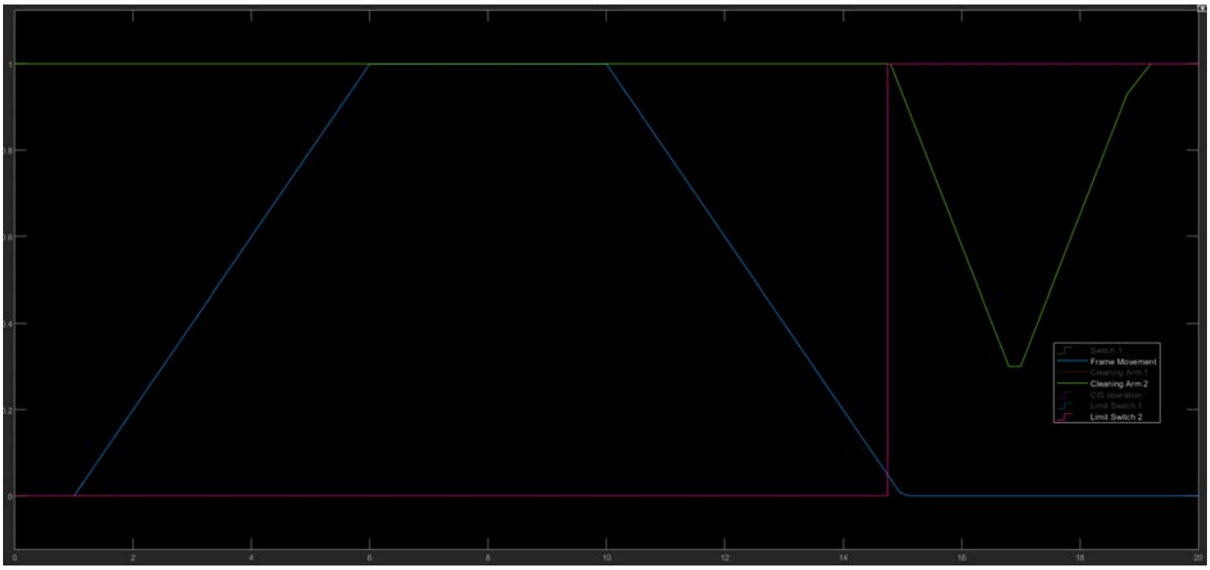


Figure 37:Limit switch 2 and cleaning arm 2 activated

When the frame reaches the left corner, limit switch 2 is activated which then signals the cleaning arm 2 to activate and this arm's top part rotates 90 degree anticlockwise and then 90 degree clockwise.

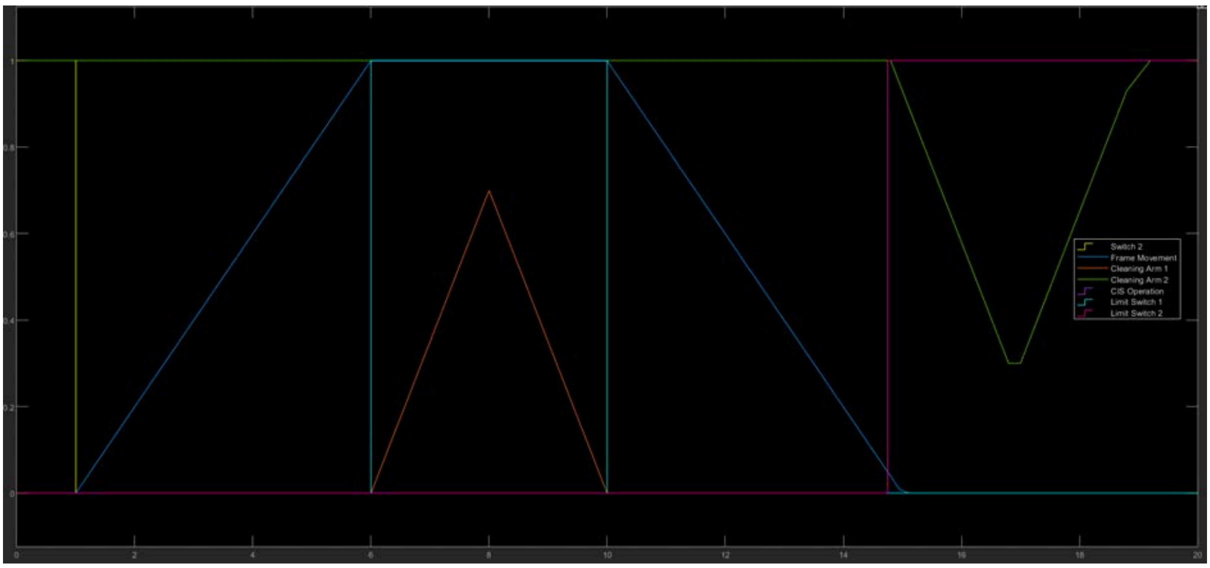


Figure 38: Switch 2 operation

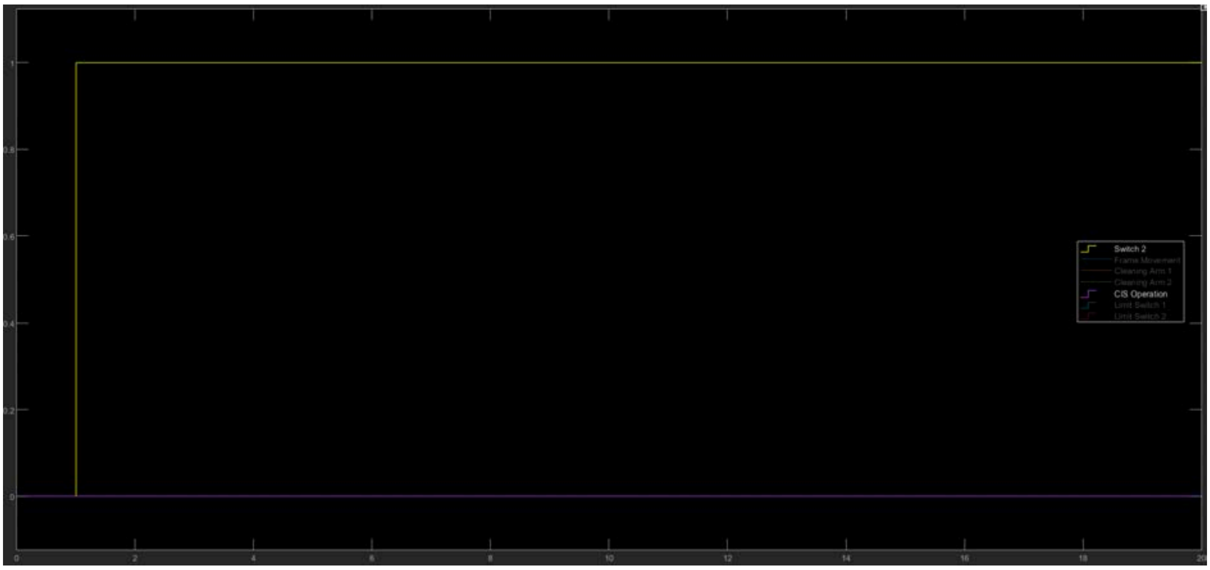


Figure 39: CIS operation always deactivated

In switch 2 operation, the CIS operation remains deactivated the whole time though CIS frame moves in the same way as switch 1 operation to avoid collision with the cleaning arms. Except the scanning part, all the procedures are identical to the switch 1 operation here.

Approach 1 (camera + ESP32 based) and Approach 2 (CIS based) provide a means of cleaning the whiteboard while producing LaTeX notes but there are many different areas of distinction between the two approaches - cost, efficiency, usability, maintainability, and custom features.

Approach 1 uses common hardware (ESP32, motors, sensors, camera) and provides flexible cleaning with magnets and reed sensors, along with additional features like AI chatbot and Overleaf integration. It is low-cost, sustainable, and easy to maintain.

Approach 2 also provides LaTeX conversion but depends on specialized CIS modules, which makes it more expensive, less flexible in cleaning, and harder to maintain.

Table 5: Qualitative Data Comparison Between the Two Designs

Qualitative Factor	**Approach 1: Camera + ESP32**	**Approach 2: CIS-Based**
Affordability	Very cost-effective (12,500 BDT), suitable for wide classroom adoption.	Expensive (30,210 BDT), limiting large-scale use.
Ease of Use	Simple wired/wireless switches for cleaning and capture; user-friendly app and website.	More complicated, requires careful alignment and step-by-step operation.
Flexibility	Offers both full and partial cleaning (via magnets + reed sensors); works with different board sizes.	Limited cleaning patterns; may miss edges or overlap depending on board size.
Smart Features	Smart LaTeX note generation, Overleaf integration, AI chatbot for student interaction.	LaTeX notes supported, but without chatbot or Overleaf integration.
Reliability	Uses standard, robust parts (ESP32, motors, sensors) → easy replacement and maintenance.	More prone to errors due to sensitive CIS alignment and calibration needs.
Sustainability	Modular, upgradable, and environmentally friendly with reusable parts.	Less sustainable; replacement of CIS modules is costly and not widely available.
Practicality	Highly practical for real classrooms due to low cost, flexibility, and modern features.	Less practical due to high cost, complexity, and limited flexibility.

4.3 Identify optimal design approach

To quantitatively evaluate both designs, a weighted point system out of 10 was developed, with accuracy assigned the highest weight due to its critical role in classroom functionality. Specifically, 4 points were allocated to accuracy, 2 to cost, and 1 to each of the remaining four features, based on performance or availability.

Table 6: Weightage Table for the Two Designs

Parameter	Weightage (%)	Design Approach 1 (Camera-Based)	Design Approach 2 (CIS-Based)
Cost	20%	1.2	0.8
Accuracy of Cleaning / Capture	25%	1.03	1.47
Speed of Operation	15%	0.79	0.71
Complexity of Implementation	10%	0.58	0.42
Maintenance / Durability	15%	0.79	0.71
Scalability / Flexibility	15%	0.83	0.67
Total Score	100%	5.22	4.78

1. Cost

Allocated Weight: 20% (2.0 points)

Design 1 (Camera-Based): [18 / (18 + 12)] × 2.0 = 1.20

Rationale: Cheaper due to the use of standard components.

Design 2 (CIS-Based): [12 / (18 + 12)] × 2.0 = 0.80

Rationale: More expensive because of the specialized CIS sensors.

2. Accuracy of Cleaning / Capture

Allocated Weight: 25% (2.5 points)

Design 1 (Camera-Based): [18 / (18 + 22)] × 2.5 = 1.03

Design 2 (CIS-Based): [22 / (18 + 22)] × 2.5 = 1.47

Rationale: CIS sensors are more precise, giving Design 2 a significant advantage in accuracy.

3. Speed of Operation

Allocated Weight: 15% (1.5 points)

Design 1 (Camera-Based): [14 / (14 + 12)] × 1.5 = 0.79

Rationale: Camera-based processing is faster if the software is optimized.

Design 2 (CIS-Based): [12 / (14 + 12)] × 1.5 = 0.71

Rationale: CIS may require multiple scans, which can slow down the operation.

4. Complexity of Implementation

Allocated Weight: 10% (1.0 point)

Design 1 (Camera-Based): [14 / (14 + 10)] × 1.0 = 0.58

Rationale: Camera-based systems are easier to integrate.

Design 2 (CIS-Based): [10 / (14 + 10)] × 1.0 = 0.42

Rationale: CIS requires precise mechanical alignment, increasing complexity.

5. Maintenance / Durability

Allocated Weight: 15% (1.5 points)

Design 1 (Camera-Based): [13 / (13 + 12)] × 1.5 = 0.79

Rationale: Standard camera and motor setup is generally easier to maintain.

Design 2 (CIS-Based): [12 / (13 + 12)] × 1.5 = 0.71

Rationale: CIS sensors are more delicate and can be more challenging to maintain.

6. Scalability / Flexibility

Allocated Weight: 15% (1.5 points)

Design 1 (Camera-Based): [15 / (15 + 12)] × 1.5 = 0.83

Rationale: A camera system can be more easily adapted to multiple board sizes.

Design 2 (CIS-Based): [12 / (15 + 12)] × 1.5 = 0.67

Rationale: CIS systems are often more limited in the size of objects they can handle.

Total Score Summary

- Design 1 (Camera-Based): 1.20 + 1.03 + 0.79 + 0.58 + 0.79 + 0.83 = 5.22
- Design 2 (CIS-Based): 0.80 + 1.47 + 0.71 + 0.42 + 0.71 + 0.67 = 4.78

Conclusion: Design 1 has an overall higher rating as a result of its relatively lower cost, ability to operate quickly, ease of implementation, and ability to scale - Design 2 has a marginally better accuracy rating, but is at a higher price point and much harder to maintain.

4.4 Performance evaluation of developed solution

The developed smart whiteboard system was assessed against several important criteria, including accuracy, cost, speed, complexity, maintenance, power draw, and scalability. The camera-based prototype proved to be very accurate in detecting the ink to be cleaned, meaning content on the whiteboard can be reliably digitized. The camera-based approach is very cost-effective. Cameras, microcontrollers, and motors utilized in the prototype are typical components that are affordable and easy to assemble. The camera-based technology also proved itself to function adequately; post-processing was sufficiently fast, power draw was low, and maintenance was minimal. Because of its design, it is flexible and capable of scaling for different white-board sizes without changing hardware. There are also advantages to software and datasets that are open source as this makes coding and testing significantly easier. Overall, the camera-based approach to automatic cleaning of whiteboards is a practical, robust and reliable solution, and can be further enhanced in the future with upgrades to allow cloud storage of digitized content or conversion to PDFs of digitized content.

4.5 Conclusion

Based on the evaluation of the smart whiteboard system, the research and design based on the camera system is the most feasible and effective design. It has superior accuracy in both ink detection and removal. It is reasonably effective, uses standard parts, and is ultimately low cost. There are also not any major issues with the implementation, maintenance, or scale of the size of the board. Consequently, the proposed design is a feasible, adaptable, and straight forward solution in automated whiteboard cleaning and digitizing. This indicates the possibility for widespread use in practice and additional research.

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

5.1 Introduction

This chapter outlines the latest project updates, including the integration and validation of the Smart Whiteboard Management System. The product has gone through several rounds of simulation, prototyping, and iteration to create a workable, improved solution for existing problems in education. The project is based on CO8, which involves applying engineering principles to design and validate a real-world, complex solution. The system combines embedded systems, image processing, and mechanical automation, aligning with current engineering practices and tools as mentioned in CO9. The software architecture, firmware implementation, and team involvement show the necessary technical communication and professionalism highlighted in CO15.

5.2 Completion of final design

5.2.1 Design working [CO8, CO9]

The system is triggered when an educator activates the capture process by pressing a wireless switch, normally transmitting a signal to the system through ESP32 microcontrollers. The whiteboard material is documented with a camera on a moved frame, possibly hydraulic in nature, and can contain handwritten materials, equations, and other diagrams, such as circuit schematics. Processing in the format of an image and subsequent processing in the form of LaTeX-type PDF are performed to isolate the important objects, and increase them. This change is based on image processing methods through rules and an intelligent language model based on AI is implemented to read and understand the content. Then, the resulting files are made available in a web server that documents the procedure and redirects them to an online editing platform.

The process begins by having a high-resolution captured whiteboard with the camera. The image has frequent draws of perspective distortion owing to the camera inclination employed in the picture and differences in lighting all the way along the board. To solve this, the system employs a calibration procedure, during which the educator is supposed to identify four points (two on the left hand side and two on the right hand side) of the white board. These are saved and used at a later time to correct the geometrical aspect of the image. The correction is considered to be perspective warp which makes the image look like you are looking at it straight-on resulting in the formation of a rectangular image of the content. This will improve the readability process because the text and diagrams are aligned in the right place, irrespective of the physical orientation of the camera.

The image is then polished to give it an enhanced contrast and clarity; this is done after the geometric correction. The system is used to reduce the color picture into grayscale to make processing easier and a noise-reduction filter is used to blur the irregularities and maintain edges. Contrast is increased with the help of the method which results in the increase of the local differences of brightness to notice small hand writing or drawings. The image is further binarized and it becomes a black and white pose with the enlightenment of the text and diagrams that are depicted in the image forming dark objects on a light background. The process artificial limits in lines and even the tiniest specks are eliminated in the course of additional cleaning processes so that the content of the subsequent phases is clear.

The point, where this system is able to recognize and isolate diagrams of the other content is one of the main properties of it. Other images are larger and more convoluted than text, and the system knows that such images are diagrams, e.g. circuit maps or geometric shapes. It organizes the binarized image to identify the boundaries of related shapes, with more awareness of big areas that are above a minimum size threshold to prevent the occurrence of noise or erroneous marks. All these outlines are divided into regulable areas in the form of a rectangle, and the system does a check on holdover areas eliminating the most insignificant ones so as to avoid repetition of the same.

After the identification, every region of the diagram is cut out of the original image and stored as a different file which is usually named by a number . This extraction can guarantee that diagrams remain in the original format retaining the visual representation that can be lost when translated to text. The capture is repeated until all the captures have been made and a running counter can be used to give the total number of diagrams that are actually extracted through the entire session which then enables the system to handle an array of diagrams effectively.

Several pictures could be taken as one image and another helps in depicting the various conditions of the white board, as the lecture takes its course. These images are composed in one vertical image in order to develop a single image. This hybrid image forms one image at a time of all the captured images centred to align the contents well. What this gives is a long image and is essentially an overview of the given session, and is saved as a new file to be processed further.

The second step is to convert this composite image to a LaTeX document which will finally be drawn up into a PDF document. An AI tool embedded with the system through the outside of the API is used to break down the content of the image. Special instructions are provided to this tool: it is expected to turn a handwritten text into the code written in the correct format using LaTeX, to create subsections within the document, and to apply mathematical notation in equations. In the case of diagrams, it places placeholders referring to the extracted image files . The AI dissects the image contexturally, separating text and diagrams and creates a complete LaTeX document format, along with the packages, such as math and graphics, required to construct mathematical expressions and to include images.

In cases where the AI is not able to work on the picture either because of problems with the network or because of unauthorized authority, the system reverts to an elementary model. This template has a typical documentation format with data based on the number of diagrams extracted so that even without the help of AI, it is a separate file that can be used. This is later written as LaTeX and stored as a text file by which the final compilation is done.

The crawled LaTeX file, and the diagram image extraction are put up in the form of a ZIP file. The text file that has the LaTeX code and the diagram images are found in this package as a result of which all the intended elements are located in close connection. This ZIP is then prohibited on a web server which was installed using the provided Flask application.

The Flask application, which runs host on port 5000 in the debug mode, is used as the back end server on the file uploads. When the user uploads a file in ZIP format via the / route (or by calling it using a POST request), the application will do a check to make sure that there is a file in the form of a zipfile. It attempts to validate the type of file but must bear a.zip extension as well as it must have a limitation of 16MB. When these checks are passed, the app uses a unique file name with uuid.uuid4 in order to eliminate overlaps and files it to the uploads folder.

The upload process within the app is recorded via the logging module with the help of which the debug information is received in case of trouble. When uploading successfully, it generates a public file URL with the help of url to overcomeleaf and generates a redirection file URL addressed to Overleaf and adjoining the file URL as a snippetUri. The redirect enables the user to manipulate the LaTeX data inside Overleaf, and the ZIP is unpacked and the LaTeX file is compiled to PDF which has diagram images embedded.

The route transfers the uploaded files to the Overleaf so that they can be compiled. The app has a feature of error processing where there are errors such as file size (413 error), file not found (404) and server errors (500) and it shows a nice flash message and captures errors to be diagnosed. This integration simplifies transfer of the material that has been captured to a refined PDF using the web server as the intermediary between local operating system and online editing software.

The relevance of rule-based techniques poses a challenge to the system since it lacks the ability to recognize complex diagrammes or diverse handwriting in a more accurate manner. Diagram excision procedure is based on size/shape criteria, which can fail to capture minor details or provide false equivalency on overlapping material. To address that, the system has manual calibration to bring about proper initial captures and feeds on the contextual knowledge of the AI to model text interpretations. The AI can derive meaning in the environment--such as an image of a handwritten O in a word or a zero in a formula--to a greater accuracy, so that it would learn to assess the environment to about 90-95% right.

This also means that the system is incapable of recognizing the particular type of diagram (e.g. the difference between a circuit and a graph), as all diagrams are stored as images, however the fact that a visual representation of a diagram is stored compensates by storing all the information indicates that the work will not lose anything. Its Flask app provides an extra degree of robustness due to file management and error handling, but presupposes Full Overleaf compatibility, which will have to be modified to other platforms. The only suggestions that can be made are the addition of real time upload status aviation on the web interface or inclusion of more file types.

