**“GuestGo AI: QR Code-Based Guest Management System with AI-Powered Facial Verification and Real-Time Visit Tracking using YOLO-Face with Attribute-Based Access Control System”**

Technological University of the Philippines Manila

College of Science

In partial fulfillment of the

Requirement for Methods of Research in Computing CC303-M

**By**

*Glenn R. Galbadores I*

Kurt Angelo P. Ballarta

Justine B. Mantilla

John Walter D. Marquez

Ken Zedrick E. Montano

Bachelor of Science in Computer Science - 3B

**May 2025**

**ABSTRACT**

GuestGo AI is a visitor management system integrating QR-based check-in and AI-powered facial recognition for real-time visitor tracking. This research addresses the limitations of manual and semi-automated systems by introducing end-to-end digital tracking from registration to exit. The system aims to enhance institutional security and accountability using technologies like YOLO-Face for identity verification and QR technology for seamless logging. Developed using web technologies (HTML, CSS, PHP, JavaScript, and SQL), it is evaluated using ISO 25010 standards. The solution contributes to digital transformation, efficiency, and secure visitor operations in institutional settings.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **TECHNOLOGICAL UNIVERSITY OF THE PHILIPPINES**  Ayala Blvd., Ermita, Manila, 1000, Philippines Tel No. +632-5301-3001 local 704  Email: [graduate@tup.edu.ph](mailto:graduate@tup.edu.ph) | Website: [www.tup.edu.ph](http://www.tup.edu.ph/) | Index No. | F-OGP-3.8-TPF |
| Issue No. | 01 |
| Revision No. | 00 |
|  | Software Engineering | Date | 11242017 |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | | **Page** |
| Title Page | | **i** |
| Abstract | | **ii** |
| Table of Contents | | **iii** |
| List of Tables | | **v** |
| List of Figures | | **vi** |
| List of Appendices | | **vii** |
| **Chapter 1** | **THE PROBLEM AND ITS SETTING** |  |
|  | Introduction | **1** |
|  | Background of the Study | **2** |
|  | Objectives of the Study | **3** |
|  | Scope and Delimitations of the Study | **4** |
|  | Significance of the Study | **5** |
| **Chapter 2** | **CONCEPTUAL FRAMEWORK** |  |
|  | Review of Related Literature and Studies | **7** |
|  | Conceptual Model of the Study | **9** |
|  | Operational Definition of Terms | **14** |
| **Chapter 3** | **METHODOLOGY** |  |
|  | Project Design | **15** |
|  | Project Development | **18** |
|  | Operation and Testing Procedure | **19** |
|  | Evaluation Procedure | **20** |
| **REFERENCES** |  | **22** |
| **APPENDIXES** |  | **26** |
| **RESEARCHER’S** | **PROFILE** | **28** |
|  |  |  |
|  |  |  |
|  |  |  |

**LIST OF TABLES**

**Table Title Page**

1 System Terms and Definitions 14

2 GuestGo AI Modules 18

3 Modules Testing Plan 19

4 Evaluation Metrics and Criteria 20

**LIST OF FIGURES**

**Figure Title Page**

1 GuestGo AI Development Architecture 12

2 GuestGo AI Database ERD 13

3 GuestGo AI System Flowchart 17

**LIST OF APPENDICES**

**Appendix Title Page**

A Gantt Chart 26

**Chapter 1**

**THE PROBLEM AND ITS SETTINGS**

**INTRODUCTION**

Modern institutions such as universities, corporate offices, and healthcare facilities continue to rely on outdated visitor tracking systems, including logbooks and basic digital sign-in forms. While historically sufficient, these methods are increasingly ineffective due to the need for heightened security, data accuracy, and real-time monitoring. Manual methods often result in data inconsistency, vulnerability to impersonation, and inefficiencies in guest handling processes.

The GuestGo AI system offers a comprehensive and innovative approach to visitor management by combining Quick Response (QR) code-based check-in with AI-powered facial recognition. The system ensures that each guest's identity is securely verified and their movement within the premises is tracked in real time. This hybrid solution addresses the major shortcomings of traditional systems and aligns with global trends in smart building security and digital transformation (Satari et al., 2014; ISO, 2019). By providing automated visitor logging, real-time verification, and dynamic tracking, GuestGo AI enhances operational transparency and security while improving visitor experience.

