



Faculty of Computer Science and Information Technology

***MULTI-MODAL AUTHENTICATION AND GEOLOCATION APPROACH
FOR ATTENDANCE VERIFICATION***

Dayang Nur Alisa Binti Abang Senawi

Bachelor of Software Engineering with Honours

2025

**MULTI-MODAL AUTHENTICATION AND GEOLOCATION APPROACH
FOR ATTENDANCE VERIFICATION**

DAYANG NUR ALISA BINTI ABANG SENAWI

This project is submitted in partial fulfilment of
the requirements for the degree of Bachelor of Software Engineering with Honours

Faculty of Computer Science and Information Technology
UNIVERSITI MALAYSIA SARAWAK

2025

**PENGESAHAN PELBAGAI MODAL DAN PENGGUNAAN GEOLOKASI
UNTUK PENGESAHAN KEHADIRAN**

DAYANG NUR ALISA BINTI ABANG SENAWI

Projek ini merupakan salah satu keperluan untuk
Ijazah Sarjana Muda
Kejuruteraan Perisian

Fakulti Sains Komputer dan Teknologi Maklumat
UNIVERSITI MALAYSIA SARAWAK

2025

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE MULTI-MODAL AUTHENTICATION AND GEOLOCATION APPROACH
FOR ATTENDANCE VERIFICATION

ACADEMIC SESSION: 2024/2025

DAYANG NUR ALISA BINTI ABANG SENAWI

(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (✓)

☐

CONFIDENTIAL

(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)

☐

RESTRICTED

(Contains restricted information as dictated by the body or organization where the research was conducted)

☒

UNRESTRICTED



(AUTHOR'S SIGNATURE)

Validated by


Assoc. Prof. Dr. Jihari Abdullah
Faculty of Computer Science & Information Technology
Universiti Malaysia Sarawak
94300 Kuching, Sarawak

(SUPERVISOR'S SIGNATURE)

Permanent Address

No. 739, Lot 883, Lorong Juara

3B, Taman Sukma, 93050,

Kuching, Sarawak

Date: 23/6/2025

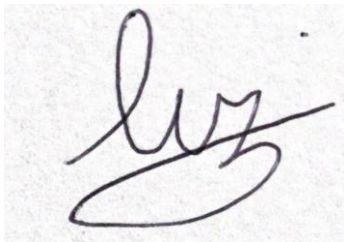
Date: 23/6/2025

Note * Thesis refers to PhD, Master, and Bachelor Degree

** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

DECLARATION

I, Dayang Nur Alisa Binti Abang Senawi with the matric number of 77919, hereby solemnly declare that the report titled '**Multi-Modal Authentication and Geolocation Approach for Attendance Verification**' is prepared and completed authentically by me in partial fulfilment of the requirements for the degree of Bachelor of Computer Science with Honours (Software Engineering) is a record of an original work carried out by me under the supervision of Associate Professor Dr. Johari bin Abdullah. I hereby affirm that the work presented in this project has not been previously or concurrently submitted for any other degree at UNIVERSITI MALAYSIA SARAWAK (UNIMAS) or any other institutions. I declare that I have not copied or plagiarize from any other sources except for quotations and citations which have been duly acknowledged.



DAYANG NUR ALISA BINTI ABANG SENAWI (77919)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

ACKNOWLEDGEMENT

All praise and gratitude belong to Allah, the Most Gracious and the Most Merciful, for providing me with the strength, patience, and wisdom to successfully complete this project within the designated timeframe. This achievement would not have been attainable without His guidance and blessings.

I want to express my sincere appreciation to all who played a role in the successful completion of this project. Their essential support, guidance, and motivation were pivotal in achieving success in this endeavour.

Firstly, I would like to extend my heartfelt thanks to my project supervisor, Associate Professor Dr. Johari bin Abdullah, for his steadfast guidance, valuable feedback, and ongoing support during this journey. His commitment and knowledge played a crucial role in guaranteeing the quality and successful completion of this project.

Furthermore, I am deeply thankful to my examiner, Ts. Abdul Rahman bin Mat, for his insightful comments and valuable suggestions that have refined my work and raised its standards. I would also like to express my heartfelt gratitude to Professor Dr. Wang Yin Chai, the project coordinator, for offering vital resources, clear instructions, and prompt guidance, all of which were instrumental in successfully managing the project within the set timeline.

On top of that, I would like to express my gratitude to Universiti Malaysia Sarawak (UNIMAS) and the Faculty of Computer Science and Information Technology (FCSIT) for providing me with the platform and resources necessary to carry out this project. Working on this project has provided an exceptional learning experience.

Additionally, I am sincerely grateful to all the respondents and participants who dedicated their time to share their insights and feedback, which greatly enhanced the research

and results of this project. A warm appreciation goes to my friends and course mates for their unwavering support, encouragement, and help during this challenging but fulfilling journey. Their presence enhanced the experience, making it more enjoyable and easier to navigate.

Finally, I want to express my heartfelt thanks to my cherished parents and family for their unwavering support, prayers, and love. Their support and faith in me have consistently served as my most significant source of inspiration. I appreciate everyone's contributions and support in bringing this project to fruition. May Allah reward each of you abundantly for your kindness and support.

ABSTRACT

Effective attendance management is essential in current office settings to improve accountability, productivity, and operational oversight. Conventional attendance systems frequently fail to effectively tackle problems like fraud, inefficiencies, and the absence of real-time data. This project aims to develop a Multi-Modal Authentication and Geolocation Approach for Attendance Verification, such as fingerprint recognition and, QR code scanning, along with geolocation tracking integration.

The system has been designed to confirm employee presence in specified work areas and to offer real-time tracking functionalities. The project utilises advanced tools such as React Native and Firebase to guarantee an efficient user experience and effective data management. Attendance reports produced by the system will assist HR personnel and managers in recognising trends and facilitating data-driven decision-making.

This mobile application seeks to establish a new benchmark in attendance verification, overcoming the shortcomings of conventional systems and fulfilling the requirements of modern workplaces. The project aims to improve workforce management efficiency and reliability, contributing to enhanced organisational productivity and security.

ABSTRAK

Pengurusan kehadiran yang berkesan adalah penting dalam persekitaran pejabat semasa untuk meningkatkan akauntabiliti, produktiviti, dan penyeliaan operasi. Sistem kehadiran konvensional sering kali gagal menangani masalah seperti penipuan, ketidakcekapan, dan ketiadaan data masa nyata dengan berkesan. Projek ini bertujuan untuk membangunkan pengesahan pelbagai modal dan pendekatan geolokasi untuk pengesahan kehadiran yang menggunakan kaedah pengesahan pelbagai mod, seperti pengecaman cap jari, dan pengimbasan kod QR, berserta integrasi penjejakan geolokasi.

Sistem ini direka untuk mengesahkan kehadiran pekerja di kawasan kerja yang ditentukan dan menawarkan fungsi penjejakan masa nyata. Projek ini menggunakan alat canggih seperti 'React Native' dan 'Firebase' untuk memastikan pengalaman pengguna yang cekap dan pengurusan data yang berkesan. Laporan kehadiran yang dihasilkan oleh sistem ini akan membantu kakitangan sumber manusia dan pengurus dalam mengenal pasti trend dan memudahkan pengambilan keputusan berdasarkan data.

Aplikasi mudah alih ini berusaha untuk menetapkan penanda aras baharu dalam pengesahan kehadiran, mengatasi kekurangan sistem konvensional dan memenuhi keperluan tempat kerja moden. Projek ini bertujuan untuk meningkatkan kecekapan dan kebolehpercayaan pengurusan tenaga kerja, menyumbang kepada peningkatan produktiviti dan keselamatan organisasi.

Table of Contents

DECLARATION.....	i
ACKNOWLEDGEMENT	ii
ABSTRACT.....	iv
ABSTRAK.....	v
LIST OF TABLES	ix
LIST OF FIGURES	x
CHAPTER 1: INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	4
1.3 Objectives.....	7
1.4 Brief Methodology	8
1.5 Scopes.....	10
1.6 Significance of Project	10
1.7 Project Schedule.....	12
1.8 Expected Outcome	13
1.9 Project Outline.....	14
1.10 Summary	16
CHAPTER 2: LITERATURE REVIEW	17
2.1 Introduction	17
2.2 Review on Similar Existing Systems	17
2.2.1 Buddy Punch.....	18
2.2.2 Jibble App.....	29
2.2.3 Deputy App.....	43
2.3 Features Comparison of the Reviewed Systems	55
2.4 Review on Tools and Technology	56
2.4.1 Software.....	56
2.4.2 Hardware	70
2.5 Summary	74
CHAPTER 3: REQUIREMENT ANALYSIS AND DESIGN	76
3.1 Introduction	76
3.2 Methodology	76
3.3 Requirements Planning	78
3.3.1 Analysis on Similar Existing Systems	79

3.3.2 Analysis on Proposed Application	80
3.3.3 Questionnaire.....	81
3.3.4 Software Requirements.....	95
3.3.5 Hardware Requirements	96
3.4 User Design	98
3.4.1 Overall System Architectural Design	99
3.4.2 Flowchart	102
3.4.3 Context Diagram.....	106
3.4.4 Data Flow Diagram (DFD).....	108
3.4.5 Entity Relationship Diagram (ERD).....	116
3.4.6 Data Dictionary.....	119
3.4.7 Wireframes.....	124
3.5 Summary	136
CHAPTER 4: IMPLEMENTATION	138
4.1 Introduction	138
4.2 Installation and Configuration of Multi-Modal Authentication and Geolocation Approach for Attendance Verification	138
4.2.1 Visual Studio Code	138
4.2.2 Expo.....	145
4.2.3 Android Studio.....	149
4.2.4 Integration Overview.....	155
4.3 Using the Multi-Modal Authentication and Geolocation Approach for Attendance Verification	158
4.3.1 Welcome Page (Join or Login)	158
4.3.2 Registration Page.....	159
4.3.3 Login Page.....	160
4.3.4 Login with Credentials	161
4.3.5 Clock-In/Clock-Out Process.....	162
4.3.6 Geolocation Feature Overview (After Successful Clock-In)	170
4.3.7 Dashboard and Reporting	173
4.3.8 Notification Feature for Presence Verification	180
4.3.9 Setting Page	183
4.3.10 User Profile Page	184
4.3.11 Config Page	186
4.3.12 Support Page.....	187

4.4 Summary	187
CHAPTER 5: TESTING.....	189
5.1 Introduction	189
5.2 Functional Testing	189
5.2.1 Component Testing.....	190
5.2.2 Integration Testing	200
5.3 Non-Functional Testing.....	203
5.3.1 Usability Testing.....	203
5.3.2 User Acceptance Testing (UAT)	206
5.4 Discussion on Results	209
5.5 Summary	210
CHAPTER 6: CONCLUSION AND FUTURE WORKS	212
6.1 Introduction	212
6.2 Objectives and Achievements	212
6.3 Project Limitations and Constraints	213
6.4 Future Work.....	215
6.5 Summary	217
REFERENCES	218
APPENDICES	225
Appendix A	225
Appendix B	228
Appendix C	230

LIST OF TABLES

Table 2.1: Features and Functionalities of Buddy Punch.....	19
Table 2.2: Benefits of Buddy Punch.	27
Table 2.3: The Limitations of Buddy Punch.	28
Table 2.4: Features and Functionalities of Jibble.....	30
Table 2.5: The Benefits of Jibble.	42
Table 2.6: The Limitations of Jibble.	43
Table 2.7: Features and Functionalities of Deputy.....	45
Table 2.8: Benefits of Deputy.	53
Table 2.9: The limitations of Deputy.	54
Table 2.10: Comparison of Features between Existing Systems and Proposed System.....	55
Table 2.11: Reasons and Justifications on Choosing React Native	59
Table 2.12: Reasons and Justifications on Choosing Android Studio	61
Table 2.13: Reasons and Justifications on Visual Studio Code.	63
Table 2.14: Reasons and Justifications on Choosing Expo.....	65
Table 2.15 : Reasons and Justifications on Choosing Supabase.....	67
Table 2.16: Reasons and Justifications on Choosing Firebase Cloud Messaging (FCM).	68
Table 2.17: Reasons and Justifications on Choosing PostgreSQL.	70
Table 2.18: Reasons and Justifications for Choosing HP Laptop 15-fc0xxx.....	71
Table 2.19: Reasons and Justifications for Choosing Huawei P30 Pro.	74
Table 3.1: Respondents' Concerns with Concluded Concerns.	93
Table 3.2: Respondents' Suggestions with Concluded Suggestions.	94
Table 3.3: Software Requirements for the proposed system.....	95
Table 3.4: Hardware Requirements for the Back-end System.	96
Table 3.5: Hardware Requirements for the Proposed Android Application.	97
Table 3.6: Employee Table of the proposed system.....	119
Table 3.7: AuthenticationLog Table of the proposed system.	120
Table 3.8: Geolocation Table of the proposed system.	121
Table 3.9: Attendance Table of the proposed system.	122
Table 3.10: Attendance Report Table of the proposed system.	123
Table 5.1: Component Testing Result for Registration Module	191
Table 5.2: Component Testing Result for Login Module	193
Table 5.3: Component Testing Result for Attendance Module (Clock-In / Clock-Out)	194
Table 5.4: Component Testing Result for Notification Module.....	196
Table 5.5: Component Testing Result for Geolocation View Module	197
Table 5.6: Component Testing Result for Dashboard & Report Module.....	198
Table 5.7: Integration Testing Result for CorePresence Application.....	201
Table 5.8: Usability Testing Result for Core Function Modules	204
Table 6.1: Details of Objectives and Achievements.	212

