I.J. Modern Education and Computer Science, **2026, 2, 38-53**
Published Online on April 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2026.02.03

Enhancing Student Engagement and Attendance in online Learning environment by using AI Enhanced Robotic Process Automation

M. Durga Prasad*
Department of computer science and Engineering, Telangana University, Telangana State, India
Email: durgamamidyala@gmail.com
ORCID iD: https://orcid.org/0009-0001-8565-3857
*Corresponding Author

Balusu Nandini
Department of computer science and Engineering, Telangana University, Telangana State, India
Email: cnuvnk@gmail.com
ORCID iD: https://orcid.org/0009-0009-4066-7970

Received: 10 April, 2025; Revised: 08 July, 2025; Accepted: 26 October, 2025; Published: 08 April, 2026

Abstract: Student engagement is a crucial aspect of higher education learning. However, it may be challenging to ensure active engagement, especially if students lack motivation to participate. This research proposes an innovative technique for increasing student participation in online sessions that incorporates real-time chat interaction, engagement reminders, and attendance tracking. Unlike traditional research, which focuses on post-session analysis, the developed Bot actively monitors student participation during session itself, providing real-time notifications to disengaged students without requiring the need for human intervention. It records attendance for each session, monitors weekly student participation, and dispatches updates via email to both instructors and students. To provide a more interesting and responsible learning environment, the Bot also utilizes AI to evaluate student responses and provide suggestions for their improvement.

Index Terms: Bot, Robotic Process Automation, Artificial Intelligence, Student Engagement

1. Introduction

Student engagement has grown in popularity within educational research and practice over the past few years [1]. Student engagement is a key component for academic achievement and motivation in higher education [2]. Student engagement is recognized as an essential element at all levels and phases of education, with a primary function to enhance the quality and effectiveness of student learning [3]. Several studies have demonstrated that students who exhibit higher levels of engagement, both in terms of their participation in online session activities and attention to academic assignments, tend to do better academically [4]. This correlation between student engagement and performance of students is clearly shown in their improved academic performance, higher retention rates, and a deeper understanding of the information being taught [5]. The correlation between student engagement and the use of learning technologies in the classroom presents a significance area of focus for academic research. [6]. Student engagement and overall learning performance are influenced by a variety of digital tools, online learning platforms, and technological advancements [7]. Increased application of technology in education presents both new challenges and opportunities. According to researchers [8], technology can increase student engagement by facilitating more interactive, individualized, and adaptable learning experiences. For years now, instructors in higher education have been using remote learning tools like video conferencing to connect with students in real time [9]. Extensive research [10] has been done on how video conferencing platforms are used in educational settings. Wang et al [11] described the development of a blended synchronous learning environment (BSLE) that combines online and face-to-face instruction. Their study proved that, BSLE is convenient to distance students, but it works well if there is active engagement, good communication, and the same attention given to both online and face-to-face students. The researchers emphasized the importance of modifying teaching methods and preserving high-quality audio to improve learning results and student

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

engagement. They also mentioned the difficulty teachers experience in managing attention among various learner groups.

Sayem et al [12] found that using video conferencing platform in foundation engineering courses boosted student satisfaction and reduced instructor workload by 25%. For conducting meetings, training sessions, and qualitative interviews organizations use video conferencing tools, such as Skype, Zoom, Google Meet, and Microsoft Teams due to their convenience and ease of use. According to Archibald et al [13], zoom was favored over other video conferencing programs for conducting qualitative interviews with nurses because of its convenience and ease of use. When assessing Indonesian teacher's preparedness for online instruction during the COVID-19 epidemic, Arifianto et al [14] found that motivation and adaptability are essential for successful participation. While teacher-directed training and motivation techniques are successful, time limits, poor engagement, and difficulties adopting traditional methods online are problems, according to Dimaculangan et al [15]. According to Salami et al [16], student's self-efficacy is adversely affected by academic stress and sadness. The collective outcomes of these research demonstrate that although online learning platforms provide ease and flexibility for both instructors and students, instructors often encounter difficulties with student engagement, technological constraints, and modifying conventional teaching techniques for virtual environments.

However, using video conferencing platforms also bring its own difficulties [17], such as time limits for free accounts, trouble keeping track of every student [18], and potential network issues affecting session continuity. Several video conferencing platforms, including Zoom, Google Meet, and Microsoft Teams, offer APIs with various useful features; however, they still present certain limitations during live sessions. For example, chat messages of video conferencing platforms are usually available only after the meeting if the host has enabled the option to save them. Likewise, most participant analysis tools can be accessed before or after the meeting, but not while it's in progress. Although the API allows you to manage user accounts, schedule meetings, and gather meeting statistics, it does not support sending live chat messages, muting participants, or controlling audio and video during an active session. To tackle the issue of interacting with video conference meetings in real-time, we have developed a Bot. This Bot have the capability to interact with the video conference application using UI automation activities. It can send chat messages and get participant's details during live ongoing meeting. Since the API of most video conferencing applications does not provide API's for real-time interaction, the integration of UI automation techniques can effectively address these limitations. Thus, providing greater flexibility in managing meetings as they occur. This method enables enhanced control during live sessions and supports the efficient execution of tasks that require immediate action. Instructor support for interaction and self-regulation plays a crucial role in determining online engagement, with self-regulation partially mediating the relationship [19]. In online learning, social presence plays a crucial role in fostering a sense of connection among students, which significantly enhances their motivation and engagement in the learning process. Although instructors primarily focus on teaching [20], maintaining regular interaction can be difficult. Still, this sense of connection enhances both the learning experience and success [21].

These findings highlight the challenges of encouraging online engagement and underscore the need for programs that support both instructors and students in overcoming these obstacles. These challenges for instructors underscore the importance of adequate support in efficiently utilizing video conference applications like Zoom, Google Meet, Microsoft Teams, and others [22]. With growing popularity of artificial intelligence (AI) has revolutionized the way organizations function by substituting human intervention with programmed processes. Robotic process automation (RPA) has been the strongest enabler of this transformation [23]. These technical developments have the greater impact in improving organizational capacities. Finding repetitive activities and creating consistent protocols in Bot development are essential to this technology's efficacy. Consequently, these Bots can be programmed to perform tasks on their own, reducing staff involvement [24].

UiPath [25], a well-known RPA tool [26] designed for large-scale end-to-end automation, is used as tool to build the proposed Bot. The developed bot facilitates effective classroom management and student engagement by integrating smoothly with video conferencing platforms. It logs into video conferencing platforms at varying and unpredictable intervals during the session to maintain continuous interaction. Since the Bot logs in at random times during the session, students are unable to predict the exact moment it will appear, which keeps them attentive and engaged throughout. The host sets these times by scheduling emails that start the Bot automatically. While preserving a level of unpredictability in the interaction flow, this approach ensures a seamless and dynamic experience. The Bot records attendance of participants as soon as it logs into the meeting and stores the data for future reference. This data is used by instructors to efficiently track student participation over time. Before the start of the session or as part of the schedule, the instructor emails a series of questions with associated keywords to check answers to a designated official email address, along with the meeting ID and verification code. When the email with desired subject line is received, the Bot automatically downloads the details and execute the process. The Bot starts up automatically to carry out the plan, obtaining the questions and uploading them in the chat during the session. Students are prompted to respond within a specified time, and if any student fails to engage, the system sends a polite reminder to student for encouraging participation.

Furthermore, the Bot use artificial intelligence to assess student's responses in chat for a particular question. It employs artificial intelligence to deliver personalized feedback that not only encourages accurate answers but also benefits students by providing constructive suggestions for incorrect responses. This helps students to see their mistakes as learning opportunities and motivates them to grow without becoming disappointed. Additionally, by giving students

prompt feedback helps students to grasp concepts better. The Bot encourages ongoing participation of students and keeps them engaged in their educational process.

After every session, the Bot records and saves attendance data in a file, making it easy for the instructors to track participation of students over time. To keep everyone informed, the Bot sends weekly attendance reminders to students as well as instructors by using Gmail API. In this way, it ensures that no one misses important updates making them to react early. By automating routine tasks, our Bot helps manage classrooms more efficiently and keeps students engaged, motivated, and aligned with their learning goals. It creates a positive and supportive environment by offering real-time feedback, personalized assistance, and simplifying administrative work. This not only helps students stay focused and make progress, but also gives instructors more time to concentrate on teaching and improving the quality of instruction. In addition to simplifying classroom tasks, the Bot helps create a more positive and focused learning environment. The bot has been thoroughly tested in various scenarios to ensure reliability and effectiveness.

