International Journal of Computer Applications (0975 - 8887)
Volume 187 - No.123, July 2026

A Low Cost Cloud based Secure Solution for Automatic Examination Seating Arrangement using Google Platform

Amar Arora
USAR, GGSIPU
East Delhi Campus, Surajmal Vihar
Delhi, INDIA

Amrit Pal Singh
USAR, GGSIPU
East Delhi Campus, Surajmal Vihar
Delhi, INDIA

ABSTRACT

In any university system, seating plan arrangement in an examination is a critical administrative task and when handled manually, it is time consuming, error-prone, and susceptible to malpractice. This article presents a low-cost cloud-based solution for automating the examination seating arrangement process using the Google Platform—specifically Google Sheets and Google Apps Script. The proposed system eliminates the need for dedicated server infrastructure by leveraging Google's free cloud services, making it accessible to institutions with limited budgets without compromising the functionality or security. The system comprises Google Sheet for initiating Google Apps Script function to initiate creation of seating arrangement comprising master examination data such as date, shift, paper id, etc. This called function internally calls a Google Script API, which collects student roll number, etc from different master sheets to create the seating plan in a separate Google Sheet. This API is protected through a shared key based authentication mechanism to ensure the security and integrity of the seating data. The system automatically generates seating plans by distributing students from different departments or courses across examination halls according to the selection of rooms made by the seating plan manager. Experimental results demonstrate that the system significantly reduces administrative workload, minimizes human error, accelerates the entire process, and improves overall examination security. The solution is scalable, platform-independent and does not require additional hardware investment beyond an internet-enabled device.

Keywords

Examination seating arrangement, Google Sheets, Google Apps Script, cloud computing, shared key protected, academic integrity, automation

1. INTRODUCTION

The conduct of exams is one of the most fundamental activities in any educational institution. This becomes more complex in university systems as a central conduct team needs to make a common seating plan for multiple University Schools containing multiple programs which are further segregated into semesters and subjects. The major requirement for the integrity of this process is the systematic arrangement of students in examination halls. A well-designed seating plan ensures optimal use of space with the condition that students from the same class or course are separated, thereby reducing opportunities for academic dishonesty, such as copying or collusion. However, in many universities or academic institutions, particularly those with limited administrative resources, seating arrangements are still performed manually, which is tedious, error-prone, difficult to scale, and very time consuming. Manual seating arrangement involves significant repetitive work: gathering student enrollment data, allocating examination rooms, distributing students, and cross checking with the master data for any left out or repetition of roll numbers. Even after manual verification, there is the possibility of a duplicate seat number, a miscounted room capacity, or a same-class cluster. It can compromise the fairness and security of the entire examination and can even lead to last minute confusion. As student populations grow and examination schedules become more complex, the inadequacy of manual methods becomes increasingly apparent [1, 9].

The rapid adoption of cloud computing and web-based collaboration platforms presents a compelling opportunity to address these shortcomings. Google Workspace, in particular, offers a suite of powerful, freely available tools—including Google Sheets and Google Apps Script—that can serve as the foundation for lightweight but robust institutional automation. Unlike enterprise software solutions that require expensive licenses, dedicated servers, and specialist IT staff, Google-platform-based solutions are accessible to virtually any institution with Internet [19] connectivity [20]. This paper proposes and demonstrates a low-cost cloud-based automatic examination seating arrangement system built entirely on the Google Platform. The system uses Google Sheets as the data layer and Google Apps Script as the computation and logic layer, enabling automated seat allocation for students from multiple university schools and their respective departments or courses, even when they share a common exam duration and limited physical infrastructure, without compromising the integrity of the examination process. To protect the integrity of the computation and logic layer, the system incorporates a shared key based authentication mechanism to interact with the API, ensuring that even the person autho-

rized to create the seating plan will not be able to access the control and logic layer.

The remainder of this paper is organized as follows. Section 2 reviews related work in the domain of automated seating arrangement during the examination process. Section 3 presents the system architecture and design. Section 4 describes the implementation in detail. Section 5 presents the results and a discussion of the system's performance. Section 6 concludes the paper and outlines directions for future work.

2. RELATED STUDY

The automating of examination seating arrangements has been a topic of research involving a decade, employing a range of algorithmic and technological approaches. This section surveys the most relevant prior work in this area.