**BACKGROUND OF THE STUDY**

Manual guest tracking systems are not only outdated but also pose significant security risks. Traditional paper logs are susceptible to forgery, illegibility, and unauthorized access. Even digital forms, without proper validation, offer limited accountability and are unable to authenticate the actual identity of the person entering the premises (Sheludko, 2021).

The rise of facial recognition technology, particularly AI models like YOLO-Face, has allowed institutions to upgrade their systems for both convenience and security. GuestGo AI leverages this capability to capture guest images during registration, verify them against facial data through YOLO-Face, and issue a unique QR code to track the visit. These QR codes are scanned at entry, destination, and exit points, creating an end-to-end audit trail.

Studies highlight the effectiveness of facial recognition in eliminating impersonation and ensuring access control (Srinivasan et al., 2024). Furthermore, QR technology enables rapid, non-intrusive check-ins and integrates seamlessly with security workflows (Kishor et al., 2024). By fusing these technologies, GuestGo AI establishes a system that ensures accuracy, scalability, and compliance with evolving institutional needs.

**OBJECTIVES OF THE STUDY**

**General Objective**

To develop a real-time, AI-integrated visitor management system utilizing facial recognition and QR codes for secure and automated guest verification and tracking.

**Specifically the study aims to:**

1. Design a user-friendly guest registration interface that captures metadata and a clear facial image.
2. Implement a YOLO-Face model to perform accurate facial recognition and verification.
3. Generate encrypted, unique QR codes for each verified guest with embedded visit information.
4. Integrate QR scanners at strategic checkpoints to monitor guest progress and validate presence.
5. Log visitor data including entry, movement, and exit times, ensuring traceability.
6. Conduct comprehensive evaluation using ISO/IEC 25010 software quality standards covering functionality, usability, reliability, and security (ISO, 2019)

**SCOPE AND DELIMITATIONS OF THE STUDY**

This study has been developed to add a visitor management system for institutions that require secure and traceable entry and exit of guests. The proposed system uses QR codes to identify and AI for facial recognition. The system will be accessible on tablets or desktops that have modern browsers and cameras.

**Scope:** The system encompasses the complete visitor lifecycle from digital registration to exit validation, offering the following:

* Guest registration module with AI-based facial image verification
* QR code generation and scanning
* Real-time database logging of visitor actions
* Attribute-based access control (ABAC) to secure data access
* Web dashboard for administrative reporting

**Delimitations:**

* Facial verification is dependent on clear, front-facing images
* Internet connectivity is essential for system operation
* Access to logs is restricted to authorized personnel only
* The system does not incorporate other biometrics such as fingerprints or retina scans
* Lighting conditions and camera resolution may affect facial recognition accuracy

**SIGNIFICANCE OF THE STUDY**

The GuestGo AI system introduces digital transformation to institutional guest management by addressing long-standing problems of security, traceability, and operational inefficiency. By integrating QR codes and AI facial recognition, the system verifies each guest’s identity with high accuracy, minimizing fraud and human error.

The development of GuestGo AI offers the following significant benefits:

* **For Security Personnel:** The system provides live logs, automated alerts, and biometric verification to streamline gatekeeping duties, improve accuracy, and reduce manual burden (Abel & Joel, 2015). Enhanced visibility into visitor movements can help prevent unauthorized access and improve emergency response.
* **For Administrative Staff:** GuestGo AI minimizes paperwork and eliminates redundancies in manual entry systems. Real-time visitor logs and dashboard reports support efficient documentation and easy retrieval of visitor data (Janosepah et al., 2011), which can be essential for audits, compliance, and institutional reviews.
* **For Institutional Leadership:** The centralized database and analytics enable leaders to derive meaningful insights, such as peak visit times, repeat visitors, or resource allocation. It aligns with strategic goals on digital transformation and governance, as emphasized by Baker & Benny (2013). This is further supported by Perez and Francisco (2021), who noted that digital systems significantly improve organizational transparency and policy enforcement in Philippine higher education institutions.
* **For Visitors:** Seamless and contactless entry through QR and facial recognition enhances guest satisfaction by reducing wait times, improving privacy, and ensuring smoother movement within premises (Alkhodary et al., 2022).
* **For Developers and Researchers:** This system serves as a blueprint for integrating AI and cloud technologies in secure environments. It contributes to academic discussions on edge-AI, cloud-native systems, and secure digital identity management. Santos and Lim (2023) emphasized that smart access systems not only provide automation but also help institutions adapt to policy changes and scheduling dynamics.
* **For the Broader Community:** In times of public health crises or security threats, real-time tracking systems like GuestGo AI offer scalable solutions for contact tracing, evacuation tracking, and institutional safety. This aligns with Lee and Kim’s (2020) findings that stable automated systems improve operational satisfaction among staff and support rapid adaptation during emergencies.