LIST OF FIGURES

Figure 1.1: Manual Punch Cards (GoGet, 2023).	1
Figure 1.2: Rapid Application Development Methodology (App Builder, 2024).	8
Figure 1.3: Gantt Chart Task Table on Project Timeline FYP 1.	13
Figure 1.4: Gantt Chart Task Table on Project Timeline FYP 2.	13
Figure 2.1: Buddy Punch app in both IOS and Android.	19
Figure 2.2: Buddy Punch authentication screen.	21
Figure 2.3: Authentication methods and devices employees will use to clock in.	22
Figure 2.4: Illustrate the login methods available in Buddy Punch: using a username and password, a 4-digit PIN, and QR codes, respectively.	22
Figure 2.5: Employee's Location Map View.	23
Figure 2.6: GPS Time Tracking App for Employee Timesheets.	24
Figure 2.7: Live location updates.	25
Figure 2.8: Importing Time Data for Report Conversion.	26
Figure 2.9: Downloading Reports in Excel.	26
Figure 2.10: Daily Hours Report for Employee Time Tracking.	27
Figure 2.11: Jibble app in both IOS and Android.	30
Figure 2.12: Jibble Web App User Interface (Kiarie, 2024).	32
Figure 2.13: Jibble Mobile App User Interface (Maina, 2024).	33
Figure 2.14: Timesheets of the employees.	34
Figure 2.15: Showing the clock-in, clock-out, and time summary features of Jibble, providing a comprehensive view of employee attendance and work hours.	34
Figure 2.16: Device Restrictions.	35
Figure 2.17: GPS Tracking Features.	36
Figure 2.18: GPS Tracking Live Location Features.	37
Figure 2.19: Geofence Features (Maina, 2024).	38
Figure 2.20: Jibble lets user generate timesheets, time tracked, and availability insight reports.	39
Figure 2.21: Exporting Tracked Time Report.	40
Figure 2.22: Reminder and Alerts custom notifications.	41
Figure 2.23: Deputy app in both IOS and Android.	44
Figure 2.24: Deputy User Interface Dashboard.	46
Figure 2.25: Shift Start, Ongoing Shift, and Weekly Shift Summary Interfaces.	47
Figure 2.26: Clock In/Out using Website.	48
Figure 2.27: Enabling Geofence in location.	49
Figure 2.28: Deputy's scheduling interface showcasing flexible shift creation, time-off management, and compliance tracking (Wahal, 2024).	49
Figure 2.29: Employees can easily mark unavailable times and request leave.	50
Figure 2.30: Analyse Employee Efficiency with Schedule, Timesheets, and Sales Insights.	51
Figure 2.31: Deputy's Report Dashboard.	52
Figure 2.32: React Native Architecture Diagram (EDUCBA ,2020).	57
Figure 2.33: React Native Logo (PS Solutions, 2025).	58
Figure 2.34: Android NDK Version (Mutinda, 2023).	60

Figure 2.35: Visual Studio Code interface showcasing AI-powered coding assistance (Code, 2025).	62
Figure 2.36: Expo CLI and connected mobile app preview for live development and debugging. (Expo, 2025).	64
Figure 2.37: Supabase interface showing real-time database tables for application data management (John, 2024).	66
Figure 2.38: Overview of Firebase Cloud Messaging service (Firebase, 2025).	68
Figure 2.39: PostgreSQL Architecture.	69
Figure 2.40: An overview of the laptop specifications	72
Figure 2.41: Huawei P30 Pro device specifications screen used (Rahman, 2019).	73
Figure 3.1: 4 phases of Rapid Application Development (Lomio & Mejidana, 2023).	77
Figure 3.2: Age and Gender of Respondents.	82
Figure 3.3: Respondent's prior experience with Attendance Systems.	83
Figure 3.4: Different kinds of attendance systems used in respondents' workplaces.	84
Figure 3.5: Opinions of respondents toward traditional attendance methods.	85
Figure 3.6: Prevalence of buddy punching in workplaces.	86
Figure 3.7: Prevalence of False Attendance Marking in Offices.	86
Figure 3.8: Frequency of Issues with Current Attendance Systems.	87
Figure 3.9: Various levels of concurrence on whether the existing attendance system accurately tracks attendance and presence.	87
Figure 3.10: Perception of Advance Attendance Features for the workplace.	88
Figure 3.11: The Role of User-Friendliness in the proposed Attendance System.	89
Figure 3.12: Key Features Prioritized in an Attendance System.	90
Figure 3.13: Frequency of Mobile Device Use for Attendance System Interaction.	91
Figure 3.14: Convenience of Attendance Tracking via Mobile App.	91
Figure 3.15: Confidence Levels in the Reliability and Security of Advanced Attendance Technologies.	92
Figure 3.16: Overall System Architectural Design.	100
Figure 3.17: Flowchart of the proposed system	103
Figure 3.18: Context Diagram.	107
Figure 3.19: Data Flow Diagram (DFD) Level 0 of the Multi-Modal Authentication and Geolocation Approach for Attendance Verification System.	109
Figure 3.20: Data Flow Diagram Level 1 for Process 1.0.	112
Figure 3.21: Data Flow Diagram Level 1 for Process 2.0.	113
Figure 3.22: Data Flow Diagram Level 1 for Process 3.0	114
Figure 3.23: Data Flow Diagram Level 1 for Process 4.0	115
Figure 3.24: Data Flow Diagram Level 1 for Process 5.0	116
Figure 3.25: Entity Relationship Diagram (ERD).	117
Figure 3.26: Register Page.	125
Figure 3.27: Sign in Page.	126
Figure 3.28: Setting Page.	127
Figure 3.29: Employee Clock In/Out Page.	127
Figure 3.30: Clock in Method Selection Page.	128
Figure 3.31: Fingerprint Method Page.	129
Figure 3.32: QR Code Clock in Page.	130
Figure 3.33: Account Settings Page.	131

Figure 3.34: Geolocation Tracking Page.	132
Figure 3.35: Report Page.	133
Figure 3.36: Employee Dashboard Page.....	134
Figure 3.37: Manager Dashboard Page.....	135
Figure 3.38: Admin Dashboard Page.	136
Figure 4.1: Visual Studio Code Official Website.	139
Figure 4.2: Visual Studio Code Setup.	140
Figure 4.3 Visual Studio Code Installer displays the default installation directory, which can be customized as needed before proceeding to the next step.	141
Figure 4.4: Additional setup options in Visual Studio Code.....	142
Figure 4.5: Installation Ready.	142
Figure 4.6: Visual Studio Code Setup Wizard.	143
Figure 4.7: Visual Studio Code Interfaces.	144
Figure 4.8: Visual Studio Code terminal confirming a successful setup.	145
Figure 4.9: Official Node.js website showing the LTS version download option.	146
Figure 4.10: Command Prompt showing successful verification of Node.js and npm installation.....	146
Figure 4.11: Installing Expo CLI globally via npm using the command prompt.	147
Figure 4.12: Initializing a new Expo project named spot using the expo init command.....	147
Figure 4.13: Navigating into the newly created project directory with cd spot.....	148
Figure 4.14: Running npm start to launch the Expo development server.	148
Figure 4.15: Expo start command window.	148
Figure 4.16: QR code from Expo CLI displayed in the terminal.....	149
Figure 4.17: Welcome to Android Studio Setup.	150
Figure 4.18: Choose Components.....	151
Figure 4.19: Configuration Settings.....	151
Figure 4.20: Install Type (Standard).	152
Figure 4.21: Verify Settings.	153
Figure 4.22: SDK's License Agreement.....	153
Figure 4.23: Downloading Components.	154
Figure 4.24: Languages & Frameworks Android SDK.	154
Figure 4.25: AVD Manager showing configured virtual devices (APIs 35 & 30).....	155
Figure 4.26: Terminal command to install React Native using Expo CLI.....	156
Figure 4.27: Terminal command showing installation of Supabase JavaScript client via npm.	156
Figure 4.28: Supabase client setup example in JavaScript.	157
Figure 4.29: Terminal command showing installation of Firebase Cloud Messaging via Expo CLI.	157
Figure 4.30: Welcome Page.	158
Figure 4.31: Registration Page.....	159
Figure 4.32: Login Page.....	160
Figure 4.33 Login with Credentials Page.	161
Figure 4.34: Main Clock Page.	162
Figure 4.35: Authentication Method Selection Page.	163
Figure 4.36: Fingerprint Authentication Prompt.....	164
Figure 4.37: Fingerprint Clock-In Success Notification.....	165

Figure 4.38: Fingerprint Clock-Out Success Notification.	166
Figure 4.39: Fingerprint Clock-Out Success Notification.	167
Figure 4.40: Fingerprint Clock-Out Success Notification.	168
Figure 4.41: Fingerprint Clock-Out Success Notification.	169
Figure 4.42: Manager Overview.	170
Figure 4.43: Admin View.	171
Figure 4.44: Employee View.....	172
Figure 4.45: Dashboard- Manager View.	173
Figure 4.46: Attendance List - Manager View.	174
Figure 4.47: Dashboard- Admin View.	175
Figure 4.48: Attendance List – Admin.	176
Figure 4.49: Dashboard – Employee View.	177
Figure 4.50: Exported CSV attendance report ready for sharing via mobile.....	178
Figure 4.51: Attendance report previewed directly on a mobile device.	179
Figure 4.52: Notification confirmation shown after an Admin/Manager sends a location request to the user.	180
Figure 4.53: In-app notification screen showing the employee's received location verification request.	181
Figure 4.54: Push notification received on the user’s device prompting location update.	182
Figure 4.55: In-app notification screen showing the employee's received location verification request.	183
Figure 4.56: Profile Page.	184
Figure 4.57: Profile Update Confirmation.	185
Figure 4.58: Config Setting Page.....	186
Figure 4.59: Support Page.....	187
Figure 5.1: General Usability of CorePresence.	207
Figure 5.2: Attendance Features for CorePresence.....	208
Figure 5.3: Interface Design of CorePresence.	208
Figure 5.4: Overall Experience of CorePresence.....	209

CHAPTER 1: INTRODUCTION

1.1 Background

Attendance refers to the regularity and consistency with which individuals participate in events or obligations, such as work or education. In workplaces and educational institutions, monitoring attendance is essential to ensure accountability and operational efficiency. Regular attendance is not only critical for success in educational environments but also plays a key role in managing employee productivity and performance in the workplace. An accurate and efficient system for tracking attendance helps organizations and schools reduce absenteeism and optimize scheduling. In the workplace, attendance involves reporting for scheduled shifts and being present for their entirety.



Figure 1.1: Manual Punch Cards (GoGet, 2023).

Attendance management systems have evolved significantly over time. Traditional methods, such as manual check-ins using paper-based records or punch cards, as shown in

Figure 1.1 above, often result in inefficiencies and inaccuracies, especially in large organizations. Inaccurate attendance records can lead to payroll errors, poor performance evaluations, and challenges in resource allocation. Moreover, Traditional methods involve the manual recording of attendance on document, which may lead to inaccuracies and the possibility of attendance falsification. As organizations strive for more efficient ways to track attendance, advanced technologies such as biometric authentication and geolocation have become essential for enhancing the accuracy and reliability of these systems.

The integration of biometric authentication into attendance systems has revolutionized the way organizations handle employee presence verification. Biometrics, including fingerprint offer a higher level of security compared to traditional methods like ID cards or passwords. According to Rahman (2021), Biometric technologies, such as fingerprint authentication, provide a secure and dependable means of authenticating individual identity, guaranteeing that only authorised individuals are permitted to record their presence. These systems also ensure that the individual physically present is the one being recorded, increasing accountability in the workplace (Memane et al., 2022; Vinod et al., 2021; Basloom et al., 2020; Adiono, 2021; Ali et al., 2020; Dutta et al., 2020). With the inclusion of biometrics, attendance systems have transitioned from simple tracking mechanisms to advanced security solutions, safeguarding both employees' time and organizational resources.

Incorporating geolocation technology into attendance tracking systems further enhances the reliability of these systems. Geolocation allows organizations to ensure that employees are physically present within the office premises during working hours. GPS-based attendance systems can verify not just the time of clock-in but also the location, adding an extra layer of verification. By combining biometrics with GPS tracking, organizations can address concerns about remote workers or those working from multiple locations. The geo-location

feature involves the ability to identify the physical location of a user's device or smartphone through geographical coordinates (Taju et al., 2024).

Real-time data collection and monitoring add a dynamic element to attendance management, providing employers with up-to-the-minute information on their workforce. This capability allows managers to instantly address attendance issues, optimize staff allocation, and improve operational efficiency. Additionally, real-time tracking facilitates better decision-making regarding resource allocation, ensuring that staff levels are adjusted based on current attendance. The inclusion of real-time monitoring, along with biometric authentication and geolocation, not only boosts operational efficiency but also helps in managing employee productivity more effectively (Chauhan et al., 2024). These technological advancements position attendance systems as essential tools in modern office environments, contributing significantly to overall organizational success.