2. Literature Review

Over the past few years, technology has made many changes and reshaped the way businesses operate. Tasks that required manual effort to perform are now handled by software bots, known as RPA. This is often described as building a “digital workforce”. These transformations do not only result in making changes in organizational structures but also affect technological environments. Tools such as cognitive software, is central to the creation of a hybrid workforce comprising human and machine capabilities [27]. Several methods have been developed to automate email systems that incorporate intelligent capabilities that enhance user experience. One of the most difficult challenges continues to be that of email spam, which is simply referred to as junk mail. Sometimes, important emails get moved into the spam folder by mistake, which causes confusion and delays. Furthermore, the huge number of incoming emails presents another major issue. Users often spend a great deal of time and effort trying to find relevant and useful emails from the sheer deluge. In order to tackle such issues, it is essential to create ways of classifying and marking emails according to their content, which would allow users to quickly spot significant messages without having to even open them.

In this regard, Izzat Alsmadi et al[28] suggested a solution based on machine learning that intended to determine the true sender of an email through learning from previous email histories. Although this solution was demonstrated to have reasonable accuracy, it remained constrained by the probabilistic nature inherent in machine learning models. Such uncertainties in identifying senders pose an issue in real-world business applications, where misclassifications would result in considerable risks. In addition, the time-consuming process of verification each sender requires turns out to be impossible. Thus, something better is needed, while offering total accuracy with reduced effort. The central hypothesis of this research is that Case-Based Reasoning (CBR) can be used to solve the email overload problem. This is supported by an analysis of email databases, which shows that previous occurrences can be used to predict future questions and responses [29]. Current email management in its present form takes a lot of effort and time from both the sender and receiver, so the reward for automating the email process may be worthwhile. To this, hybrid approaches have been proposed to facilitate the identification of areas where automation would be most beneficial to users [30].

Furthermore, this research explores the development of innovative, end-to-end systems capable of automatically generating concise email responses [31]. The research also examines the definition of RPA. RPA is an information technology used in computer software that facilitates it to be easy to implement, design, and administer software robots, which mimic human behavior while communicating with computer systems. RPA is utilized intensively in several industries for automating mundane activities and enhancing workflow efficiency. In the context of Industry 4.0, Jorge Ribeiro et al [32] wrote about the implications of integrating RPA with AI practices, particularly identifying how it redefine ERP-based processes in terms of enhanced user involvement, optimization, and process intellect. Sofia Khan [23] has suggested applying RPA bots for education to help teachers provide more efficient lessons through the assessment of lesson quality. Additionally, there is an increasing amount of research focusing on the development of educational Bots, which are intended to reduce the administrative burden and increase the efficiency of educational systems [33]. Automated processes like managing student information enable these bots to maintain data accuracy and integrity, thus providing teachers an opportunity to spend more time on teaching. Similarly, Arpit khare et al [34] created an RPA bot that automates email correspondence effectively. The bot safely accesses mailboxes, filters and classifies important emails, labels them accordingly, downloads attachments, handles files, and moves unnecessary emails to trash. This tool was programmed and tested using UiPath to enhance system performance.

RPA and AI are transforming how educators manage participation and engagement in virtual classrooms. The Bot created for this research combines these technologies to automate critical functions such as login into platforms like Zoom, registering attendance, and gathering student feedback in real time. The AI component examines student’s feedback using massive language models like OpenAI [35]. The response from OpenAI is validated using keyword driven approach. Using keywords, it assesses key ideas and suggests changes as necessary, encouraging students to reflect more deeply and polish their responses [36].

This level of automation not only reduces teacher’s administrative workload, but also ensures that engagement tracking is done consistently and without manual intervention [37]. The unique feature of this technology is its capacity to classify students as "Present”, P_Inactive," or "Absent" during live sessions according to behavioral traits. To better understand each student’s overall performance, we paired their engagement data with their GPA, offering a clearer view

of both academic progress and behavioral patterns. We used three machine learning models named Isolation Forest, One-Class SVM, and DBSCAN, to find students who might be at risk or disengaged. We decided to use a consensus approach, meaning a student would only be flagged if at least two models agreed. That is because each model has its own way of spotting outliers, so relying on just one wouldn't be reliable. This makes the identification process more reliable by highlighting students who consistently exhibit troubling patterns and lowering false positives.

The approach's basis is Engagement Theory, which emphasizes learning via technology-assisted, meaningful, and interactive activities [38]. The Bot created in this study is specifically designed for synchronous learning environment, where students and instructors interact together. On the other hand, adaptive learning platforms like Squirrel AI place an emphasis on long-term mastery through self-paced, individualized training [39]. It manages classroom dynamics in real time, enabling educators to monitor student participation and offer prompt assistance. This technology brings together automation, semantic analysis, and student behavior tracking to improve how learning systems respond in real time. By doing so, it addresses a major gap in today's educational tools by offering intelligent feedback during live interactions. To emphasize these contrasts, Table 1 compares the proposed interaction Bot to three well-known educational platforms: Kahoot, Nearpod [40], and Squirrel AI.

Table 1. Feature comparison of the Bot with existing educational platforms

Feature	Proposed Bot	Kahoot	Nearpod	Squirrel AI
Input Type	Open-ended responses evaluated using OpenAI LLM	Multiple-choice, polls	Mostly multiple-choice, some open response	Structured inputs based on knowledge points
Feedback Mechanism	Real-time feedback using semantic analysis and keyword validation	Instant scoring feedback	Instructor-guided feedback	Feedback based on performance trends
Automation Level	Fully automated, runs without instructor involvement	Requires a host to run live sessions	Instructor-led or scheduled presentations	Fully automated personalized learning
Error Handling & Reliability	Built-in error handling with retry logic (via RPA tools like UiPath)	Minimal; issues may go unnoticed	Basic; depends on device and internet	Handled internally by system infrastructure
Outlier Detection	Yes—detects struggling students using AI and behavior patterns	No	No	Yes—based on academic performance over time
Use Case Focus	Real-time engagement, feedback, and student monitoring	Game-based quizzes and motivation	Interactive lesson delivery	Mastery-based adaptive learning

By allowing students to type complete, open-ended responses rather than simply choosing from previously determined options, the Proposed Bot differs from the other tools. It makes learning more interactive and individualized by providing immediate, helpful feedback using sophisticated language understanding. This Bot operates independently and keeps things going without continuous direction, in contrast to platforms like Kahoot or Nearpod that frequently require a teacher to lead the session. By examining how they react, it can also determine which students may be having difficulty and assist in determining who requires additional assistance. It's a strong choice for real-time learning, feedback, and monitoring student progress all without the need for human intervention.

3. Methodology

Among the several video conferencing platforms available, we have selected Zoom for our research due to its widespread usage. The diagram in Figure 1 depicts the architecture of the developed Bot, which automates administrative tasks for guaranteed student engagement in zoom like online meetings. This solution utilizes the potential of RPA through the UiPath platform, executed on a Windows operating system. UiPath is an RPA tool which automates redundant tasks, hence allowing organizations to streamline workflows and minimize the workload.

3.1 Bot Initialization and Email Processing

The Bot is designed to trigger automatically when it receives an email with a particular subject line, like "Student Engagement-CCBot". The UiPath Orchestrator, the central console for running robotic processes, is set to trigger the Bot when it receives an email with the preset subject. The users can either send the email at the moment or schedule it at a later time, depending on when they want the Bot to be triggered. After sending the email, the Orchestrator senses its receipt, analyzes the subject line, and if it is the preset subject, the Bot is triggered to perform the tasks assigned to it. This feature gives the users the choice of triggering the Bot either in real time or at a determined point in time, thus creating a smooth and effective automation process with little or no human intervention. The setup of the parameters to configure the trigger can be done through the interface of the orchestrator, as shown in Figure 2.

After activation, the Bot will login to the user's machine in unattended mode by submitting credentials. For protection of sensitive data, the UiPath Robotic Enterprise Framework (REFramework) is employed that makes that user credentials and other sensitive data are handled securely. Instead of hardcoding credentials within the workflow, the Bot retrieves this information from encrypted sources such as Orchestrator assets, thereby maintaining security levels high.