Most of the initial efforts were focused on web-based database-driven applications. Aravinth et al. [6] developed an exam hall seating system using PHP, demonstrating the feasibility of automating seat assignments through a web interface. Adetona et al. [1] developed a comprehensive web-based application that incorporated departmental separation to curb malpractice, and subsequently refined this into a fully implemented system [2] that addressed real-world constraints such as varying hall capacities and multi-subject exams. The proposal by Chaki and Anirban [9] focused on minimizing the inter-department proximity and development of an efficient algorithm for centralized examination systems. In a recent proposal by Alawode and Adegboye [4] demonstrated superior results over greedy approaches in the seating arrangement problem. Agalday and Nizam [3] extended this work by applying genetic algorithms and investigating the effect of parameter optimization on solution quality, achieving measurable performance gains through careful tuning of crossover and mutation rates. Kishore et al. [22] explored the area of exam hall logistics and developed an enhanced automation system, and Anjum et al. [5] addressed hall allotment and seating arrangement within a single integrated framework. Moreover, Dhotre et al. [11] and Inamdar et al. [18] each presented their respective automatic seating arrangement systems along with invigilator or faculty duty allocation as a separate task. More recently, artificial intelligence and deep learning have also been used to enhance the solution. Dharanya et al. [10] proposed a face recognition-based system that ensures that the student actually sits in their allocated position as per the seating plan, adding a biometric layer of security. Khiangte et al. [21] explored the use of prompt engineering with large language models to generate seating plans, representing a novel application of generative AI to institutional administration. Pathan and Reshamwala [23] described a digital allocation system specifically designed to enhance exam integrity through algorithmic dispersion. Ayob and Malik [8] formalized the examination-room assignment problem as a combinatorial optimization model, providing a theoretical underpinning for many practical approaches. Kadarkarai et al. [20] presented a recent automated system that further refined the scheduling constraints for multi-subject scenarios. The presented literature demonstrates the various advancements in the area of seating plan automation including the machine learning and generative AI based solutions as well. However, a major limitation of the surveyed literature is the underlying high development cost and infrastructure requirements. Most of the proposed systems require dedicated web servers, proprietary software, or costly hardware. These costs further escalate with the solutions including machine learning and Generative AI with the requirement of the GPU based execution. This creates barriers to adoption for institutions with limited IT budgets. The present work addresses this gap by constructing a fully functional, secure seating arrangement system using only freely available Google cloud services, requiring no additional infrastructure beyond a standard internet-connected device.

3. SYSTEM ARCHITECTURE AND DESIGN

The proposed system comprises the following components:

(1) Master Student Data in Google Docs [14] [C1]

(2) Master Data Sheet Containing Room Structures, Room Dimension Master Data, Program Master, etc. in Google Sheet [15][C2]

(3) Google Apps Script [13] in the Master Data Sheet [C3]

(4) Google Cloud Standalone API [12][C4]

Initially, the setup requires the student data as per the examination schedule, i.e. list of roll numbers, name, course code, paper id, paper code, etc, which is normally provided by the examination department. These data are stored in separate Google Docs [C1] segregated according to the program code. These data act as master student data to obtain the required roll numbers for the room allocation. Thereafter, the complete examination schedule is stored in a separate Google Sheet [C2] containing meta data of room structure, such as number of rooms, column and rows inside a room, program and master student data [C1] file mapping, etc. It also serves as a gateway to the system where the initial request for a seating plan generation is executed via the Google Apps Script function attached within Google Sheet [C3]. The Apps Script function in C3 internally makes an authenticated API call to Google Cloud Standalone API Script [C4] through a token generated using HMACSha256Signature [16] using shared key, random nonce and timestamp. This token also has a maximum validity of 5 minutes to prevent any replay attack [17]. Once authenticated, the API opens the POST request data which contains the details such as Program Id, Date and session of exam, Paper Id, Batch, etc. which identifies the unique Google Docs [C1], and from the same the roll numbers for the scheduled exam are fetched via pattern matching. Once these data are fetched, the Standalone API [C4] fetches the room numbers and their structure from C2 and creates a new Google Sheet in the predefined Google Drive Folder to create a replica of the Room Structure and starts filling the designated rooms with the fetched roll numbers and other required details. Once all the roll numbers are filled, the Standalone API ends its operation and returns a success message. The Google Apps Script on the Master Data Sheet [C3] returns a success message to the user to confirm the successful generation of the seating plan. This architecture ensures the security and isolation of critical modules without affecting user accessibility.