An effective a sense of self and attendance management system is essential to mitigate the rising issues of imitating others, theft, and absenteeism in workplaces, while also addressing the heightened security demands in sectors such as healthcare, finance, government, and education (Teye, 2022). The Advanced Attendance and Presence Verification System Using Multi-modal Authentication and Geolocation for Office Environments project aims to address key challenges, one of which is buddy punching, a common form of time theft. Buddy punching occurs when workers falsely clock in or out for one another, it is a widespread problem with workplace efficiency and ethics (Manikam, 2024). To address these problems, the proposed solution incorporates multimodal authentication technologies, including biometrics and geolocation, to verify that personnel are physically present while clocking in and out. This method seeks to improve security, dependability, and operational efficiency, directly

confronting the shortcomings of conventional attendance systems and alleviating problems such as buddy punching (Manikam, 2024).

Therefore, an Advanced Attendance and Presence Verification System Using Multimodal Authentication and Geolocation for Office Environments is proposed to address the growing need for secure and accurate attendance tracking in office environments. This system will integrate multimodal authentication methods, such as biometric verification, and geolocation services to ensure that employees are physically present within predefined office boundaries. The system will capture and authenticate attendance data in real-time, leveraging GPS tracking and biometric identifiers like fingerprint. By doing so, it will eliminate issues like "buddy punching" and fraudulent attendance, while also providing management with insightful analytics on employee attendance patterns. This approach not only improves the accuracy of attendance records but also streamlines payroll processing and enhances workforce management. The system ultimately aims to increase organizational efficiency by ensuring that attendance records are both secure and reliable, contributing to better decision-making in resource allocation and productivity monitoring.

1.2 Problem Statement

Accurate attendance tracking is crucial in today's workplace to guarantee operational effectiveness and productivity. Manual sign-ins and simple card systems are examples of traditional attendance management techniques that are frequently insufficient for handling the diversified and increasingly mobile workforce of today. This is because taking attendance by manually had numerous drawbacks, such as cost, fraudulent attendance, and the possibility of inaccurate results (Surve et al., 2020). Particularly in larger organisations where it is more difficult to verify individual attendance records, these systems are vulnerable to manipulation

and inaccuracy. In addition to making it more difficult to verify employee attendance, this deficiency adds inefficiencies and extra administrative labour, which has a detrimental effect on workforce management. Organisations are finding it more and more difficult to adopt accurate and efficient attendance systems that go beyond antiquated manual techniques considering these constraints.

“Buddy punching,” a prevalent type of attendance fraud in which workers clock in for one another, is one of the major problems that organisations deal with. By paying workers for hours they haven't actually worked, these methods encourage time theft, which raises payroll expenses and reduces productivity. Furthermore, irregular attendance tracking and absence can lower overall productivity, especially in labour-intensive industries. Research indicates that 75% of organisations incur financial losses attributable to buddy punching, averaging an estimated \$1,560 lost per employee per year (Manikam, 2024). This problem is brought to light by Matt Rissell, CEO of QuickBooks Time, who states, "I was missing almost 30 hours each week, essentially covering the salary of a nonexistent full-time employee." (Rissell, 2016). This emphasizes how expensive it is to track attendance incorrectly and engage in fraudulent activity. Moreover, employees steal an average of 4.5 hours weekly, which amounts to six weeks of unapproved leave per year, thereby affecting productivity and team morale (Manikam, 2024). The issue is particularly severe in high-risk sectors like retail and hospitality, where buddy punching rates may attain 40% (Manikam, 2024). These difficulties demonstrate the need for a more dependable system that can reliably verify employee presence and stop fraudulent clock-ins. How can businesses improve the integrity of recorded attendance data and lessen attendance fraud?

The difficulty of accurately confirming both location and identity in attendance tracking has increased as businesses embrace flexible scheduling and hybrid work models. These circumstances can be challenging to handle, especially if there are unclear limits and problems

with office attendance aren't resolved almost once, which could cause them to become the norm (Carter, 2024). Although biometric technologies such as fingerprints provide distinct IDs for personnel verification, their dependence on one-mode verification frequently fails to verify physical presence at certain places. Organizations may experience productivity losses and attendance fraud because of this restriction. Businesses have difficulties making sure that workers are physically present at designated work locations when they clock in if they do not have an integrated system that combines geolocation and biometric verification. Improving the precision and dependability of attendance data requires addressing these issues.

Contemporary attendance systems frequently lack the real-time insights and actionable data essential for efficient labor management. The absence of direct visibility hinders managers from effectively monitoring attendance trends, promptly resolving concerns, and making informed staffing choices. In the absence of these capabilities, managers cannot proactively arrange and oversee their workers, resulting in increased operational inefficiencies and potentially elevated costs. Finding trends in tardiness or absence is much easier when attendance is monitored in real-time (Jain, 2024). These difficulties underscore the urgent requirement for an improved attendance system that facilitates data-driven workforce management.

The accelerated development of office technology, especially in biometric and geolocation systems, presents considerable privacy and data security problems. These systems must adhere to stringent data protection regulations to ensure the security of employee information and foster confidence. However, real-time location tracking engenders privacy problems, since employees may experience discomfort due to continuous surveillance. Achieving equilibrium among security, privacy, and compliance is crucial for fostering broad acceptance of new technologies while addressing organizational requirements and honouring

employee rights. Addressing these issues is essential for attaining successful and compliant attendance monitoring solutions.

The suggested Advanced Attendance and Presence Verification System Using Multi-Modal Authentication and Geolocation seeks to overcome these problems by offering a safe, precise, and effective attendance management solution. This solution streamlines labour management improves accountability in office settings and lowers attendance fraud by fusing biometric techniques with GPS-based position verification. By doing this, it facilitates more effective resource allocation, lowers administrative burden, and conforms to data privacy regulations. This strategy not only improves operational effectiveness but also cultivates an accountable culture, opening the door to a more transparent and productive workplace.

1.3 Objectives

The primary objective of this project is to design and develop an advanced attendance and presence verification system utilizing multi-modal authentication for office environments. The system will accomplish the following objectives:

1. To improve the precision of tracking employee attendance by using automated, trustworthy recording and validation methods.
2. To equip managers with immediate access to staff attendance information for enhanced productivity and effective workforce distribution.
3. To assist HR personnel in analysing attendance statistics and identifying areas for productivity development, with a focus on time management and absence.
4. To mitigate the risk of attendance fraud through the implementation of secure, multi-modal verification methods that guarantee precise attendance records.

The project seeks to deliver an efficient and secure attendance management solution, thereby improving accountability, operational efficiency, and organisational productivity.

1.4 Brief Methodology

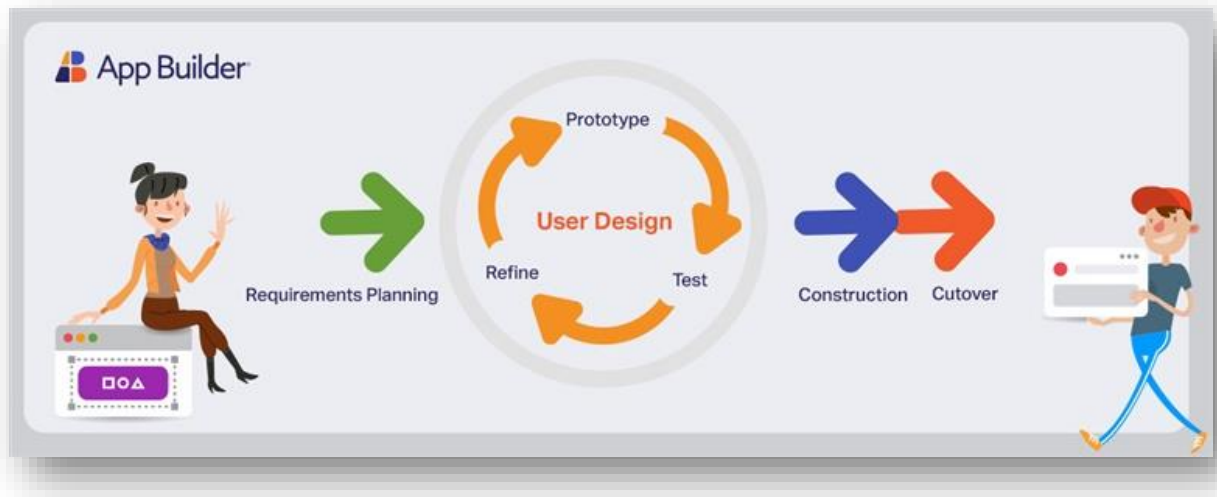


Figure 1.2: Rapid Application Development Methodology (App Builder, 2024).

In this proposed project, the Advanced Attendance and Presence Verification System Using Multi-Modal Authentication and Geolocation for Office Environments would be developed using the Rapid Application Development (RAD) approach as shown in Figure 1.2 above. The RAD framework was established by technology consultant and author James Martin in 1991, who recognised and leveraged software's boundless adaptability to formulate development frameworks (Martin, 1991). RAD is ideal for this project since it focuses on rapid prototyping and iterative development, allowing for quicker delivery while ensuring the system fulfils user requirements and expectations. Given the project's emphasis on real-time attendance verification, RAD enables the quick development of functional prototypes that can be modified and validated with user feedback, resulting in a dependable, high-quality application. According to the literature, RAD is centred on user interaction and prototyping,

where the development life cycle's phases of analysis, design, build, and testing are condensed into a series of brief, iterative development cycles (Berger et al., 2004). The stated shortcomings of the conventional lifecycle are addressed by this method, which promotes efficacy and self-correction as each increment is enhanced (Berger et al., 2004).

The analysis and rapid design phase, which starts the RAD process, is where project requirements are set, including the goals of implementing geolocation and multi-modal authentication. To ensure that the project's objectives are in line with academic standards, this step involves close collaboration with the Final Year Project (FYP) supervisor. It identifies and plans out essential features including fingerprint authentication, QR code scanning, and geolocation tracking.

Subsequently, the project moves into the prototyping phase, during which a collection of designs and functional prototypes is created. React Native functions as the principal framework for mobile application development on Android platform, with HTML, CSS, and JavaScript augmenting front-end design, while Firebase and Supabase (a backend-as-a-service built on PostgreSQL) oversee back-end operations. The prototypes are progressively improved according to comments from the FYP supervisor, ensuring that both the user interface and functionality align with the established project objectives.

The construction phase follows, during RAD's rapid construction phase, the emphasis switches to creating the real application (Kolosky, 2024). During this phase, the prototypes are turned into a functional app that includes key authentication and geolocation features. Regular meetings with the FYP supervisor give critical feedback for refining and improving system operation, resulting in stable and dependable performance.

The subsequent phase involves full testing of the system to verify its functioning, correctness, and user experience. Multi-modal authentication and geolocation functionalities are evaluated in real workplace environments to verify their efficacy. Feedback collected from

FYP supervisor during this phase informs final improvements prior to system deployment. Lastly, the deployment step is putting the system into use and performing maintenance to fix any problems that may come up. The project is concluded when a final evaluation with the FYP supervisor verifies the system's efficacy and preparedness.

1.5 Scopes

The project aims to develop a sophisticated attendance and presence verification system specifically for office-based businesses that depend on effective personnel management, including service-oriented enterprises, corporate firms, or administrative divisions that uses geolocation and multi-modal authentication to accurately track employee attendance in workplace environments. In order to verify employees' credentials and locations during attendance logging, the system will integrate contemporary authentication techniques including fingerprint verification and QR code scanning with geolocation tracking. The target audience consists of employees who will gain from a modern, user-friendly attendance solution that improves security and accountability, as well as management and HR teams who will use the system for efficient attendance monitoring and real-time insights. Using React Native and Supabase (a backend-as-a-service built on PostgreSQL) for safe data management, the project will concentrate on designing, developing, and launching a cross-platform mobile application that works with Android.

1.6 Significance of Project

For organisations, proper attendance tracking is crucial for sustaining productivity and reducing financial losses resulting from absenteeism and attendance fraud. Organisations can ensure accurate attendance records by installing a system that employs multi-modal

authentication, including fingerprint scanning and geolocation verification. This precision diminishes payroll inaccuracies and the likelihood of "buddy punching," fostering a more transparent and efficient workplace. Moreover, this method facilitates adherence to labour regulations by guaranteeing that records conform to legal standards, which is particularly advantageous for firms in highly regulated sectors.

Human Resources and management teams gain advantages from immediate information about attendance patterns. Utilising sophisticated, data-driven solutions to monitor employee attendance enables better informed decisions regarding resource allocation, workforce scheduling, and performance assessments. This feature allows organisations to discern patterns, such as recurrent absence or tardiness, enabling them to implement proactive measures to enhance employee engagement and workplace efficiency. The accessibility of actionable data enables HR professionals to tackle workforce concerns prior to their effect on organisational performance.

Employees benefit from a reliable and precise attendance verification system that recognises their time and dedication. This solution enhances confidence between employees and management by eliminating manual sign-ins and reducing payroll errors, ensuring accurate attendance records and equitable compensation for time worked. Moreover, real-time tracking guarantees employees that their personal data, including geolocation, is utilised exclusively for attendance purposes, thereby strengthening the organization's dedication to data privacy.