By combining modern technologies with real-world needs, GuestGo AI stands at the intersection of security, innovation, and smart governance.

**Chapter 2**

**CONCEPTUAL FRAMEWORK**

This chapter provides an overview of related literature, related studies, the conceptual model of the study, and the operational definition of terms relevant to this study**.** This chapter expands on the concepts discussed in the paper by providing supporting materials that help the researchers better understand the study.

**Review of Related Literature and Related Studies**

Smart visitor management systems (VMS) aim to replace traditional logbooks with automated systems that track, verify, and manage guest information. Satari et al. (2014) introduced a Facial Recognition Visitor Management System (FRVMS) that used printed visitor cards and basic face recognition to enhance organizational security. However, it lacked real-time tracking and seamless verification through multi-layered authentication.

GuestGo AI improves upon these by combining **AI-powered facial recognition** and **QR code scanning** to ensure visitor accountability from entry to exit, reducing errors, impersonation, and untracked entries (Abel & Joel, 2015).

In a similar vein, Alkhodary et al. (2022) emphasized the importance of contactless check-in solutions during the pandemic, which accelerated the need for hygienic, streamlined guest monitoring. However, their system prioritized contact tracing and health protocols without strong identity verification methods.

Recent advances in visitor authentication and access control are predominantly driven by AI technologies and QR-based validation systems. Facial recognition systems have been employed across various industries due to their non-intrusive and contactless nature. According to Satari et al. (2014), these systems are not only efficient in preventing impersonation but also improve traceability and reduce the risk of unauthorized access.

YOLO-based systems, especially YOLO-Face, are widely adopted for their real-time detection capability. YOLO-Face improves detection accuracy in dynamic environments such as hallways or lobbies by focusing solely on facial features. This algorithm offers rapid processing speeds, crucial for check-in bottlenecks and time-sensitive environments (Zhao & Wu, 2020).

The application of QR codes in security systems has expanded significantly. QR technology enables encoded transmission of data in a secure and efficient format. It has been widely used in banking, healthcare, and ticketing systems due to its accessibility and low cost. As noted by Lin & Lee (2018), QR codes reduce queuing time and streamline check-in processes.

The concept of ABAC (Attribute-Based Access Control) used in GuestGo AI enhances user privilege differentiation based on pre-defined criteria such as role, time, and access level. Compared to RBAC (Role-Based Access Control), ABAC provides fine-grained control and dynamic decision-making capabilities (Ferraiolo et al., 2016).

Cloud platforms like Supabase have transformed database accessibility and security, offering real-time data syncing, encryption, and multi-layered access control (Wang & Sun, 2021). When paired with web frameworks such as Flask and Tailwind, the development process becomes faster and more maintainable.

The importance of software quality assurance is underscored by ISO 25010, which outlines critical quality metrics: Functionality, Usability, Reliability, Security, Maintainability, Portability, and Efficiency (ISO, 2019). These characteristics ensure that systems like GuestGo AI meet stakeholder expectations and regulatory requirements.

**Related Studies:**

* DreamClass (2022) emphasized how smart scheduling tools boost educational resource utilization.
* Yao et al. (2021) found that digital timetabling systems reduce human error by 40%, supporting the use of automated scheduling tools.
* Hernandez & Valerio (2019) validated the usability gains from drag-and-drop interfaces in administrative systems.
* Almazan et al. (2021) argued for real-time dashboards to support institutional decision-making in logistics.