4. IMPLEMENTATION DETAILS

The project was implemented completely in Google Cloud and Google Drive ecosystem starting with a master Google Sheets namely "Seating Plan Automation" consisting of following sheets:

(1) Shift Master: It contains mapping of shift Id with Shift details, e.g. Shift ID 1 corresponds to 10:00 AM to 01:00 PM (Morning) and so on.

(2) Program Master: It contains mapping between Program Code and File Name of Google Docs where student data needs to be fetched as discussed in section 3.

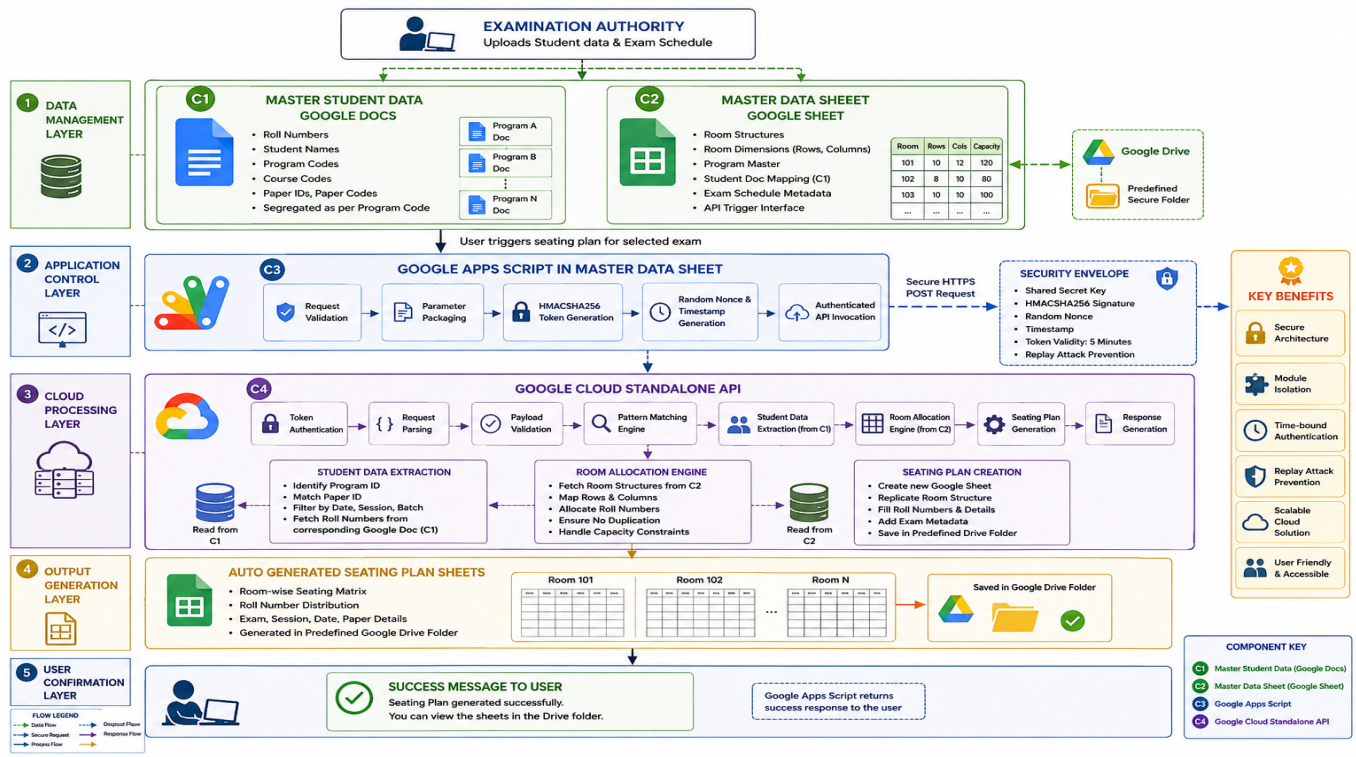

Fig. 1. System architecture of the proposed Google-platform-based seating arrangement solution, showing the data management, application control, cloud processing, and output layers (C1–C4)

(3) Room Master: It contains the meta data (ID Room Name, StartRow, EndRow, StartColumn, EndColumn, DateRowNumber, DateColumnNumber ShiftRowNumber, ShiftColumnNumber, ProgramNameRowNumber, ProgramNameColNumber SummaryRowNumber, Filling Priority) for each room in the master seating plan. It helps in identification of correct position to fill the roll numbers according to the selection of a specific Room ID.