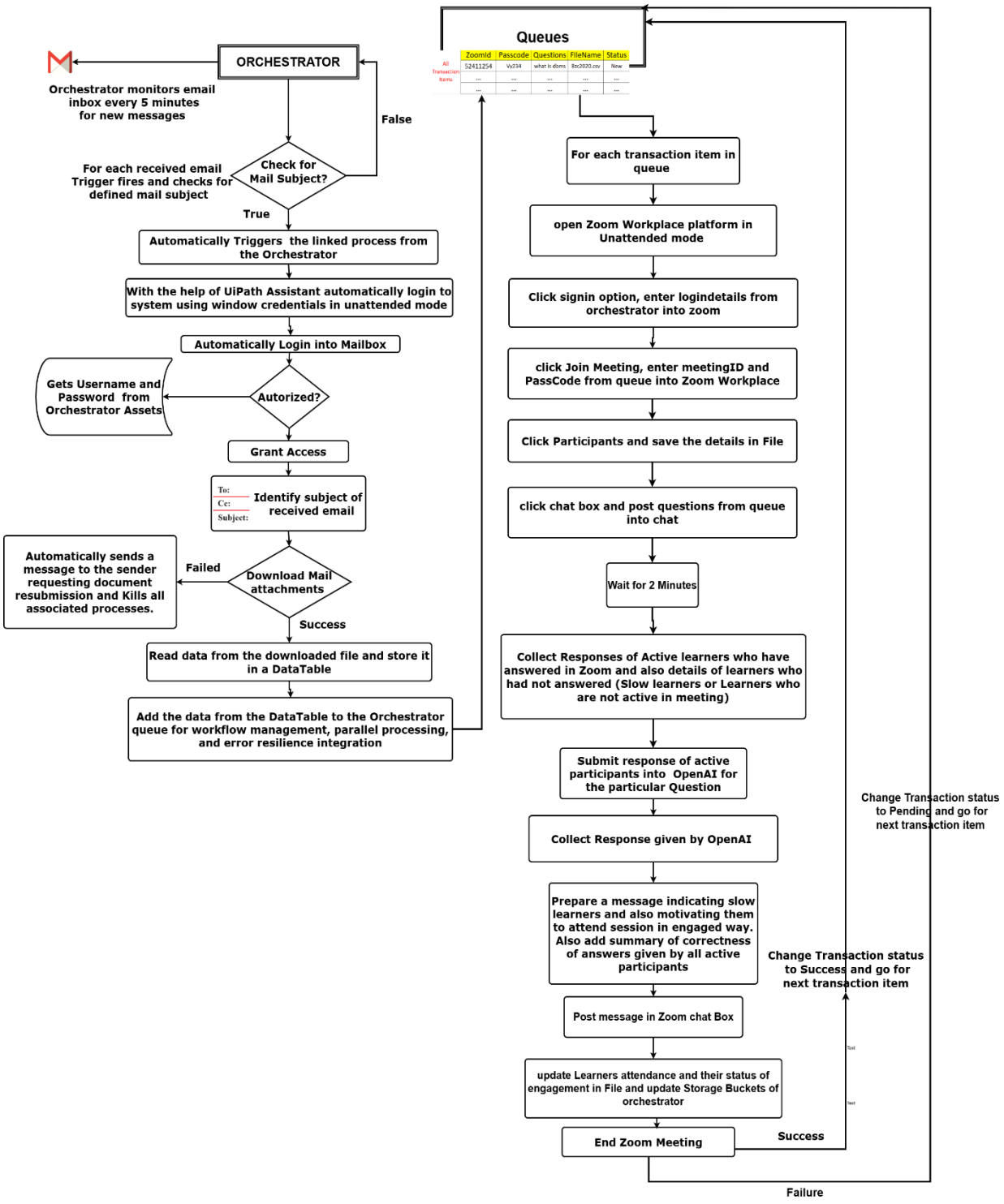


Fig. 1. System design of proposed Bot



Fig. 2. The Orchestrator in an active state, waiting for an email with a specified subject to trigger the bot once the event occurs.

3.2 Email interaction and attachment processing

The Bot interacts with the Gmail server via the IMAP Protocol (Internet Message Access Protocol). IMAP protocol offers a secure connection to retrieve emails from the inbox. Having arrived at the mailbox, the Bot retrieves a list of emails via the Get IMAP Mail Message activity in UiPath, configured with parameters such as Gmail server details (imap.gmail.com), name of the email folder (Inbox), and the needed authentication details. Once emails are fetched, the Bot verifies for attachments. If any attachments satisfy the required criteria, they are stored to a specific folder on the system. If any attachment does not satisfy the specified format or content requirements, Bot sends an automatic follow-up email to the sender asking for the proper file. The Get IMAP Mail Message activity property window is depicted in Figure 3.

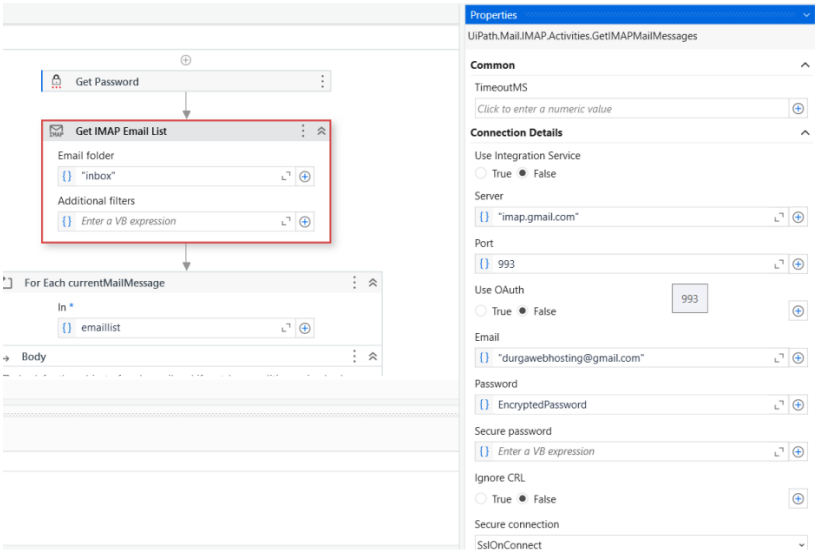


Fig. 3. Property window of get imap email list activity

3.3 File handling and data extraction

The attachments downloaded from Gmail have vital information, such as Zoom ID, vv code, questions, keywords and file location. Once the Bot downloads attachments, it automatically processes the content of these files into a Datatable. A Datatable is a data format in UiPath that holds tabular data, just like an Excel sheet. Then data from datatable is then added to a Queue for further processing.

3.4 Queue management and user interface automation

The data in the Datatable is moved to the queue in bulk with the help of the "Bulk Add Queue" activity, as seen in the Figure 4. UiPath queues are employed to structure, administer, and monitor data to allow for efficient processing, enhanced handling of errors, and scaling of automation activities effectively.

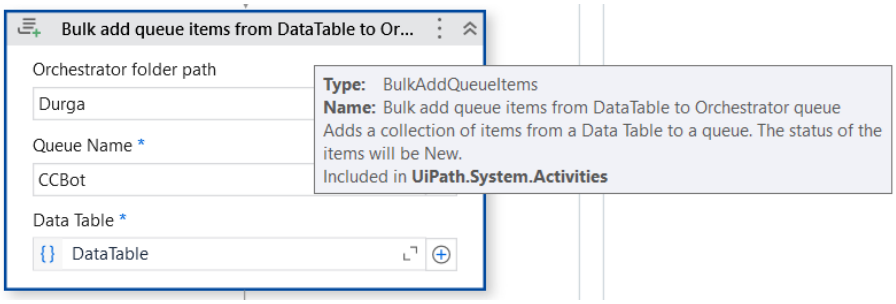


Fig. 4. The Bot organizes downloaded content into queues for optimal control by employing the bulk add queue activity

UiPath uses Selectors as exact locators for UI objects, which provide direct interaction. However, where UI layouts are changing or not sufficiently stable, Anchors provide a better solution by allowing the bot to find a target object relative to an unmovable reference point on the screen, which provides better automation stability, as shown in Figure 5. The Bot launches Zoom Workspace application and clicks on "Sign in" button through click activity. The moment the sign-in page is launched, it types the host's email address and password into their own fields through UiPath's "Type into" activity. Once the credentials are typed in correctly next it proceeds and clicks the "Sign In" button to complete the login process. This automation accelerates the entire process, which is faster and more efficient in gaining access to the

Zoom Workspace with no direct intervention on the user's part. Within the Zoom meeting, the Bot clicks the participant's icon and get details of all participants into DataTable using Find children activity. Exact interaction with such items is done by using a combination of target and anchor selectors. The target selector determines the target item, whereas the anchor provides a contextual hint so that the click is performed on the correct item. This improves the stability of the automation and makes it robust against application layout changes.