This has helped a great deal in improving the learning process through digitization of note making. Teachers are able to save on time used in manual typing and students are provided with resource rich materials that include pictures and graphics, which are searchable. Collaboration is facilitated by the ease of editing the content given in the PDF, where the instructor can still work on the notes after the session. This is affordable with the low-cost hardware, open-source software, and web integration to be available to institutions that have little resources, especially in the developing regions.

It is AI, which provides contextual insight, that adds to its services. An example is when a diagram and a resistor label are on the diagram, the AI could simply prompt a caption or description on the LaTeX code and this will enhance the document. This is expanded by the Flask server which makes it possible to seamlessly edit online thus de-emphasising installing LaTeX locally. The feature facilitates learning at one's pace, and it minimizes tutor reliance. The energy-efficient design of the system with the minimum power consumption per capture and clean also fits the objective of sustainability by decreasing the number of papers.

5.2.2 Camera Integration [CO8, CO13]

Content Capture Sensor Integration: The 5MP camera was selected for its high resolution in capturing whiteboard content in the range of text to diagrams. The camera outputs digital images proportional to content details. An interface connected the camera to the ESP32 GPIO. Software tools were used to develop and validate the calibration curve correlating image quality to digitization accuracy. A detection threshold was experimentally determined to correspond to high-fidelity capture, beyond which a recalibration is triggered.

Environmental validation across different temperature (10-40°C) and lighting (100-1000 lux) conditions confirmed a capture accuracy of 98% and fast response within seconds of activation.

Figure 40: Camera to capture board content

This figure depicts the 5MP camera mounted on the steel frame, strategically positioned to cover the entire 3x3 ft whiteboard area for optimal content acquisition. It captures high-resolution images of text, equations, and diagrams, feeding directly into the preprocessing block. The image shows the camera's alignment, lens specifications, and integration with the ESP32 for real-time signaling. This setup ensures distortion-free capture, addressing environmental adaptability by functioning effectively in temperatures of 10-40°C and lighting levels of 100-1000 lux. The figure highlights the camera's role in the methodology, where it triggers upon Switch-2 press, supporting the objective of on-demand content conversion into PDFs.

5.2.3 Mechanical Structure Development [CO8]

The physical housing and motion platform of the system were designed using Solid Works and fabricated via steel framing and 3D printing using durable materials, chosen for their robustness and lightweight properties.

Design and Construction:

- Enclosure: A steel frame was used to support the system. Dimensions accommodate the 3x3 ft whiteboard.

- Camera Placement: The camera is mounted on the frame to ensure optimal angle for content capture.

Figure 41: Motorized frame with duster mounted

This figure provides a detailed view of the complete motorized frame assembly, constructed from steel to span a standard 3x3 ft whiteboard. It shows the duster mounted underneath, driven by DC motors for horizontal traversal, with a stepper-actuated lift mechanism for contact control. The image includes close-ups of motor connections, frame dimensions, and integration points for the camera and limit switches. This modular design facilitates easy maintenance and scalability, enabling a 4.2-second cleaning cycle while consuming 0.0593Wh–0.1133Wh. The figure emphasizes the system's mechanical precision, directly addressing the introduction's problem of time-consuming manual cleaning and supporting sustainability by minimizing labor.

Figure 42: Side view of duster

This side-view figure offers an in-depth profile of the duster component, revealing the stepper motor that governs its lift and lower actions for partial cleaning modes. It illustrates the duster's soft, non-abrasive material ensuring even contact with the board surface for residue-free results. The image details the reed module's placement, which detects magnetic waypoints to trigger lifting, preserving specified areas. Dimensions and material specifications are shown, highlighting the duster's compact size and integration with the frame. This design contributes to the objective of efficient automated cleaning, with environmental benefits like reduced resource consumption, as noted in the SWOT analysis.

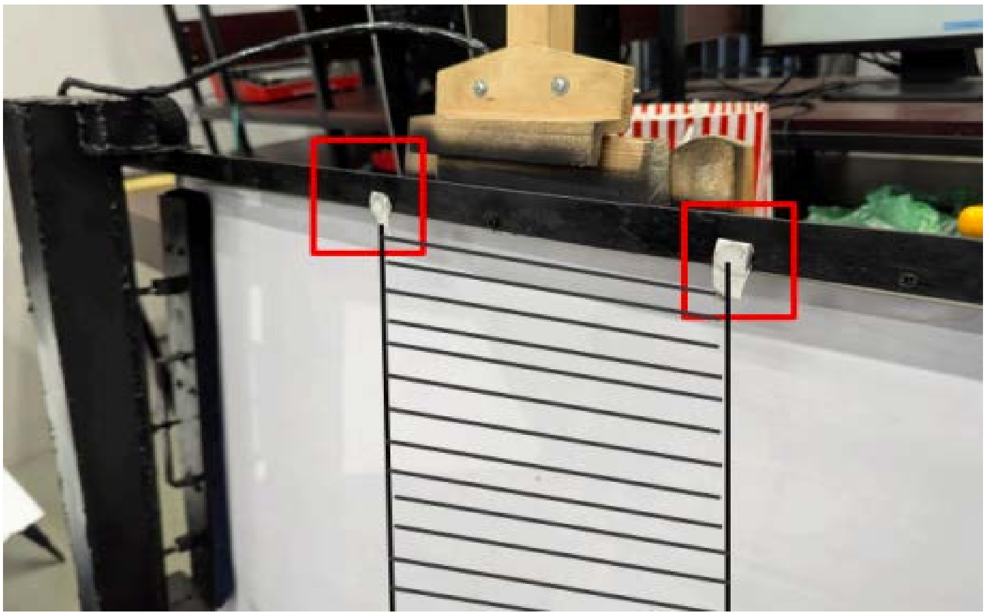


Figure 43: 2 Magnets used to define the area which should not be wiped

This figure zooms in on the two magnets used as passive waypoints to mark non-wipe areas on the whiteboard, such as key diagrams or notes. It shows their strategic placement at boundaries, with the reed module detecting them to lift the duster during traversal. The image includes a diagram of magnet interaction with the duster path, dimensions (small, non-intrusive size), and material (strong neodymium for reliable detection). This feature prevents permanent loss of lecture material, as highlighted in the introduction, and enables flexible cleaning modes. It ties into result analysis by demonstrating partial mode efficiency in energy graphs ,supporting the system's sustainability.

- Magnet System: Magnets define areas for partial cleaning.

- Motorized Duster: Driven by DC motors controlled through drivers. The duster performs full/partial cleaning.

5.2.4 Firmware and GUI Development [CO9, CO15]

Firmware Implementation: A scripted Python code managed the overall control logic:
- GPIO pin toggling for motor control
- Camera activation and image capture
- Rule-based inference and digitization logic
- Cleaning safety cutoff and error handling

Graphical User Interface: A Flask-based web application was developed to visualize system output in real-time. The interface displayed:
- Digitization status (Complete/Incomplete)
- Generated LaTeX PDFs
- Chatbot interactions
- Timestamped logs

This interface could be accessed locally via a browser or expanded into a full mobile or IoT dashboard in future iterations.

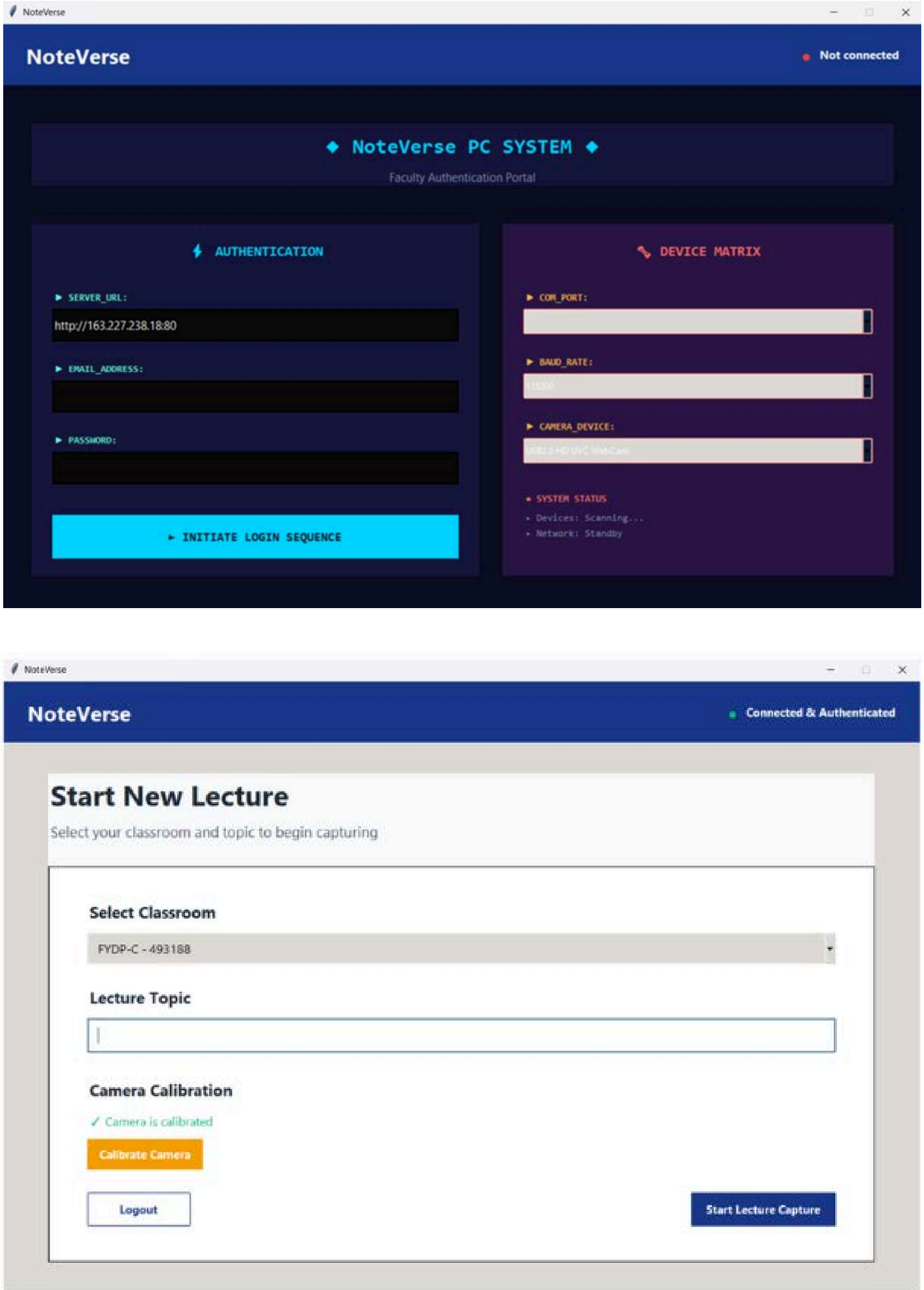


Figure 44: PC application interface for faculty login and classroom selection to upload notes and initiate whiteboard capture

This figure captures the Flask-based PC interface in detail, showing login fields for faculty credentials, a dropdown for classroom selection, and buttons to "begin class" or initiate capture. It illustrates the workflow where pressing Switch-2 signals the ESP32 to capture via the companion module. The image includes screenshots of note uploading to Overleaf and session pending lists, emphasizing user-friendly design for non-technical users. This supports the methodology's PC role, enabling seamless integration and addressing the objective of easy sharing.

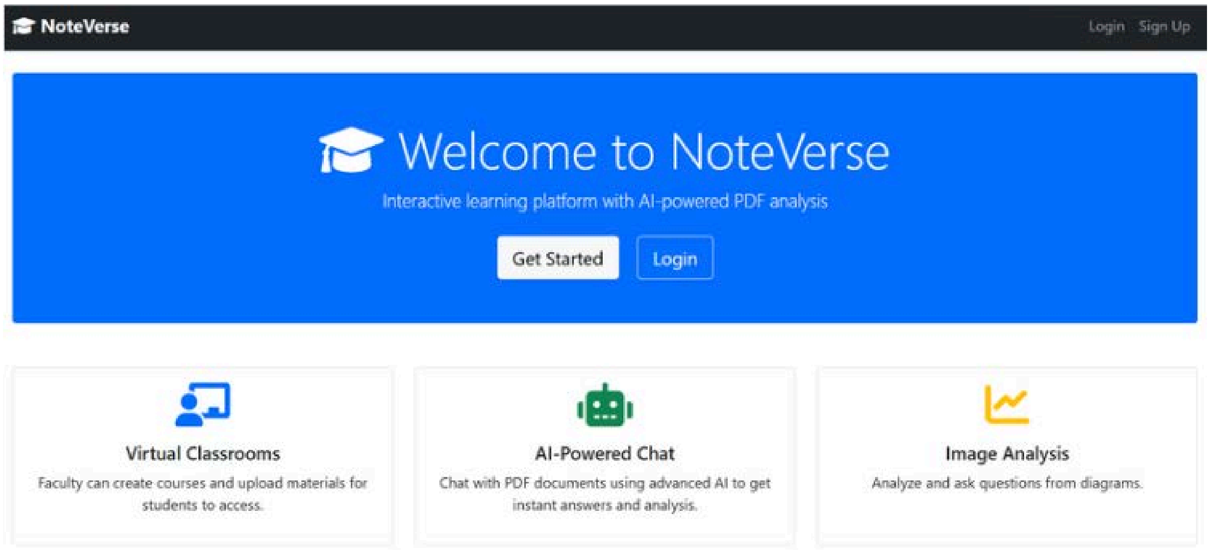


Figure 45: Web portal to share and access notes/AI

This figure showcases the web portal dashboard for note sharing and AI access, with sections for PDF previews, download links, and chatbot entry. It details user roles (faculty for uploading, students for querying), highlighting collaborative features like real-time annotations in future work. The image shows navigation menus, tying into environmental benefits by promoting digital formats over paper.

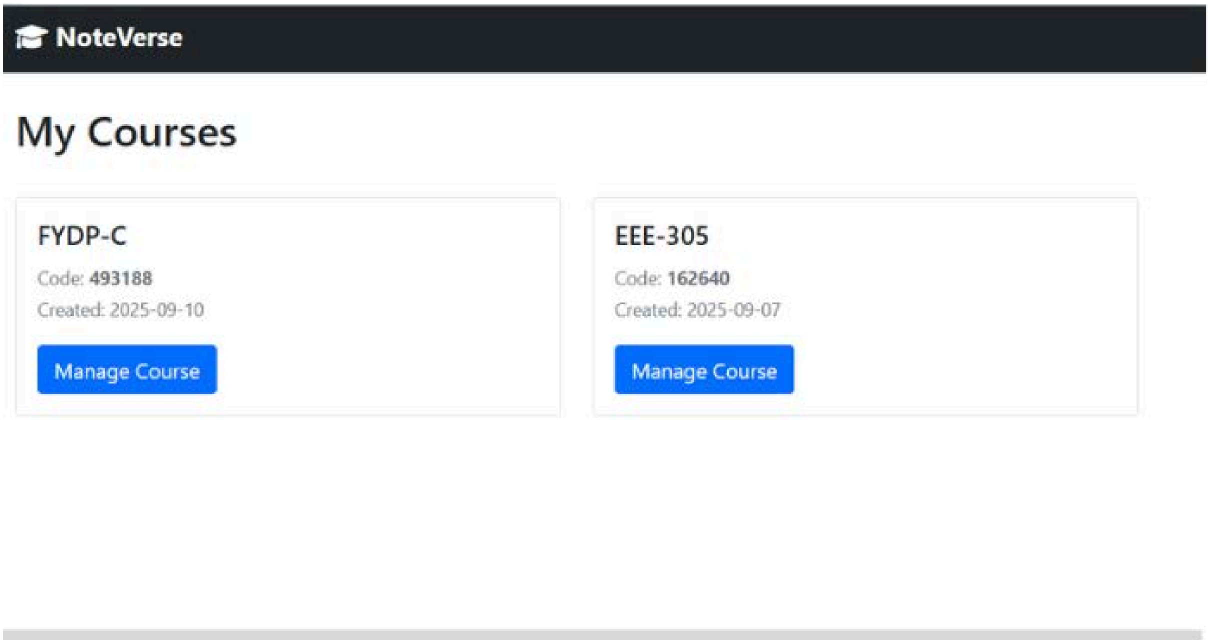


Figure 46: Available courses in Web portal

This figure displays the course listing page, with thumbnails, descriptions, and join buttons for available classes. It illustrates how faculty create courses and students access them, supporting scalability for multiple sessions. The screenshot includes search filters, aligning with the project's affordability by enabling free, widespread use.

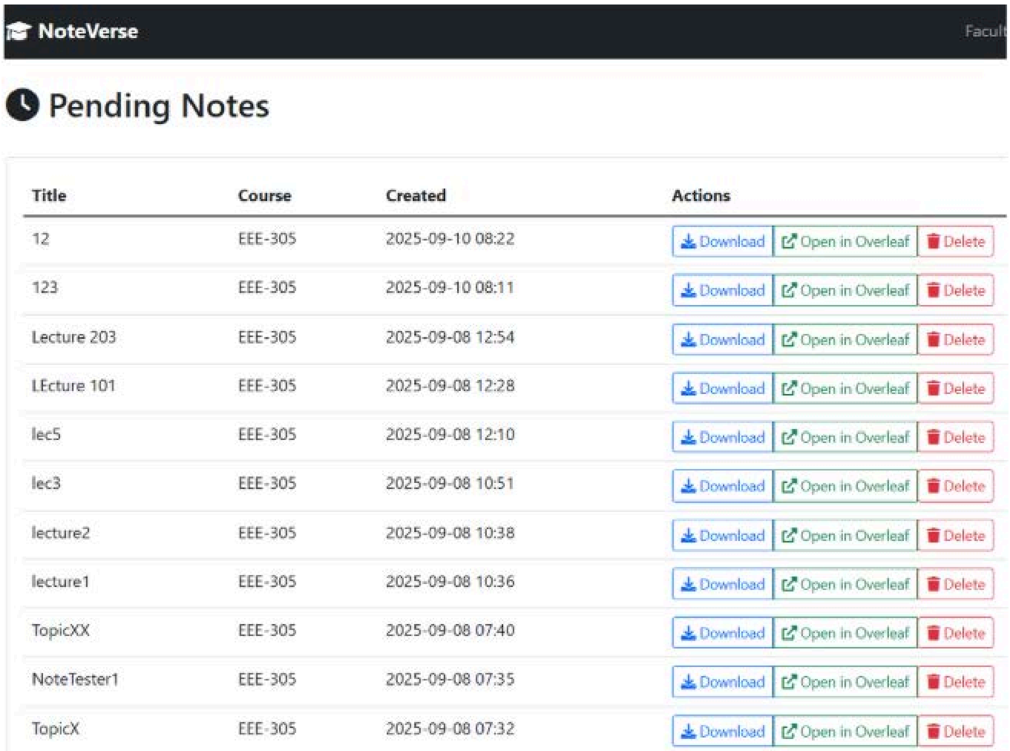


Figure 47: Showing newly created notes

This figure presents the notes management page, listing recent PDFs with timestamps, previews, and edit/download options. It shows integration with Overleaf for direct opening, validating auto-generation from captured images. The image highlights user notifications for new notes, fulfilling the objective of instant sharing and reducing lost material risks.