(4) Master Seating Plan: It is the main template that contains the blank seating plans for every room according to the seating layout.

(5) Subject Summary Location Master: It is similar to the Room Master, but it contains mapping of the Subject Summary Id and Paper Code Number to fill the summary of roll numbers for a paper code which is filled in Subject summary master.

(6) Subject Summary Master: This acts as a summary sheet of each paper code mentioning the rooms and strength per room mentioned in column fashion. It also acts as a receiving document for the conduct team while handing over the sheets to the concerned faculty.

(7) Master Data: It acts as the master where the whole exam schedule is mentioned containing data such as Semester, Exam Date, Shift, Total Strength, Paper Id, Paper Code, Paper Name, Program Code, Batch, Program Short Name.

The Master Data also contains a button which is connected to a Google Apps Script function to execute the seat filling by calling Standalone API. Below these details, the seating plan manager first fills all the Room IDs in sequence and then selects the respective column and clicks the seating plan generation button. The script once initiated first fetches all the details from the selected column and looks for the respective Ids in respective Masters to search for respective roll numbers in the designated files, and once the roll numbers have been extracted, it starts the creation of new Google Sheet with the copy of the Master Seating Plan and Subject Summary Master which act as the blank templates to be filled with roll numbers as per selected sequence of room numbers. The execution details are further explained with system snapshots in the next section.

5. RESULTS

The initiation of the seating plan allocation starts with the selection of the requisite column from the Master Data sheet as shown in S.No. 1 ”INITIATION - COLUMN SELECTION” at Figure 2. It is evident from the figure that once a column is selected and the button on the left top portion labeled ”Create Seating Plan for the Selected Column” is pressed, the meta data of the examination are fetched along with the Room ID listed below the meta data by the authorized personnel. As the room numbers start to fill, their color will change from white to Yellow as shown in S.No. 2 ”REAL TIME ALLOCATION PROGRESS” in Figure 2 while it is partially filled and it turns to green once the room capacity for the current course roll numbers has been reached. This allows the user to monitor the real time progress of the seating plan completion. Once the execution of the entire column completes, if the last room is still not completely filled, it will remain yellow, indicating the availability of space in the room for the paper code under consideration. The system also displays the number of students that were filled out of