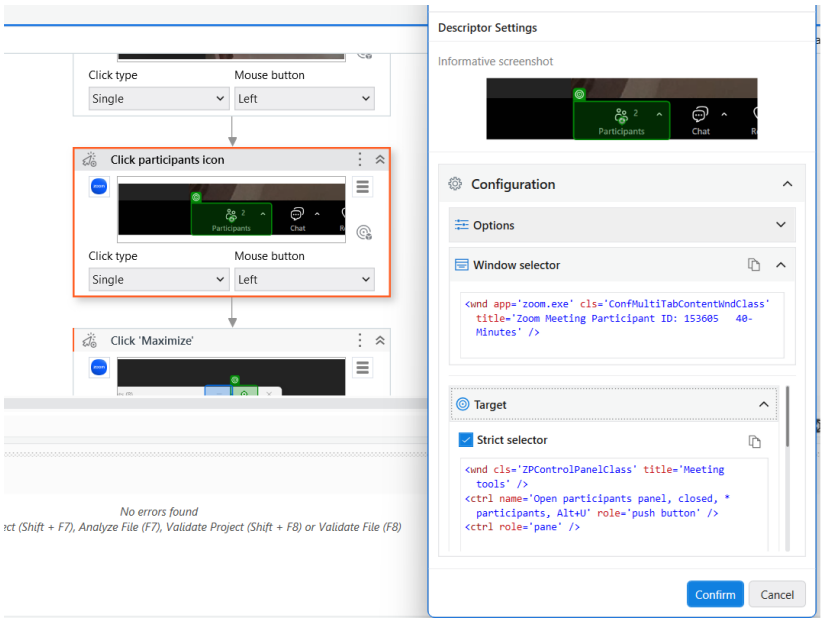


Fig. 5. Marking targets and anchors to identify input elements within the target application using a unified targeting approach

Upon collecting participant's details, it selects the "Chat" icon using click activity. It then posts a question taken from the queue into the chat box by using the Type into activity. The students can then look and respond to this question in real-time. After posting a question in chat box, the Bot waits for a response over the course of two minutes, allowing sufficient time for students to engage. It then collects the responses from students who have replied within this window, recording each participant's name and message into a well-organized data table. Once the response period concludes, the Bot carefully monitors each participant's attendance during the Zoom meeting. Those who are not present are marked as "Absent," while participant's who attend the meeting are recorded as "Present". In addition, Bot identifies participants who were Present but not engaged in the meeting. These participants are labelled as "Present-Inactive". It sends them a polite reminder, informing them of the lack of a reply.

Moreover, the Bot uses AI to analyze student responses and generates personalized feedback to motivate and encourage further engagement. A hybrid model is used by combining OpenAI large language model(LLM) with a keyword based validation mechanism. The language model employed by the system is OpenAI, which was chosen for its ability to understand and evaluate subject-specific content.

We added a keyword validation stage to guarantee that the responses are correct and cover all relevant points. Associated Keywords for a question are provided by instructor at initial state when providing meeting details through mail. The system validates OpenAI responses with associated keywords. If any essential phrases are missing, the system automatically sends OpenAI a modified HTTP request with an updated prompt seeking a more detailed and conceptually accurate explanation. The developed Bot consists of a main workflow that invokes the "AI_sequence" workflow. A workflow named "AI_sequence" workflow is created with two arguments: one as an input (say, Message_input) and the other as an output (say, out_Message) as shown in Figure 6.

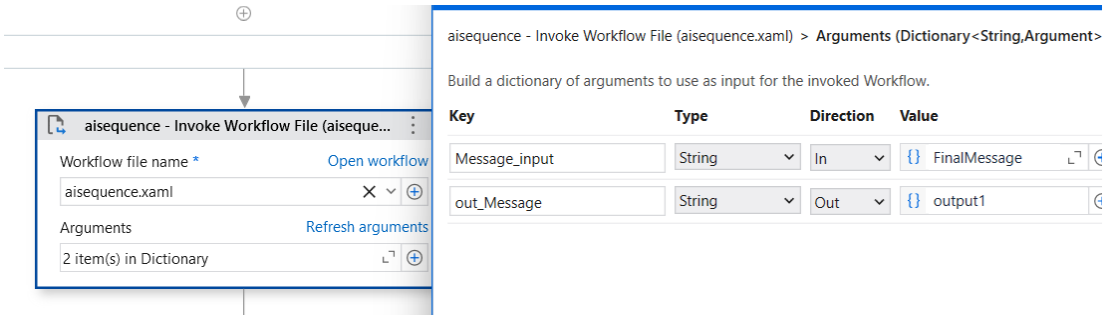


Fig. 6. AI_sequence workflow with defined arguments.

The Message_input argument takes its input from the FinalMessage variable, which is the response of the participants to a question, and passes it to OpenAI for analysis. The workflow utilizes an HTTP request activity to connect to connect to OpenAI's API, which includes necessary parameters like an API key as shown in Figure 7.

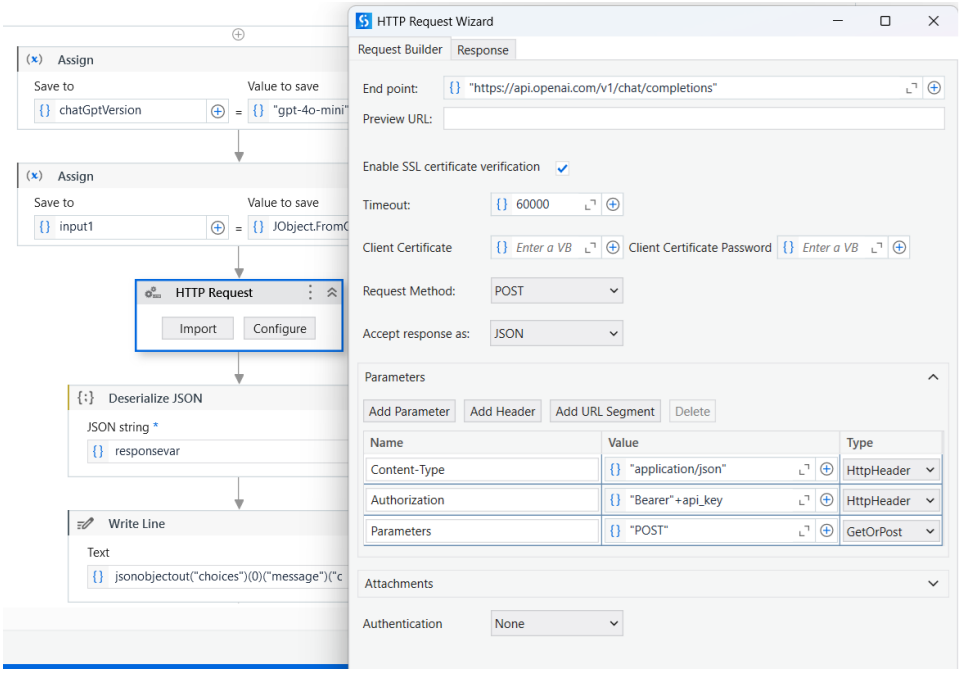


Fig. 7. Http request wizard to connect OpenAI

The input argument is sent as part of the request, and the response from OpenAI is stored in the output argument. The OpenAI output is processed using a keyword-driven technique, in which the Python code looks for keywords based on the question. If any keywords are lacking, the system will send an HTTP request to OpenAI with the missing information for the defined answer. After all processing, the output argument which holds the participant's response feedback is processed and typed into the Chat window of the Zoom application. This has led students to believe that their answers are being thoroughly reviewed and checked, encouraging them to be more engaged in session. The figure 8 illustrates students responding to a question posed by Bot, followed by Bot displaying feedback after reviewing their answers. Additionally, it highlights the students who remained inactive during this process (i.e. Slow Responders).