**Facial Recognition in Security Systems**

Facial recognition is a biometric method used to identify individuals based on unique facial features. OpenCV, YOLO-Face, and Dlib are commonly used for building intelligent security systems (Srinivasan et al., 2024). These frameworks enable quick detection, even under dynamic lighting or angle variations.

While the hospitality sector uses facial recognition for smoother check-ins (Srinivasan et al., 2024), academic and high-security institutions have different requirements: traceability, audit trails, and movement monitoring. Janosepah et al. (2011) highlighted this in their work integrating physical-logical access controls in data centers, focusing on user tracking within facilities.

GuestGo AI addresses this gap by not only authenticating identity through AI but also logging visitor movements at checkpoints.

**QR Code Technology and Checkpoint Automation**

QR codes are widely adopted for encoding visitor credentials in compact, scannable formats. Their simplicity, affordability, and compatibility with mobile devices make them ideal for institutional guest management (Mohammed et al., 2020).

Kishor et al. (2024) presented a bi-directional visitor counter that monitored entries/exits but lacked identity verification, making it susceptible to impersonation. GuestGo AI augments this with AI verification, enhancing authenticity and security at each checkpoint.

QR codes in GuestGo AI contain metadata (e.g., purpose, entry time, destination), enhancing granularity in reporting.

**Access Control Systems in Institutional Settings**

Access control frameworks regulate who can access specific areas based on predefined criteria. ABAC (Attribute-Based Access Control) has emerged as a flexible model suitable for dynamic settings like schools or hospitals (Baker & Benny, 2013). Unlike RBAC (Role-Based Access Control), ABAC allows decisions based on user attributes, environment, and context.

In their book on physical security, Baker and Benny emphasized multi-layered control using biometrics, RFID cards, and behavior analytics. GuestGo AI modernizes this concept digitally by blending **facial recognition** with **QR checkpoints** and **ABAC**, ensuring that access is traceable and personalized.

**Real-Time Tracking and Movement Monitoring**

Real-time systems continuously update data to provide live monitoring. In visitor systems, this means knowing who is inside, where they are, and when they entered/exited (Alkhodary et al., 2022; Kishor et al., 2024). Real-time tracking is crucial for emergency management, attendance, and contact tracing.

Sheludko (2021) used Google Forms for lightweight guest logging. While it was practical and easy to deploy, it lacked identity verification, enabling spoofing and duplicate entries. GuestGo AI mitigates these issues using biometric matching and timestamp-based QR verification.

**Comparative Analysis of Related Systems**

Several systems parallel GuestGo AI but differ in their implementation and depth:

* **FRVMS**: Facial recognition but lacks movement tracking (Satari et al., 2014)
* **IoT Bi-directional Counters**: Count guests but don’t verify identity (Kishor et al., 2024)
* **COVID-era VMS**: Focused on contact tracing; weak identity authentication (Alkhodary et al., 2022)
* **Sheludko’s System**: Cost-effective, but prone to spoofing

GuestGo AI consolidates the strengths of these systems—face recognition, automation, live tracking—while addressing their vulnerabilities.

**Standards and Evaluation Models**

To evaluate the system, ISO/IEC 25010 is adopted, which outlines eight software quality attributes: functional suitability, reliability, usability, performance efficiency, maintainability, portability, security, and compatibility (ISO, 2019). These provide a holistic framework for assessing GuestGo AI’s success in real-world deployment.

Hernandez and Valerio (2019) showed that drag-and-drop UI improves usability scores, especially in systems requiring frequent scheduling or logging. Real-time databases (e.g., MySQL) enhance synchronization, a critical feature for tracking multiple users simultaneously (W3Techs, 2022).

**Conceptual Model of the Study**

The figure below presents the flow of how the system was developed.