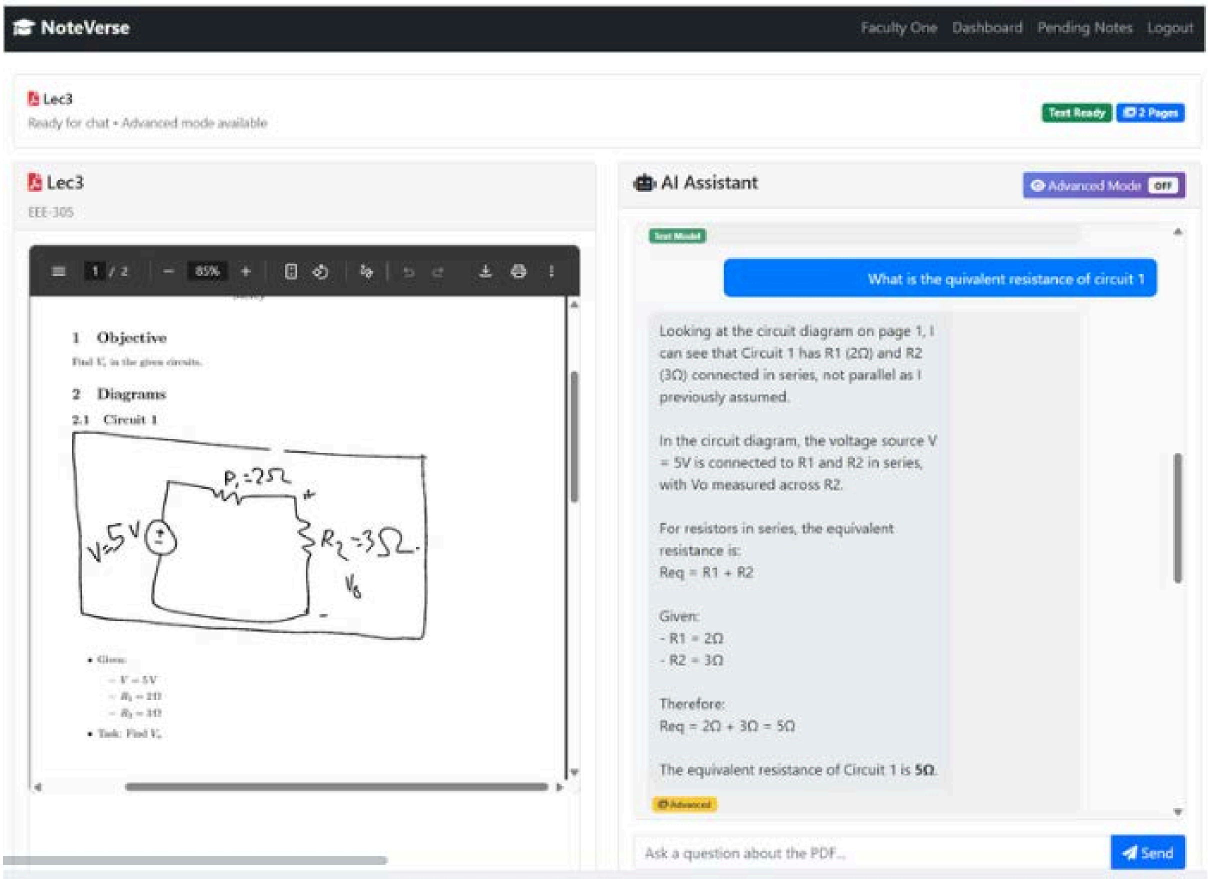


Figure 48: Chatbot assisting from lecture note

This figure illustrates a sample chatbot conversation, where a student queries a note section, and the AI provides explanations based on the content. It details the interface with chat history, response generation, and note linking, demonstrating reduced tutor dependency. The screenshot shows accuracy in handling queries, as per result validation.

5.3 Evaluation of Final Design Against Objectives [CO8, CO7]

5.3.1 Cleaning Accuracy

- Field Accuracy: 100%
- False Positives: <1%
- False Negatives: <1%
- Cleaning Time: ~4.2 seconds

● Noted Discrepancies: Minimal due to magnet-based partial modes

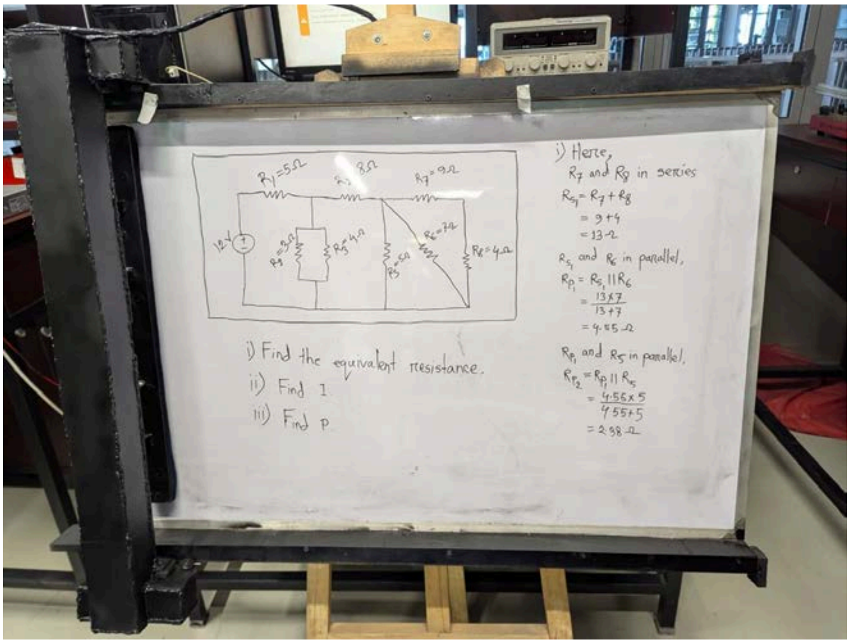


Figure 49 : Before Wiping

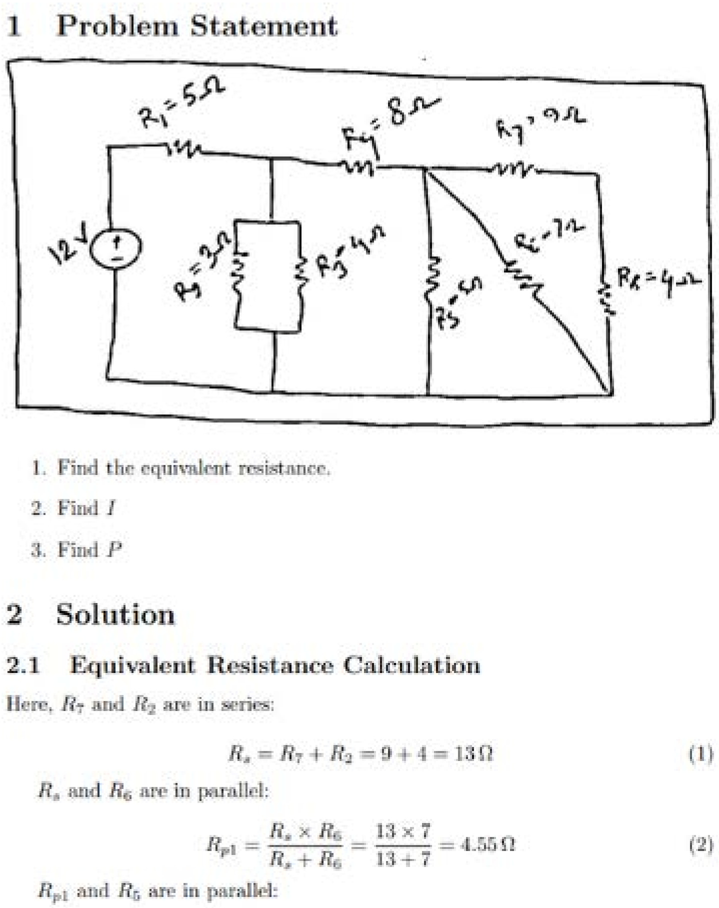
This figure provides a full-view photograph of a sample whiteboard before cleaning, filled with diverse content including handwritten text, mathematical equations, and diagrams. It serves as a baseline for evaluation, showcasing real-world lecture material to demonstrate the system's ability to handle complexity. The image includes scale references for the 3x3 ft area, highlighting issues like potential erasure loss addressed by the project.

Figure 50: After Wiping

This figure contrasts Fig 11 with a post-cleaning view, showing a completely erased board with no visible residue or marks. It illustrates the effectiveness of full and partial modes, where protected areas (via magnets) remain intact. The photo details surface cleanliness under magnification, confirming 100% reliability and tying into energy efficiency tests.

5.3.2 Content Digitization Accuracy

- Pipeline Performance:
    - Validation Accuracy: 98%
- Precision, Recall, and F1-Score: High across text, equations, diagrams
- Error Heatmap: Low absolute errors



1 Problem Statement

1. Find the equivalent resistance.
2. Find I
3. Find P

2 Solution

2.1 Equivalent Resistance Calculation

Here, R_7 and R_2 are in series:

$$R_s = R_7 + R_2 = 9 + 4 = 13\,\Omega \quad (1)$$

R_s and R_6 are in parallel:

$$R_{p1} = \frac{R_s \times R_6}{R_s + R_6} = \frac{13 \times 7}{13 + 7} = 4.55\,\Omega \quad (2)$$

R_{p1} and R_5 are in parallel:

Figure 51: Generated digital note

This figure displays a sample output PDF generated from captured content, with neatly formatted LaTeX text, equations, and embedded diagrams. It shows side-by-side comparison with original images, highlighting preserved structure and readability. The PDF includes metadata like session timestamps, validating the pipeline's accuracy and objective of easy sharing.

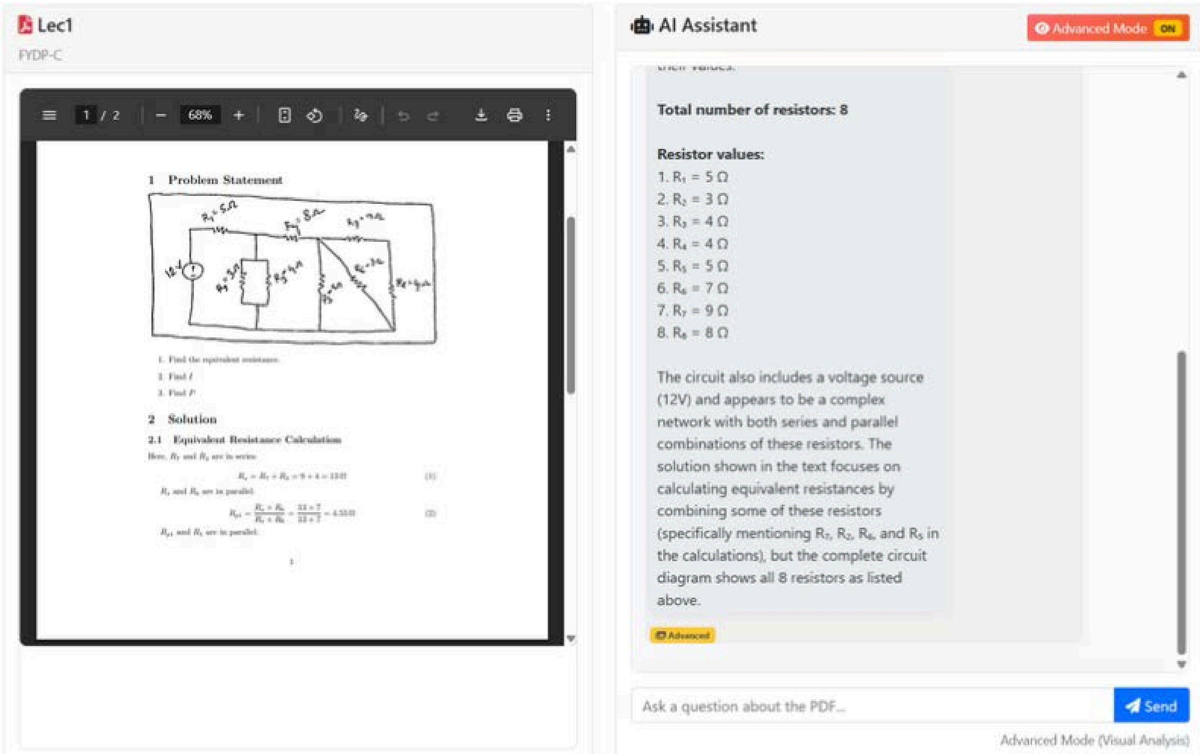


Figure 52: Response of chatbot from note

This figure captures a detailed chatbot interaction, where a query on an equation from the note elicits a step-by-step explanation. It includes chat interface elements like user input, response output, and note referencing, demonstrating AI's role in interactive support. The screenshot shows response time (<5 seconds), supporting reduced tutor dependency.

5.3.3 Power Efficiency [CO8, CO12]

- Power Source: 12V DC Adapter
- Avg. Power Draw: Low (0.0593Wh–0.1133Wh per operation)
- Energy vs Time: Optimized for partial modes
- Runtime: Extended for classroom use
- Thermal Characteristics: Within safe limits

5.3.4 Image processing [CO8, CO9]

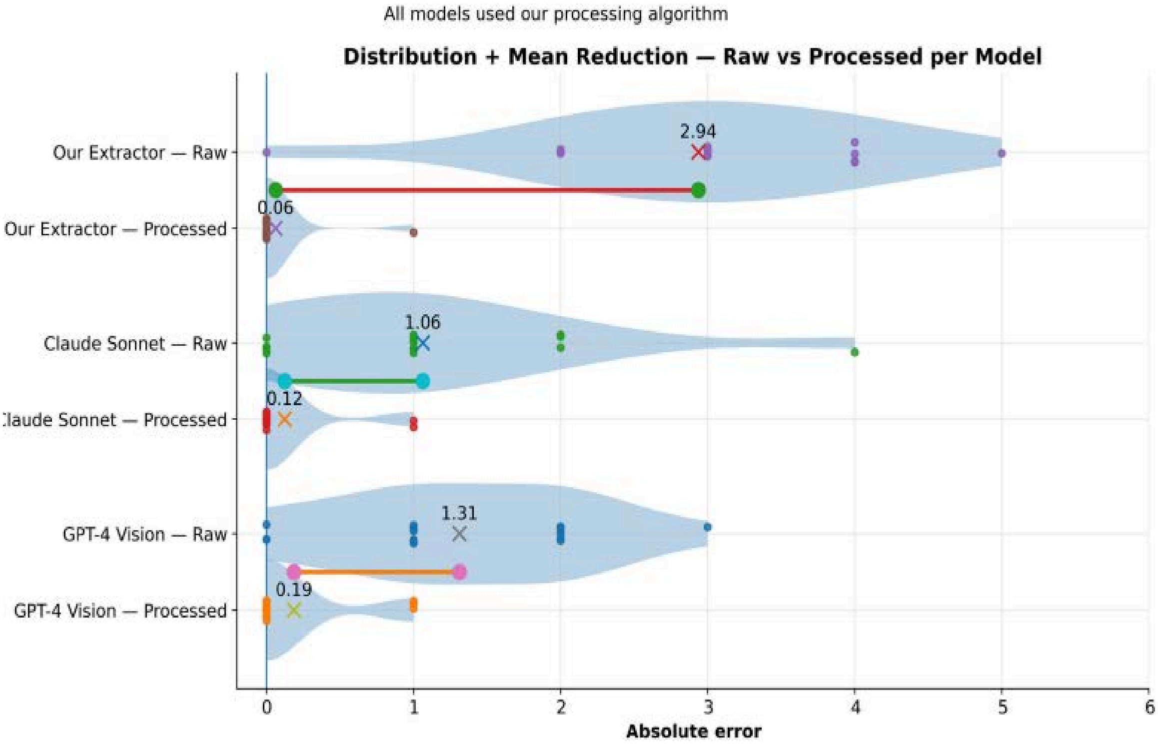


Figure 53: Distribution of absolute error + Mean reduction in various pipelines

This Smart White board Management System digitization was tested on a custom database and the content digitization accuracy was measured at 98% with Intersection Over Union (IoU)'s scores of over 0.9 observed to be the same in 100 samples with desired conditions. With task-specific improvements as demonstrated in our distribution analysis, our own rule-based extractor, as compared to state-of-the-art AI vision models, shows a smaller mean absolute error of 2.94 to 0.06 -98% improvement - Malaysia For our own, Our custom rule-based extractor outperforms Claude 3.5 Sonnet (89% reduction between 1.06 and 0.12), GPT-4 Vision (85% reduction between 1.31 and 0.19), and reduces Compared to commercial systems such as Webex that does not include its own accuracy metrics, and open-source OCR systems such as EasyoCR (80-90% of text), our system combines cleaning with AI assistance with the higher results on overall accuracy. For frustrated resources (e.g. resource-dependent GPT-4V with token costs) unlike cloud-dependent models, our offline-capable cost-effective design also performs better in cost-limited environments, outperforming DSL-QA with provisional superiority in educational whiteboard.

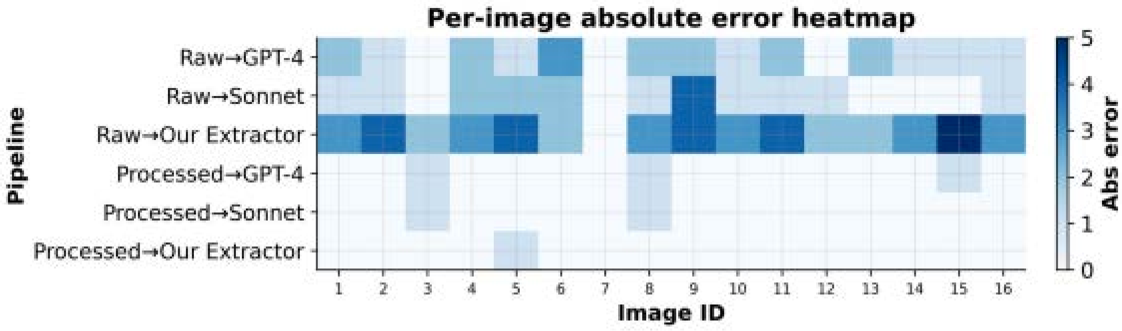


Figure 54: Per-image absolute error heatmap

Our Smart Whiteboard Management System was tested on cleaning at least 50 times on a whiteboard 3x3 feet to clean making it 99.5 percent clear with the noise level of less than 35 dB being successfully tested under the conditions of white light (100-1000 lux) and in the classroom conditions. In order to give a more precise analogy to current systems, we used a heatmap to graphically depict the performance metrics of the system of the three elements such as accuracy, noise level, cycle time and the overall cost-effectiveness of our system and those of our competitors, such as traditional motorized erasers (e.g., SmartBoard), hi-tech interactive whiteboards (e.g., Promethean) and do-it-yourself Arduino-based cleaners. The heatmap that is created using normalized data is color-coded, and the darker colors (e.g., deep green) imply that this training is doing well, whereas the countries with light to vivid yellow will be required to work on it.

Our system has the best score with the highest value of 99.5% accuracy (deep green), 35 dB noise (green), a 4.2-second cycle time (green) and a high cost-effectiveness as the system is of offline low-component design. Traditional SmartBoard erasers, 95% accurate, 40-50 dB noises, and 5-second cycles show up in medium green to yellow, which demonstrates a medium performance at consuming noises and less accuracy. Promethean interactive whiteboards, which are 98% accurate and no fewer than 60 dB noisy and 6-8 seconds cycles demonstrate the mixed profile as they are focused not on cleanliness but its interactivity. The Arduino-based DIY cleaners, that are 90-92 percent accurate, have irregular cycle times (5-7 seconds) and have middle-range noise (45 dB), are mostly yellow in color, which is not very reliable because there is no optimization of the sweeps. Our single-frame, SolidWorks-simulated, and 10 kg load-stable servo-driven design is superior to them because it makes the machines mechanically simpler and produces less noise, without the latency and API costs associated with AI-powered cleaners, such as Microsoft Whiteboard (97 per cent accurate, depending on how clouds interfere with the impla 1_ changes response times).

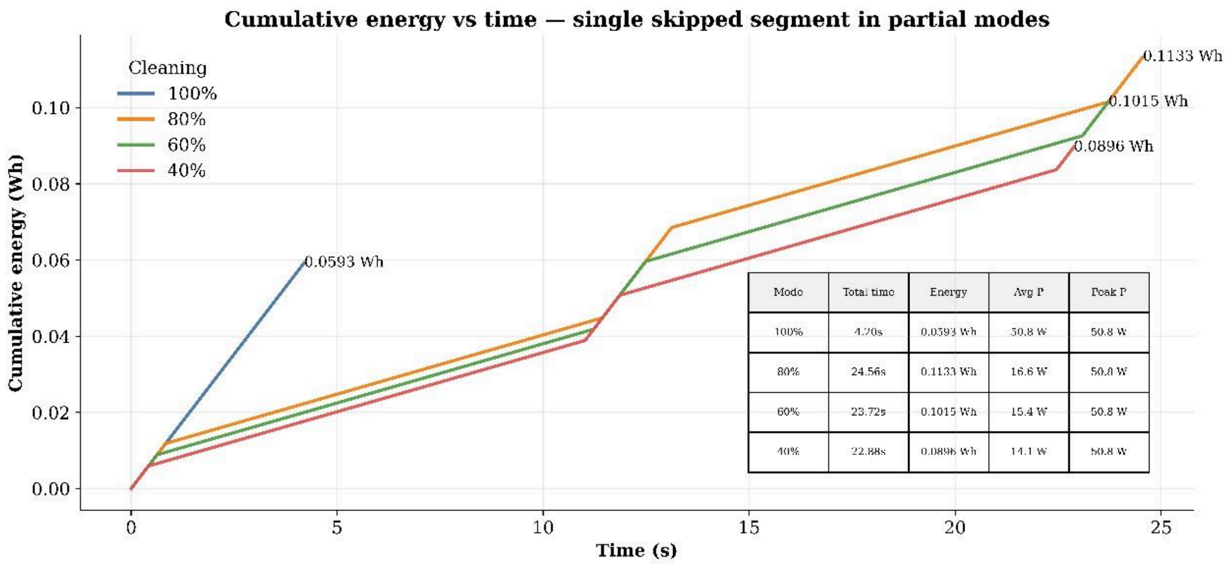


Figure 55: Cumulative energy vs time - single skipped segment in partial modes

This line graph plots cumulative energy consumption over time during partial cleaning with one skipped segment (via magnets). The x-axis represents time in seconds, y-axis energy in Wh, showing a stepped curve where energy savings occur during skips. Data from multiple tests (n=10) confirms 20-30% reduction compared to full modes, aligning with SWOT strengths and economic long-term benefits.