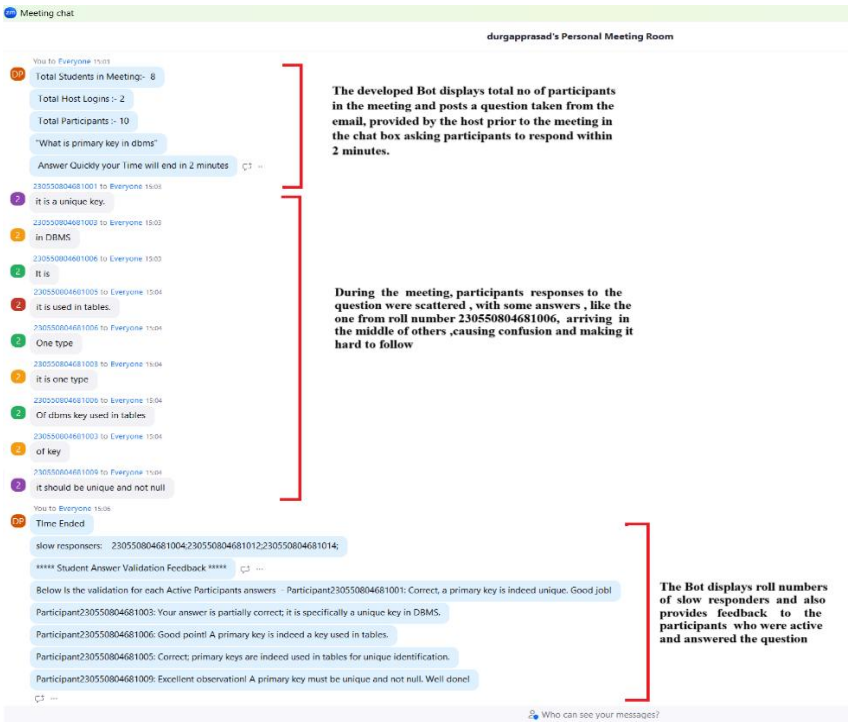


Fig. 8. Student Engagement and Feedback Process.

Figure 8 illustrates the Bot identifying slow responders and simultaneously giving feedback to those who are actively engaging in the meeting. This approach helps motivate everyone to stay involved and encourages greater participation in upcoming sessions.

3.5 Excel file handling and reporting

The Bot uses UiPath Excel management activities to effectively capture the details of the students. It fills the rows with appropriate information regarding each student's attendance. The student attendance is divided into three categories: Present, Present-Inactive, and Absent. Through UiPath automation, the Bot makes sure that all the data is accurately captured in the Excel sheet. For each and every meeting, whenever Bot will trigger, it will add a new column in the Excel sheet with respect to session date and time as column name. This way, every student's information is properly linked with the respective meeting date and time. The student hall ticket number is used to modify the concerned row, keeping a systematic record of response and attendance. UiPath incorporates sophisticated error handling methods to ensure that the Bot keeps running optimally even if there are errors. In case of a failed operation, e.g., an error in handling the user interface or fetching the data from the application, the Bot can retry the action multiple times. In addition, complete Exception Handling allows any errors to be logged and reported so that they can be resolved quickly.

3.6 Transaction updates and report generation

After the Bot finishes its operation for a specific Zoom meeting, it updates the status of the queue item inside the UiPath Orchestrator. The transaction status reflects if the processing went smoothly(Successful) or any error occurred (Failed). The Bot will create reports summarizing the student attendance. Additionally, the Bot proceeds execution with the next transaction item in Queue.

3.7 Error handling and system reliability

We developed the automation using UiPath retry features to handle problems like a bad internet connection or system errors. If a task fails, it can try again according to the rules we set because we used Orchestrator queues with automatic retry settings. Try-catch blocks and retry scopes were also incorporated into the workflow to address transient issues, such as a slow response or connection timeout, by trying the action again before giving up. This configuration makes the Bot more dependable by enabling it to fix common mistakes without assistance. UiPath queues are employed to structure, administer, and monitor data to allow for efficient processing, enhanced handling of errors, and scaling of automation activities effectively.

3.8 Outlier detection for finding students at risk

An approach that combines a three-level attendance classification (Present, P_Inactive, and Absent) with GPA is presented for identifying students who are at risk. Three machine learning models are used to find strange behavioral patterns: DBSCAN, isolation forest, and one-class SVM. A consensus approach decreases false positives and increases reliability by only alerting students when at least two models agree. The method helps in identifying students who might genuinely require early assistance or intervention.

Algorithm 1. Algorithm of the developed Bot

Step1: Configure the Bot for logging windows by entering username and passwords from orchestrator assets.
Step2: Configure the Get IMAP mail message activity.
Step3: Configure the Bot to take email-id and password from orchestrator assets or config file in RE-framework.
Step4: out_mail: = List<MailMessage>
Step5: For each mail-in out_mail repeat
    if is_important(mail)
    open the mail
    Download attachments File.
Step6: Open the downloaded file which contains zoom meeting details and transfer it into queue for better Management.
Step7: Use Get Transaction item to get information from queue.
Step8: For each Transaction Item
    perform UI automation activities to login into the meeting using Transaction item.
    Perform data scraping to get all participants details.
    Post Questions in chat and collect responses of all participants and saves accordingly the participants status.
    Use OpenAI to analyze response of participants and send response in participants in chat to encourage for further engagement.
    Perform Excel management activities for inserting or updating participant's status.
Step9: After completing step8, Generate and update reports which are useful for weekly report of participants periodically.
Step10: Send mail to all Faculties and students of their attendance status and engagement percentage.
End

4. Results and Discussions

The Bot has been tested in UiPath 2023.8.0 on a 64-bit Windows 10-based system with Intel Core i5 processor and 8 GB RAM. The whole purpose of these tests is that the Bot behaves as expected under all defined tasks. Telangana University Faculty Members were asked to send emails under the subject 'Student Engagement-CC Bot", with attachments containing information for opening the Zoom meeting. A few emails were specifically sent with varied subject lines so that the Bot would be tested for whether it could recognize relevant content or not. When the correct email was received, the orchestrator initiated the process of the Bot.

Upon receiving the appropriate email, the Bot downloaded the attachments of the emails, subsequently logged into the email account and continued with the specified workflow. It retrieved Zoom meeting-related data from the files that were downloaded, successfully logged into zoom meeting, conducted UI automation to scrape information from a Zoom Workspace application, and finally saved processed data in specified Excel files. The Bot was successfully able to sort emails, download files, and scrape and organize data without human intervention.

4.1 Performance evaluation

The performance of the Bot that has been developed was assessed according to some important quality-of-service parameters:

Table 2. Quality-of –service parameters used for performance evaluation

Sno	Quality-of-service Parameter	Description
1	Efficiency	The ratio of valuable tasks completed by the Bot in comparison to manual effort
2	Time Consumption	The amount of time taken by the Bot versus human labor to perform the same tasks
3	Accuracy	The Bot's ability to process data and complete tasks without errors.
4	Precision	The Bot's capability to consistently perform tasks with minimal errors and deviations.

The Bot was tested with two scenarios:

Case I – Manual Process: Faculty members asked to manually collect data from emails, login into Zoom app and retrieve participant’s details and update into an Excel sheet. Then asked to analyze students who didn't reply. It became such a monotonous task to analyze student's replies because they were responding in zig zag pattern. This consumed all the meeting time and they couldn't focus in delivering lecture.

Case II – Automated Process (Bot): The Bot was used, where it automatically downloads attachments, scrapes data from Zoom Application, and save the data into Excel files. This method significantly reduced the need for human intervention.

1. Efficiency

In Case I, human fallibilities, including inconsistencies, mistakes, and emotional biases, hindered manual processes, leading to unpredictable results. In contrast, the Bot's automation in Case II showed a considerable boost in efficiency, executing tasks consistently and without any mistakes that usually happen in manual processing.

2. Time consumption

In Case I, the manual process consumed much more time, especially with large datasets. For instance, manually processing students who were present in meeting and finding students inactive during meeting have wasted lot of meeting time. The Instructors work was challenging as chat messages often come in one after another, with other student’s chat messages mixing in. This makes it difficult to follow the conversation and respond to questions in an organized way. However, In Case II, using the Bot finished the entire process in 3 minutes and 55 seconds, with 2 minutes spent idle while waiting for student’s responses. This means the Bot completed the task in just 1 minute and 55 seconds (115 seconds) of active time, highlighting the substantial time efficiency that automation can offer. The system’s ability to operate within this time range proves that the Bot is not only efficient but also scalable for larger sets of participants in Zoom meeting. The output of the Bot is shown in Figure 9.