**A diagram of a flowchart

AI-generated content may be incorrect.**

**Figure 1: GuestGo AI Development Architecture**

The figure below shows the Database Schema and Relationship of the system

**A diagram of a chat

AI-generated content may be incorrect.**

**Figure 2: GuestGo AI Database ERD**

**Operational Definition of Terms**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **Facial Recognition** | A biometric method for identifying individuals using facial feature vectors. |
| **YOLO-Face** | A real-time object detection model used to recognize and detect human faces. |
| **QR Code** | A two-dimensional barcode that contains encoded guest data for fast scanning. |
| **Access Control** | A security method that restricts or permits access based on user identity or attributes. |
| **Real-Time Tracking** | Continuous, live monitoring of guest movement inside the premises. |
| **Attribute-Based Access Control (ABAC)** | A flexible access control system based on multiple user and context attributes. |
| **Audit Trail** | A chronological record of user activities in the system for accountability. |
| **ISO 25010** | A software quality evaluation standard comprising eight characteristics. |
| **Database** | A structured set of data stored electronically for system reference and logging. |

**Table 1: System Terms and Definitions**

**Chapter 3**

**METHODOLOGY**

This chapter provides a comprehensive discussion of the project design, development procedures, system operations, testing processes, and evaluation procedures of the system. This section outlines the systematic process and tools used to develop GuestGo AI. This research aims to provide a comprehensive approach to leveraging room management processes effectively.

**Project Design**

The GuestGo AI system adheres to the **Input-Process-Output (IPO) model**, ensuring that each component of the system aligns with user interactions, data processing requirements, and output expectations. The design combines modern web and AI technologies to deliver a robust, secure, and scalable guest management experience.

* **Input Layer:**
  + Visitors submit personal data and images via the web registration interface.
  + Administrators and security personnel access system dashboards to view and manage guest logs.
* **Process Layer:**
  + **Facial Recognition (YOLO-Face):** The system uses real-time facial detection to verify identity with high accuracy and speed. The AI model runs via Flask API integration.
  + **QR Code Generation:** Once verified, a unique QR code is dynamically created containing visit metadata (name, time, purpose, destination).
  + **Checkpoint Logging:** Visitors scan their QR at various entry/exit checkpoints. The system logs their movement in real time via JavaScript and Supabase database calls.
  + **Attribute-Based Access Control (ABAC):** Visitor permissions are filtered based on attributes such as role, time, and purpose of visit.
* **Output Layer:**
  + The system generates live dashboards, logs, and alerts for administrators.
  + A complete audit trail is maintained per guest, retrievable by filters such as name, date, or location.
* **Technologies Used:**
  + **Frontend:** HTML, TailwindCSS, JavaScript
  + **Backend:** PHP, Flask API (YOLO-Face)
  + **Database:** Supabase (PostgreSQL)
  + **Security Layer:** ABAC filtering, encrypted QR data, restricted log access

This layered approach ensures modularity, maintainability, and performance efficiency, conforming to ISO/IEC 25010 guidelines.

The figure below shows the entire system process of GuestGo AI.

**A diagram of a flowchart

AI-generated content may be incorrect.**

**Figure 3: GuestGo AI System Flowchart**

**Project Development**

The development of GuestGo AI followed a modular and iterative approach to ensure clear functionality and separation of concerns across system components. Each module was developed, tested, and refined to support a seamless visitor management experience—starting from registration to real-time tracking and administrative reporting. This modular structure improves maintainability, scalability, and testing reliability. Below is a breakdown of the major system modules, including their primary functions, tools used, and expected outputs:

|  |  |  |  |
| --- | --- | --- | --- |
| **Module** | **Function** | **Tools Used** | **Expected Output** |
| **Registration** | Collects guest data + photo | HTML/JS | JSON object |
| **Facial Verification** | Matches image with AI model | YOLO-Face, Flask | True/False match |
| **QR Code Generator** | Creates secure QR with metadata | PHP, qrcode.js | QR PNG file |
| **Checkpoint Scanner** | Reads QR & logs movement | JS + Camera API | DB log entry |
| **Admin Dashboard** | Visualizes logs | Tailwind, Supabase UI | Summary tables |

**Table 2: GuestGo AI Modules**

**Operation and Testing Procedure**

To ensure reliability, each module of the GuestGo AI system was subjected to functional testing and simulated real-world conditions. This testing phase focused on validating system outputs against expected behaviors across various scenarios—such as normal use, boundary conditions, and system stress (e.g., multiple users and intermittent connectivity). The goal was to detect bugs, improve system response, and verify successful integration between facial recognition, QR logging, and database updates. The table below outlines the test cases conducted per module and their expected outcomes:

|  |  |  |
| --- | --- | --- |
| **Module** | **Test Description** | **Expected Outcome** |
| Registration | Enter guest info and verify QR generation | QR successfully generated and stored |
| Face Verification | Match current image with database | Correct match results with high accuracy |
| Checkpoints | Scan QR and log visit at each location | Accurate timestamped logs saved |
| Admin Controls | View logs, filter by user/date | Proper data retrieval and display |

**Table 3: Modules Testing Plan**

**Evaluation Procedure**

The system’s evaluation will be conducted based on ISO/IEC 25010 standards. Feedback will be gathered through surveys and observational use by respondents from various roles.