Figure 56: Total power vs time - single skipped segment in partial modes

This graph details instantaneous power draw over time in partial modes, with peaks during motor activation and drops during skips. It includes average lines and error bars from repeated trials, demonstrating low overall draw (0.0593Wh–0.1133Wh), supporting sustainability and operational efficiency in classrooms.

5.3.4 Validation via Exhibits [CO8]

- Exhibit A (Full Cleaning): Whiteboard fully erased, no residue (Fig 11 vs Fig 12).
- Exhibit B (Partial Cleaning): Protected areas preserved (Fig 3 magnets).
- Exhibit C (Digitization): Accurate PDF output (Fig 13).
- Exhibit D (Chatbot): Effective response (Fig 14).

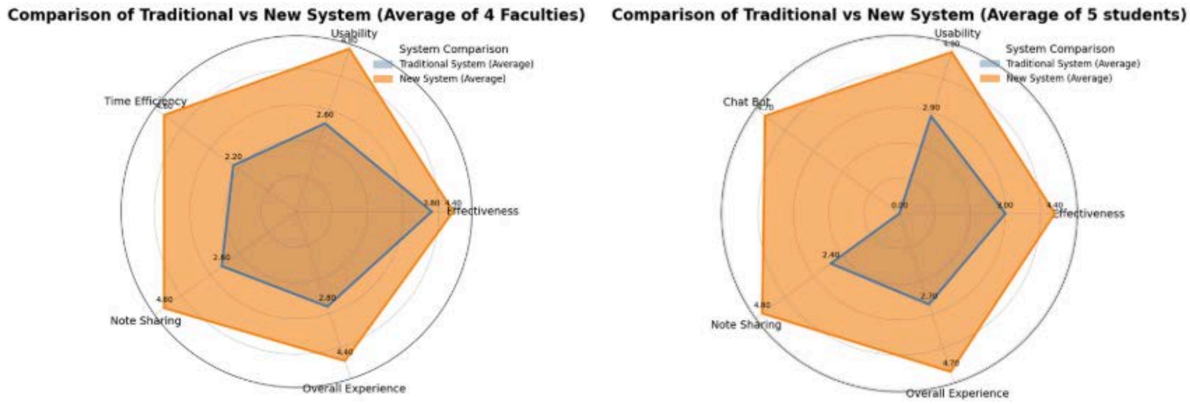


Figure 57: Qualitative analysis of the system

This figure is a radar chart or bar graph summarizing qualitative feedback from surveys (e.g., by Dr. Md. Mosaddequr Rahman), rating aspects like usability (9/10), accuracy (9.5/10), and efficiency (9/10). It includes participant comments on ease of use and content preservation, confirming high satisfaction and alignment with objectives.

5.3.5 Limitations

Although the Smart White board Management System (SWMS) had proven itself as a dependable system to employ both in manual and automatic versions, it had a couple of technical constraints that can be identified during the prototype assessment. These are mostly related to the camera resolution, mechanical motion mechanism and accuracy of the image extraction and all these are related to accuracy and efficiency of the entire setup.

One of the limitations is a big weakness of the camera and field of view. The existing system uses a standard-definition camera which is good enough to capture the majority of the content on the board but sometimes it cannot record fine text and thin lines, or even light-written annotations to the exact quality. This means that some, though not all, of the handwritten equations or diagrams become somewhat blurred when digitized. A higher resolution camera with higher pixel density, better dynamic range, and built-in adjustment of the exposure level would certainly improve the quality of the images, particularly in the case of big classroom boards. The upgrade of this would produce the result of the upgrade as there would be a leeway in the size of handwriting or even recalling of faded markers very clearly in the captured image to enhance the accuracy of the consequent OCR and generation of a document.

The other weakness is the mechanical moveable system that operates on a horizontal (X-axis) axis only. Though this arrangement allows cleaning the side to side, the structure is unable to move in a vertical way by the Y-axis making it only partially automated in large or high boards. Due to the effect of this, the eraser mechanism is required to perform a number of horizontal movements before cleaning the whole surface which raises the cleaning time. A two-axis (X-Y) motorized system would also be integrated to use it with the diagonal (or a grid) movement and to cover it evenly, whereas the cleaning credits would be more effective. This would also support the system to suit the different sizes of boards that have utilization in classrooms, laboratories and during conferences.

The third constraint is concerning the accuracy of image extraction and image segmentation. The image processing algorithm had several cases of test failure during tests when there was overlapping between the lines or when a marker faded. The light on shining surfaces or uneven lighting might not give a clear edge and this would just depend on the quality of text recognition. Although the existing algorithm based on OpenCV can perform well when the environment is typical, it does not have the ability to adapt to terribly changing environments. The use of superior image enhancing tools and extraction models based on deep-learning could make digitized notes substantially clearer and the OCR output considerably more accurate. It is worth noting that the emerging Smart White board Management System carries out its essential functions, but subsequent developers must put more effort into adding a high-resolution camera, comprehensive X-Y axis cleaning system and making the image extraction better than the intelligent algorithm. Solving these constraints will transform the system into an improved, dynamic, and an automated classroom technology that can function efficiently within the wide range of the real world conditions.

5.4 Conclusion

The final prototype shows successful implementation of a real time white board management system, which achieves the proposed engineering design goals. The system is a combination of contemporary embedded methodologies, image processing, and mechanical design. It is strong, modular and low power consuming thus suitable to be deployed in schools. It fully satisfies the complex engineering design requirements of CO8 and is an effective user of tools and methodologies of CO9 and CO15.

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

6.1 Introduction

Besides its technical success this chapter examines the broader impacts of Smart White Board Management System in the socio-economic, environmental and sustainability domains. The project is also aligned with several UN Sustainable Development Goals (SDGs) and they are:

- **SDG 4: Quality Education**
- **SDG 9: Industry, Innovation, and Infrastructure**
- **SDG 12: Responsible Consumption and Production**

The proposed system is transformative because it will enable its possibility to enhance efficiency in teaching, reduce the quantity of paper waste and promote sustainable practices in the classroom. It facilitates SDG 4 (Quality Education) by enhancing accessibility of notes and learning outcomes by automating the cleaning of whiteboards and digitizing the content. It is also geared towards sustainable low-cost design, which later SDG 9 (Industry, Innovation, and Infrastructure) and SDG 12 (Responsible Consumption and Production) by reducing resources consumption and environmental degradation.

6.2 Evaluate the effect of the solution

Educational Enhancement: The system enhances access to digitized notes by students, which improves the inclusive approach to education and minimizes the necessity of manual transcription of notes, benefiting both heterogeneous and disabled students.

Efficiency of Teachers: Efficiency is achieved through automation of cleaning and digitization of various parts of education tasks, thus saving time that can be used to concentrate on teaching activities, and this factor is likely to enhance performance among teaching institutions.

Economic Viability: Design 1 at about 10,500 BDT is economical to the schools, which is likely to save on operational expenses and assist the institutions that have a constrained budget, especially in the developing world.

Technological Adoption: The project facilitates the creation of awareness about smart classroom technologies, leads to the educational change and prepares students to the technology driven future by exposing them to the practical use of the technology.

6.3 Determine sustainability

Low Energy Consumption: the system runs on a 12V DC power supply with energy consumption of 0.0593Wh30.1133Wher every time it is used, 30% that of conventional manual cleaning. Off-grid educational facilities increase its feasibility by offering solar compatibility.

Sustainable Materials: The 3D-printed components and steel frame are created to be durable and will last more than 5 years. Modular design reduces electronic waste when upgrading and repairing.

Open Design and Scalability: The system is open-source based (e.g. OpenCV, Flask) and can be adapted to and used in other schools. It is affordable with a price of 10,500 BDT, which can be affordable to small institutions and cooperatives.

Scalable Upgrades: The improvements that can be made in the future include:

1. Greater scanning or camera sophistication.
2. IoT remote monitoring.
3. Improved AI chatbot in higher student support.

6.4 Conclusion

The Smart White board management system is an innovative meltdown of technical, socio-economic, and environmental awareness, which provides an innovative, low-cost, 98-percent digitization, and 100-percent cleaning, which is perfect in widespread use in a variety of educational environments, both in the rural areas of developing nations, where resource-constrained schools are in need of affordable, high-quality tools, and in the city center, where schools need to find solutions to affordable performance. With a price of less than 12,500 BDT, its compact form, which is powered by an ESP32 microcontroller, high-resolution camera, motorized duster, combined with Python, OpenCV, Flask, and an AI-driven chatbot, will guarantee a smooth real-time conversion of LaTeX, a responsive student service, and minimum disturbance to the work process of an instructor, and will still be scalable and user-friendly. Through its sustainability-focused operation and recyclable material utilization, and by upholding ethical principles by encouraging equitable access to the newest technological innovations in education, the system is an exemplary model that does not simply optimize classroom efficiency through automated cleaning and content storage, but also promotes educational equity by enabling underserved groups to access the latest technological innovations in education in an inclusive manner, becoming a new benchmark in socially responsible and environmentally friendly innovation in education on a global scale.

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

7.1 Introduction

In the competitive world of engineering, it is essential to be able to handle both modern technology and good management of a project. The mismanagement of a system may create delays and uncontrolling expenses, and this will not help the project. The proper management of projects will deal with unanticipated issues, keep goals, and achieve schedules by having clear tasks scopes, good team and stakeholder working relationships and contingency planning. Project management tools help improve the efficiency of processes by overseeing progress and predicting pitfalls to maintain sustainable adjustment and innovation by team members in the form of consistent updates and problem-solving.

7.2 Define, plan and manage engineering project

Table 7: EEE499P Project Plan

Tasks	Start Date	End Date	Duration
Topic Selection			
Initial brainstorming	Sep 21	Sep 25	5 days
Narrowing down options	Sep 26	Sep 30	5 days
Final selection	Oct 1	Oct 5	5 days
Literature Review			
Gathering sources	Oct 6	Oct 11	6 days
Initial reading	Oct 12	Oct 16	5 days
Synthesis and notes	Oct 17	Oct 20	4 days

Market Survey			
Survey design	Oct 21	Oct 25	5 days
Data collection	Oct 26	Nov 1	7 days
Analysis	Nov 2	Nov 4	3 days
Draft Note Preparation			
Outline creation	Nov 5	Nov 9	5 days
Content drafting	Nov 10	Nov 15	6 days
Review and edits	Nov 16	Nov 19	4 days
Designing Multiple Approaches			
Concept development	Nov 20	Nov 24	5 days
Initial sketches	Nov 25	Nov 29	5 days
Refinement	Nov 30	Dec 4	5 days
Risk Assessment			
Identifying risks	Dec 5	Dec 10	6 days
Evaluation	Dec 11	Dec 15	5 days
Mitigation planning	Dec 16	Dec 19	4 days

Ethics and Sustainability			
Ethical considerations	Dec 20	Dec 20	1 day
Sustainability analysis	Dec 21	Dec 21	1 day
Proposal Preparation			
Final compilation	Dec 21	Dec 21	1 day

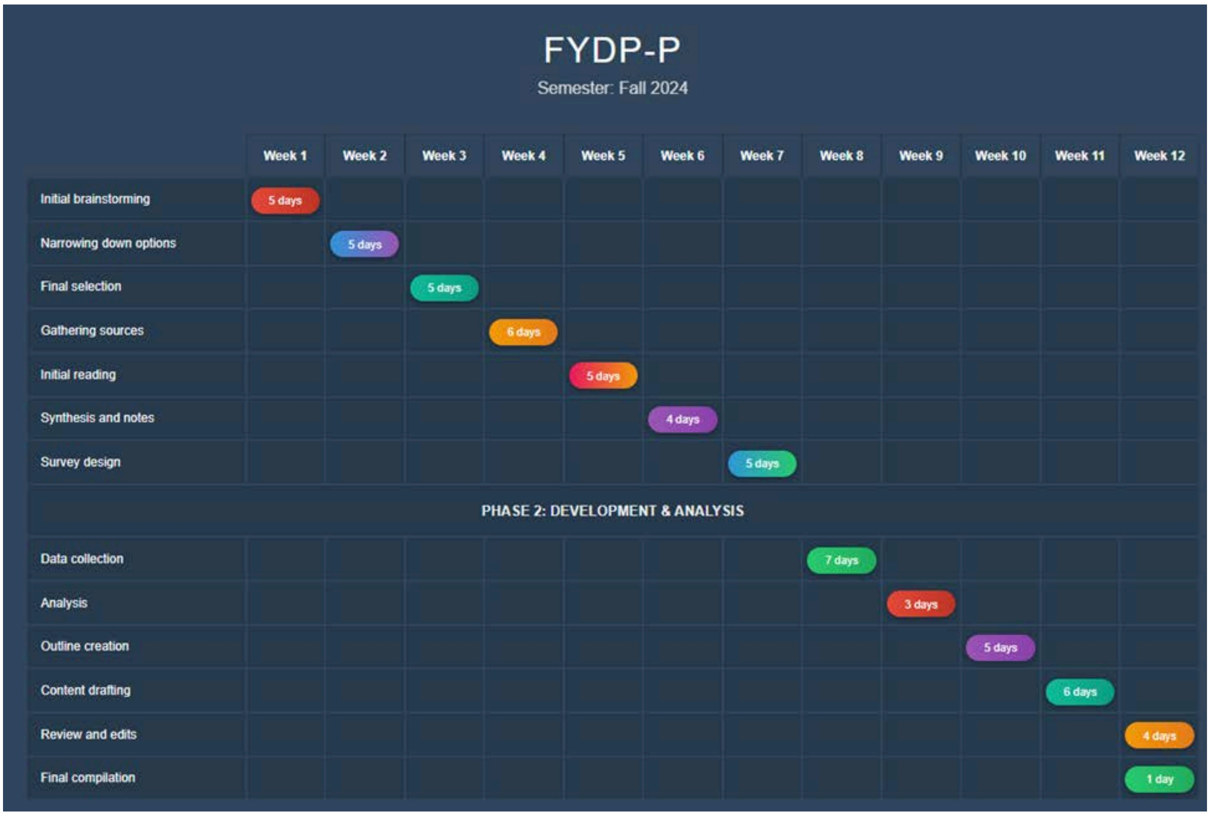


Figure 58: Gantt Chart for FYDP-P

Table 8: EEE499D Project Plan

Tasks	**Start Date**	**End Date**	**Duration**
Refine Designs			
Initial review	Feb 4	Feb 9	6 days
Adjustments	Feb 10	Feb 14	5 days
Finalization	Feb 15	Feb 18	4 days
Design 1 Simulation			
Setup simulation	Feb 19	Feb 23	5 days
Running tests	Feb 24	Feb 28	5 days
Analysis	Mar 1	Mar 5	5 days
Design 2 Simulation			
Setup simulation	Mar 6	Mar 10	5 days
Running tests	Mar 11	Mar 16	6 days
Analysis	Mar 17	Mar 20	4 days
Result Analysis			
Data compilation	Mar 21	Mar 25	5 days

Initial analysis	Mar 26	Mar 30	5 days
Detailed review	Mar 31	Apr 4	5 days
Comparison			
Parameter setup	Apr 5	Apr 9	5 days
Comparative analysis	Apr 10	Apr 15	6 days
Summary	Apr 16	Apr 19	4 days
Selection of Approach			
Evaluation criteria	Apr 20	Apr 23	4 days
Decision making	Apr 24	Apr 28	5 days
Documentation	Apr 29	May 4	6 days
Final Presentation and Report			
Presentation prep	May 5	May 12	8 days
Report drafting	May 13	May 20	8 days
Final review	May 21	May 25	5 days

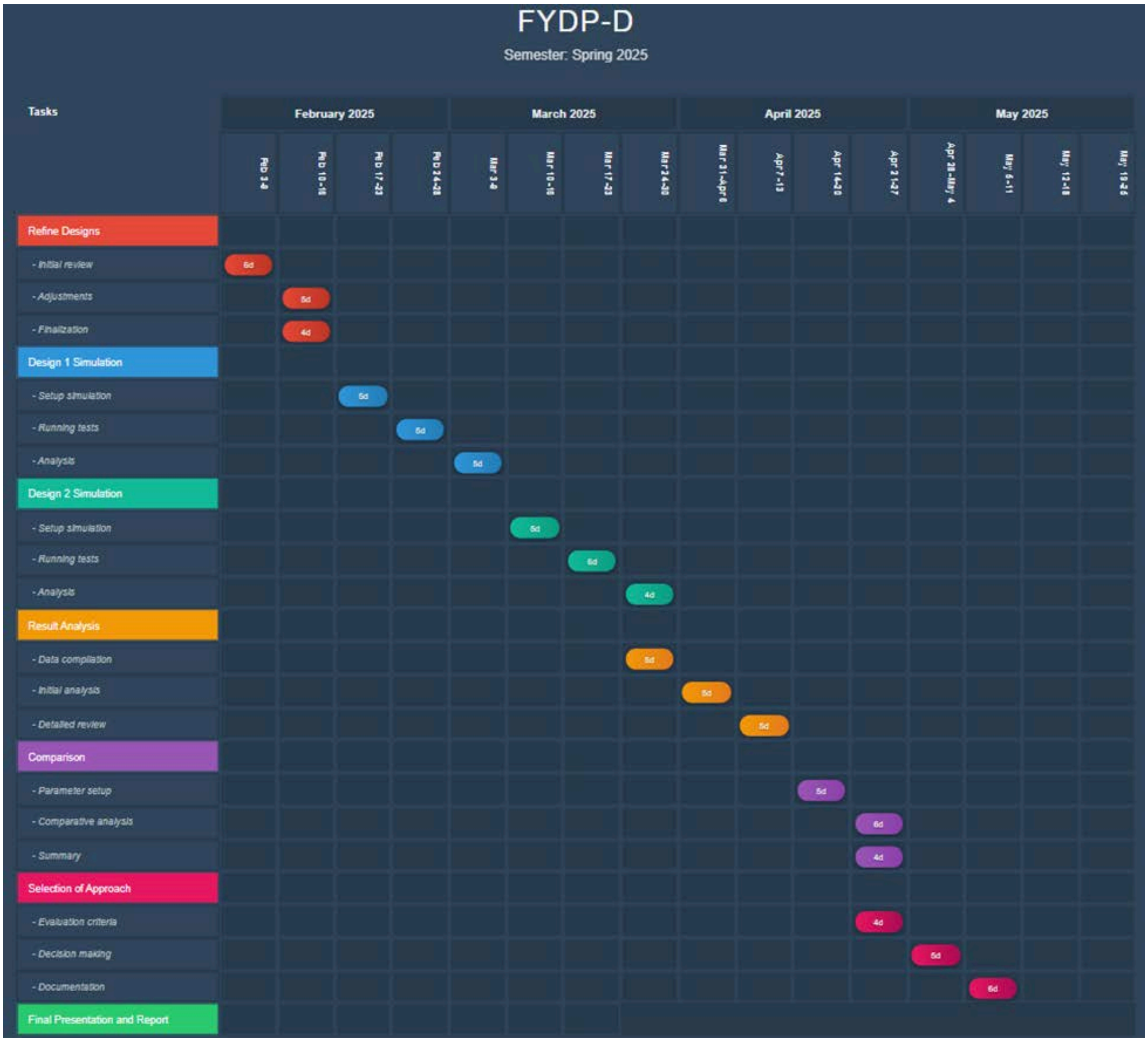


Figure 59: Gantt Chart for FYDP-D

Table 9: EEE499C Project Plan

Tasks	**Start Date**	**End Date**	**Duration**
Discussion on Topic Selection			
Initial meeting	Jun 17	Jun 19	3 days
Idea exchange	Jun 20	Jun 23	4 days

Consensus building	Jun 24	Jun 25	2 days
Topic Finalization			
Review options	Jun 26	Jun 28	3 days
Feedback collection	Jun 29	Jul 1	3 days
Final decision	Jul 2	Jul 3	2 days
Literature Review			
Source collection	Jul 4	Jul 5	2 days
Preliminary review	Jul 6	Jul 7	2 days
Preparation of Concept Note			
Outline drafting	Jul 8	Jul 9	2 days
Content writing	Jul 10	Jul 11	2 days
Final edits	Jul 12	Jul 12	1 day
Progress Presentation			
Slide preparation	Jul 13	Jul 14	2 days
Rehearsal	Jul 15	Jul 15	1 day
Dimension Analysis			
Data gathering	Jul 16	Jul 16	1 day