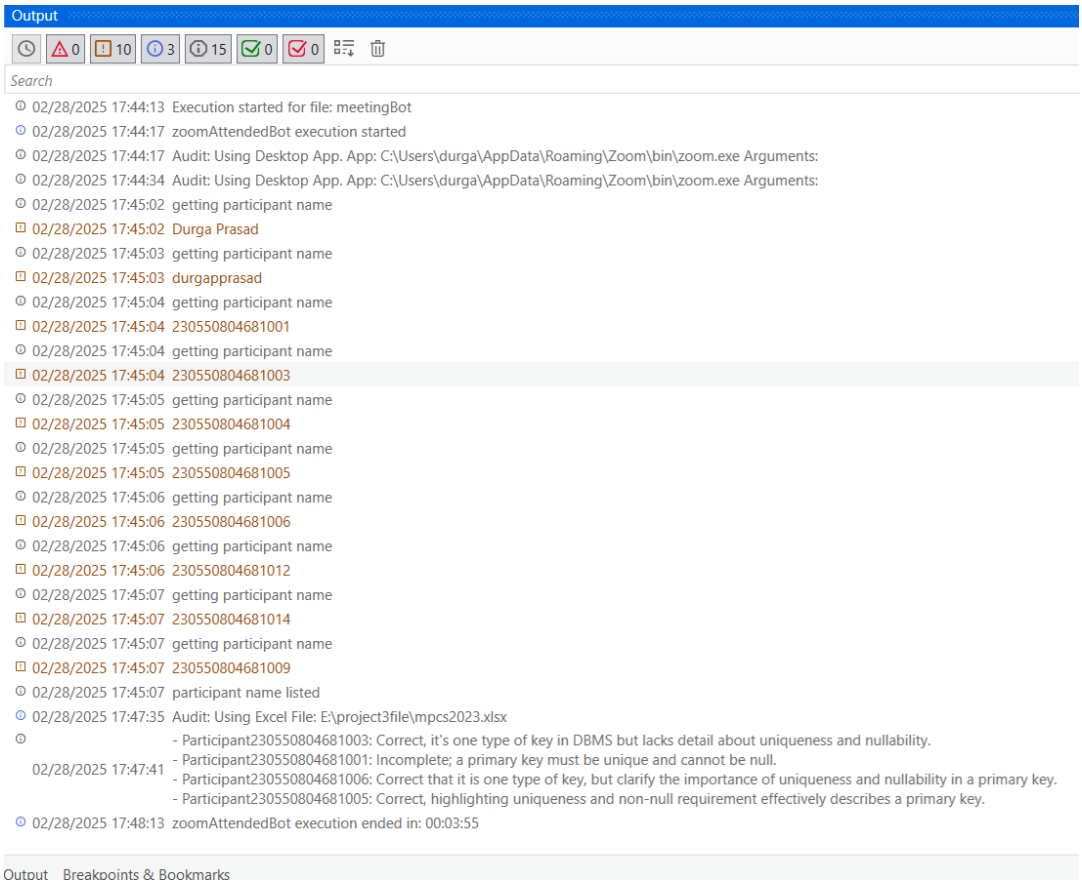


Fig. 9. Output Log file of the developed Bot.

3. Accuracy

When tasks were done manually (Case I), there were unavoidable mistakes caused by human errors like misreading emails or misspelling student IDs and their Present or Inactive Status. In contrast, the automated process that the Bot followed in Case II always guaranteed high accuracy. All student IDs were entered correctly with their status of presence along with time of meeting, and the results were stored in the corresponding Excel files without any discrepancies. This highlights the higher accuracy achieved by the Bot compared to manual processing. Bot maintaining student's attendance with their presence status (Present, P_Inactive, Absent). For every meeting, new column is added and the Bot takes the meeting time as column header automatically as shown in Figure 10.

Hallticket No	2025-02-14_12-53	2025-02-15_11-58	2025-02-17_12-16	2025-02-18_12-49	2025-02-19_13-03	2025-02-18_13-06	2025-02-22_13-58	2025-02-24_14-19	2025-02-25_14-33	2025-02-28_11-47
230550804681001	AB	P	AB	P	AB	P	P	P	AB	P
230550804681002	P	P	P-InActive	P-InActive	P	P-InActive	P-InActive	P-InActive	P	AB
230550804681003	AB	P-InActive	AB	P-InActive	AB	P-InActive	P	P	P	P
230550804681004	AB	P	P	P	P	P	P-InActive	P	P	P-InActive
230550804681005	P	AB	P	AB	P	AB	P	AB	P-InActive	P
230550804681006	P-InActive	P	P-InActive	P-InActive	P-InActive	P	P	P	P	P
230550804681007	P	P	AB	AB	AB	AB	P-InActive	AB	P	AB
230550804681008	AB	P-InActive	P	P	P	P-InActive	AB	AB	AB	AB
230550804681009	P	AB	AB	AB	AB	AB	P	AB	AB	P-InActive
230550804681010	AB	P	AB	AB	P	AB	P	P	P-InActive	AB
230550804681011	P-InActive	P-InActive	P	P	P-InActive	P	P-InActive	P-InActive	P	AB
230550804681012	AB	P-InActive	AB	P-InActive	P	P	AB	P	AB	P-InActive
230550804681013	P	P	P	P	P	AB	P	AB	P-InActive	AB
230550804681014	AB	P-InActive	AB	P	P-InActive	P	AB	P	AB	P-InActive

Fig. 10. Bot maintaining student's Attendance details with date & time of meeting taken automatically as column header.

4. Precision

The Bot executed its operations with minimal deviation from the desired results, exhibiting excellent precision. The automated process is efficiently optimized to prevent mistakes, particularly when dealing with dynamic content such as web pages with rapidly changing components, based on its capability to finish actions with high reliability.

4.2 New insights into three category classifications

Previously, in online meetings "Present" was misleading because students could be logged in meeting but be disengaged. Now, we can see who are truly participating and who are idle(P_Inactive). We can see from the Figure 11 that even some of the student's attendance is 100% their active engagement is 60% indicating their inactive status. This allows targeted Inventions for inactive students before they become fully Absent.

HtNo	Total Days	No of Days Present and active (P)	No of Days Present but Not Active (P-Inactive)	No of Days Absent	Attendance%	Active participation%
230550804681001	10	6	0	4	60	100
230550804681002	10	4	5	1	90	44
230550804681003	10	4	3	3	70	57
230550804681004	10	7	2	1	90	78
230550804681005	10	5	1	4	60	83
230550804681006	10	6	4	0	100	60
230550804681007	10	3	1	6	40	75
230550804681008	10	3	2	5	50	60
230550804681009	10	3	0	7	30	100
230550804681010	10	4	1	5	50	80
230550804681011	10	4	5	1	90	44
230550804681012	10	3	3	4	60	50
230550804681013	10	6	1	3	70	86
230550804681014	10	3	3	4	60	50

Fig. 11. Bot also maintains weekly Attendance of students along with Active participation percentage.

4.3 Computational complexity

In the case of UiPath RPA, computational complexity refers to the efficiency of the Bot's execution of tasks. It comprises aspects like the use of resources (CPU, memory) and the duration for which the process is completed. Yet, common computational analysis approaches such as Big O notation are not directly extendable to RPA solutions. The performance of the Bot was validated by executing data processing for various Zoom meetings, with findings proving that the Bot took only 115-140 seconds of active time where additional 2 minutes is Bot's idle time. The performance shows that the Bot works well even with big datasets.

4.4 Early Identification of Students Needing Support

GPA scores and attendance data were used to analyze 902 students over a 30-day period. The three levels of attendance behavior—Present, P_Inactive, and Absent—were numerically encoded for machine learning analysis. There were also other behavioral characteristics like the number of inactive days, the number of absent days, and an attendance trend that contrasted recent and past attendance. Despite having about three inactive and six absent days, the average student had a high attendance score of 0.75. With an average GPA of 0.35, the scores showed a wide range of academic performance levels. Although some students saw a decline in attendance, the attendance trend metric revealed that many students started to improve towards the end of the 30-day period, suggesting a gradual improvement in engagement. Students with unusual attendance-performance patterns were identified using three anomaly detection models: Isolation Forest, One-Class SVM, and DBSCAN. The results are displayed in the figure 12 below.

```
Total students analyzed: 902

Summary Statistics:
                      mean       std   min     max
attendance_score  0.749834  0.076971  0.50   0.950
inactive_count    3.129712  1.675648  0.00   9.000
absent_count      5.940133  2.277323  0.00  13.000
gpa_score         0.350865  0.394337  0.00   0.984
attendance_trend  0.005931  0.180749 -0.55   0.500

Outliers Detected Per Model:
                   Model  Outliers Detected
0       Isolation Forest                136
1          One-Class SVM                137
2                 DBSCAN                 62
3  Consensus (≥2 models)                112
4        Soft (≥1 model)                173
```

Fig. 12. Attendance trends and model wise outlier detection with consensus approach.