This technology improves the experience of distant or mobile employees in hybrid and flexible work environments by enabling them to check attendance effortlessly from specified locations. This strategy facilitates the shift to contemporary, flexible work structures while preserving the organization's supervision. Employees gain from a flexible and secure method of recording their attendance, which aligns with the organization's operational requirements and promotes a culture of adaptability and transparency.

By incorporating modern technology in order to improve security, accuracy, and productivity, the suggested system offers an innovative approach to attendance monitoring that overcomes the shortcomings of manual solutions. This solution does double duty, it reduces the financial losses caused by attendance fraud and sets the stage for better workforce management by creating a data-rich environment that helps with strategic HR decisions, makes people more accountable, and builds continuous improvement.

1.7 Project Schedule

The project schedule offers an ordered timeline to develop the Advanced Attendance and Presence Verification System Using Geolocation and Multi-Modal Authentication. The schedule for development is set out using a Gantt chart that is in line with the Rapid Application Development (RAD) methodology and covers approximately nine months spanning two semesters. Prototyping, testing, requirements planning, and user design are important stages that enable concurrent and iterative task completion for effective time management. To guarantee that all project milestones are met within the allotted period, a more thorough calendar is provided in Appendix A and B, while the Figure 1.3 and Figure 1.4 shows a brief Gantt Chart task table on an overview of the project schedule.

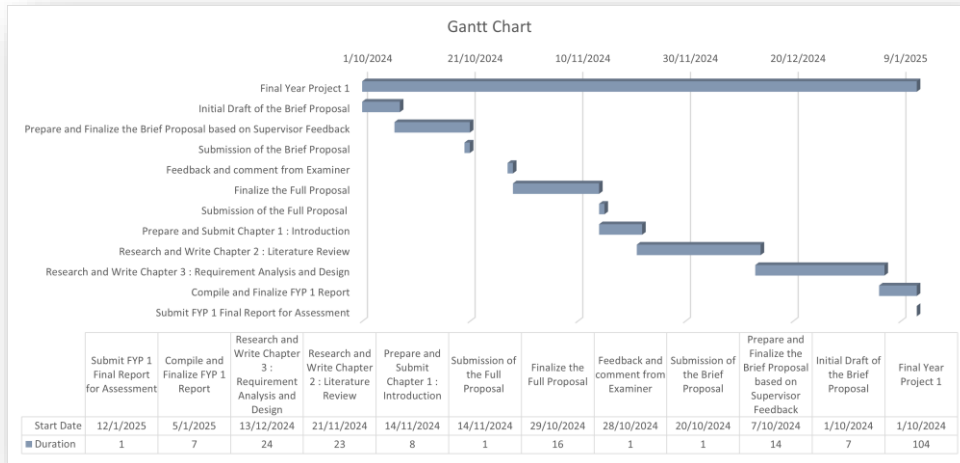


Figure 1.3: Gantt Chart Task Table on Project Timeline FYP 1.

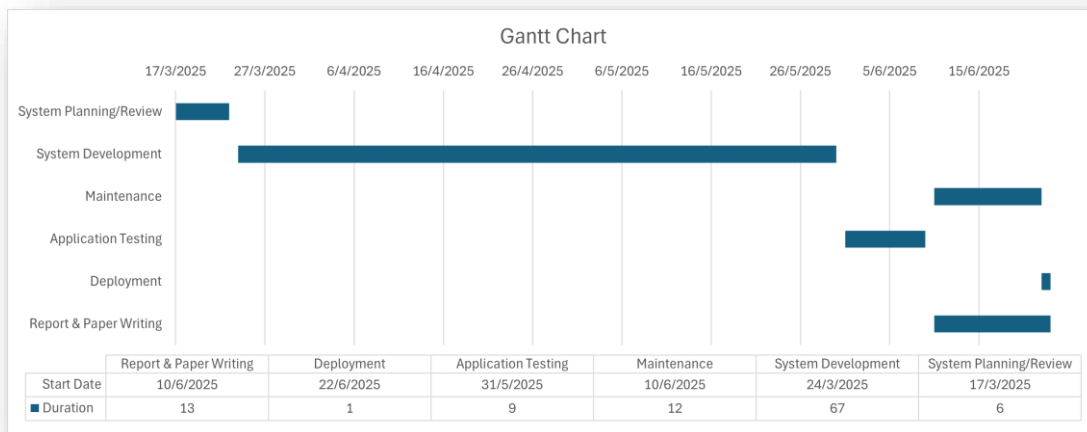


Figure 1.4: Gantt Chart Task Table on Project Timeline FYP 2.

1.8 Expected Outcome

This project will produce an enhanced attendance and presence verification system for workplace environments, employing multi-modal authentication techniques, such as biometric verification and geolocation tracking. This solution will enhance attendance accuracy, offering organizations dependable records and real-time data insights. Integrating these skills will assist

management and HR teams in making data-driven choices, optimizing labor allocation, and addressing attendance trends that affect productivity.

In addition, the project will meet its goals of making a secure, user-friendly system for managing attendance that also meets data privacy standards. Full recording of the system's design process, along with feedback from users and performance reviews, will prove that it works. The goal is to have a dependable and effective system that not only makes tracking attendance better but also helps make the workplace more accountable, which leads to better use of resources and higher operating efficiency.

1.9 Project Outline

This report consists of six chapters detailing the development and implementation of an advanced attendance and presence verification system aimed at improving attendance accuracy and mitigating fraud in workplace environments through multi-modal authentication and geolocation monitoring.

Chapter 1: The Introduction gives an entire overview of the project by talking about its background, problem statement, goals, methodology, scope, their significance, timeline, expected outcomes, and outline. This chapter puts the problem in its proper place by talking about problems that already exist with keeping meticulous records of attendance. The methodology is a summary of the approach that was chosen, and the objectives spell out the exact goals. The project's limits are set by the scope, and the system's expected benefits are emphasised by the significance. The timeline shows a structured plan, and the predicted outcomes list the things that the application is supposed to do.

Chapter 2: Literature Review analyses previous studies, commercial applications, and research that is pertinent to attendance verification systems. The functionality and technology are analysed in this chapter, which contrasts the suggested system with current solutions. The

shortcomings of existing systems are identified, and the ways in which the suggested system fills up these gaps are discussed. The review, which is based on conference papers, industry reports, and journal publications, also looks at the tools and frameworks used for authentication technologies including biometrics and geolocation.

Chapter 3: Requirement Analysis and Design outlines the system development process, encompassing the gathering of requirements and the formulation of system architecture. Design components, such as architectural design, flowcharts, data flow diagrams (DFD), entity-relationship diagrams (ERD), and activity diagrams, depict the system's architecture and functionality. Interface prototypes offer a glimpse of the application's user experience, illustrating how the system will engage with users to facilitate smooth attendance verification.

Chapter 4: Implementation steps involved in developing the system using the design presented in Chapter 3 are described in this chapter. It discusses the technologies, frameworks, and tools utilized in the system's development as well as how important elements like geolocation tracking and biometric identification were integrated. It also describes how various system modules work together and interact to accomplish the desired outcomes.

Chapter 5: For testing, the main goal of this chapter is to confirm the system's dependability, performance, and functionality. It describes the methods, testing settings, and evaluation metrics that are employed to evaluate the system. There is discussion of a variety of testing techniques, including user acceptability testing, integration testing, and unit testing. To make sure the system satisfies its goals and operates at its best, testing phase feedback is examined and improved.

Chapter 6: Conclusion and Future Work is a summary of the project results that shows what was done to meet the project goals. It talks about the problems that were encountered and the lessons that were learnt. Some ideas for future work are talked about, such as ways to

improve the system's usability, scale, and usefulness. Researchers and writers who want to improve attendance verification systems can learn from this chapter. It shows new ways to improve attendance tracking in workplaces that are becoming more mobile and flexible.

1.10 Summary

Chapter 1 introduces CorePresence, a mobile system for attendance and presence verification, developed to meet the evolving needs of modern workplace environments. With the widespread adoption of flexible and remote work arrangements, many traditional attendance systems struggle to deliver accurate and secure tracking of employee presence.

This chapter outlines the purpose behind the project and identifies the core issue — standard attendance methods are often inefficient, lack flexibility, and offer limited verification options. To overcome these shortcomings, the project introduces a mobile application that incorporates multi-modal clock-in features such as QR code scanning and fingerprint authentication, along with geolocation capabilities for real-time presence verification.

Besides outlining the project's aim and key objectives, this section emphasizes its potential to enhance efficiency, security, and convenience in attendance management. In addition, it highlights the significance of the project as a practical solution that aligns with current workplace trends and technological advancements.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This chapter offers a comprehensive examination of current attendance and verification systems to elucidate their characteristics, capabilities, and constraints. This analysis focuses on three analogous applications: Buddy Punch, Jibble, and Deputy, in conjunction with the proposed project. A comparative table has been created to elucidate the similarities and differences between the current solutions and the suggested application via the examination of these systems. This analysis seeks to discover chances for improving the planned project by including advanced functionality and fixing deficiencies noted in the examined systems. This chapter also assesses the selection of hardware and software components necessary for the creation of the proposed system. Particular emphasis is placed on identifying appropriate technologies, encompassing devices and platforms, that correspond with the system's objectives. The advantages of the evaluated systems are integrated into the design parameters, guaranteeing that the suggested solution retains competitiveness while providing distinctive features like multi-modal authentication and geolocation-based attendance. In conclusion, the chapter consolidates findings from the comparative analysis and technological assessment, presenting a unified discourse on the project's distinctive attributes and benefits. This foundation will direct the implementation and guarantee that the proposed system achieves its intended objectives efficiently.

2.2 Review on Similar Existing Systems

This section examines three prominent attendance management systems which are Buddy Punch, Jibble, and Deputy, which possess comparable features to the proposed system. These applications were selected for their innovative methodologies and technological

attributes, establishing them as benchmarks in attendance monitoring and personnel management sectors. Each system provides distinct functions customized for various organizational requirements, integrating simplicity, precision, and flexibility.

Buddy Punch is acknowledged for its intuitive interface and fundamental attendance monitoring features, rendering it an appropriate choice for enterprises in pursuit of effective time management solutions. Conversely, Jibble emphasizes sophisticated functionalities like geolocation monitoring, serving enterprises with contemporary biometric requirements. Deputy is a comprehensive platform that combines attendance monitoring with employee scheduling and workforce management, offering a unified solution for firms seeking to optimize.

These systems underscore progress in attendance management, providing insights into diverse technical applications. They provide essential references for evaluating the strengths and weaknesses of current solutions, while guiding the development of the proposed system to tackle specific difficulties and incorporate extra functionalities.

2.2.1 Buddy Punch

Buddy Punch is a workforce management tool that aims to improve attendance tracking and avoid fraudulent behaviours such as buddy punching. The app's capabilities include geolocation monitoring, biometric clock-ins and time tracking, making it perfect for organizations that require exact attendance verification. According to Kiarie (2024), based on its features and capabilities, the software is appropriate for companies looking to precisely track the time and whereabouts of field employees. According to Czerwonka (2024), In response to a demand for a perfect time-tracking software that tracks workers in real time while they

multitask, Buddy Punch developed. Buddy Punch is compatible with all operating systems and devices, with the best results achieved using Google Chrome.

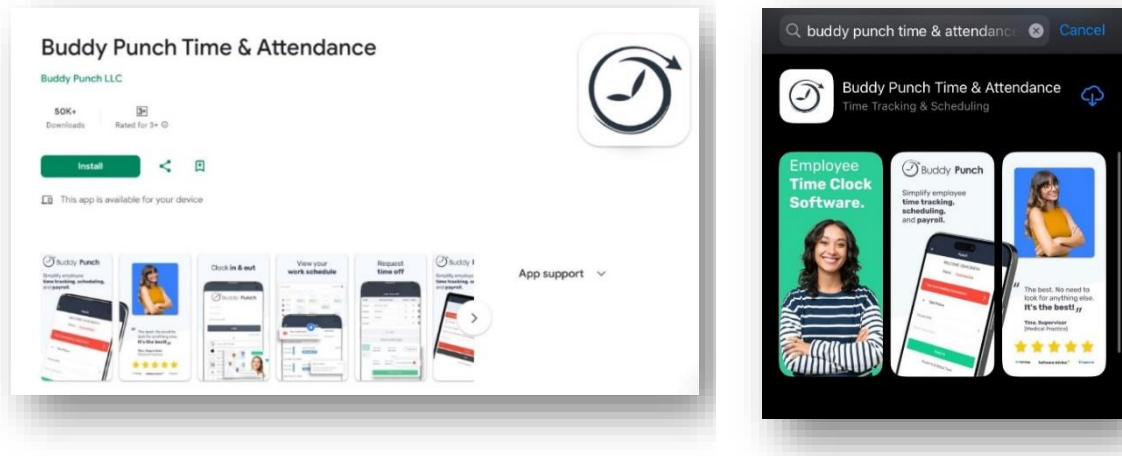


Figure 2.1: Buddy Punch app in both IOS and Android.

Buddy Punch is very adaptable and perfect for companies that want to make tracking time, integrating payroll, and managing employees easier while cutting down on labour costs. Also, the app makes sure that employers follow labour laws by automatically calculating overtime and keeping thorough records of what employees do. Features and Functionalities of Buddy Punch are outlined in Table 2.1.