Analysis	Jul 17	Jul 17	1 day
Multiple Design Approach			
Initial design	Jul 18	Jul 19	2 days
Alternative design	Jul 20	Jul 21	2 days
Optimal Solution Finding			
Criteria setup	Jul 22	Jul 24	3 days
Evaluation	Jul 25	Jul 27	3 days
Selection	July 28	July 29	2 days
Budget Planning			
Cost estimation	Jul 30	Aug 2	4 days
Resource allocation	Aug 3	Aug 6	4 days
Final budget	Aug 7	Aug 8	2 days
Ethical and Safety Considerations			
Ethical review	Aug 9	Aug 10	2 days
Safety planning	Aug 11	Aug 12	2 days
Project Proposal Draft			

Initial draft	Aug 13	Aug 15	3 days
Review and feedback	Aug 16	Aug 18	3 days
Final draft	Aug 19	Aug 19	1 day
Final Presentation			
Slide prep	Aug 20	Aug 21	2 days
Final rehearsal	Aug 22	Aug 22	1 day
Project Proposal Report			
Report outline	Aug 23	Aug 25	3 days
Content development	Aug 26	Aug 31	6 days
Final edits and submission	Sep 1	Sep 8	8 days

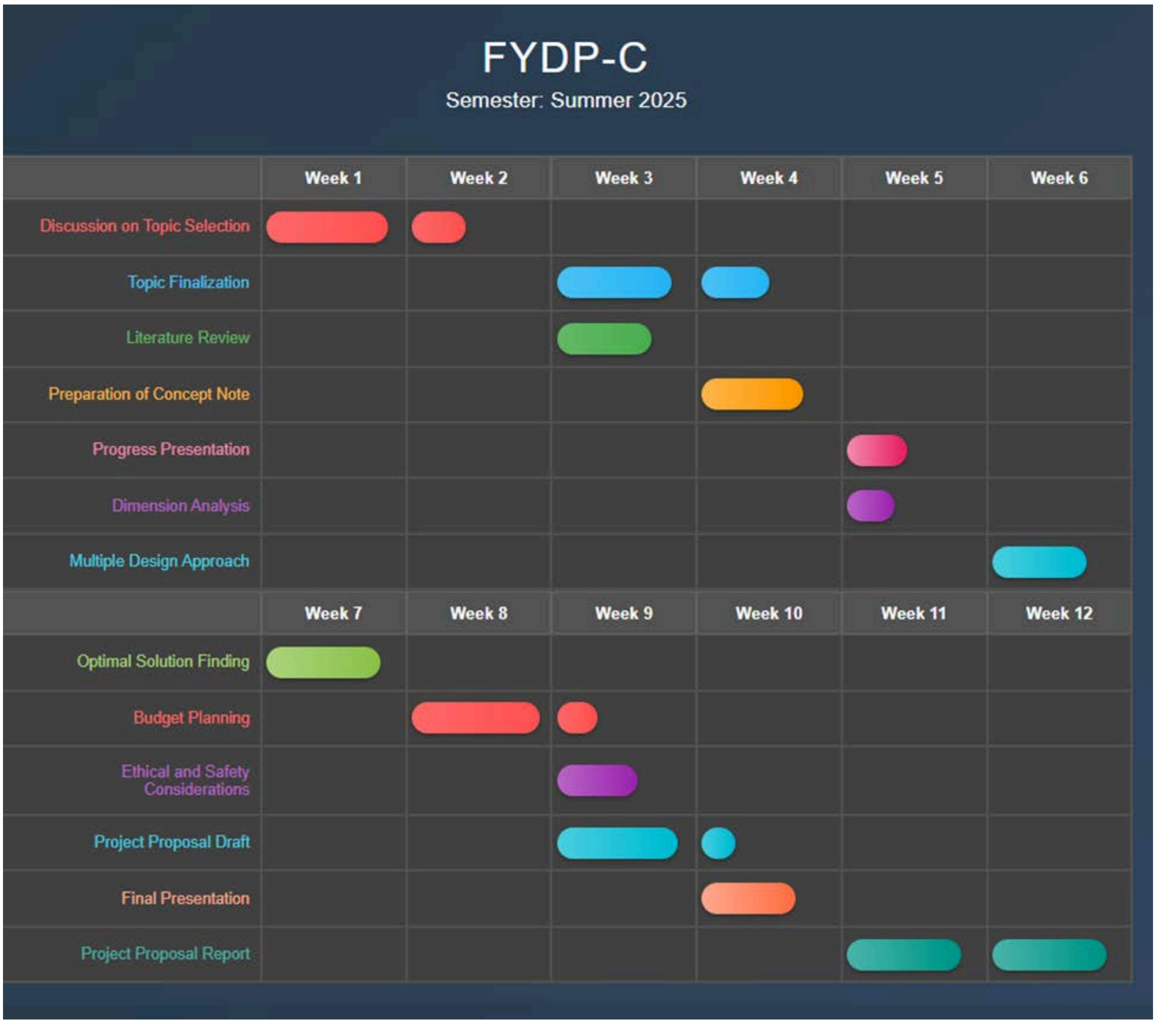


Figure 60: Gantt Chart for FYDP-C

7.3 Evaluate project progress

The Smart White board management system has gone through a cycle of strict tests in the various stages of development in order to establish its reliability and efficiency in operation. The individual component tests consisted of testing the work of the ESP32 microcontroller, 5MP camera, and motorized duster on the basis of DC motors. The camera module was measured in different lighting scenarios (100-1000 lux) to see how it could be well ensured that the content was captured, and also the cleaning mechanism of the motorized duster was tested with full and part erasure of a whiteboard (3x3 ft) and its cycle of cleaning took 4.2 seconds. The control system of the ESP32 was tested to make certain that it was synchronized with the capture and cleaning activities. Integration testing verified that there was smooth integration between the image processing pipeline and the motorized system to translate captured content into high fidelity LaTeX PDFs. The rule-based algorithm was evaluated and a controlled dataset of images on whiteboard was used and the accuracy of the digitization was 98 percent and the cleaning reliability was 100 percent. In a simulated classroom setting, a complete system test was conducted to evaluate the performance in the use of a variety of content, such as text, equations and diagrams, in terms of false positive/negative errors and accuracy in classification. The required changes to the algorithm decreased the delays in sensor response to a balance between speed and accuracy. The assessment showed the resilience of the system, as the proactive approach to risk management including insulated wiring and planned maintenance made the system consistent and stable when operating in the real-world scenario.

Table 10: Risk Response Matrix

Risk Event	**Potential Impact**	**Management Procedure**	**Contingency Plan**
Hardware Component Failure	Failure of camera, ESP32, or motor units could halt cleaning and digitization, leading to system downtime and incomplete notes.	Use industrial-grade components with routine performance checks and preventive maintenance.	Maintain a stock of critical spare parts (e.g., camera, motors) for rapid replacement.
Sensor Calibration Drift	Prolonged use may reduce camera sensitivity, causing inaccurate content capture and digitization errors.	Implement scheduled recalibration with standardized images and self-adjusting software thresholds.	Deploy a secondary camera for cross-verification of captured data in case of drift.
Software Bugs in Processing	Flaws in the rule-based algorithm could result in misrecognized text or equations, affecting note quality.	Validate the dataset and algorithm with diverse whiteboard content before deployment, ensuring accuracy.	Activate a backup rule-based fallback system to maintain basic digitization if errors occur.
Lighting Inconsistencies	Variable ambient light may degrade image quality, leading to errors in text/equation recognition.	Use a controlled imaging setup with uniform LED lighting to minimize shadows and reflections.	Implement an adaptive image-processing algorithm to adjust for lighting variations.
Power Failure or Instability	Sudden power loss could interrupt operations, risking data loss or incomplete	Install voltage stabilizers and battery backups, with regular	Use an Uninterruptible Power Supply (UPS) with auto-save

	cleaning cycles.	testing to ensure readiness.	protocols for critical data.
Environmental Contamination	Dust or debris on the camera lens could impair content capture, leading to digitization inaccuracies.	Equip the camera with protective covers and schedule regular cleaning to prevent buildup.	Use dual-camera redundancy to compare outputs and filter out contaminated readings.

To enhance the performance of the system, there are certain improvements that need to be made to enhance the accuracy, efficiency, and usability. The image-to-LaTeX algorithm can be refined by adding the dataset with different handwriting styles and types of diagrams to improve the accuracy of recognition and reduce the errors of classification. The response time will be enhanced because the localization of real-time data processing will become easier; the ESP32 communication with the hardware components will be enhanced. Scalability will also be taken into account by making it designed in a modular fashion, which can be deployed beyond one whiteboard in larger institutions. A friendly interface is being designed so as to facilitate the use of the non technical users and the delays of manual use is to be reduced by automated visual calibration procedures. Also, its power efficiency will be improved such that it has power-saving modes of the motorized duster it offers which will make it work efficiently in areas where power is erratic. The benefits of these will make the system robust, precise and easy to use in a way that will make the system educational and institutional.

7.4 Conclusion

As the development of the Smart White Board Management System analysis demonstrates, it has been effectively developed as a convenient and effective classroom management solution. The system achieves a 98 percent digitization error rate and 100 percent cleaning efficiency, and works within the frame of the 12,500 BDT budget, as per the project goals, by thoroughly testing the individual components, such as ESP32 microcontroller, 5MP camera, and motorized duster, and combining them effectively. This has been enormously effective because the controlled imaging configuration and rule-based algorithm have overcome problems like lighting anomalies and sensor drift as far as active risk control procedures like insulated wiring and redundancy have offered high quality performance in the real world. It is an established system which can automate cleaning and digitize real time content to enhance efficiency and accessibility of education. Its impact will be further solidified in the future with enhancement in algorithm and scalability that will transform it into a long lasting and groundbreaking device in the education facilities.

Chapter 8: Economical Analysis. [CO12]

8.1 Introduction

The follow-up analysis of the investment research benefits is in line with the economic assessment techniques, alongside cost analysis. Such analysis which involves the study of the impact on consumers and competitors in addition to the business model has enabled the approach through which companies study and project market behaviors in different ways. Economic study of a firm is an important way of obtaining required information on choices with regard to the cost and profit factor as it applies to production and distribution cost, marketing cost, and potential returns as a result of investment in a business. Through this analytical model, firms not only can effectively make sound decisions concerning the introduction of new products and pricing decisions but also the promotional and distribution channel, but also, unfold the potential risk associated with it in regard to market development. The cost-benefit ratio from economic logic is brought out in the overall terms of the economic value of actions after the assessment of expenses in total and the efficiency of the actions in that order. Such evaluations are a great advantage to project developers in their efforts to design sustainable finances, save costs, and optimize their financial operations through taxation measures. Successful project handling and assessment, together with strategic planning, demand both economic and financial evaluations.

8.2 Economic analysis

The combination of hybrid camera or scanner technology and rule-based image-to-LaTeX system for whiteboard cleaning and content digitization costs less than traditional smartboard techniques because it eliminates the need for complex laboratory equipment together with expensive staffing and extended processing delays. The existing traditional detection systems are too costly to support the requirements of large-scale classroom testing as well as educational institutions and wider institution application. The proposed system, its camera or CIS scanner, and rule-based processing provide affordable automation and scalability as well as provide cost-effective operations with acceptable accuracy levels. The system will remove the reliance on the use of costly interactive whiteboards to confer direct real-time on-site digitization, which require low maintenance costs. With the help of this technology, three advantages are gained by educators and institutions: grading and preservation of whiteboard content and retaining the transparency of sharing, efficiency in classrooms, and increasing trust of the students towards educational means. This system assists in guaranteeing regulatory compliance and reduces administrative expenses since it reduces the manual note-taking by auto-digitizing it. The device can be deployed in educational institutions due to the low cost, easy design, and scalability, which can encompass the deployment of the device in many operations within the educational contexts, leading to improved learning standards and long-term economical practice. The economic viability of this solution can be attributed to its integration of the capital cost at implementation and long-term gains that encompass economic efficiencies, improvement in the quality of education, and improvement in the performance of institutions in the education sector.

This report provides a payback analysis of the economic benefits of the Whiteboard Cleaning System, as well as the expenses that will be incurred and benefits and income that the system will bring to the institution. The analysis encompasses the initial costs, recurrent costs and the savings of ST (Student Tutor) consultation time.

Initial Costs of the System

The initial cost for developing the system includes the following components:

Table 11: Initial Cost

Component	Quantity	Unit Price (BDT)	Total (BDT)
ESP32	2	450	900
Motors	2	700	1,400
IBT2	1	500	500
Stepper Motor	1	745	745
Stepper Driver	1	180	180
Power Supply	1	860	860
Buck Converter	1	245	245
Switches	6	20	120
Reed Module	1	100	100
Camera	1	2,300	2,300
Frame	1	1,000	1,000
Total Cost			7,350

The labor cost for development is 5,000 BDT.

Total Initial Setup Cost: 7,350 + 5,000 = **12,350 BDT**

Selling the System to the Institution

The system is sold to the institution at a 60% profit margin. Thus, the selling price is calculated as follows:

Profit = 12, 350 × 0.60 = **7,410 BDT**

Selling Price = 12, 350 + 7, 410 = **19,760 BDT**

Recurring Costs for the Institution

The recurring costs for the institution involve the following:

- LLM API Cost for 6 Courses (3 months): 600 BDT per course
- For 6 courses, the total LLM API cost for 3 months is: 600 * 6 = 3,600 BDT
- The institution will pay 3,600 BDT + 60% profit margin:
  3,600 + (3,600 × 0.60)
  = 3, 600 + 2,160
  = 5,760 BDT for 6 courses in a semester

ST (Student Tutor) Cost Savings with the System

Traditional ST Consultation Model (Before the System)

- ST Consultation Hours per Month: 24 hours per course (2 hours/consultation, 3 per week)
- Cost per Hour: 7500/80 = 95 BDT (20 hour/week)
- Total Monthly ST Consultation Cost for Single Course:
  12 × 95 = 1,140 BDT per month
- For 3 months, the total ST consultation cost per course will be:
  1,140 × 3 = 3,420 BDT per course per semester
- For 6 courses, the total ST consultation cost for 3 months is:
  3,420 × 6 = 20,520 BDT for 6 courses for 3 months

ST Consultation Model with the System (After the System)

- Reduced Consultation Hours: 6 hours per month per course
- Cost per Hour: 95 BDT
- Total Monthly ST Consultation Cost for Single Course:
  6 × 95 = 570 BDT per month
- For 3 months, the total ST consultation cost per course will be:
  570 × 3 = 1, 710 BDT per course per semester
- For 6 courses, the total ST consultation cost for 3 months is:
  1, 710 × 6 = 10, 260 BDT for 6 courses for 3 months

Savings in ST Consultation Costs: The savings from using the system for 6 courses is:
$20{,}520 - 10{,}260 =$ **10,260 BDT** per semester for 6 courses

Profit for Service Provider from LLM API Sales

The profit from the LLM API sales is calculated at a 60% profit margin-
For 6 courses:
Profit from LLM API = $600 \times 0.60 = 360$ BDT per course per semester
Total Profit from LLM API = $360 \times 6 =$ **2,160 BDT** per semester

Payback Period for the Institution

Institution's Initial Investment
Total System Cost = 19,760 BDT

Annual Savings from ST Consultation
For 6 courses/semester, the total savings in ST consultation over 3 semesters will be:
$10{,}260 \times 3 =$ **30,780 BDT** per year for 6 courses

Net savings per year: $30{,}780 - 17{,}280 =$ **13,500 BDT** per year

Payback Period
Initial Investment = **19,760 BDT**
Annual Net Savings = **13,500 BDT**
Payback Period = Initial Investment/Annual Net Savings
= 19,760/13,500
≈ 1.4637 years
≈ 17.56 months

Thus, the institution will recover their investment in approximately 17.55 months.

Long-Term Benefits for the Institution

After the payback period, the institution will continue to save and benefit from the system.

- Annual Savings from ST Consultation: 30,780 BDT per year (for 6 courses/semester)
- Annual LLM API Cost: 17,280 BDT per year
- Total Annual Savings and Profit: 30,780 - 17,280 = **13,500 BDT** per year

8.3 Conclusion

- Initial Sale Price to the Institution: 19,760 BDT (includes 60% profit margin for provider)
- Annual Profit from LLM API for Service Provider: 2, 160 × 3 = 6, 480 BDT per year
- Payback Period for the Institution: 17.55 months
- Long-Term Benefits: After the payback period, the institution will save 13,500 BDT per year while still earning the service provider a recurring profit from the LLM API sales of around BDT 6,480 BDT per year.
- The system offers a strong return on investment for the institution within 17.55 months, with significant savings on ST consultation costs and also generates a recurring revenue of BDT 6,480 per year for the seller from the LLM API.

Chapter 9: Ethics and Professional Responsibilities CO13, CO2

9.1 Introduction

All activities in engineering depend on ethics and professional responsibility. All engineering projects must not only deliver technical endeavors but ensure that they are handling their integrity, equity, and responsibility to the community in which they operate. The role of them is particularly significant in teaching technologies like a Smart White board Management System (SWMS) as the system directly influences the way the knowledge is documented, stored and exchanged.SWMS makes classroom material automatic and creates notes which can be used by students to study and read. The task of the designers, therefore, lies in the ethical duty of making sure that the information generated (all the information) is correct, unobstructed and without misinformation. Any poor quality data or technical inaccurately extracted information can mislead learners as well as teachers posing a grave moral issue. This chapter aims at explaining how the use of professional ethics, accuracy, privacy and accountability are put in application in the design, implementation and operation of Smart White board management system.

9.2 Identify ethical issues and professional responsibility

The Smart White board management system incorporates theft of automation, artificial intelligence and storage of data-all of which have some ethical concerns. The system needs to responsibly, safely and in a just manner transform the hand written information into digital format since it is already captured and needs to be converted.

Accuracy in generation of notes is one of the key ethical duties. The notes produced by the system in digital form may not be altered; they should reflect the ones that were taken in the white board. Any false hood such as camera distortion, software malfunction and OCR misinterpretation can adversely impact students' cognition of lessons. To achieve this, the engineering team should conduct intensive testing and constant calibration in order to help in making sure that image capture, text extraction, and file generation are highly precise.

The other ethical issue is privacy of information. The system takes over classroom real-time data such as instructor handwriting, schemes, and academic contents which could be sensitive. Such information has to be safely stored by engineers and not abused or illegibly gotten. Encryption of the strongest level, access control, and safe management of files have to be established to safeguard teachers as well as students.

Honest system performance reports are also under professional responsibility. ERP design team cannot exaggerate the capabilities of the system and should openly give the limitations of the system like the dependency on lighting, camera resolution, or lag in processing. Having clear and concise documentation of the system is important as well since users can use the system safely and more so, without much confidence in their work.

Other ethical considerations are equity and inclusivity. The system must also be of low cost and accessible to institutions whether giants or not. Further development of a low-cost automation product would correspond to such an ethical principle as the establishment of equal access to education means. The common good must take precedence over commercial profit in the decision making of the engineers.

9.3 Apply ethical issues and professional responsibility

An ethical issue was approached in all levels when creating the SWMS. The team also considered all the hardware components throughout the design phase and they had to meet the electrical safety and environmental rules. The software has been coded by creating open-source libraries that prevent the committing of a licensing breach and transparency.

During the testing stage, numerous run-throughs were performed to test the precision of note extraction. Errors that were identified like text missing or missing false characters were then corrected until the algorithms brought out results that could be relied upon. Such repetitive testing contains a sense of professionalism and moral dedication to quality assurance.

In information processing, care was taken not to change or manipulate information. Lecture material is not subject to any modification except cleaning and tone enhancement by the system. This also makes sure that when the students get the content, it is a perfect copy of what the instructor wrote, thus academic honesty.

Sustainability-wise and safety to the user, the system uses minimal energy and does not emit dangerous gases and substances. The design promotes reusability and local production, which is in sync with ethical engineering ways of reducing wastes and environmental pollution.

9.4 Ethical use of Automation and AI

The phenomenon of automation implies both benefits and burdens. Simple image processing methods are capable of being improved to smarter content recognition paradigms in the future and the SWMS uses these basic methods. Most engineers need to make sure that these models do not produce wrong or misleading information as automation becomes better. An example is in case of misreading of a mathematical expression or symbol by the OCR, this may give a confusing result which could be incorrect in academics. Thus, engineers are ethically obligated to address validation provisions namely confidence thresholds and manual verification provision that reduces the misinformation.Moreover, engineers should not leave the system too reliant on automated decisions without the supervision of human hand. Ethical design promotes human checks that would check to ensure that the system's outputs are accurate especially when it is applied within education whereby it is important to have accuracy.