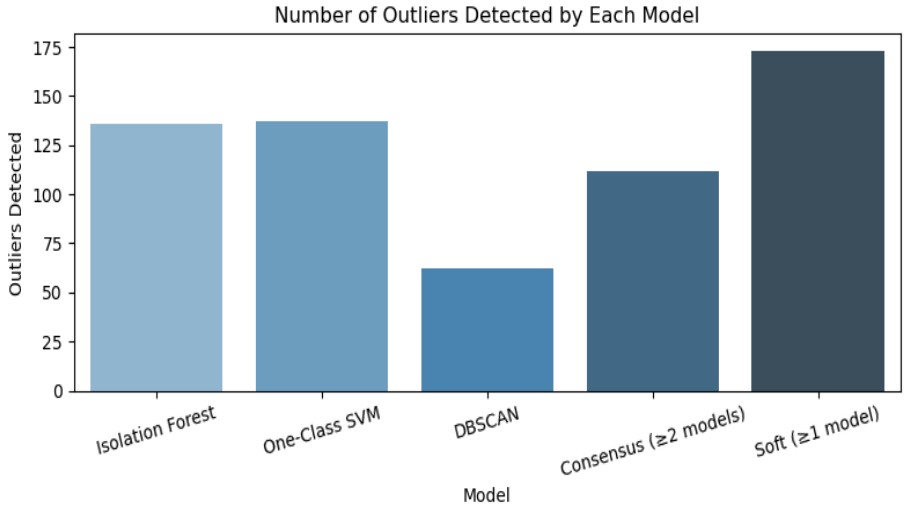


Fig. 13. A bar chart showing number of outlier's detected by each model.

4.5 User perception: Student and Faculty

To gather feedback, instructors and students had informal discussions after live sessions. verbal comments and usage observations were documented in order to understand user experience and perception. Both faculty and student feedback indicated that they had a good experience using the system during live sessions. Students thought that class activities were better structured, particularly in terms of facilitating discussions and providing prompt answers to queries. Particularly in larger groups, this structure made students feel simpler to participate and maintain focus.

On the other hand, when the instructor sends the session's meeting ID, passcode, and questions with keywords to the official email address, the Bot is automatically activated via email. The instructor's device is not impacted because it joins the Zoom session independently from a different computer. UiPath Orchestrator, which offers total control, error handling, and retry mechanisms, ensures dependability and manages the entire automation. The Bot instance joins the meeting to gather important data like participant responses, chat messages, and attendance. Instructors can view the Bot's conversation in the Zoom chat window at any time and are free to use their systems as normal. This enables them to keep an eye on interactions, confirm the accuracy of responses, and follow the engagement's progression without having to step in. Instructors reported that it freed them up to focus more on instruction by reducing repetitive duties like tracking attendance, keeping a watch on participation, and collecting responses. Additionally, they also appreciated having full oversight, with the ability to review and adjust any automated output if needed. All things considered, the system was viewed as a beneficial by the instructor which enhanced class flow while maintaining the instructor's central role.

5. Conclusion

The Bot, which was developed by using UiPath, was a powerful and efficient tool for automating routine work on video conferencing applications. By incorporating email management, UI automation, data scraping, and Excel file management, the Bot significantly reduced manual labor, improved efficiency, and improved accuracy resulting in greater student engagement. The performance test demonstrated that the Bot could accomplish tasks within minutes as opposed to manual methods while generating high accuracy. Moreover, the fact that the Bot can handle huge data and run consistently on multiple runs demonstrates that the Bot can be utilized on a large scale in educational institutions.

Author Contributions Statement

M. Durga Prasad – Conceptualization, Methodology, Data Curation, Software Implementation, Model training, validation, Performance Evaluation, Formal Analysis, Visualization and writing.

Balusu Nandini – Supervision, Writing, validation, Performance Evaluation, Review and Editing and Project Management.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors declare no conflicts of interest.

Funding Declaration

This research received no external funding.

Data Availability Statement

Data sharing is not applicable to this article.

Ethical Declarations

The authors declare that all procedures performed in this study involving students were conducted in accordance with ethical standards. Informed consent was obtained from all participants involved in the study.

Acknowledgments

We sincerely thank the experts for their professional evaluation and valuable recommendations, which have contributed to improving the quality of the experiment and the reliability of its results.

Declaration of Generative AI in Scholarly Writing

The authors declare that no generative artificial intelligence (AI) were used in the writing process of this manuscript.

Abbreviations

The following abbreviations are used in this manuscript:

AI - Artificial Intelligence
API - Application Programming Interface
BSLE - blended synchronous learning environment
CBR - Case-Based Reasoning
ERP - Enterprise Resource Planning
GPA - Grade Point Average
IMAP - Internet Message Access Protocol
LLM - Large Language Model
RPA - Robotic process automation
UI - User Interface

References

[1] Gloria Aparicio, Txomin Iturralde, and Amaia Maseda, “A holistic bibliometric overview of the student engagement research field,” Journal of Further and Higher Education, vol. 45, no. 4, pp. 540–557, Apr. 2021, doi: 10.1080/0309877X.2020.1795092.
[2] Katherine Wimpenny and Maggi Savin-Baden, “Alienation, agency and authenticity: a synthesis of the literature on student engagement,” Teaching in Higher Education, vol. 18, no. 3, pp. 311–326, Apr. 2013, doi: 10.1080/13562517.2012.725223.
[3] Amy L. Reschly and Sandra L. Christenson, Eds., Handbook of Research on Student Engagement. Cham: Springer International Publishing, 2022. doi: 10.1007/978-3-031-07853-8.
[4] Jennifer A Fredricks, Phyllis C Blumenfeld, and Alison H Paris, “School Engagement: Potential of the Concept, State of the Evidence,” Review of Educational Research, vol. 74, no. 1, pp. 59–109, Mar. 2004, doi: 10.3102/00346543074001059.
[5] Vicki Trowler, “Student engagement literature review,” vol. 11, no. 1, pp. 1–15, 2010.
[6] Melissa Bond, Katja Buntins, Svenja Bedenlier, Olaf Zawacki-Richter, and Michael Kerres, “Mapping research in student engagement and educational technology in higher education: a systematic evidence map,” Int J Educ Technol High Educ, vol. 17, no. 1, p. 2, Dec. 2020, doi: 10.1186/s41239-019-0176-8
[7] Florence Martin, Ting Sun, and Carl D. Westine, “A systematic review of research on online teaching and learning from 2009 to 2018,” *Computers & Education*, vol. 159, p. 104009, Dec. 2020, doi: 10.1016/j.compedu.2020.104009
[8] Laura A. Schindler, Gary J. Burkholder, Osama A. Morad, and Craig Marsh, “Computer-based technology and student engagement: a critical review of the literature,” Int J Educ Technol High Educ, vol. 14, no. 1, p. 25, Dec. 2017, doi: 10.1186/s41239-017-0063-0.
[9] Panagiotes S. Anastasiades et al., “Interactive Videoconferencing for collaborative learning at a distance in the school of 21st century: A case study in elementary schools in Greece,” Computers & Education, vol. 54, no. 2, pp. 321–339, Feb. 2010, doi: 10.1016/j.compedu.2009.08.016.