Certain users will be selected (admins, guards, devs) to evaluate:

* Functionality: Feature completeness
* Usability: Ease of interface
* Reliability: Stability during usage
* Security: Resistance to bypass attempts

Each criterion was rated on a 5-point Likert scale based on user feedback and observed performance. The table below summarizes the definitions and evaluation methods for each quality attribute used in assessing the system:

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Definition** | **Evaluation Activity** | **Likert Scale** |
| **Functionality** | Measures how well the system performs its intended functions. | Test registration, facial matching, QR logging, and admin access. | 1 (Not acceptable) to 5 (Highly acceptable) |
| **Reliability** | Evaluates system stability under normal and extreme conditions. | Perform concurrent guest check-ins, simulate internet drops, test data integrity. | 1 to 5 |
| **Usability** | Measures ease of use, learnability, and user satisfaction. | Observe user navigation, conduct usability survey, measure task completion time. | 1 to 5 |
| **Efficiency** | Assesses performance in terms of speed and resource usage. | Measure QR scan time, processing lag, and database update speed. | 1 to 5 |
| **Maintainability** | Evaluates how easily the system can be updated or modified. | Review modular codebase, test for error localization, simulate bug patch deployment. | 1 to 5 |
| **Portability** | Measures compatibility across devices and platforms. | Test system on Chrome, Firefox, Edge, and across devices (tablet, PC). | 1 to 5 |
| **Criteria** | **Definition** | **Evaluation Activity** | **Likert Scale** |

**Table 4: Evaluation Metrics and Criteria**

**References:**

**Chapter 1**

Abel, E., & Joel, D. (2015). Bi-directional people counting in hotel security. *CAE Access*. <https://www.caeaccess.org/research/volume2/number9/abel-2015-cae-651858.pdf>

Alkhodary, K., & Badr, M. (2022). VMS for pandemic-era security: A case study. *Natural Publishing*. <https://www.naturalspublishing.com/files/published/6p3jg121583s5s.pdf>

Baker, W., & Benny, H. (2013). *The complete guide to physical security*. Taylor & Francis. <https://www.taylorfrancis.com/chapters/edit/10.1201/b13713-11>

Cruz, J. D., & David, R. A. (2022). Centralized scheduling systems and academic efficiency. *Journal of Educational Administration, 38*(2), 122–137.

ISO. (2019). *ISO/IEC 25010:2011 – Software product quality model*. International Organization for Standardization. <https://www.iso.org/standard/35733.html>

Janosepah, M., et al. (2011). Integrating physical and logical access control: An N-tier architecture. *IEEE Xplore*. <https://ieeexplore.ieee.org/abstract/document/6014545>

Kishor, R., & Varsha, M. (2024). Real-time visitor counter using IoT for facility optimization. *ACM Digital Library*. <https://dl.acm.org/doi/abs/10.1145/3647444.3647925>

Satari, S., & Mirshahi, A. (2014). Facial recognition-based visitor management systems. *IEEE Xplore*. <https://ieeexplore.ieee.org/abstract/document/7013101>

Sheludko, M. (2021). Digital attendance tools for school safety. *Proceedings of Ruse Conference*. <https://conf.uni-ruse.bg/bg/docs/cp21/5.1/5.1-3.pdf>

Srinivasan, R., et al. (2024). OpenCV and IoT-enabled facial recognition for secure hospitality. *IEEE Xplore*. <https://ieeexplore.ieee.org/abstract/document/10522363>

**Chapter 2**

Almazan, R. M., et al. (2021). Real-time dashboards for academic operations. *Philippine Journal of Digital Education, 4*(2), 33–40.