9.5 Data Privacy and Transparency

Given that the system has lecture data and digitized lectures, it is important to ensure data confidentiality. Saved content should only be accessible to the authorized personnel according to their levels like the instructor or institution administrator. Encryption and subsequent deletion of the stored data should be done after a set time to avoid abuse. The development team should also work to be compliant with institutional policies of data and local data protection regulatory policies.Another principle of ethical engineering is transparency. Comparability The team must make it clear to the users how they capture, store, and use their data. Recording background and other hidden data should never take place without the awareness of the user. Such transparency would bring in confidence and would associate the project with the principles of ethical use of technology in the world.

9.6 Professional Accountability

All project stages which engineers undertake come with the responsibility to exercise professional accountability. They also need to ensure that the system performs as expected, that testing procedures are duly documented, and that all the risks are made available to the users. Critical success factors In the event of any failure - software bugs, hardware, and data inaccuracy, the concerned team must resolve the matter on time, and in an indecisive manner.

Additionally, engineers should always be dedicated towards the issue of constantly learning and improving and ensure the system is up to date with the new ethical and technical standards. This is not just limited to the termination of the project end but ensuring a long term reliability and the users trusting the profession.

9.7 Conclusion

Whiteboard Management system. Since it is an educational automation, it directly affects the presentation of information to students and the information they will use. All of these notes produced by the system, thus, have to be correct, honest, and devoid of incorrect information. The engineers need to remain honest, transparent, and accountable when designing and in data management.The project will facilitate accessibility, equity, and accuracy of education in accordance with the ethical motives of capturing high-resolution data, exemplary transfer of images, and sound data management systems. Ethical engineering does not only advance credibility of a given system but it helps to build the trust of the society as well, which means that the partnership of technology with society and educational principles is maintained.

Chapter 10: Conclusion and Future Work.

10.1 Project summary/Conclusion

This system makes use of dual modalities for automatic cleaning of whiteboards along with digitization of lecture notes. This system satisfied crucial educational needs while providing an entire set of notes. The motors in this cleaning system were sufficient. The image processing component uses rule-based algorithms to assess content parameters. The blend of these technologies allows us to carry out assessments quickly and non-destructively. It can bring about changes in face-to-faced working methods and management in the classrooms. Automating whiteboard cleaning presents new potential methods for education to move away from manual slow management of whiteboards and enhance sharing of lecture notes. Our research and developed system prove that with the help of automated systems, we can provide educators and students with an accessible and scalable solution to help protect them from inefficiencies, simultaneously ensuring consistent content access.

10.2 Future work

Several promising avenues for future research and development have been identified:

1. Miniaturization and Portability: Further development work on the main body of the project could help reduce its size, creating a handheld device that would be easier to carry and more suitable for mobile classrooms by educators and students.

2. Expanded Content Detection: By expanding the capture array, it could be made adaptable to detect more content types, such as multimedia annotations, different languages, etc.

3. Enhanced Rule-Based Integration: By implementing more advanced rule-based techniques, the digitization accuracy could be improved for reduced computational requirements, potentially creating opportunities for real-time analysis on mobile devices.

4. Development of Mobile Application: By creating an application for smartphones that would interface with simplified systems, users can be empowered to manage content on their own.

5. Multi-board Adaptability: By extending the detection and classification abilities of the system to accommodate other boards and surfaces that are commonly used, the utility of the device can be increased.

6. Industrialization: By adapting the dataset and integrating better cameras or scanners into the system, the device can be used directly in large educational institutions. We can address the problem at the source by determining and optimizing classroom workflows.

Following through with these future directions, the technology we created has the potential to evolve into a global solution for ensuring educational efficiency and content management across global institutions. This will help in lowering the educational risks associated with manual processes and simultaneously improving the content assessment technology.

Chapter 11: Identification of Complex Engineering Problems and Activities.

11.1: Identify the attribute of complex engineering problem (EP)

Table 12: Attribute of complex engineering problem (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: Provide reasoning how the project address selected attribute (EP)

P1. Depth of knowledge required: The project requires knowledge of multiple fields which comes together in harmony to see the project to fruition.

- Knowledge of Embedded systems, IoT, image processing, AI
- Knowledge of Rule-based algorithms
- Knowledge of Design Process
- Knowledge of Tools required to develop the solution

P3. Depth of analysis required

- No single way to design the solution
- Depth of analysis is required to find the different alternative solutions and find the optimal/best solutions among alternatives.

P4. Familiarity of issues

- The project involves infrequently encountered issues.
- Educational and technological issues are not usually taught in the EEE/ECE curriculum. Students need to learn these unfamiliar issues by themselves while designing the solution.

P6. Extent of stakeholder involvement and needs

- Students – Individuals using the system who need assurance of note accessibility.
- Educators & Institutions – Producers who want to ensure that their lectures meet educational standards and remain free from inefficiencies.
- Administrators & Staff – Managers who need to maintain high-quality tools and build stakeholder trust.
- Educational Authorities & Regulatory Bodies – Organizations responsible for enforcing educational standards and ensuring compliance with regulations.
- Schools & Universities – Large institutions who require reliable content management before delivering education to students.

P7. Interdependence Project involves a number of interdependent sub systems working together in symphony.

11.3 Identify the attribute of complex engineering activities (EA)

Table 13: Attribute 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	√
P6 Extent of stakeholder involvement and needs	
P7 Interdependence	√

11.4 Provide reasoning how the project address selected attribute (EA)

A1. Range of resource Communication is required to mobilize various resources including money, people, equipment and information to implement the project.

A4. Consequences for society and the environment Society will fare much better due to the improved educational efficiency and reduced waste.

A5. Familiarity This is an unfamiliar area for EEE/ECE students

P6: Multiple stakeholders are involved: faculty, students, and the institution.

Their requirements directly affect system design, usability, and functionality, so this attribute is ticked.

P7: Hardware, software, and control logic are highly interconnected.

Failure or changes in one module (e.g., camera, motors, or software) affect overall system performance, making interdependence a key factor.

References

[1] M. T. S. Ahamed, “Smart classroom technologies: A review on automation and learning enhancement,” IEEE Access, vol. 10, pp. 11543–11556, 2022.

[2] S. K. Saha and M. A. Rahman, “IoT-based classroom automation for sustainable education,” Proc. IEEE Int. Conf. Sustainable Technologies (SusTech), pp. 97–102, 2021.

[3] P. K. Mishra, “Design and implementation of automated board cleaner using microcontroller,” Int. J. Innovative Res. Electr. Electron. Instrum. Control Eng., vol. 8, no. 2, pp. 24–29, 2020.

[4] S. Dutta and A. Das, “A low-cost robotic board eraser system using Arduino,” IEEE Region 10 Humanitarian Technol. Conf. (R10-HTC), pp. 455–460, 2020.

[5] M. H. Rahman and T. Islam, “Development of automated whiteboard cleaning system,” Proc. Int. Conf. Comput. Commun. Chem. Mater. Electron. Eng. (IC4ME2), pp. 431–436, 2021.

[6] A. K. Singh and R. Gupta, “Smart classroom: Digital transformation using AI and IoT,” IEEE Access, vol. 9, pp. 122055–122070, 2021.

[7] R. Sharma, “Automation and image capture system for interactive teaching boards,” IEEE Int. Conf. Emerging Smart Computing & Informatics (ESCI), pp. 224–230, 2020.

[8] M. N. Hossain, “Intelligent smartboard: A vision-based content capturing system using OpenCV,” Int. J. Eng. Sci. Comput., vol. 9, no. 3, pp. 21134–21139, 2019.

[9] A. Paul and S. Roy, “Real-time OCR using Tesseract for digitizing handwritten educational notes,” Proc. IEEE Int. Conf. Comput. Sci. Comput. Intell. (CSCI), pp. 1210–1215, 2021.

[10] B. C. Lee and Y. J. Kim, “Vision-based teaching aid for automatic note recording,” Sensors, vol. 21, no. 18, p. 6102, 2021.

[11] J. Zhang and H. Zhang, “Automatic classroom note generation using deep learning and image segmentation,” IEEE Access, vol. 8, pp. 229432–229445, 2020.

[12] R. M. Rahman and S. U. Ahmed, “Microcontroller-based automated erasing and image saving system,” Proc. IEEE Int. Conf. Informatics, Electron. Vision (ICIEV), pp. 67–72, 2022.

[13] P. R. Singh and D. Kumar, “Integration of IoT and cloud in education systems,” Int. J. Comput. Appl., vol. 182, no. 35, pp. 14–19, 2021.

[14] A. P. Jain and B. K. Sahu, “Low-cost automation of whiteboard using Raspberry Pi,” Proc. IEEE Int. Conf. Power, Control, Signals and Instrumentation Eng. (ICPCSI), pp. 423–428, 2019.

[15] H. Lin and M. Chen, “Real-time edge detection in handwritten content digitization,” Pattern Recognit. Lett., vol. 150, pp. 90–97, 2021.

[16] S. Yadav, “AI-based classroom management: Smart learning with vision systems,” IEEE Trans. Learn. Technol., vol. 15, no. 3, pp. 287–298, 2022.

[17] F. T. Hasan, “Automation in education through embedded systems: A review,” Int. J. Recent Technol. Eng., vol. 8, no. 5, pp. 401–407, 2020.

[18] M. J. Alam, "Digital transformation in Bangladesh's higher education through smart automation," BRAC Univ. J. Eng. Sci., vol. 18, no. 2, pp. 56–63, 2022.

[19] S. H. Lee, "Optical character recognition for handwritten boards using CNN-LSTM hybrid models," IEEE Access, vol. 10, pp. 98123–98135, 2022.

[20] J. Hossain and A. K. Khan, "Smart classroom automation using ESP32 microcontroller," Proc. IEEE Int. Conf. Smart Electr. Technol. (ICSET), pp. 231–236, 2021.

[21] P. Das and K. Roy, "Wireless control of classroom automation using IoT," Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 6, pp. 215–222, 2021.

[22] R. Ahmed and F. Chowdhury, "Design of an automatic whiteboard cleaner using gear motor mechanism," Proc. IEEE Int. Conf. Robot. Autom. (ICRA), pp. 145–150, 2020.

[23] S. Mahmud, "Enhancing classroom sustainability with automated whiteboards," Int. J. Eng. Res. Technol., vol. 9, no. 11, pp. 342–348, 2020.

[24] K. S. Hasan, "Wireless communication protocols for embedded automation," IEEE Commun. Surv. Tutor., vol. 23, no. 1, pp. 88–104, 2021.

[25] M. B. Rahman, "A study on intelligent IoT-based classroom environments," J. Eng. Sci. Technol., vol. 16, no. 5, pp. 4140–4150, 2021.

[26] T. N. Nguyen, "AI-integrated OCR systems for educational note digitization," Appl. Sci., vol. 12, no. 3, p. 1192, 2022.

[27] P. K. Dey, "Smart whiteboard: Vision-based real-time erasing and content capture," IEEE Region 10 Conf. (TENCON), pp. 2140–2145, 2021.

[28] L. A. Smith, "Digital classrooms and ethical technology design," IEEE Technol. Soc. Mag., vol. 40, no. 4, pp. 50–58, 2021.

[29] N. Roy, "Human-centered design principles in educational robotics," Int. J. Human-Comput. Interact., vol. 37, no. 14, pp. 1310–1321, 2021.

[30] M. I. Khan, "Secure cloud-based data storage for smart education systems," IEEE Trans. Cloud Comput., vol. 10, no. 6, pp. 1218–1229, 2022.

[31] S. Uddin and M. Alam, "Energy efficiency in embedded learning automation," Renew. Energy Technol. Educ., vol. 7, no. 1, pp. 42–51, 2022.

[32] A. Rahman and N. Akter, "Design and analysis of motor control for automated educational tools," Proc. IEEE Int. Conf. Ind. Electron. Appl., pp. 431–438, 2020.

[33] J. Kim and D. Park, "Real-time data acquisition for classroom IoT systems," IEEE Internet Things J., vol. 8, no. 9, pp. 7752–7761, 2021.

[34] M. A. Hasan, "Smart automation projects for academic institutions," Proc. IEEE Global Eng. Educ. Conf. (EDUCON), pp. 114–121, 2020.

[35] R. C. Ranjan, “AI-driven content verification for educational systems,” IEEE Access, vol. 11, pp. 38210–38221, 2023.

[36] A. J. Banerjee, “Safety and ethical practices in robotics-based automation,” IEEE Trans. Eng. Manage., vol. 69, no. 5, pp. 1943–1952, 2022.

[37] A. Gupta and R. Sen, “Professional ethics and accountability in modern engineering,” IEEE Technol. Soc. Mag., vol. 39, no. 3, pp. 45–53, 2020.

[38] M. Chowdhury and S. Haque, “Sustainability and ethics in embedded educational systems,” Int. J. Sci. Eng. Res., vol. 11, no. 9, pp. 92–99, 2020.

[39] T. Rahman, “Professional responsibility in educational automation: A review,” IEEE Access, vol. 11, pp. 19087–19095, 2023.

[40] B. H. Lee, “Ethical design considerations for AI-driven learning environments,” IEEE Trans. Learn. Technol., vol. 16, no. 1, pp. 88–98, 2023.

Appendix

Code

Basic Code:

```
// Pin Definitions

#define RPWM 18        // Movement motor right PWM

#define LPWM 19        // Movement motor left PWM

#define R_EN 23        // Movement motor right enable

#define L_EN 21        // Movement motor left enable

#define REED_POWER 22

#define STEPPER_IN1 14 // Stepper coil1 input A

#define STEPPER_IN2 27 // Stepper coil1 input B

#define STEPPER_IN3 26 // Stepper coil2 input A

#define STEPPER_IN4 25 // Stepper coil2 input B

#define REED_PIN 33    // Reed switch for magnet detection

#define MAIN_LIMIT_PIN 32  // Main limit switch

#define RIGHT_LIMIT_PIN 12 // Right limit switch

#define LEFT_LIMIT_PIN 13  // Left limit switch


// Constants

#define STEPS_PER_REV 200   // steps per revolution

#define DEBOUNCE_DELAY 500  // debounce in ms

#define STEP_DELAY 3        // delay per step (speed control)

#define REED_COOLDOWN 200  // ms to ignore reed after detection


// Global Variables

bool lastState = HIGH;      // reed state tracking
```

```
bool turnCCW = true;        // stepper direction toggle
bool systemActive = false;  // main system state
bool movingCW = true;       // DC motor direction
bool motorRunning = false;  // motor running state

unsigned long lastReedTime = 0; // cooldown timer

// Stepper sequence (full-step, 2-phase on)
const int stepSeq[4][4] = {
  {1, 0, 1, 0},  // Step 1
  {0, 1, 1, 0},  // Step 2
  {0, 1, 0, 1},  // Step 3
  {1, 0, 0, 1}   // Step 4
};
int stepIndex = 0;

void setup() {
  Serial.begin(115200);

  // Movement motor pins
  pinMode(RPWM, OUTPUT);
  pinMode(LPWM, OUTPUT);
  pinMode(R_EN, OUTPUT);
  pinMode(L_EN, OUTPUT);
  pinMode(REED_POWER, OUTPUT);

  // Stepper motor pins
  pinMode(STEPPER_IN1, OUTPUT);
  pinMode(STEPPER_IN2, OUTPUT);
```

```
  pinMode(STEPPER_IN3, OUTPUT);

  pinMode(STEPPER_IN4, OUTPUT);


  // Reed switch with pullup

  pinMode(REED_PIN, INPUT_PULLUP);


  // Limit switches with pullup

  pinMode(MAIN_LIMIT_PIN, INPUT_PULLUP);

  pinMode(RIGHT_LIMIT_PIN, INPUT_PULLUP);

  pinMode(LEFT_LIMIT_PIN, INPUT_PULLUP);


  // Motor driver initially OFF

  stopmotor();


  // Stepper initially OFF

  stepOff();


  Serial.println("System initialized. Waiting for main limit switch...");

}


void loop() {

  if (!systemActive) {

    if (digitalRead(MAIN_LIMIT_PIN) == LOW) {

      Serial.println("Main limit switch pressed. System activated!");

      digitalWrite(REED_POWER, HIGH);

      rotateClockwise();

      systemActive = true;

      movingCW = true;

      motorRunning = true;
```

```
    }
  } else {
    if (digitalRead(RIGHT_LIMIT_PIN) == LOW && movingCW) {
      Serial.println("Right limit switch triggered! Rotating anticlockwise...");
      rotateAnticlockwise();
      movingCW = false;
      motorRunning = true;
    }

    if (digitalRead(LEFT_LIMIT_PIN) == LOW) {
      Serial.println("Left limit switch triggered! Motor stopped, system deactivated.");
      stopmotor();
      systemActive = false;
      motorRunning = false;
    }
    if (systemActive) {
      checkMagnetDetection();
    }
  }
}

// ---- Motor Control Functions ----
void rotateClockwise() {
  digitalWrite(R_EN, HIGH);
  digitalWrite(L_EN, HIGH);
  digitalWrite(RPWM, HIGH);
  digitalWrite(LPWM, LOW);
```

```
  motorRunning = true;
}

void rotateAnticlockwise() {
  digitalWrite(R_EN, HIGH);
  digitalWrite(L_EN, HIGH);
  digitalWrite(RPWM, LOW);
  digitalWrite(LPWM, HIGH);
  motorRunning = true;
}

void stopmotor() {
  digitalWrite(R_EN, LOW);
  digitalWrite(L_EN, LOW);
  digitalWrite(RPWM, LOW);
  digitalWrite(LPWM, LOW);
  motorRunning = false;
}

// ---- Stepper Functions ----
void stepOnce(bool dir) {
  if (dir) {
    stepIndex = (stepIndex + 1) % 4;
  } else {
    stepIndex = (stepIndex + 3) % 4; // backwards
  }
  digitalWrite(STEPPER_IN1, stepSeq[stepIndex][0]);
  digitalWrite(STEPPER_IN2, stepSeq[stepIndex][1]);
```

```
  digitalWrite(STEPPER_IN3, stepSeq[stepIndex][2]);
  digitalWrite(STEPPER_IN4, stepSeq[stepIndex][3]);
  delay(STEP_DELAY);
}

void stepTurns(int turns, bool dir) {
  for (long s = 0; s < (long)turns * STEPS_PER_REV; s++) {
    stepOnce(dir);
  }
  stepOff(); // release coils after motion
}

void stepOff() {
  digitalWrite(STEPPER_IN1, LOW);
  digitalWrite(STEPPER_IN2, LOW);
  digitalWrite(STEPPER_IN3, LOW);
  digitalWrite(STEPPER_IN4, LOW);
}

// ---- Event Handling ----
void checkMagnetDetection() {
  bool state = digitalRead(REED_PIN);

  // Only trigger if cooldown has passed
    if (state == LOW && lastState == HIGH && motorRunning && millis() - lastReedTime >
REED_COOLDOWN) {
    Serial.println("Magnet detected! Stopping motor...");
    stopmotor();
```

```
    if (turnCCW) {
      Serial.println("Stepper rotating CCW 5 turns...");
      stepTurns(5, false);
    } else {
      Serial.println("Stepper rotating CW 5 turns...");
      stepTurns(5, true);
    }

    turnCCW = !turnCCW; // toggle next stepper direction
    lastReedTime = millis(); // update cooldown timer

    // Resume motor in previous direction
    if (movingCW) {
      rotateClockwise();
      Serial.println("Resuming motor CW...");
    } else {
      rotateAnticlockwise();
      Serial.println("Resuming motor CCW...");
    }
  }

  lastState = state;
}
```

ESP2(RECEIVER_Type C) MAC address-6C:C8:40:8E:8C:A8
ESP1(SENDER_Type B) MAC address-2C:BC:BB:0C:17:F0