[10] Cheryl Mader and Kavin Ming, "Videoconferencing: A New Opportunity to Facilitate Learning," The Clearing House: A Journal of Educational Strategies, Issues and Ideas, vol. 88, no. 4, pp. 109–116, Jul. 2015, doi: 10.1080/00098655.2015.1043974.
[11] Qiyun Wang, Changqin Huang, and Choon Lang Quek, "Students' perspectives on the design and implementation of a blended synchronous learning environment," AJET, Apr. 2017, doi: 10.14742/ajet.3404.
[12] Abu Shadat Muhammad Sayem, Benjamin Taylor, Mitchell McClanachan, and Umme Mumtahina, "Effective use of Zoom technology and instructional videos to improve engagement and success of distance students in Engineering," presented at the 28th annual conference of the Australasian association for engineering education (AAEE 2017), 2017.
[13] Mandy M. Archibald, Rachel C. Ambagtsheer, Mavourneen G. Casey, and Michael Lawless, "Using Zoom Videoconferencing for Qualitative Data Collection: Perceptions and Experiences of Researchers and Participants," International Journal of Qualitative Methods, vol. 18, p. 1609406919874596, Jan. 2019, doi: 10.1177/1609406919874596.
[14] Chandra Fitra Arifianto, Mutawali, and Heny Subekti, "The Teachers' Online Readiness: an evaluation of Online Learning during Covid-19 Pandemic in Indonesia," ijsl, vol. 1, no. 3, pp. 270–282, Aug. 2021, doi: 10.47134/ijsl.v1i3.63.
[15] Kristine Anne Dimaculangan, Hanifa Hadji Abas, and Celestie S. Quinto, "Narrative Study of Teaching Strategies and Challenges Encountered by Teachers in Synchronous Online Classes," ijsl, vol. 2, no. 2, pp. 201–216, Apr. 2022, doi: 10.47134/ijsl.v2i2.113.
[16] Mutiu Salami, Rahmattullah Khan, Muhammed Yusuf, Asma Perveen, and Mohammed Y.M. Mai, "Impact of Perceived Academic Stress and Depression on Self Efficacy Beliefs among University Students during Online Learning in Peninsula, Malaysia," ijsl, vol. 1, no. 3, pp. 260–269, Aug. 2021, doi: 10.47134/ijsl.v1i3.53.
[17] Qonita A'isy, Fitrotul Mufaridah, and Nurkamilah Nurkamilah, "The Challenges of Teaching Speaking to High School Students through Zoom Video Conference Application," jpbi, vol. 1, no. 2, p. 8, Oct. 2023, doi: 10.47134/jpbi.v1i2.49.
[18] Derar Serhan, "Transitioning from Face-to-Face to Remote Learning: Students' Attitudes and Perceptions of using Zoom during COVID-19 Pandemic," IJTES, vol. 4, no. 4, pp. 335–342, Sep. 2020, doi: 10.46328/ijtes.v4i4.148.
[19] Jose Ma Gopez and Beatriz Gopez, "Instructor scaffolding for interaction and online student engagement among a sample of college students in the Philippines: the mediating role of self-regulation," Eur J Psychol Educ, vol. 39, no. 2, pp. 1069–1091, Jun. 2024, doi: 10.1007/s10212-023-00728-y.
[20] Regina L. Garza Mitchell, Whitney DeCamp, Brian S. Horvitz, Megan Grunert Kowalske, and Cherrelle Singleton, "'I'm Not Teaching Them Per Se': Designing and Delivering Asynchronous Undergraduate Online STEM Courses," Innov High Educ, vol. 49, no. 1, pp. 91–111, Feb. 2024, doi: 10.1007/s10755-023-09670-9.
[21] Xindi Kong, Hongyu Liang, Chunsheng Wu, Zheyan Li, and Yuxin Xie, "The association between perceived teacher emotional support and online learning engagement in high school students: the chain mediating effect of social presence and online learning self-efficacy," Eur J Psychol Educ, vol. 40, no. 1, p. 33, Mar. 2025, doi: 10.1007/s10212-024-00932-4.
[22] Julius Zulu, Patricia Phiri Nalube, Robert Changwe, and Simeon Mbewe, "The challenges and opportunities of using Zoom App in the teaching and learning of mathematics in Higher Education Institutions (HEIS) during COVID-19 pandemic: Lecturers' and students' perspective," International Journal of Research and Innovation in Applied Science (IJRIAS), vol. 6, no. 7, pp. 92–101, 2021.
[23] Sofia Khan, R. K. Tailor, Ranu Pareek, Rashmi Gujrati, and Hayri Uygun, "Application of Robotic Process Automation in education sector," Journal of Information and Optimization Sciences, vol. 43, no. 7, pp. 1815–1834, Oct. 2022, doi: 10.1080/02522667.2022.2128534.
[24] Nishant Jha, Deepak Prashar, and Amandeep Nagpal, "Combining Artificial Intelligence with Robotic Process Automation—An Intelligent Automation Approach," in Deep Learning and Big Data for Intelligent Transportation, vol. 945, Khaled R. Ahmed and Aboul Ella Hassanien, Eds., in Studies in Computational Intelligence, vol. 945. , Cham: Springer International Publishing, 2021, pp. 245–264. doi: 10.1007/978-3-030-65661-4_12.
[25] U. Inc, "UiPath automation platform: drive AI transformation with agentic automation | UiPath." Accessed: Apr. 02, 2025. [Online]. Available: https://www.uipath.com
[26] Judith Wewerka and Manfred Reichert, "Robotic Process Automation -- A Systematic Literature Review and Assessment Framework," 2020, arXiv. doi: 10.48550/ARXIV.2012.11951.
[27] University of Arkansas (U.S.), Mary Lacity, Leslie Willcocks, and London School of Economics and Political Science (U.K.), "Becoming Strategic with Intelligent Automation," MSQE, pp. 169–182, 2021, doi: 10.17705/2msqe.00047.
[28] Izzat Alsmadi and Ikdam Alhami, "Clustering and classification of email contents," Journal of King Saud University - Computer and Information Sciences, vol. 27, no. 1, pp. 46–57, Jan. 2015, doi: 10.1016/j.jksuci.2014.03.014.
[29] M. Asif Naeem, I. Wayan S. Linggawa, Aftab A. Mughal, Christof Lutteroth, and Gerald Weber, "A Smart Email Client Prototype for Effective Reuse of Past Replies," IEEE Access, vol. 6, pp. 69453–69471, 2018, doi: 10.1109/ACCESS.2018.2878523.
[30] Soya Park, Amy X. Zhang, Luke S. Murray, and David R. Karger, "Opportunities for Automating Email Processing: A Need-Finding Study," in Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Glasgow Scotland Uk: ACM, May 2019, pp. 1–12. doi: 10.1145/3290605.3300604.
[31] Min Peng, Binlong Gao, Jiahui Zhu, Jiajia Huang, Mengting Yuan, and Fei Li, "High quality information extraction and query-oriented summarization for automatic query-reply in social network," Expert Systems with Applications, vol. 44, pp. 92–101, Feb. 2016, doi: 10.1016/j.eswa.2015.08.056.
[32] Jorge Ribeiro, Rui Lima, Tiago Eckhardt, and Sara Paiva, "Robotic Process Automation and Artificial Intelligence in Industry 4.0 – A Literature review," *Procedia Computer Science*, vol. 181, pp. 51–58, 2021, doi: 10.1016/j.procs.2021.01.104
[33] Mamidyala Durga Prasad and Balusu Nandini, "Faculty Assistant Bot-automation of administrative activities using robotic process automation," IJECE, vol. 14, no. 6, p. 6797, Dec. 2024, doi: 10.11591/ijece.v14i6.pp6797-6806.
[34] Arpit Khare, Sudhakar Singh, Richa Mishra, Shiv Prakash, and Pratibha Dixit, "E-Mail Assistant – Automation of E-Mail Handling and Management using Robotic Process Automation," in 2022 International Conference on Decision Aid Sciences and Applications (DASA), Chiangrai, Thailand: IEEE, Mar. 2022, pp. 511–516. doi: 10.1109/DASA54658.2022.9765017.

[35] OpenAI, “OpenAI API Documentation.” OpenAI, Sep. 03, 2025. [Online]. Available: https://platform.openai.com/docs
[36] W. Morris, S. Crossley, L. Holmes, C. Ou, M. Dascalu, and D. McNamara, “Formative feedback on student-authored summaries in intelligent textbooks using large language models,” International Journal of Artificial Intelligence in Education, pp. 1–22, 2024.
[37] S. Svetsky and O. Moravcik, “The implementation of digital technology for automation of teaching processes,” presented at the 2016 Future Technologies Conference (FTC), IEEE, 2016, pp. 340–348.
[38] G. Kearsley and B. Shneiderman, “Engagement theory: A framework for technology-based teaching and learning,” Educational technology, vol. 38, no. 5, pp. 20–23, 1998.
[39] W. Cui, Z. Xue, and K.-P. Thai, “Performance comparison of an AI-based adaptive learning system in China,” presented at the 2018 Chinese automation congress (CAC), IEEE, 2018, pp. 3170–3175.
[40] A. Al Redhaei, M. Awad, and K. Salameh, “Assessing the Impact of Gamification in Higher Education: An Experimental Study using Kahoot! and Nearpod During COVID-19 Pandemic,” presented at the 2022 Advances in Science and Engineering Technology International Conferences (ASET), IEEE, 2022, pp. 1–6.

Authors’ Profiles



M. Durga Prasad is Pursuing Ph.D. in Computer Science & Engineering from Department of Computer Science & Engineering, Telangana University, Telangana State, India. He completed his B.Tech. Computer Science and Information Technology and M, Tech. Computer Science and Engineering from Jawaharlal Nehru Technological University, Hyderabad. His research work mainly focuses on machine learning and artificial intelligence.



Dr. B. Nandini has completed her Ph.D. from Jawaharlal Nehru Technological University, Hyderabad. Her current research focuses on artificial intelligence, block chain technologies, machine learning, deep learning and computer networks. Currently, she is working as Associate Professor in the Department of Computer Science & Engineering, Telangana University, Telangana State, India. She has published papers in SCI, SCOPUS, UGC CARE listed national, international Journals and conferences.

How to cite this paper: M. Durga Prasad, Balusu Nandini, "Enhancing Student Engagement and Attendance in online Learning environment by using AI Enhanced Robotic Process Automation", International Journal of Modern Education and Computer Science(IJMECS), Vol.18, No.2, pp. 38-53, 2026. DOI:10.5815/ijmecs.2026.02.03