Table 2.1: Features and Functionalities of Buddy Punch.

Features	Functionalities
Time Tracking	- Employees may clock in and out via mobile apps, web browsers - Monitor staff attendance, breaks, and overtime in real time.
Geolocation & IP Address	- Use GPS tracking to keep an eye on where employees clock in from and limit access based on IP addresses. - Allow employees to clock in only from designated locations to stop misuse.
Payroll Management	Automatically calculate normal and overtime hours, as well as breaks.
Scheduling	- Use drag-and-drop features to create and distribute employee work schedules. - Receive notifications for schedule changes and manage availability request.

Overtime Alerts	• Receive alerts when employees near overtime thresholds. • Establish regulations to prohibit clock-ins prior to designated shifts.
Customizable Reports	• Create customised reports to track attendance, overtime, and labour expenditures.
Notifications	• Get alerts in real time for log-in/log-out events, missed punches, or overtime breaches.
Multi-Device Compatibility	• Access the platform on desktops, smartphones, or tablets with seamless sync.
Integration Options	• Integrate with tools for human resources and accounting functions.
Privacy Settings	• Control access to data for administrators, managers, and staff.

Login and Clock-In Interface

According to Figure 2.2, users need to log in or create an account to access the Buddy Punch application. The login interface provides various authentication methods, such as email, QR codes, or biometric identification. The geolocation feature guarantees that employee's clock in inside a designated zone, as seen in the interface. One disadvantage mentioned is the restricted offline capabilities, which may impede operations in places with low connectivity. Account creation is restricted to the web desktop version. The security features provide reliable identity verification; however, the mandatory registration process may hinder users from promptly accessing the app's features. The application provides a 14-day free trial for users to assess premium features. Upon conclusion of the trial period, a subscription is necessary to maintain access to advanced features. The subscription model may deter casual users or individuals reluctant to make a financial commitment following the trial period.

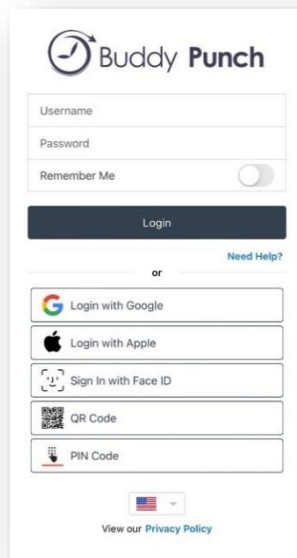


Figure 2.2: Buddy Punch authentication screen.

When establishing an account for Buddy Punch, a company manager is required to select the authentication methods and devices that employees will use to log in, as illustrated in Figure 2.3. Employees have the option of logging in with a 4-digit PIN, QR codes, or a username and password. Furthermore, the manager will designate the devices including desktops, mobile phones, and tablets that employees will employ to log in. Figure 2.3 illustrates this configuration, which guarantees a secure and adaptable time-tracking system that is customised to the organization's requirements. Employees are provided with numerous methods of accessing the application on a variety of devices.

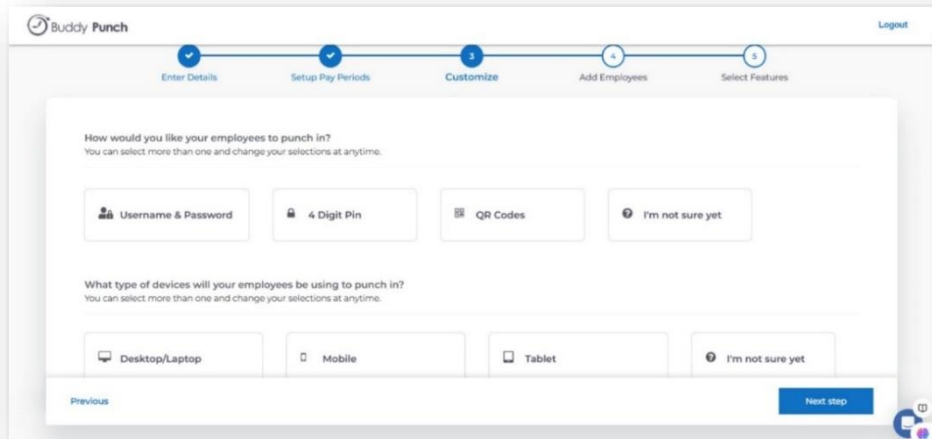


Figure 2.3: Authentication methods and devices employees will use to clock in.

Referring to Figure 2.4, employees can log in to Buddy Punch using various authentication methods. These include the standard username and password, a 4-digit PIN for quicker access, and QR code scanning, which offers an efficient and secure alternative. This flexibility in login options enhances user convenience and supports different levels of security.

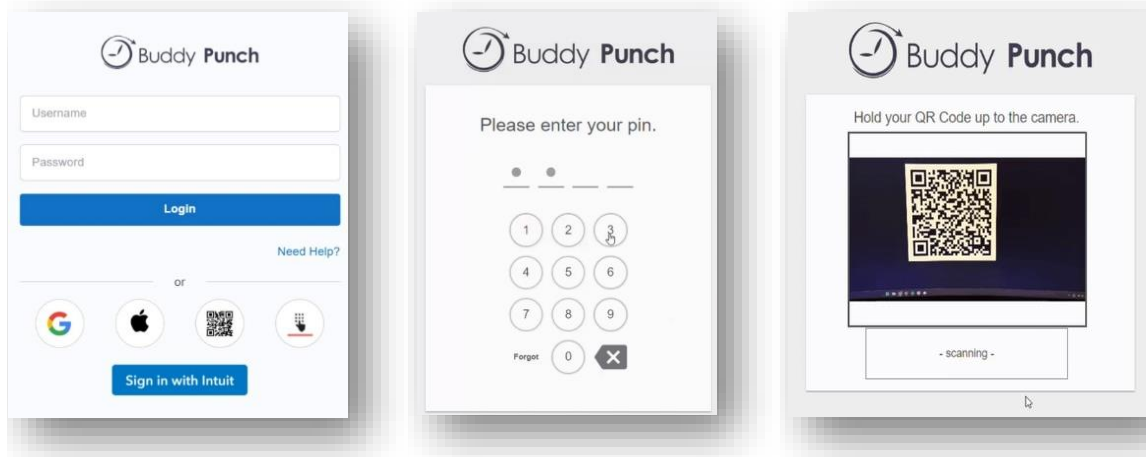


Figure 2.4: Illustrate the login methods available in Buddy Punch: using a username and password, a 4-digit PIN, and QR codes, respectively.

Real-time GPS tracking

The next features is Buddy Punch's real-time GPS tracking, which is designed to give admins and managers accurate information about where their team members are at all times, but only while they are working. This feature makes sure that employee's clock in and out within a certain area of land, called a geofence. This helps stop time theft and makes sure that employees are at the right job site. Also, the app gives employers real-time information on where their employees are, so they can make sure that everyone is where they need to be. Employers can see where their team members are on an interactive map with the map view tool. This makes planning and coordinating better. Employers have the capability to observe the geographical positions of their team members through an interactive map utilising the map view tool, as illustrated in Figure 2.5 Employee's Location Map View.

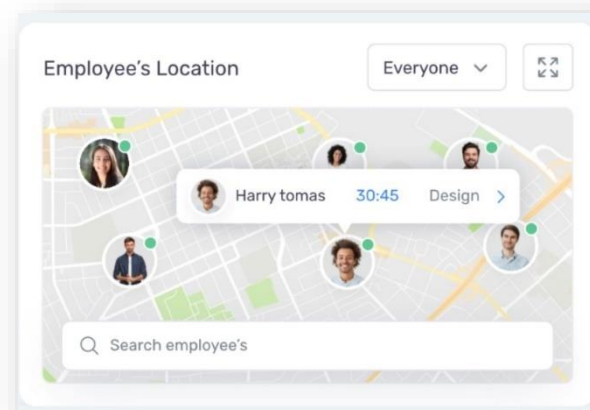


Figure 2.5: Employee's Location Map View.

Figure 2.6 illustrates the integration of GPS data into employee timesheets through the GPS Time Tracking App, emphasising its role in enhancing reporting accuracy.

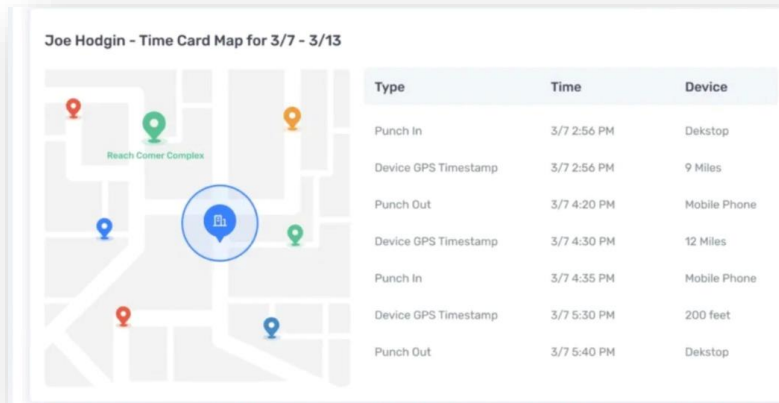


Figure 2.6: GPS Time Tracking App for Employee Timesheets.

Figure 2.7 shows Live location updates illustrate the mechanism through which employers obtain real-time updates, providing a precise understanding of the current locations of their workforce. However, how well it works depends on how well the mobile network is connected. If the signal is weak, the features may not work as well. Overall, this GPS feature gives user a useful tool for keeping track of time and location, which makes security better and increases team management.

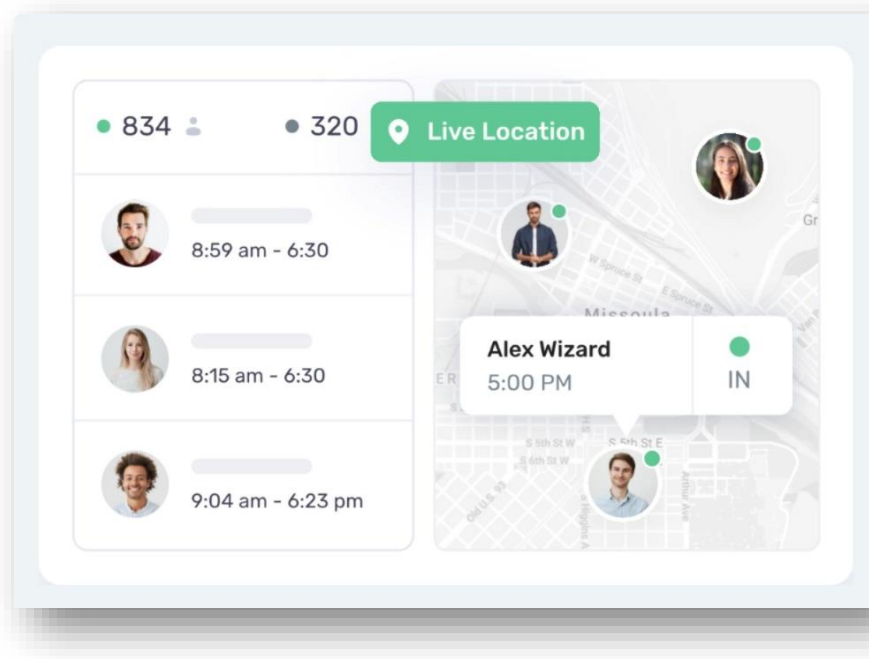


Figure 2.7: Live location updates.

Reporting and Analytics

Buddy Punch's reporting features give managers and administrators detailed time and attendance data. Users can import time data to create detailed reports on employee hours, overtime, and other parameters. Data can be easily imported into structured reports using Figure 2.8. This tool gives administrators precise, up-to-date reports.

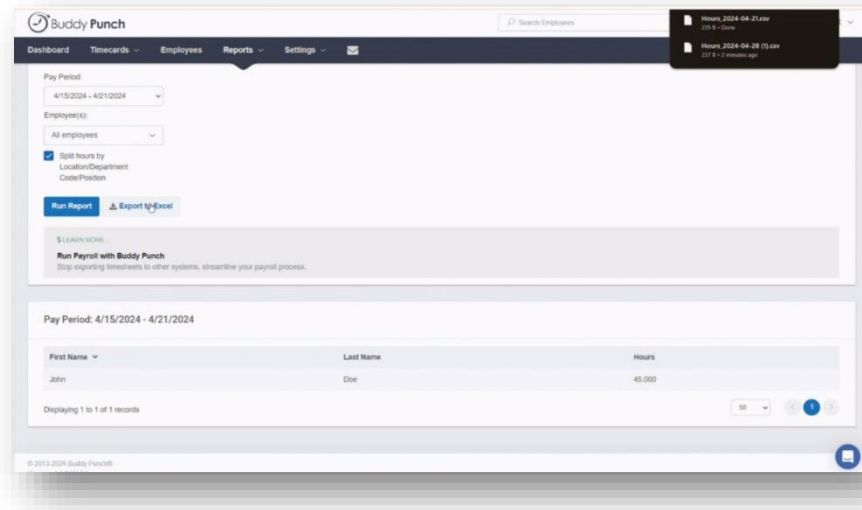


Figure 2.8: Importing Time Data for Report Conversion.