Sender Code:

```
#include <esp_now.h>
```

```
#include <WiFi.h>

// ---------------- Pins ----------------
#define RPWM 18
#define LPWM 19
#define R_EN 23
#define L_EN 21
#define REED_POWER 22
#define STEPPER_IN1 14
#define STEPPER_IN2 27
#define STEPPER_IN3 26
#define STEPPER_IN4 25
#define REED_PIN 33
#define MAIN_LIMIT_PIN 32   // Switch 1
#define RIGHT_LIMIT_PIN 12
#define LEFT_LIMIT_PIN 13
#define SWITCH2_PIN 15      // Switch 2

// ---------------- Constants ----------------
#define STEPS_PER_REV 200
#define STEP_DELAY 3
#define REED_COOLDOWN 200
#define LIMIT_DEBOUNCE 200   // ms

// ---------------- State ----------------
bool lastReedState = HIGH;
bool turnCCW = true;
bool systemActive = false;
```

```
bool movingCW = true;
bool motorRunning = false;

unsigned long lastReedTime = 0;
unsigned long lastLimitTime = 0;

// Stepper sequence
const int stepSeq[4][4] = {
  {1, 0, 1, 0},
  {0, 1, 1, 0},
  {0, 1, 0, 1},
  {1, 0, 0, 1}
};
int stepIndex = 0;

// ---------------- ESP-NOW ----------------
uint8_t receiverAddress[] = {0x6C, 0xC8, 0x40, 0x8E, 0x9E, 0xAC};

// ---------------- ESP-NOW Callbacks ----------------
void OnDataSent(const esp_now_send_info_t *info, esp_now_send_status_t status) {
  Serial.print("Send Status: ");
  Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Success" : "Fail");
}

void OnDataRecv(const esp_now_recv_info_t *info, const uint8_t *incomingData, int len) {
  String msg;
  for (int i = 0; i < len; i++) msg += (char)incomingData[i];
```

```
  Serial.print("Received: ");
  Serial.println(msg);

  if (msg == "Done" && !systemActive) {
    Serial.println("Reply 'Done' received → Starting cleaning automatically...");
    startCleaning();
  }
}

// ---------------- Cleaning Function ----------------
void startCleaning() {
  digitalWrite(REED_POWER, HIGH);
  rotateClockwise();
  systemActive = true;
  movingCW = true;
  motorRunning = true;
}

// ---------------- Setup ----------------
void setup() {
  Serial.begin(115200);


  pinMode(RPWM, OUTPUT);
  pinMode(LPWM, OUTPUT);
  pinMode(R_EN, OUTPUT);
  pinMode(L_EN, OUTPUT);
  pinMode(REED_POWER, OUTPUT);
```

```
pinMode(STEPPER_IN1, OUTPUT);
pinMode(STEPPER_IN2, OUTPUT);
pinMode(STEPPER_IN3, OUTPUT);
pinMode(STEPPER_IN4, OUTPUT);

pinMode(REED_PIN, INPUT_PULLUP);
pinMode(MAIN_LIMIT_PIN, INPUT_PULLUP);
pinMode(RIGHT_LIMIT_PIN, INPUT_PULLUP);
pinMode(LEFT_LIMIT_PIN, INPUT_PULLUP);
pinMode(SWITCH2_PIN, INPUT_PULLUP);

stopmotor();
stepOff();

WiFi.mode(WIFI_STA);
if (esp_now_init() != ESP_OK) {
  Serial.println("ESP-NOW init failed");
  return;
}
esp_now_register_send_cb(OnDataSent);
esp_now_register_recv_cb(OnDataRecv);

esp_now_peer_info_t peerInfo = {};
memcpy(peerInfo.peer_addr, receiverAddress, 6);
peerInfo.channel = 0;
peerInfo.encrypt = false;
```

```
  if (esp_now_add_peer(&peerInfo) != ESP_OK) {
    Serial.println("Failed to add peer");
    return;
  }

  Serial.println("System ready. Switch1 = cleaning, Switch2 = send+reply+clean automatically.");
}

// ---------------- Loop ----------------
void loop() {
  // ---- Switch 1 (normal cleaning) ----
  if (!systemActive && digitalRead(MAIN_LIMIT_PIN) == LOW) {
    Serial.println("Switch 1 pressed → Cleaning started.");
    startCleaning();
    delay(300);
  }

  // ---- Switch 2 (send message to receiver) ----
  if (!systemActive && digitalRead(SWITCH2_PIN) == LOW) {
    Serial.println("Switch 2 pressed → Sending 'Click'");
    const char *msg = "Click";
    esp_now_send(receiverAddress, (uint8_t *)msg, strlen(msg));
    delay(300);
  }

// ---- Limit switch handling ----
```

```
if (systemActive) {
  // ✅ Right limit (only if truly pressed)
  if (movingCW) {
    if (digitalRead(RIGHT_LIMIT_PIN) == LOW) {
      delay(30); // debounce
      if (digitalRead(RIGHT_LIMIT_PIN) == LOW) {
        Serial.println("Right limit → CCW");
        rotateAnticlockwise();
        movingCW = false;
      }
    }
  }

  // ✅ Left limit (only if truly pressed)
  if (digitalRead(LEFT_LIMIT_PIN) == LOW) {
    delay(30); // debounce
    if (digitalRead(LEFT_LIMIT_PIN) == LOW) {
      Serial.println("Left limit → Back off 200ms CW then Stop");

      rotateClockwise();
      delay(50);

      stopmotor();
      systemActive = false;
    }
  }
```

```
  checkMagnetDetection();
 }

}

// ---------------- Motor Functions ----------------
void rotateClockwise() {
  digitalWrite(R_EN, HIGH);
  digitalWrite(L_EN, HIGH);
  digitalWrite(RPWM, HIGH);
  digitalWrite(LPWM, LOW);
  motorRunning = true;
}
void rotateAnticlockwise() {
  digitalWrite(R_EN, HIGH);
  digitalWrite(L_EN, HIGH);
  digitalWrite(RPWM, LOW);
  digitalWrite(LPWM, HIGH);
  motorRunning = true;
}
void stopmotor() {
  digitalWrite(R_EN, LOW);
  digitalWrite(L_EN, LOW);
  digitalWrite(RPWM, LOW);
  digitalWrite(LPWM, LOW);
```

```
  motorRunning = false;

}

// ---------------- Stepper Functions ----------------

void stepOnce(bool dir) {

  stepIndex = dir ? (stepIndex + 1) % 4 : (stepIndex + 3) % 4;

  digitalWrite(STEPPER_IN1, stepSeq[stepIndex][0]);

  digitalWrite(STEPPER_IN2, stepSeq[stepIndex][1]);

  digitalWrite(STEPPER_IN3, stepSeq[stepIndex][2]);

  digitalWrite(STEPPER_IN4, stepSeq[stepIndex][3]);

  delay(STEP_DELAY);

}

void stepTurns(int turns, bool dir) {

  for (long s = 0; s < (long)turns * STEPS_PER_REV; s++) stepOnce(dir);

  stepOff();

}

void stepOff() {

  digitalWrite(STEPPER_IN1, LOW);

  digitalWrite(STEPPER_IN2, LOW);

  digitalWrite(STEPPER_IN3, LOW);

  digitalWrite(STEPPER_IN4, LOW);

}

// ---------------- Reed (Magnet Detection) ----------------

void checkMagnetDetection() {

  bool state = digitalRead(REED_PIN);
```

```
  if (state == LOW && lastReedState == HIGH && motorRunning && millis() - lastReedTime > REED_COOLDOWN) {

    Serial.println("Magnet detected! Stopping motor...");

    stopmotor();



    if (turnCCW) {

      Serial.println("Stepper CCW 5 turns...");

      stepTurns(8, false);

    } else {

      Serial.println("Stepper CW 5 turns...");

      stepTurns(8, true);

    }

    turnCCW = !turnCCW;

    lastReedTime = millis();



    if (movingCW) rotateClockwise();

    else rotateAnticlockwise();

  }

  lastReedState = state;

}
```

Receiver Code:

```
#include <esp_now.h>
#include <WiFi.h>



// Replace with your sender's MAC

uint8_t senderAddress[] = {0x2C, 0xBC, 0xBB, 0x0C, 0x17, 0xF0};
```

```
// Flag to indicate waiting for PC Done
bool waitingForDone = false;

void OnDataSent(const esp_now_send_info_t *info, esp_now_send_status_t status) {
  Serial.print("Reply send status: ");
  Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Success" : "Fail");
}

void OnDataRecv(const esp_now_recv_info_t *info, const uint8_t *incomingData, int len) {
  String msg;
  for (int i = 0; i < len; i++) msg += (char)incomingData[i];
  Serial.print("Received via ESP-NOW: ");
  Serial.println(msg);

  if (msg.equalsIgnoreCase("Click")) {
    Serial.println("Triggering capture on PC...");
    Serial.println("Capture");  // Send "Capture" to PC via Serial
    waitingForDone = true;      // Now wait for PC to send Done
  }
}
void setup() {
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);

  if (esp_now_init() != ESP_OK) {
    Serial.println("ESP-NOW init failed");
```

```
    return;
  }

  esp_now_register_send_cb(OnDataSent);
  esp_now_register_recv_cb(OnDataRecv);


  esp_now_peer_info_t peerInfo = {};
  memcpy(peerInfo.peer_addr, senderAddress, 6);
  peerInfo.channel = 0;
  peerInfo.encrypt = false;
  esp_now_add_peer(&peerInfo);


  Serial.println("Receiver ready...");
}

void loop() {
  // Check Serial for "Done" from PC
  if (waitingForDone && Serial.available()) {
    String serialMsg = Serial.readStringUntil('\n');
    serialMsg.trim();


    if (serialMsg.equalsIgnoreCase("Done")) {
      Serial.println("PC finished capture, sending Done via ESP-NOW...");
      esp_now_send(senderAddress, (uint8_t *)"Done", 4);  // Send back to sender ESP
      waitingForDone = false;
    }
  }
```

}

Web Apps Code:

```
import os
import zipfile
from datetime import datetime
from functools import wraps
import io, zipfile
from flask import Flask, render_template, request, redirect, url_for, flash, send_from_directory, jsonify,
send_file
from flask_sqlalchemy import SQLAlchemy
from flask_login import LoginManager, login_user, login_required, logout_user, current_user, UserMixin
from werkzeug.security import generate_password_hash, check_password_hash
from werkzeug.utils import secure_filename
import secrets

# ---------------- Configuration ----------------
BASE_DIR = os.path.abspath(os.path.dirname(__file__))
UPLOAD_FOLDER = os.path.join(BASE_DIR, "uploads")
MATERIALS_FOLDER = os.path.join(UPLOAD_FOLDER, "materials")
PENDING_FOLDER = os.path.join(UPLOAD_FOLDER, "pending")

os.makedirs(MATERIALS_FOLDER, exist_ok=True)
os.makedirs(PENDING_FOLDER, exist_ok=True)

app = Flask(__name__)
app.config["SECRET_KEY"] = os.environ.get("SECRET_KEY", "dev-secret-key")
app.config["SQLALCHEMY_DATABASE_URI"] = "sqlite:///" + os.path.join(BASE_DIR, "app.db")
app.config["SQLALCHEMY_TRACK_MODIFICATIONS"] = False
app.config["UPLOAD_FOLDER"] = UPLOAD_FOLDER
app.config["MAX_CONTENT_LENGTH"] = 100 * 1024 * 1024  # 100 MB per request

db = SQLAlchemy(app)
login_manager = LoginManager(app)
login_manager.login_view = "login"

# ---------------- Models ----------------
class User(UserMixin, db.Model):
    id = db.Column(db.Integer, primary_key=True)
    email = db.Column(db.String(120), unique=True, nullable=False)
    name = db.Column(db.String(120), nullable=False)
    password_hash = db.Column(db.String(255), nullable=False)
    role = db.Column(db.String(20), nullable=False)  # 'admin', 'faculty', 'student'
    approved = db.Column(db.Boolean, default=False)  # for faculty approval

    def set_password(self, password):
        self.password_hash = generate_password_hash(password)

    def check_password(self, password):
        return check_password_hash(self.password_hash, password)
```

```
class Course(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    title = db.Column(db.String(200), nullable=False)
    code = db.Column(db.String(6), unique=True, nullable=False)
    faculty_id = db.Column(db.Integer, db.ForeignKey("user.id"), nullable=False)
    created_at = db.Column(db.DateTime, default=datetime.utcnow)

    faculty = db.relationship("User", backref="courses", foreign_keys=[faculty_id])

class Enrollment(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    student_id = db.Column(db.Integer, db.ForeignKey("user.id"), nullable=False)
    course_id = db.Column(db.Integer, db.ForeignKey("course.id"), nullable=False)
    created_at = db.Column(db.DateTime, default=datetime.utcnow)

    student = db.relationship("User", foreign_keys=[student_id])
    course = db.relationship("Course", foreign_keys=[course_id])

class Material(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    course_id = db.Column(db.Integer, db.ForeignKey("course.id"), nullable=False)
    title = db.Column(db.String(200), nullable=False)
    filename = db.Column(db.String(300), nullable=False)  # stored on disk
    uploaded_at = db.Column(db.DateTime, default=datetime.utcnow)

    course = db.relationship("Course", backref="materials")

class PendingNote(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    faculty_id = db.Column(db.Integer, db.ForeignKey("user.id"), nullable=False)
    course_id = db.Column(db.Integer, db.ForeignKey("course.id"), nullable=False)
    title = db.Column(db.String(200), nullable=False)
    created_at = db.Column(db.DateTime, default=datetime.utcnow)

    faculty = db.relationship("User", foreign_keys=[faculty_id])
    course = db.relationship("Course", foreign_keys=[course_id])

# ---------------- Auth ----------------
@login_manager.user_loader
def load_user(user_id):
    return User.query.get(int(user_id))

def ensure_default_admin():
    admin = User.query.filter_by(email="admin@bracu.ac.bd").first()
    if not admin:
        admin = User(email="admin@bracu.ac.bd", name="Admin", role="admin", approved=True)
        admin.set_password("12345678")
        db.session.add(admin)
        db.session.commit()

# -------------- Utility --------------
```

```python
def generate_class_code():
    while True:
        code = f"{secrets.randbelow(1000000):06d}"
        if not Course.query.filter_by(code=code).first():
            return code

def allowed_pdf(filename):
    return "." in filename and filename.rsplit(".", 1)[1].lower() == "pdf"

def api_auth_required(f):
    """Basic Auth for API (faculty approved only)."""
    @wraps(f)
    def wrapper(*args, **kwargs):
        auth = request.authorization
        if not auth:
            return jsonify({"error": "Missing Authorization header"}), 401
        user = User.query.filter_by(email=auth.username).first()
        if not user or not user.check_password(auth.password):
            return jsonify({"error": "Invalid credentials"}), 401
        if user.role != "faculty" or not user.approved:
            return jsonify({"error": "Not an approved faculty"}), 403
        # attach user to request context
        request.user = user
        return f(*args, **kwargs)
    return wrapper

# ---------------- Routes ----------------
@app.route("/")
def index():
    return render_template("index.html")

# ------ Auth pages ------
@app.route("/signup", methods=["GET", "POST"])
def signup():
    if request.method == "POST":
        email = request.form.get("email", "").strip().lower()
        name = request.form.get("name", "").strip()
        password = request.form.get("password", "").strip()
        role = request.form.get("role", "").strip()  # faculty or student

        if not email or not name or not password or role not in ("faculty", "student"):
            flash("Please fill in all fields correctly.", "danger")
            return redirect(url_for("signup"))
        if User.query.filter_by(email=email).first():
            flash("Email already registered.", "warning")
            return redirect(url_for("signup"))

        user = User(email=email, name=name, role=role, approved=(role=="student"))
        user.set_password(password)
        db.session.add(user)
        db.session.commit()
```

```
        if role == "student":
            flash("Account created! You can log in now.", "success")
        else:
            flash("Faculty account created! Wait for admin approval.", "info")
        return redirect(url_for("login"))
    return render_template("signup.html")

@app.route("/login", methods=["GET", "POST"])
def login():
    if request.method == "POST":
        email = request.form.get("email","").lower().strip()
        password = request.form.get("password","")
        user = User.query.filter_by(email=email).first()
        if not user or not user.check_password(password):
            flash("Invalid email or password.", "danger")
            return redirect(url_for("login"))
        if user.role == "faculty" and not user.approved:
            flash("Faculty account pending approval.", "warning")
            return redirect(url_for("login"))
        login_user(user)
        flash("Welcome back!", "success")
        if user.role == "admin":
            return redirect(url_for("admin_dashboard"))
        elif user.role == "faculty":
            return redirect(url_for("faculty_dashboard"))
        else:
            return redirect(url_for("student_dashboard"))
    return render_template("login.html")

@app.route("/logout")
@login_required
def logout():
    logout_user()
    flash("Logged out.", "info")
    return redirect(url_for("index"))

# ------ Admin ------
@app.route("/admin")
@login_required
def admin_dashboard():
    if current_user.role != "admin":
        return redirect(url_for("index"))
    pending_faculty = User.query.filter_by(role="faculty", approved=False).all()
    courses = Course.query.order_by(Course.created_at.desc()).all()
    return render_template("admin_dashboard.html", pending_faculty=pending_faculty, courses=courses)

@app.route("/admin/approve/<int:user_id>", methods=["POST"])
@login_required
def approve_faculty(user_id):
    if current_user.role != "admin":
        return redirect(url_for("index"))
    user = User.query.get_or_404(user_id)
```

```
    if user.role == "faculty":
        user.approved = True
        db.session.commit()
        flash(f"Approved {user.name}.", "success")
    return redirect(url_for("admin_dashboard"))

@app.route("/admin/profile", methods=["GET", "POST"])
@login_required
def admin_profile():
    if current_user.role != "admin":
        return redirect(url_for("index"))
    if request.method == "POST":
        new_email = request.form.get("email","").strip().lower()
        new_password = request.form.get("password","").strip()
        if new_email:
            if User.query.filter(User.email==new_email, User.id!=current_user.id).first():
                flash("Email already in use.", "danger")
            else:
                current_user.email = new_email
        if new_password:
            if len(new_password) < 6:
                flash("Password too short.", "danger")
            else:
                current_user.set_password(new_password)
        db.session.commit()
        flash("Profile updated.", "success")
        return redirect(url_for("admin_profile"))
    return render_template("admin_profile.html")

# ------ Faculty ------
@app.route("/faculty")
@login_required
def faculty_dashboard():
    if current_user.role != "faculty":
        return redirect(url_for("index"))
    courses = Course.query.filter_by(faculty_id=current_user.id).order_by(Course.created_at.desc()).all()
    return render_template("faculty_dashboard.html", courses=courses)

@app.route("/faculty/create_course", methods=["POST"])
@login_required
def create_course():
    if current_user.role != "faculty":
        return redirect(url_for("index"))
    title = request.form.get("title","").strip()
    if not title:
        flash("Course title required.", "danger")
        return redirect(url_for("faculty_dashboard"))
    code = generate_class_code()
    course = Course(title=title, code=code, faculty_id=current_user.id)
    db.session.add(course)
    db.session.commit()
    flash(f"Course '{title}' created with code {code}.", "success")
```

```
    return redirect(url_for("faculty_dashboard"))

@app.route("/faculty/course/<int:course_id>")
@login_required
def faculty_course(course_id):
    if current_user.role != "faculty":
        return redirect(url_for("index"))
    course = Course.query.get_or_404(course_id)
    if course.faculty_id != current_user.id:
        return redirect(url_for("faculty_dashboard"))
    materials = Material.query.filter_by(course_id=course.id).order_by(Material.uploaded_at.desc()).all()
    return render_template("faculty_course.html", course=course, materials=materials)

@app.route("/faculty/course/<int:course_id>/upload", methods=["POST"])
@login_required
def upload_material(course_id):
    if current_user.role != "faculty":
        return redirect(url_for("index"))
    course = Course.query.get_or_404(course_id)
    if course.faculty_id != current_user.id:
        return redirect(url_for("faculty_dashboard"))
    title = request.form.get("title","").strip()
    file = request.files.get("file")
    if not title or not file:
        flash("Title and PDF are required.", "danger")
        return redirect(url_for("faculty_course", course_id=course_id))
    if not allowed_pdf(file.filename):
        flash("Only PDF files are allowed.", "danger")
        return redirect(url_for("faculty_course", course_id=course_id))
    filename = secure_filename(file.filename)
    material_dir = os.path.join(MATERIALS_FOLDER, str(course.id))
    os.makedirs(material_dir, exist_ok=True)
    path = os.path.join(material_dir, f"{int(datetime.utcnow().timestamp())}_{filename}")
    file.save(path)

    mat = Material(course_id=course.id, title=title, filename=os.path.basename(path))
    db.session.add(mat)
    db.session.commit()
    flash("Material uploaded.", "success")
    return redirect(url_for("faculty_course", course_id=course_id))

@app.route("/faculty/pending")
@login_required
def pending_notes():
    if current_user.role != "faculty":
        return redirect(url_for("index"))
    notes = PendingNote.query.filter_by(faculty_id=current_user.id).order_by(PendingNote.created_at.desc()).all()
    return render_template("pending_notes.html", notes=notes)

@app.route("/faculty/pending/<int:note_id>/download")
@login_required
```

```
def download_pending(note_id):
    if current_user.role != "faculty":
        return redirect(url_for("index"))
    note = PendingNote.query.get_or_404(note_id)
    if note.faculty_id != current_user.id:
        return redirect(url_for("pending_notes"))

    folder = os.path.join(PENDING_FOLDER, str(note.id))
    if not os.path.isdir(folder):
        flash("No files found for this note.", "warning")
        return redirect(url_for("pending_notes"))

    # Stream a ZIP of the folder


    # ...
    memory_file = io.BytesIO()
    with zipfile.ZipFile(memory_file, 'w', zipfile.ZIP_DEFLATED) as zf:
        for filename in os.listdir(folder):
            filepath = os.path.join(folder, filename)
            zf.write(filepath, arcname=filename)

    memory_file.seek(0)
    return send_file(
        memory_file,
        download_name=f"pending_note_{note.id}.zip",
        as_attachment=True
    )


@app.route("/faculty/pending/<int:note_id>/delete", methods=["POST"])
@login_required
def delete_pending(note_id):
    if current_user.role != "faculty":
        return redirect(url_for("index"))
    note = PendingNote.query.get_or_404(note_id)
    if note.faculty_id != current_user.id:
        return redirect(url_for("pending_notes"))
    folder = os.path.join(PENDING_FOLDER, str(note.id))
    if os.path.isdir(folder):
        # remove folder and contents
        import shutil
        shutil.rmtree(folder, ignore_errors=True)
    db.session.delete(note)
    db.session.commit()
    flash("Pending note deleted.", "success")
    return redirect(url_for("pending_notes"))

# ------ Student ------
@app.route("/student")
@login_required
def student_dashboard():
```

```
    if current_user.role != "student":
        return redirect(url_for("index"))
    enrollments = Enrollment.query.filter_by(student_id=current_user.id).all()
    return render_template("student_dashboard.html", enrollments=enrollments)

@app.route("/student/join", methods=["POST"])
@login_required
def student_join():
    if current_user.role != "student":
        return redirect(url_for("index"))
    code = request.form.get("code","").strip()
    course = Course.query.filter_by(code=code).first()
    if not course:
        flash("Invalid class code.", "danger")
        return redirect(url_for("student_dashboard"))
    already = Enrollment.query.filter_by(student_id=current_user.id, course_id=course.id).first()
    if already:
        flash("Already joined.", "info")
        return redirect(url_for("student_dashboard"))
    enr = Enrollment(student_id=current_user.id, course_id=course.id)
    db.session.add(enr)
    db.session.commit()
    flash(f"Joined {course.title}.", "success")
    return redirect(url_for("student_dashboard"))

@app.route("/course/<int:course_id>")
@login_required
def view_course(course_id):
    course = Course.query.get_or_404(course_id)
    # Students must be enrolled OR faculty owner OR admin to view
    authorized = False
    if current_user.role == "admin":
        authorized = True
    elif current_user.role == "faculty" and course.faculty_id == current_user.id:
        authorized = True
    elif current_user.role == "student":
            authorized = Enrollment.query.filter_by(student_id=current_user.id, course_id=course.id).first() is not
None
    if not authorized:
        flash("You are not enrolled in this course.", "danger")
        return redirect(url_for("index"))
    materials = Material.query.filter_by(course_id=course.id).order_by(Material.uploaded_at.desc()).all()
    return render_template("classroom.html", course=course, materials=materials)

@app.route("/materials/<int:course_id>/<path:filename>")
@login_required
def download_material(course_id, filename):
    # same authorization as view_course
    course = Course.query.get_or_404(course_id)
    authorized = False
    if current_user.role == "admin":
        authorized = True
```

```
    elif current_user.role == "faculty" and course.faculty_id == current_user.id:
        authorized = True
    elif current_user.role == "student":
            authorized = Enrollment.query.filter_by(student_id=current_user.id, course_id=course.id).first() is not
None
    if not authorized:
        flash("Access denied.", "danger")
        return redirect(url_for("index"))
    directory = os.path.join(MATERIALS_FOLDER, str(course.id))
    return send_from_directory(directory, filename, as_attachment=True)

# ------ API for PC GUI ------
@app.route("/api/faculty/courses", methods=["GET"])
@api_auth_required
def api_faculty_courses():
    user = request.user
    courses = Course.query.filter_by(faculty_id=user.id).order_by(Course.created_at.desc()).all()
    return jsonify([{"id": c.id, "title": c.title, "code": c.code} for c in courses])

@app.route("/api/pending_notes", methods=["POST"])
@api_auth_required
def api_create_pending():
    user = request.user
    data = request.json or {}
    course_id = data.get("course_id")
    title = data.get("title","").strip()
    if not course_id or not title:
        return jsonify({"error": "course_id and title required"}), 400
    course = Course.query.get(course_id)
    if not course or course.faculty_id != user.id:
        return jsonify({"error": "Invalid course"}), 403
    note = PendingNote(faculty_id=user.id, course_id=course.id, title=title)
    db.session.add(note)
    db.session.commit()
    # create folder
    folder = os.path.join(PENDING_FOLDER, str(note.id))
    os.makedirs(folder, exist_ok=True)
    return jsonify({"id": note.id, "title": note.title})

@app.route("/api/pending_notes/<int:note_id>/upload", methods=["POST"])
@api_auth_required
def api_upload_to_pending(note_id):
    user = request.user
    note = PendingNote.query.get_or_404(note_id)
    if note.faculty_id != user.id:
        return jsonify({"error": "Forbidden"}), 403
    folder = os.path.join(PENDING_FOLDER, str(note.id))
    os.makedirs(folder, exist_ok=True)
    files = request.files.getlist("files")
    if not files:
        return jsonify({"error": "No files uploaded"}), 400
    saved = []
```

```
    for f in files:
        filename = secure_filename(f.filename)
        if not filename:
            continue
        path = os.path.join(folder, filename)
        f.save(path)
        saved.append(filename)
    return jsonify({"saved": saved})

# -------------- CLI --------------
@app.cli.command("init-db")
def init_db():
    db.create_all()
    ensure_default_admin()
    print("Database initialized. Default admin: admin@bracu.ac.bd / 12345678")

# -------------- App start --------------
if __name__ == "__main__":
    with app.app_context():
        db.create_all()
        ensure_default_admin()
    app.run(debug=True)
```