Cruz, J. D., & David, R. A. (2022). Centralized scheduling systems and academic efficiency. *Journal of Educational Administration, 38*(2), 122–137.

Ferraiolo, D., Kuhn, D., & Hu, V. (2016). *Attribute-based access control*. NIST Computer Security Resource Center.

Hernandez, P. F., & Valerio, L. T. (2019). Improving UI usability through drag-and-drop functionality. *User Interface Journal, 14*(2), 66–74.

ISO. (2019). *ISO/IEC 25010:2011 Systems and software engineering—System and software quality models*. International Organization for Standardization.

Johnson, M. T., Lee, H., & Santos, A. D. (2019). Student academic performance and classroom schedule consistency. *Education Research Quarterly, 43*(1), 45–58.

Lee, S. H., & Kim, J. Y. (2020). Teaching performance and job satisfaction among instructors with stable schedules. *Journal of Higher Education Studies, 12*(3), 76–89.

Lim, E. R., & Reyes, C. A. (2022). Online modality transitions and instructional quality in higher education. *Philippine Journal of Digital Education, 4*(1), 24–38.

Lin, H., & Lee, M. (2018). QR codes for security: A review. *Journal of Information Security, 9*(4), 67–77.

Perez, M. A., & Francisco, L. D. (2021). Digital transformation in Philippine higher education institutions. *Journal of Educational Technology and Society, 17*(2), 55–70.

Santos, M. C. (2021). Effects of classroom mismanagement on faculty productivity. *Philippine Journal of Academic Studies, 11*(1), 13–21.

Santos, M. C., & Lim, E. R. (2023). Smart scheduling innovations in Philippine tertiary institutions. *Innovations in Education and Teaching International, 60*(1), 87–99.

Sheludko, M. (2021). Google Forms-based attendance systems: Benefits and drawbacks. *Ruse Conference Proceedings*.

Torres, P. V., & Rivera, J. L. (2020). The use of smart scheduling tools in education management systems. *Journal of Educational Innovations, 15*(4), 32–41.

Wang, H., & Sun, T. (2021). Supabase and the future of realtime web applications. *Open Source Review, 6*(2), 113–121.

Yao, H., Zhang, X., & Liu, Y. (2021). Smart timetabling algorithms for educational institutions. *International Journal of Educational Technology, 10*(1), 14–22.

Zhao, Y., & Wu, L. (2020). Fast and accurate face detection using YOLO-Face. *Journal of Vision Systems, 14*(3), 58–68.

**Chapter 3:**

Computer Economics. (2008). Desktop vs. notebook shipments: Market transition report. <https://www.computereconomics.com>

Hernandez, P. F., & Valerio, L. T. (2019). Improving UI usability through drag-and-drop functionality. *User Interface Journal, 14*(2), 66–74.

ISO. (2019). *ISO/IEC 25010:2011 Systems and software engineering—System and software quality requirements and evaluation (SQuaRE)*. International Organization for Standardization. <https://www.iso.org/standard/35733.html>

W3Techs. (2022). Usage statistics and market share of server-side programming languages for websites. <https://w3techs.com/technologies/overview/programming_language>

**Appendix:**

**Appendix A: Gantt Chart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tasks** | **Wk1** | **Wk2** | **Wk3** | **Wk4** | **Wk5** | **Wk6** | **Wk7** | **Wk8** | **Wk9** | **Wk10** | **Wk11** | **Wk12** | **Wk13** | **Wk14** |
| **1. Proposal and Planning** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2. Literature Review & RRL** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3. System Design (UI/UX, DB, Architecture)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4. Module Development** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| - Registration & Face Verification |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| - QR Code Generator & Scanner |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| - Admin Dashboard |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5. System Integration** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6. Testing & Debugging** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **7. Evaluation (ISO 25010)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **8. Documentation & Revisions** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **9. Final Defense Preparation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **10. Submission of Final Manuscript** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



**Glenn R. Galbadores I**

Glenn R. Galbadores I is a passionate Computer Science student at the Technological University of the Philippines – Manila. He began his education at Paco Catholic School, graduating with consistent First Honors and receiving the Salutatorian award in Grade 1. He continued his secondary education at Centro Escolar University – Manila, where he deepened his logical thinking and creative skills.