Figure 2.9 Downloading Reports in Excel shows how users can download reports in Excel format for analysis and record-keeping after importing data.

The screenshot shows an Excel spreadsheet titled 'Hours Summary Example'. The spreadsheet has columns for 'First Name', 'Last Name', 'Employee ID', 'Location', 'Department', 'Status', 'Reg', 'OT', 'Total PTO', 'PTO', 'Vacation', 'Total', 'Est Pay', 'Total UTO', and 'Personal Time'. The data is organized by employee, with rows for each employee's hours and PTO. The total for all employees is shown at the bottom.

	First Name	Last Name	Employee ID	Location	Department	Status	Reg	OT	Total PTO	PTO	Vacation	Total	Est Pay	Total UTO	Personal Time
1	John	Doe	12345				0	0	16	16		16	384.00	0	
2	John	Doe	12345	Foundry	Client A		8	3.483	0			11.483	317.39	0	
3	John	Doe	12345	Foundry	Client B		8	2.85	0			10.85	294.60	0	
4	John	Doe	12345	Foundry	Job XYZ		8	1.4	0			9.4	242.40	0	
5	John	Doe	12345	Job Site 123	Job XYZ		8	1.483	0			9.483	245.39	0	
6	John	Doe	12345	Satellite Branch	Client B		8	2.233	0			10.233	272.39	0	
7	John	Iverson					0	0	0			0	0.00	24	
8	John	Iverson		Main Office	Client B		5	0	0			5	70.00	0	
9	John	Iverson		Main Office	Data Analyst		6.417	0	0			6.417	89.94	0	
10	John	Iverson		Main Office	Finance		24	0	0			24	336.00	0	
11	Sarah	Johnson	34523				0	0	40	40		40	600.00	0	
12	Steven	Laramy	593045	Main Office	Data Analyst		16	0	0			16	296.00	0	
13	Steven	Laramy	593045	Main Office	Job XYZ		16	0	0			16	296.00	0	
14	Steven	Laramy	593045	Satellite Branch	Client A		16	0	0			16	296.00	0	
15	Steven	Laramy	593045	Satellite Branch	Client B		16	0	0			16	296.00	0	
16	Dean	Smith	593073	Client XYZ	10-Minute Break		0	0	0			0	0.00	0	
17	Total (1/17/2021 - 1/27/2021)						139.417	11.449	56	16	40	206.866	4036	48	

Figure 2.9: Downloading Reports in Excel.

Managers use the Daily Hours Report, shown in Figure 2.10 Daily Hours, to budget and plan payroll by viewing employee hours, including PTO. This tool gives administrators precise, up-to-date reports, increasing labour management and financial planning.

Daily Hours Export to: [CSV](#) [Excel](#) [PDF](#) [Print](#)

Start Date: End Date: [Submit](#)

Name	Employee ID	Reg	OT	PTO	Total
Bart star	0034	8.00	1.250	0.00	9.250
—	—	—	—	—	—
—	—	—	—	—	—

Figure 2.10: Daily Hours Report for Employee Time Tracking.

Benefits of Buddy Punch

The benefits of Buddy Punch, as outlined in Table 2.2 Buddy Punch is a time-tracking tool that include features like geofencing to guarantee employees clock in and leave at defined places, for safe attendance verification. It integrates with platforms like Zapier, allowing for easy payroll processing and task automation. Its user-friendly interface reduces the learning curve for employees, and features such as bespoke scheduling and payroll reporting enable flexible shift management and optimised payroll operations. The benefits of Buddy Punch are outlined in Table 2.2

Table 2.2: Benefits of Buddy Punch.

Benefits	Explanation
Geofencing	Administrators can specify location limitations for employees to clock in and out within a specific area.
Integration Support	Works with tools like Zapier to automate tasks and payroll apps to make processing easy.
Ease of Use	The platform is made with an easy-to-use interface so that workers can use it with minimal instruction.

Custom Scheduling	• Allows for flexible scheduling and accurate shift tracking.
Payroll Reporting	• Produces reports to optimise payroll procedures and reduce inaccuracies.

Limitations of Buddy Punching

However, Buddy Punch has several downsides, detailed in Table 2.3. Highlight challenges such as mobile app is restricted, with fewer reporting and account administration features than the web version. Users have reported punching out issues in which timers fail to stop properly, resulting in discrepancies in timesheets. The UI lacks customisation, making the dashboard and timecards less user-friendly. Furthermore, reporting options are rudimentary, with no visual representation or customisable filters, complicating full data analysis. These constraints can reduce efficiency for larger teams or sophisticated usage scenarios. In conclusion, Buddy Punch offers a strong foundation for workforce attendance management, however, because to its limitations, it is not as well suited for environments in the workplace that are focused on security and are dynamic. The limitations of Buddy Punch are outlined in Table 2.3.

Table 2.3: The Limitations of Buddy Punch.

Limitations	Explanation
Dependence on Internet	• Consistent internet access is necessary for functionality, which restricts usage in remote or low-connectivity places.
Punching Out Difficulties	• Users experience issues where the timer continues running after punching out, causing timesheet inaccuracies. Editing time entries can also be cumbersome, especially for new users.
System Glitches	• Occasional bugs and delays in logging in or generating reports impact the app's reliability.
Limited Notifications	• The absence of notification features makes it difficult to remind employees about attendance or time management.
Basic Reporting Features	• Reports don't have charts or graphs to show what they state, which makes it hard to see trends. Users are unable to rapidly identify patterns due to the inability to view data trends through charts or graphs. • Reports can't be customised either, and data has to be made separately for time and attendance, which is inefficient.

Dull Interface	• There are few customisation possibilities, and the UI is simple. Especially for large teams, adding new team members necessitates entering a lot of data, which takes time.
Limited Mobile App	• Many of the features found in the desktop or web versions are missing from the mobile app, such as extensive reporting, account settings changes, and access to employee profiles. This limits its usefulness to consumers who rely on mobile devices for time tracking.

2.2.2 Jibble App

Designed to meet the needs of businesses of all sizes and types, Jibble is a powerful and flexible time-tracking and attendance control system (Kiarie, 2024). Advanced technologies like geolocation, and geofencing are used by Jibble to make sure accurate attendance records while also improving security and accountability. Jibble gives on-site, remote, and field workers a lot of access options because it's a mobile app, a web platform, and a kiosk app. As part of the commitment to efficient labour management, Jibble also provides tools for tracking absences, scheduling employees, and making custom Paid Time Off (PTO) policies, which ensure that operations run smoothly (Qureshi, 2024). It has different levels of pricing, with a free plan that has a lot of features and more expensive plans that don't. This means that both small businesses and big corporations can use it. Focussing on correct and unchangeable data, Jibble helps businesses track output more easily and connects easily with other programs, providing a complete answer for managing today's workers. Figure 2.11 shows the Jibble app on both iOS and Android stores.

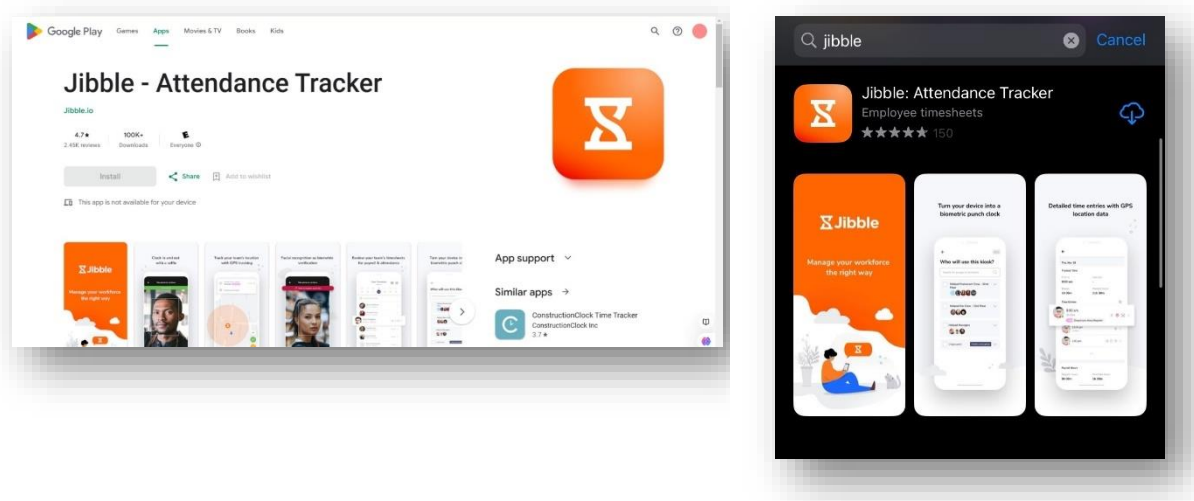


Figure 2.11: Jibble app in both IOS and Android.

Table 2.4 summarise the features and functionalities of Jibble, a powerful time-tracking and attendance management tool. Each component is detailed in detail, demonstrating how Jibble helps firms streamline personnel management by providing features like as real-time tracking, geolocation, project tracking, and extensive analytics. The features are designed to meet a variety of operational requirements, including employee scheduling and labour hours tracking for billing purposes.

Table 2.4: Features and Functionalities of Jibble.

Features	Functionalities
Time Tracking	Employees can clock in and out via mobile apps, web browsers, and kiosk devices. When the internet is reconnected, time entries sync immediately, ensuring that data is collected accurately.
Geolocation & Geofencing	- Tracks employee locations as they clock in and out - The geofencing feature guarantees that attendance is recorded only within predefined geographical bounds, which is useful for remote or field-based teams.
Workforce Scheduling & Leave Management	An easy-to-use tool makes it easier to make schedules for employees, calculates overtime automatically, and handles requests for time off. Managers are capable of quickly say whether to respond to requests for time off.
Project & Task-Based Tracking	Tracking time spent on certain projects or tasks provides insights into productivity, resource allocation, and job efficiency. Effective for categorising and analysing work hours.

Comprehensive Reporting & Analytics	Generates detailed reports on attendance records, timesheets, and productivity, exportable in formats like CSV and XLS. Includes analytics for identifying trends, detecting anomalies, and integrating with tools like Google Workspace for seamless workflows.
Work Hours Tracker	It keeps track of hours by activity, project, or client, which is great for billing and seeing how time is distributed among clients or projects.
Customizable Plans & Features	Gives a range of options to meet different business needs and budgets. The free plan includes basic capabilities like unlimited time tracking and geolocation. The premium and ultimate plans unlock additional features like limitless geofences, priority support, expanded data storage, and custom branding.
Multi-Device Compatibility	Access the platform on desktops, smartphones, or tablets with seamless sync.
Notifications	Get alerts in real time for clock-in/clock-out events, missed punches, or overtime breaches.

Jibble Web App

Jibble's user-friendly and intuitive web interface, as shown in Figure 2.12, which was designed to make workforce administration easier, is depicted in the figure below. The system displays a dashboard with three different views which is Organization, Managed by Me, and Personal when users log in as managers or administrators. By providing a thorough 360-degree view of both individual and organisational performance, these viewpoints facilitate smooth navigation and let users move between jobs and tasks with ease.

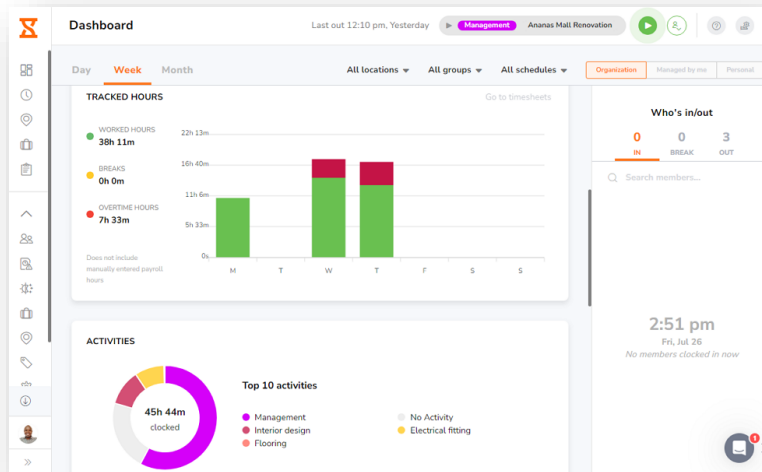


Figure 2.12: Jibble Web App User Interface (Kiarie, 2024).

Jibble Mobile App

Jibble's mobile app interface as shown in Figure 2.13, which works with iOS and Android smartphones, including iPhones and iPads, is depicted in the figure below. The app's smooth operation enables users to effectively track time and manage attendance while on the go. All users, administrators or staff are guaranteed an intuitive experience courtesy to its simple layout and clean style.