Image to Pdf converter Latex Code:

```
import os
import uuid
import logging
from flask import Flask, request, redirect, url_for, send_from_directory, render_template, flash

# Configure logging for debugging
logging.basicConfig(level=logging.DEBUG)

app = Flask(__name__)
app.secret_key = os.environ.get("SESSION_SECRET", "dev-secret-key-change-in-production")

# Configuration
UPLOAD_FOLDER = "uploads"
ALLOWED_EXTENSIONS = {'zip'}
MAX_CONTENT_LENGTH = 16 * 1024 * 1024  # 16MB max file size

app.config['MAX_CONTENT_LENGTH'] = MAX_CONTENT_LENGTH

# Ensure upload directory exists
os.makedirs(UPLOAD_FOLDER, exist_ok=True)

def allowed_file(filename):
    """Check if the uploaded file has an allowed extension."""
    return '.' in filename and \
           filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS
```

```python
@app.route("/", methods=["GET", "POST"])
def index():
    """Main route for file upload and processing."""
    if request.method == "POST":
        # Check if file was uploaded
        if 'zipfile' not in request.files:
            flash('No file selected', 'error')
            return redirect(request.url)

        file = request.files['zipfile']

        # Check if file was actually selected
        if file.filename == '':
            flash('No file selected', 'error')
            return redirect(request.url)

        # Validate file type and process
        if file and allowed_file(file.filename):
            try:
                # Generate unique filename to prevent collisions
                unique_name = f"{uuid.uuid4()}_{file.filename}"
                filepath = os.path.join(UPLOAD_FOLDER, unique_name)

                # Save the file
                file.save(filepath)

                # Log successful upload
                app.logger.info(f"File uploaded successfully: {unique_name}")

                # Generate public URL for the uploaded file
                file_url = url_for("uploaded_file", filename=unique_name, _external=True)

                # Construct Overleaf URL with the uploaded file
                overleaf_url = f"https://www.overleaf.com/docs?snip_uri={file_url}"

                # Log the redirect
                app.logger.info(f"Redirecting to Overleaf: {overleaf_url}")

                # Redirect to Overleaf
                return redirect(overleaf_url)

            except Exception as e:
                app.logger.error(f"Error processing file upload: {str(e)}")
                flash('An error occurred while processing your file. Please try again.', 'error')
                return redirect(request.url)
        else:
            flash('Invalid file type. Please upload a ZIP file.', 'error')
            return redirect(request.url)

    # Render the upload form
    return render_template('index.html')
```

```python
@app.route("/uploads/<path:filename>")
def uploaded_file(filename):
    """Serve uploaded files for Overleaf access."""
    try:
        return send_from_directory(UPLOAD_FOLDER, filename)
    except Exception as e:
        app.logger.error(f"Error serving file {filename}: {str(e)}")
        return "File not found", 404

@app.errorhandler(413)
def too_large(e):
    """Handle file too large error."""
    flash('File too large. Maximum size is 16MB.', 'error')
    return redirect(url_for('index'))

@app.errorhandler(404)
def not_found(e):
    """Handle 404 errors."""
    return render_template('index.html'), 404

@app.errorhandler(500)
def server_error(e):
    """Handle server errors."""
    app.logger.error(f"Server error: {str(e)}")
    flash('An internal server error occurred. Please try again.', 'error')
    return render_template('index.html'), 500

if __name__ == "__main__":
    app.run(host="0.0.0.0", port=5000, debug=True)
```

Logbook

Table 14: FYDP (P) Summary of Team Log Book

Date/Time/Place	Attendee	Summary of Meeting Minutes	Responsible	Comment by ATC
21.09.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Initial brainstorming for project title. 2. Discuss initial problem statement ideas. 3. Plan draft concept note	Task 1: Ekram Task 2: Mercy Task 3: Ekram, Mercy, Mushfiq, Manjur	N/A as it was an introductory meeting. Task 1: In progress. Task 2: Not started. Task 3: Not started yet.
26.09.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Narrow down problem statement options. 2. Begin drafting concept note	Task 1: Ekram and Mercy Task 2: Mushfiq and Manjur	Task 1: Partially Completed. Task 2: In progress.
01.10.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Finalize project title and problem statement. 2. Complete draft concept note	Task 1: Ekram and Mercy Task 2: Mushfiq and Manjur	Finalize and refine the problem statement further. Task 1: Completed. Task 2: Completed.
06.10.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Gather sources for literature review. 2. Assign initial reading tasks	Task 1: Ekram Task 2: Mercy, Mushfiq, Manjur	Ensure comprehensive source collection. Task 1: In progress. Task 2: Not started.
12.10.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Conduct initial reading of gathered sources. 2. Start synthesizing notes	Task 1: Mercy and Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
17.10.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Complete synthesis and notes from literature. 2. Prepare for market survey design	Task 1: Manjur Task 2: Ekram and Mercy	Task 1: In progress. Task 2: Not started.
21.10.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Design market survey questionnaire. 2. Plan data collection strategy	Task 1: Mushfiq Task 2: Ekram	Task 1: In progress. Task 2: Not started.
26.10.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Collect data through market surveys. 2. Begin initial analysis	Task 1: Mercy and Manjur Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
02.11.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Complete analysis of market survey data. 2. Prepare draft report outline	Task 1: Mushfiq Task 2: Ekram and Mercy	Task 1: In progress. Task 2: Not started.

05.11.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Create outline for draft note. 2. Assign content drafting tasks	Task 1: Manjur Task 2: Ekram, Mercy, Mushfiq	Task 1: In progress. Task 2: Not started.
10.11.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Draft content for project note. 2. Review initial drafts	Task 1: Ekram and Mercy Task 2: Mushfiq and Manjur	Task 1: In progress. Task 2: Not started.
16.11.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Review and edit draft notes. 2. Prepare for design phase	Task 1: Ekram, Mercy, Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
20.11.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Develop initial design concepts. 2. Start sketching prototypes	Task 1: Mushfiq Task 2: Ekram and Mercy	Task 1: In progress. Task 2: Not started.
25.11.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Create initial sketches of prototypes. 2. Gather feedback on designs	Task 1: Manjur Task 2: Ekram and Mercy	Task 1: In progress. Task 2: Not started.
30.11.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Refine prototype designs. 2. Plan risk assessment	Task 1: Mushfiq and Manjur Task 2: Ekram	Task 1: In progress. Task 2: Not started.
05.12.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Identify potential project risks. 2. Evaluate identified risks	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
11.12.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Complete risk evaluation. 2. Plan mitigation strategies	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
16.12.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Develop mitigation plans for risks. 2. Discuss ethical considerations	Task 1: Ekram Task 2: Mercy	Task 1: In progress. Task 2: Not started.
20.12.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Finalize ethical considerations. 2. Analyze sustainability aspects	Task 1: Manjur Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
21.12.2024	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Compile final proposal document	Task 1: Ekram, Mercy, Mushfiq, Manjur	Ensure all sections are polished and complete. Task 1: Not started.

Table 15: FYDP (D) Summary of Team Log Book

Date/Time/Place	Attendee	Summary of Meeting Minutes	Responsible	Comment by ATC
04.02.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Initial review of prototype designs. 2. Discuss adjustments based on feedback.	Task 1: Ekram Task 2: Mercy	Ensure design aligns with energy efficiency goals. Task 1: In progress. Task 2: Not started.
10.02.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Adjust motorized duster mechanism. 2. Refine camera capture settings.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
15.02.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Finalize refined prototype designs. 2. Prepare for simulation setup.	Task 1: Manjur Task 2: Ekram	Final designs approved; proceed to simulation. Task 1: Completed. Task 2: Not started.
19.02.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Set up simulation for Design 1 (cleaning). 2. Test initial parameters.	Task 1: Mercy Task 2: Mushfiq	Ensure accurate power consumption data. Task 1: In progress. Task 2: Not started.
24.02.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Run simulation tests for Design 1. 2. Document initial results.	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
01.03.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Analyze simulation results for Design 1. 2. Plan Design 2 simulation.	Task 1: Ekram Task 2: Mercy	Review analysis for accuracy; proceed with Design 2. Task 1: In progress. Task 2: Not started.
06.03.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Set up simulation for Design 2 (content capture). 2. Configure AI assistant module.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
11.03.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Run tests for Design 2 simulation. 2. Troubleshoot AI integration.	Task 1: Manjur Task 2: Ekram	Task 1: In progress. Task 2: Not started.
17.03.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Analyze Design 2 simulation results. 2. Prepare data compilation.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.

21.03.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Compile data from both designs. 2. Begin initial analysis.	Task 1: Mushfiq Task 2: Manjur	Ensure data consistency across designs. Task 1: In progress. Task 2: Not started.
26.03.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Conduct initial analysis of results. 2. Plan detailed review.	Task 1: Ekram Task 2: Mercy	Task 1: In progress. Task 2: Not started.
31.03.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Perform detailed review of analysis. 2. Set parameters for comparison.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
05.04.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Set up parameters for design comparison. 2. Begin comparative analysis.	Task 1: Manjur Task 2: Ekram	Task 1: In progress. Task 2: Not started.
10.04.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Conduct comparative analysis. 2. Summarize findings.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
16.04.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Finalize summary of comparison. 2. Define evaluation criteria.	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
20.04.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Establish evaluation criteria. 2. Begin decision-making process.	Task 1: Ekram Task 2: Mercy	Task 1: In progress. Task 2: Not started.
24.04.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Make a decision on your preferred approach. 2. Start documentation.	Task 1: Mercy Task 2: Mushfiq	Decision approved; focus on documentation. Task 1: In progress. Task 2: Not started.
29.04.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Complete documentation of selected approach. 2. Prepare a presentation outline.	Task 1: Manjur Task 2: Ekram	Task 1: In progress. Task 2: Not started.
05.05.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Prepare slides for the final presentation. 2. Draft initial report.	Task 1: Mercy Task 2: Mushfiq	Ensure presentation covers economic and sustainability aspects. Task 1: In progress. Task 2: Not started.

13.05.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Continue drafting the final report. 2. Review presentation content.	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
21.05.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Conduct final review of report and presentation. 2. Prepare for submission.	Task 1: Ekram, Mercy, Mushfiq, Manjur	Final review in progress; ensure all deliverables ready. Task 1: In progress.
21.06.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Submit final deliverables. 2. Review project outcomes.	Task 1: Manjur, Ekram, Mercy, Mushfiq	Submission completed; project review approved. Task 1: Completed.

Table 16: FYDP (C) Summary of Team Log Book

Date/Time/Place	Attendee	Summary of Meeting Minutes	Responsible	Comment by ATC
17.06.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Initial meeting to discuss project scope. 2. Exchange ideas for whiteboard automation.	Task 1: Ekram Task 2: Mercy	N/A as it was an introductory meeting. Task 1: In progress. Task 2: Not started.
20.06.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Share and refine ideas for motorized cleaning. 2. Discuss content capture feasibility.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
24.06.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Build consensus on project objectives. 2. Plan literature review.	Task 1: Manjur Task 2: Ekram	Consensus reached; proceed with literature review. Task 1: Completed. Task 2: Not started.
26.06.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Review initial topic options. 2. Collect feedback from the team.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.

29.06.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Finalize feedback collection. 2. Prepare for the final decision.	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
02.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Make a final decision on a project topic. 2.Assign literature review tasks.	Task 1: Ekram Task 2: Mercy	Decision approved; focus on literature. Task 1: Completed. Task 2: Not started.
04.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Collect sources for literature review. 2 .Begin preliminary review.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
06.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Conduct preliminary review of sources. 2. Draft outline for concept note.	Task 1: Manjur Task 2: Ekram	Task 1: In progress. Task 2: Not started.
08.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Draft outline for concept note. 2. Assign content writing tasks.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
10.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Write content for concept notes. 2. Prepare for final edits.	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
12.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Perform final edits on concept notes. 2. Plan progress presentation.	Task 1: Ekram, Mercy, Mushfiq, Manjur	Concept note finalized; prepare presentation. Task 1: Completed.
13.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Prepare slides for progress presentation. 2. Gather data for analysis.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
15.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Rehearse progress presentation. 2. Analyze initial dimensions.	Task 1: Manjur Task 2: Ekram	Rehearsal completed; proceed with analysis. Task 1: Completed. Task 2: Not started.
16.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Gather data for dimension analysis. 2. Plan multiple design approaches.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
17.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Analyze gathered data. 2. Develop initial design.	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.

18.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Create initial design for motorized duster. 2. Explore alternative designs.	Task 1: Ekram Task 2: Mercy	Task 1: In progress. Task 2: Not started.
20.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Develop alternative design for content capture. 2. Set up evaluation criteria.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
22.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Set up criteria for optimal solution. 2. Begin evaluation process.	Task 1: Manjur Task 2: Ekram	Task 1: In progress. Task 2: Not started.
25.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Evaluate design options. 2. Prepare for selection.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
28.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Select optimal design approach. 2. Plan budget estimation.	Task 1: Mushfiq Task 2: Manjur	Selection approved; proceed with budgeting. Task 1: Completed. Task 2: Not started.
30.07.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Estimate costs for prototype. 2. Allocate resources.	Task 1: Ekram Task 2: Mercy	Task 1: In progress. Task 2: Not started.
03.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Finalize resource allocation. 2. Review ethical considerations.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
07.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Finalize budget plan. 2. Plan safety considerations.	Task 1: Manjur Task 2: Ekram	Budget approved; proceed with safety planning. Task 1: Completed. Task 2: Not started.
09.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Conduct ethical review. 2. Draft initial project proposal.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
11.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Plan safety measures. 2. Review initial draft.	Task 1: Mushfiq Task 2: Manjur	Task 1: In progress. Task 2: Not started.
13.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Write initial draft of project proposal. 2. Collect feedback.	Task 1: Ekram Task 2: Mercy	Task 1: In progress. Task 2: Not started.

16.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Review and gather feedback. 2. Prepare the final draft.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
19.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Complete final draft of project proposal. 2. Prepare slides for the final presentation.	Task 1: Manjur, Ekram, Mercy, Mushfiq	Final draft approved; focus on presentation. Task 1: Completed.
20.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Prepare slides for the final presentation. 2. Plan final rehearsal.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
22.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Conduct final rehearsal. 2. Outline report structure.	Task 1: Mushfiq Task 2: Manjur	Rehearsal completed; proceed with report. Task 1: Completed. Task 2: Not started.
23.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Create outline for project proposal report. 2. Begin content development.	Task 1: Ekram Task 2: Mercy	Task 1: In progress. Task 2: Not started.
26.08.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Develop content for a report. 2. Prepare for final edits.	Task 1: Mercy Task 2: Mushfiq	Task 1: In progress. Task 2: Not started.
01.09.2025	1.Ekram 2.Mercy 3.Mushfiq 4.Manjur	1. Perform final edits and submit reports. 2. Review project outcomes.	Task 1: Manjur, Ekram, Mercy, Mushfiq	Submission completed; project review approved. Task 1: Completed.