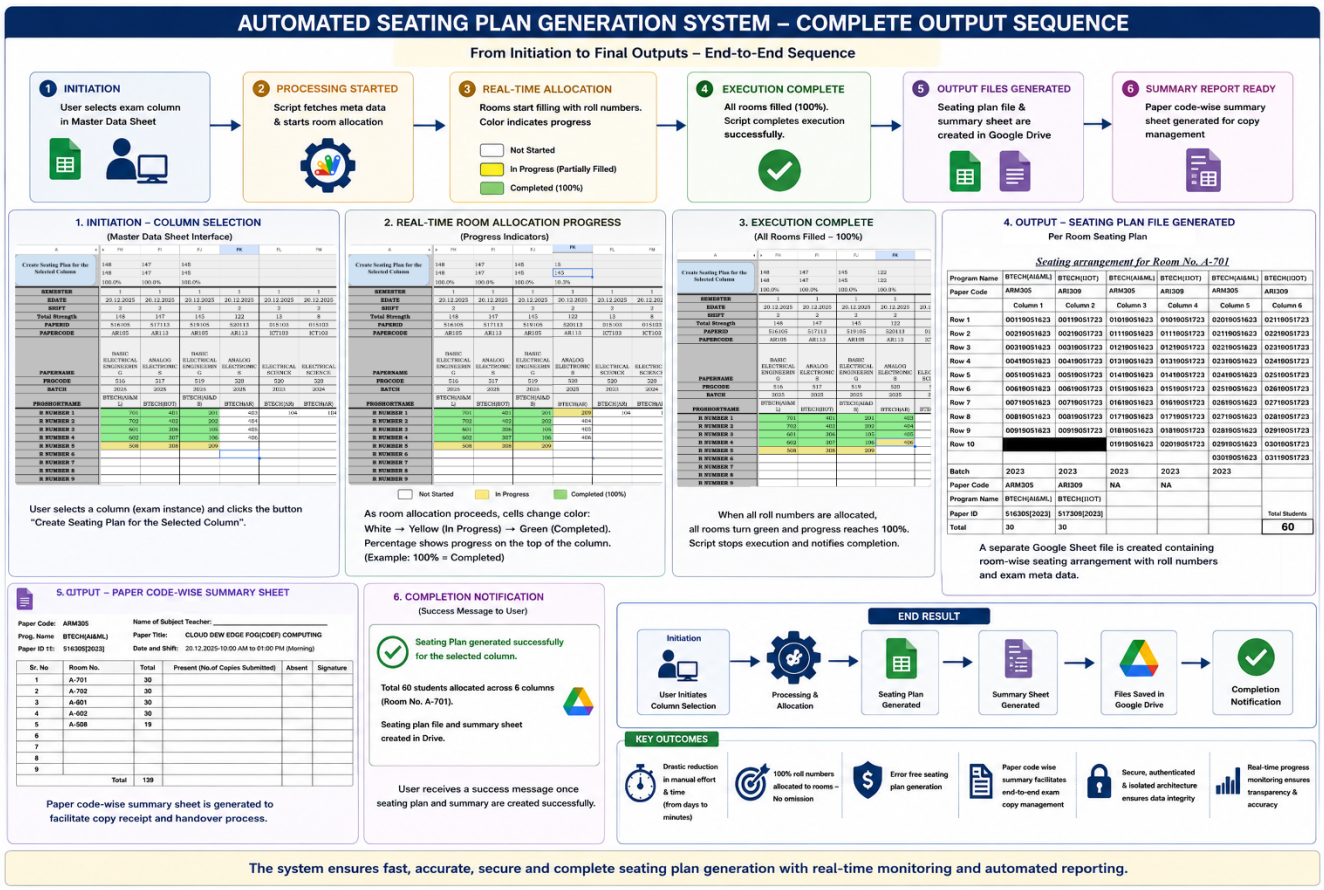

Fig. 2. End-to-end execution sequence of the seating plan generation process, from column selection to seating plan and summary sheet generation

the total number of students in the current execution by displaying the percentage filled at the top of the column. It ensures that every roll number scheduled for the exam has been allocated a room and leaves no room for error by chance.

The script completes execution once the allocation reaches 100%, and then the system notifies the user with a completion notification message. As a result, a separate file containing the seating plan as per the roll numbers is created as given in S.No. 4 "OUTPUT - SEATING PLAN GENERATED" in Figure 2.

While creating the seating plan, the system also creates a paper code wise summary sheet as shown at S.No. 5 "OUTPUT - PAPER CODE-WISE SUMMARY SHEET" in Figure 2 which facilitates receiving of the copies while handing them over to the concerned faculty member. The current system creates error-free seating plans in a fraction of the time that manual preparation used to take — a process that previously required several days. In addition, its paper code wise summary report ensures error-free handling of the physical movement of answer copies throughout the end-to-end examination conduct process.

6. CONCLUSION

This paper has presented a low cost, cloud based solution for automated examination seating arrangement built on the Google Platform. By leveraging Google Sheets as the data and presentation layer and Google Apps Script as the computation layer, the system achieves full automation of the seating plan generation drastically reducing generation time, eradicating any human error without any requirement for dedicated server infrastructure or software licensing costs.

Incorporating shared-key authentication, based on an HMAC-Sha256Signature [16] token, ensures that access to administrative functions is restricted to authorized personnel — providing an effective, lightweight security layer suited to the institutional context. Experimental evaluation confirms that the system correctly enforces seating constraints involving multiple university schools and programs, provides the required flexibility to select rooms as and when required, performs within acceptable time limits for typical examination scenarios, and produces output suitable for direct use by invigilators, students, and examination branches of the university. These results demonstrate that high-quality examination management tools need not be expensive or complex; the Google Platform provides sufficient capability for robust, secure, and scalable solutions.

Future work will explore role based access control [7], and the generation of QR-code-enabled digital seat tickets to further streamline the examination process.

7. REFERENCES

[1] Sunday Adetona, Edison Hassan, Raifu Salawu, and Samuel Omolola. The development of a web-based application of examination seating arrangement for student. *ABUAD Journal of Engineering Research and Development*, 3(1):23–33, 2020.