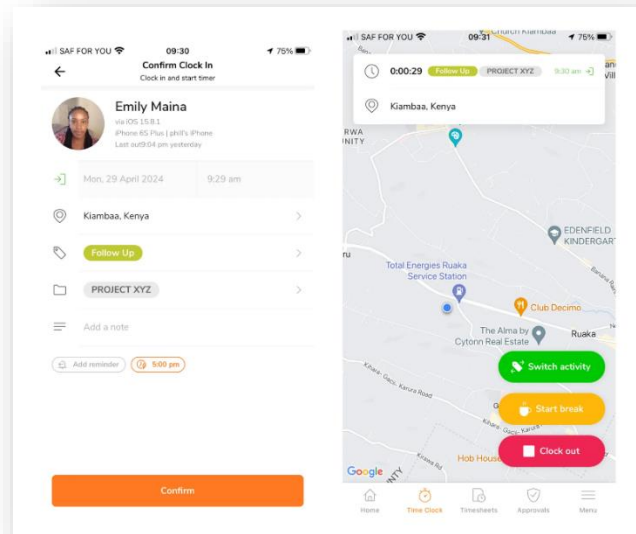


Figure 2.13: Jibble Mobile App User Interface (Maina, 2024)

Time Tracking

Employees might simply clock in and out of their shifts using desktop computers, mobile apps, or kiosk devices by using Jibble's time tracking function as shown in Figure 2.14. The system captures precise time entries, giving real-time information for tracking productivity and attendance. Businesses can effectively manage employee hours by allowing employees to track their time spent on particular projects or tasks. Even when connectivity is lost, the feature ensures that no data is lost by supporting offline functionality and syncing when the internet connection is restored.

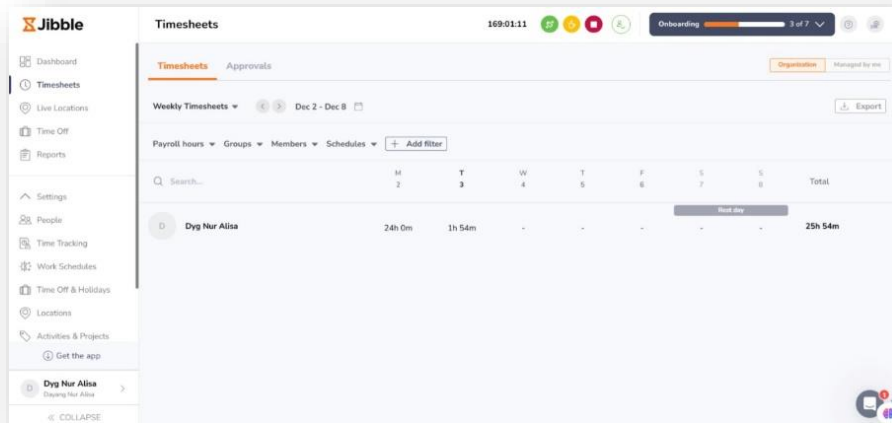


Figure 2.14: Timesheets of the employees.

Figure 2.15 shows how Jibble's clock-in and clock-out work. Users can record their attendance by clicking the clock-in button, which shows the current time. When workers are done, they may click the "stop" button to clock out. Once an employee clocks out, the app automatically creates a time summary that shows how many hours they worked that day. This makes it easy to keep track of their attendance and work hours.

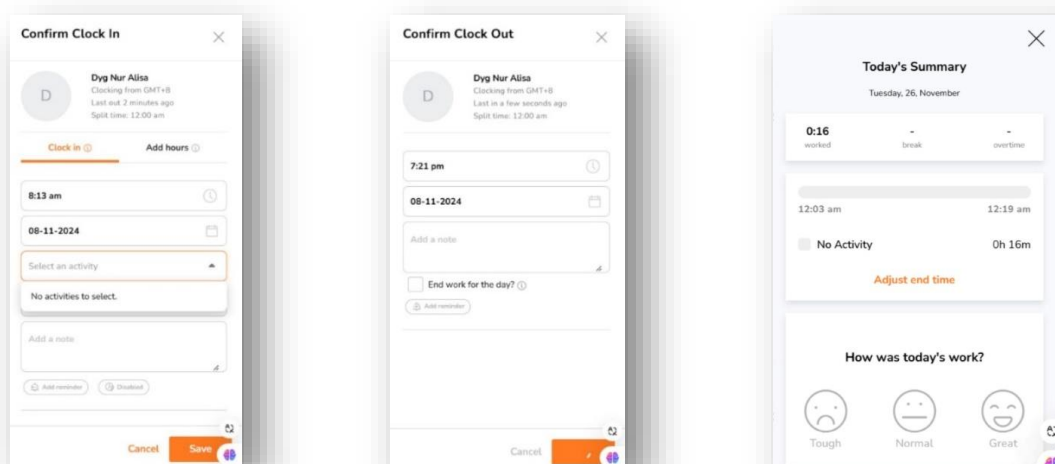


Figure 2.15: Showing the clock-in, clock-out, and time summary features of Jibble, providing a comprehensive view of employee attendance and work hours.

Admins and managers can select the mode and device/operating system by which employees clock in and out on Jibble, whether it is through the mobile app, mobile kiosk, online app, or desktop app. This guarantees that they can prevent time theft and address possible concerns, resulting in a secure and efficient attendance management system. As illustrated in Figure 2.16, device restrictions can be implemented to limit access based on specific hardware, helping to ensure that attendance can only be recorded using authorized devices.

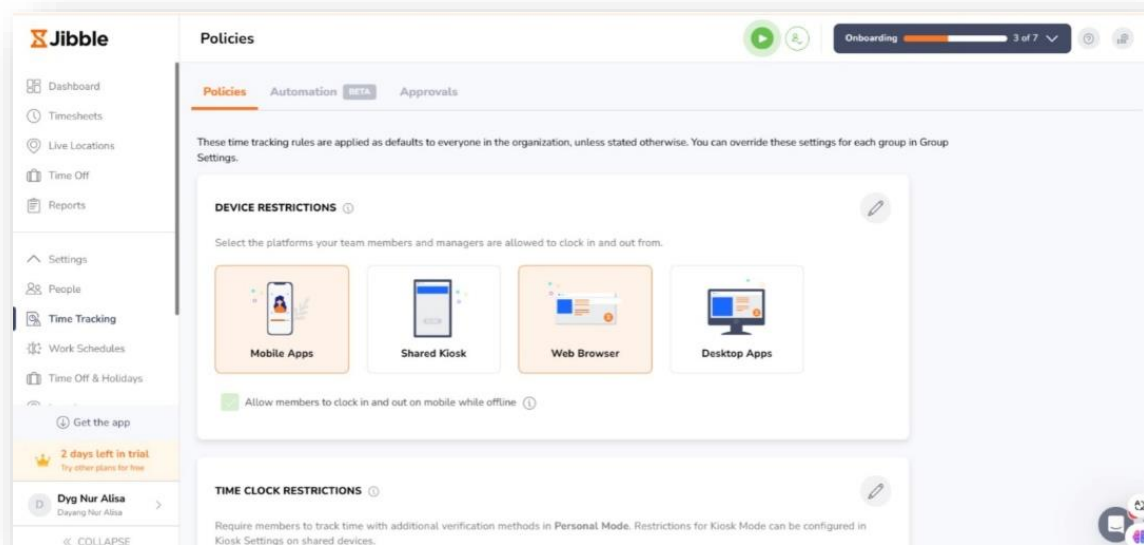


Figure 2.16: Device Restrictions.

GPS Tracking

Jibble's GPS tracking feature allows employers to confirm the precise location of employees during clock-ins and clock-outs, thereby guaranteeing that they are present at the designated worksite. This is especially beneficial for remote or field-based teams, as it provides real-time monitoring to monitor geographic distribution. Jibble ensures the accuracy of attendance data by limiting clock-ins and clock-outs to approved locations through the use of geolocation technology. The timesheets section of both the web and mobile applications

displays the GPS data collected, thereby ensuring that businesses have access to attendance records that are both transparent and dependable. Figure 2.17 demonstrates the GPS tracking features, which allow the system to accurately monitor the real-time location of users during clock-in and clock-out activities.

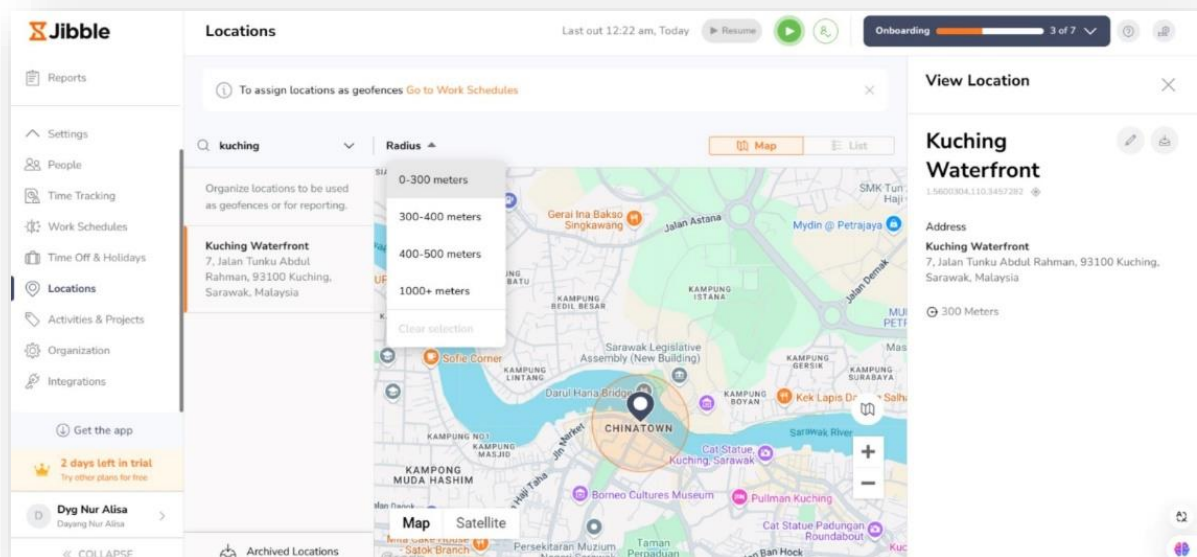


Figure 2.17: GPS Tracking Features.

Using Jibble's live location dashboard, administrators and managers can monitor the current position of their staff members. This gives managers the assurance that they are aware of the actual places of their personnel at any given moment. However, this tool is only accessible to enterprises who have subscribed to the premium plan. It provides an additional layer of advanced tracking for companies that have a greater need to keep a closer eye on their personnel. Figure 2.18 shows Jibble's live location feature, which allows managers to see the real-time location of staff. This feature is only available with the premium plan.

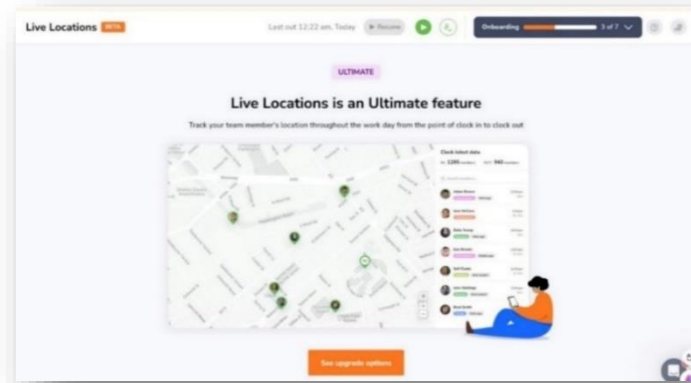


Figure 2.18: GPS Tracking Live Location Features.

Geofencing

Figure 2.19 shows Geofencing feature in Jibble that improves location-based attendance tracking by defining virtual boundaries around the workspace. Employees can only record their attendance while they are within these geofenced zones. This functionality is especially handy for firms with many locations or mobile staff who only need to check in when they arrive at the correct spot. Geofencing prevents unauthorised clock-ins and guarantees data integrity by connecting attendance to real-time locations.

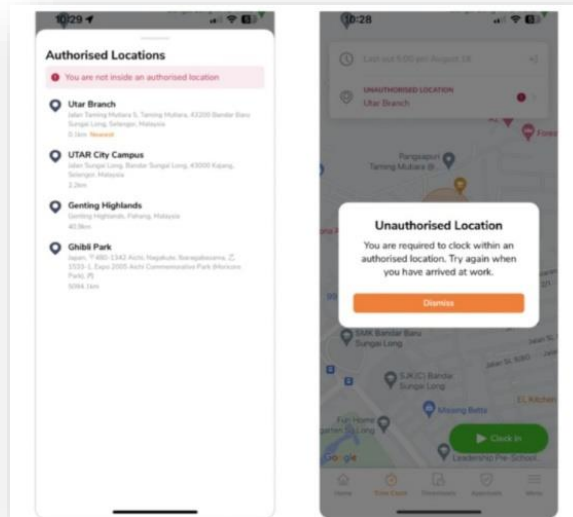


Figure 2.19: Geofence Features (Maina, 2024).

Analytics and Reports

Jibble offers thorough analytics and reports that show how employees are managing their time, showing up for work, and being productive. These reports are very important for HR departments and managers to find trends, like a lot of people being late or not showing up for work, and use the data to make choices that will make the workforce more productive. According to Kiarie (2024), Jibble's reporting system can send data in both CSV and XLS formats, which makes it easy to connect to payroll systems. This function is essential for businesses that want to be more efficient because it lets them keep track of attendance and performance in real time. As shown in Figure 2.20, Jibble allows users to generate timesheets, track time, and view availability insights, helping managers monitor work hours more effectively.