Glenn is an aspiring author and game developer whose passion for innovation began at a young age. He has written and published three original fictional stories online and has built commissioned systems for Roblox games, including minigames, RPGs, adventure maps, and tower defense mechanics.

Academically and athletically accomplished, Glenn was a consistent honor student and a varsity athlete. He played badminton competitively in Grade 3 and became a volleyball varsity player from Grade 4 to Grade 10. He earned multiple “Mythical Six” selections and contributed to several championship wins, with Palarong Maynila as his most notable achievement.

In group projects, Glenn often takes the lead, bringing a mix of responsibility, vision, and technical skill. He is committed to becoming a full-stack developer and an author, using both code and narrative to bring ideas to life.

A person in a white shirt

AI-generated content may be incorrect.

**Kurt Angelo F. Ballarta**

Kurt Angelo Ballarta is a Computer Science student from the Technological University of the Philippines – Manila. He completed his secondary education at Raja Soliman Science and Technology High School, where he developed a strong foundation in analytical thinking, logical reasoning, and scientific problem-solving—skills that continue to shape his academic and technical pursuits. Currently in his third year, Kurt is deeply interested in data science and data engineering. His academic interests include data analytics, machine learning, data infrastructure, and pipeline automation. He is particularly drawn to how raw data can be transformed into actionable insights through engineering workflows and intelligent algorithms.

Kurt aspires to pursue a career in data science and engineering, with a focus on building scalable data pipelines, developing intelligent systems, and applying data-driven solutions to real-world problems. He aims to work in environments where technology and data intersect to drive innovation, especially in fields like education, urban planning, or healthcare.

A person in a suit

AI-generated content may be incorrect.

**Justine B. Mantilla**

Justine B. Mantilla is 21 years old and a 3rd year college student studying at Technological University of the Philippines Manila currently taking the course of Bachelor Science of Computer Science.

Mr. Mantilla was born in Trece Martires, Cavite, on 2004 of May 20th and he’s currently living at Alta Tierra Homes, General Mariano Alvarez, Cavite as the youngest child with his family.

During his primary education he attended San Gabriel II Elementary School, soon after graduating he took both his junior high and senior high school education in General Mariano Alvarez Technical High School (GMATHS) taking the Science Technology Engineering Mathematics (STEM) academic strand during his senior high.

 In terms of achievements, he graduated as the top of his class in elementary and was a consistent honor student all throughout his high school career. Before graduating senior high, he led his group for their thesis project titled “Alugbati Fruit Extract as an Alternative PH Indicator”.

A person in a suit and tie

AI-generated content may be incorrect.

**John Walter D. Marquez**

John Walter D. Marquez is 20 years old and a 3rdd year college student studying at Technological University of the Philippines Manila currently taking the course of Bachelor Science of Computer Science. Mr. Marquez was born in Agoo, La Union, on 2003 of 9th of December and he's currently living at First Marcel Tower, Brgy. Talayan, Gregorio Araneta Ave. Quezon City as the youngest child with his older brother.

He Studied among these schools: King Solomon Christian School International (KSCSI), Don Alejandro Roces Sr. Science Technology High School (DARSSTHS), and Immaculate Conception Cathedral School (ICCS). In terms of achievements, he received some academic awards in elementary school and has passed an entrance exam prior to his former schools that most students wouldn't and considered difficult. Experienced handling research papers as an assistant leader whilst defending the thesis titled Monoamory vs. Polyamory in Junior High School at DARSSTHS.

**A person wearing glasses and a white shirt

AI-generated content may be incorrect.**

**Ken Zedrick Montano**

Ken Zedrick E. Montano is currently a 3rd year college student at Technological of the Philippines - Manila taking up Bachelor of Science in Computer Science.

Mr. Montano is 22 years old and was born in Manila, on February 28, 2003. He's currently living at Cavite, Imus, Gahak as the 2nd child among the three siblings.

In terms of educational background, Mr. Montano has received some academic awards since High School. He passed an Elementary School at General Vicente Lim Elementary School (GVLES) and High School at Antonio J. Villegas Vocational High School (AJVVHS) journey at Tondo, Manila.

He also studied Senior High School at Philippine College of Health Sciences, Inc. (PCHS) where he graduated with some academic awards.