[2] Sunday Oladejo Adetona and Faith Akindayomi Akintoye. Design and implementation of an examination seating arrangement application to curb examination malpractice. *ABUAD Journal of Engineering Research and Development*, 6(2):143–156, 2023.

[3] Fatih Ağalday and Ali Nizam. Performance improvement of genetic algorithm based exam seating solution by parameter optimization. *Journal of Innovative Science and Engineering*, 6(2):220–232, 2022.

[4] AJ Alawode and AJ Adegboye. Automatic seating arrangement system using tabu search algorithm. *FEPI-JOPAS*, 1(1):127–133, 2019.

[5] Shazia Anjum, Madhuri Devi Chodey, and Afzal C Muneeb. Automation of exam hall allotment and seating arrangement. *International Journal of Engineering Research & Technology*, 10(06):447–452, 2021.

[6] SS Aravinth, G Pavithra, M Myvizhimalar, D Divya, and M Rathinakrithika. Exam hall seating arrangement system using php. *International Journal of Innovative Research in Technology*, 1(11):258–261, 2014.

[7] Amar Arora and Anjana Gosain. Dynamic trust emergency role-based access control (dte-rbac). *International Journal of Computer Applications*, 175(24):44–50, October 2020.

[8] Masri Ayob and Ariff Malik. A new model for an examination-room assignment problem. *International Journal of Computer Science and Network Security*, 11(10):187–190, 2011.

[9] Prosanta Kumar Chaki and Shikha Anirban. Algorithm for efficient seating plan for centralized exam system. In *2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*, pages 320–325. IEEE, 2016.

[10] C Dharanya, N Saravanan, T Hemalatha, K Dinesh, and M Jagan. Face recognition for exam hall seating arrangement using deep learning algorithm. In *2024 4th International Conference on Pervasive Computing and Social Networking (ICPCSN)*, pages 130–133. IEEE, 2024.

[11] DR Dhotre, Sakshi Makwane, and Priyanka Lahase. Automatic exam seating arrangement system. *proceedings of (IRJET)-International Research Journal of Engineering and Technology*, 6, 2019.

[12] Google. Apps script api. `https://developers.google.com/apps-script/api`, 2026. Accessed: 2026-05-20.

[13] Google. Google apps script. `https://developers.google.com/apps-script`, 2026. Accessed: 2026-05-20.

[14] Google. Google docs. `https://docs.google.com`, 2026. Accessed: 2026-05-20.

[15] Google. Google sheets. `https://sheets.google.com`, 2026. Accessed: 2026-05-20.

[16] Google Developers. computehmacsha256signature. `https://developers.google.com/apps-script/reference/utilities/utilities#computeHmacSha256Signature(String,String,Charset)`. Accessed: 2026-05-20.

[17] Boudhayan Gupta. A replay-attack resistant message authentication scheme using time-based keying hash functions and unique message identifiers. *arXiv preprint arXiv:1602.02148*, 2016.

[18] Apurva Inamdar, Anand Gangar, Arun Gupta, and Varsha Shrivastava. Automatic exam seating & teacher duty allocation system. In *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pages 1302–1306. IEEE, 2018.

[19] V Jain and R Gupta. An algorithm for the stacking problem of asymmetric merged tunnelling. In *Proceedings-2013 International Conference on Machine Intelligence Research and Advancement, ICMIRA 2013*, pages 416–421, 2014.

[20] Pandiaraj Kadarkarai, Munibhagyala Nikhil, Panditi Sukumar, Edagotti Sathish Kumar, and Ginkala Muneswar. Automated seating arrangement for conducting examination. In *2025 International Conference on Multi-Agent Systems for Collaborative Intelligence (ICMSCI)*, pages 1245–1250. IEEE, 2025.

[21] Vanlalhmangaiha Khiangte, Vanlalhruaia, and T Gunendra Singh. Streamlining examination seating arrangements through prompt engineering: An innovative approach. In *NIELIT's International Conference on Communication, Electronics and Digital Technologies*, pages 577–586. Springer, 2024.

[22] AH Nandhu Kishore, A Sasireka, and K Vijay. Design and development of enhanced exam hall seating arrangement automation system. *INFORMATION TECHNOLOGY IN INDUSTRY*, 9(2):814–820, 2021.

[23] Mohd Shafi Pathan and Kashish Rajankumar Reshamwala. A digital examination seating allocation system to enhance exam integrity. *J. Smart Sens. Comput.*, 1(1):25206, 2025.