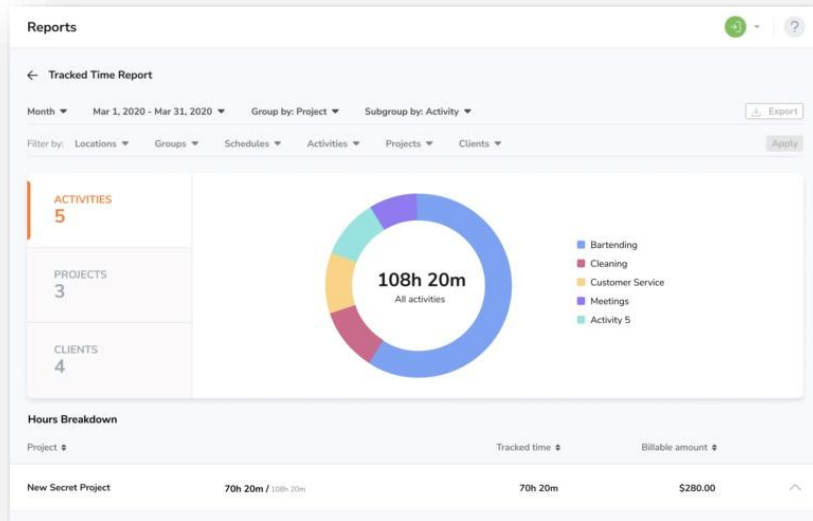


Figure 2.20: Jibble lets user generate timesheets, time tracked, and availability insight reports.

Figure 2.21 shows Reports can be created and exported in Excel or CSV formats for additional analysis. A basic report summary and, if desired, a list of the team's time entries are included in the CSV export. On the other hand, in addition to the optional time entries list, the Excel export offers a more thorough summary. To make sure the report satisfies particular needs, criteria like format and filters can be changed before exporting.

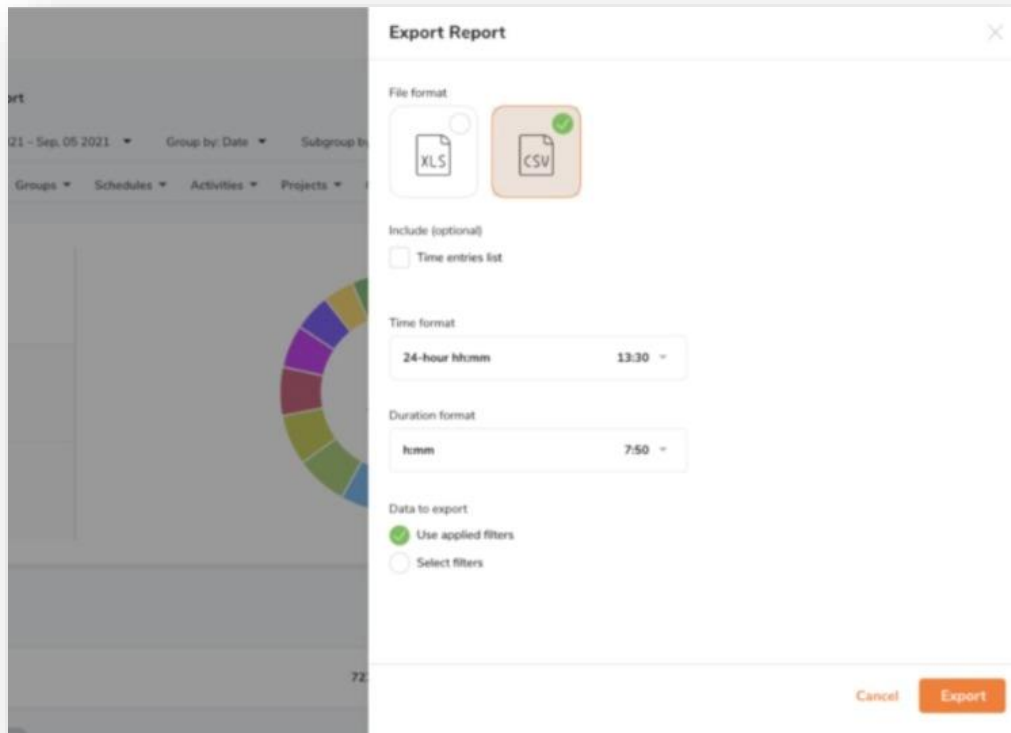


Figure 2.21: Exporting Tracked Time Report.

Smart Reminder Notifications

Jibble's smart reminder notifications helps employees in adhering to work schedules and minimising tardiness. Employees are kept informed of their shifts and responsibilities via the system's automated reminders for clock-in and break periods. By alerting staff members in advance of impending work obligations, this feature improves timeliness and assists managers in lowering absenteeism (Maina, 2024). Businesses with personnel dispersed across numerous locations can benefit from the ability to customise notifications for multiple locations. Figure 2.22 shows Jibble's custom reminder and alert notifications, which help employees stay on schedule by sending automated clock-in and break reminders. This feature supports better timeliness and can be tailored for multiple locations.

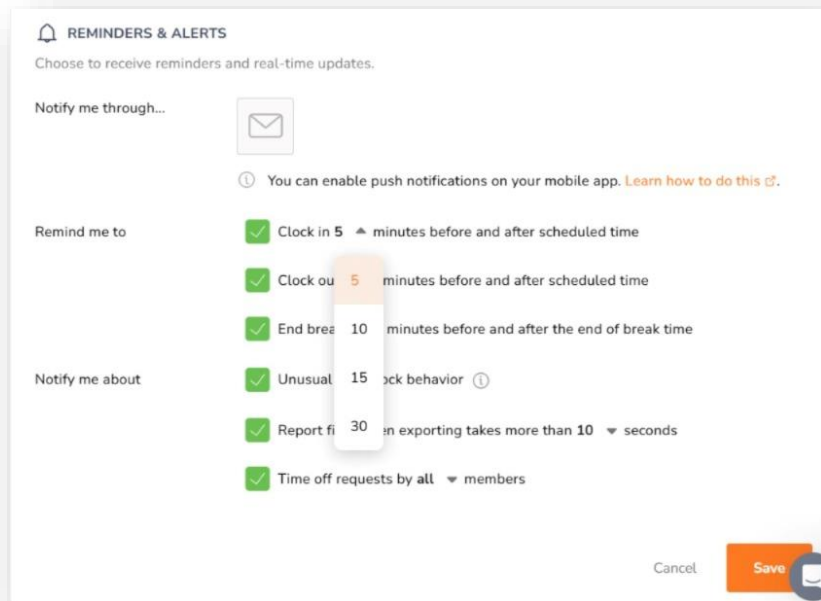


Figure 2.22: Reminder and Alerts custom notifications.

Benefits of Jibble

The table highlights various important advantages of Jibble, demonstrating its adaptability and usefulness for diverse organisational requirements. The Comprehensive Free Plan includes features such as unlimited time monitoring and geolocation, making it an excellent choice for small organisations. Multi-Device Support provides flexibility, allowing employees to clock in and out through mobile apps, desktops, browsers, or kiosks. Geofencing capabilities enable clock-ins and outs to take place exclusively at specified locations, reducing the potential for misuse. The software is designed for easy integration, functioning smoothly with Google Workspace and various other tools to streamline HR processes. Furthermore, its intuitive design guarantees accessibility for users with different levels of technical knowledge. Ultimately, Offline Functionality allows for attendance tracking regardless of internet

availability, guaranteeing dependability in every situation. The benefits of Jibble are outlined in Table 2.5.

Table 2.5: The Benefits of Jibble.

Benefits	Explanation
Comprehensive Free Plan	The free package provides unlimited time monitoring, geolocation, and integration, making it suitable for small organisations.
Multi-Device Support	Mobile applications, desktop browsers, and kiosks are all options for employees to use to record their attendance, providing freedom and convenience.
Geofencing Capabilities	Employees are permitted to clock in or out solely at designated locations, minimising the risk of misuse.
Integration-Friendly	Jibble works well with Google Workspace and other tools, making things easier for HR and management.
User-Friendly Design	The app features an intuitive interface that ensures ease of use, even for those without technical expertise.
Offline Functionality	The offline option allows for attendance tracking even when internet access is absent.

Limitations of Jibble

Jibble has notable drawbacks, detailed in Table 2.6 such as the limited features available in the free plan. Advanced functionalities like geofencing, custom branding, and extended reporting are restricted to costly premium plans, which could pose a challenge for small business budgets. Although it is powerful, its advanced features like project tracking and geofencing present a significant learning curve. The app relies significantly on internet connectivity for essential features such as live location tracking, and users have noted occasional synchronization delays that impede real-time tracking. Moreover, there are privacy issues stemming from the gathering of sensitive information such as location. These factors could influence its appropriateness for companies looking for economical and easy-to-use options. The limitations of Jibble are outlined in Table 2.6.

Table 2.6: The Limitations of Jibble.

Limitations	Explanation
Limited Features in Free Plan	• Paid plans are required to access advanced features like multiple geofences, extensive reporting, customised branding, and personalisation.
High Costs for Advanced Features	• Premium plans and extra subscriptions, such as live location tracking, may pose a financial challenge for small businesses or startups.
Learning Curve for Advanced Features	• New users might need some time to adjust to complicated functions such as project tracking, geofencing, and advanced reporting.
Dependency on Internet	• Essential features such as live location tracking and synchronisation necessitate an active internet connection, even though an offline mode is available.
Synchronization Delays	• Old & recent users have reported some occasional delays in synchronisation, impacting the accuracy of real-time tracking and reporting.
Privacy Concerns	• The application gathers sensitive information, including location, which raises potential concerns regarding privacy and data security.

2.2.3 Deputy App

Deputy is a cloud-based platform for time tracking and productivity improvement developed by Deputechtechnologies (Qureshi, 2024). Deputy is an effective and versatile workforce management tool that caters to the needs of organizations of various sizes and sectors. It offers a full set of capabilities for streamlining critical activities like as personnel scheduling, time tracking, work delegation, and labor compliance. By automating and improving these operations, Deputy decreases administrative hassles while keeping organizations legally compliant and efficient. Deputy is ideal for fast-paced, dynamic work situations, thanks to advanced features such as AI-powered scheduling that forecasts labor demands based on demand, and biometric clock-in systems that include geofencing. Furthermore, its seamless interface with payroll and human resources systems like Xero, makes it an excellent alternative for enterprises looking for a centralized and user-friendly labor

management solution. Figure 2.23 displays that the Deputy app is accessible on a variety of platforms, including Android, iOS, desktop browsers, and a dedicated desktop application.

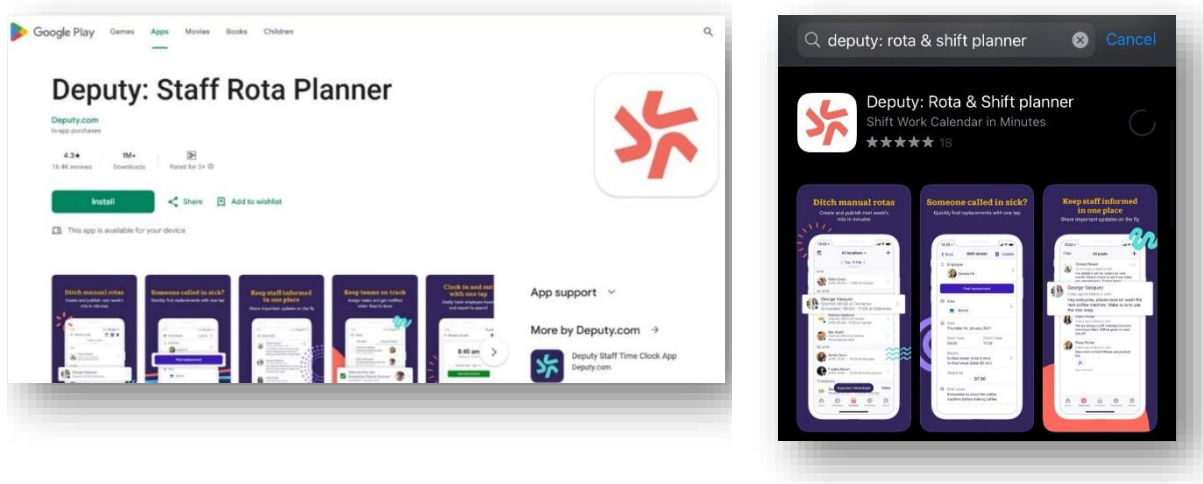


Figure 2.23: Deputy app in both IOS and Android.

The Tables 2.7 summarise the features and functionalities of the Deputy app includes a comprehensive suite of tools aimed to improve workforce management, such as time monitoring with GPS, AI-powered scheduling to optimize shift planning, and break compliance management to enforce legal break periods. It provides real-time updates for task management, payroll connectivity with platforms like as QuickBooks and ADP, and leave management with automated conflict resolution. The app improves staff collaboration with a newsfeed and private messaging; however, the message interface lacks privacy. It also includes compliance capabilities to assure adherence to labour regulations, multi-device accessibility across tablets, mobile apps, and the web, and powerful reporting and analytics features, with personalized reporting presently in beta.

Table 2.7: Features and Functionalities of Deputy.

Features	Functionalities
Time Tracking	Numerous clock-in options (e.g PIN, mobile app) with GPS tracking and geofencing for attendance verification
AI-Powered Scheduling	Automates shift planning based on sales, demand, and labor rules, and lets users choose which shifts they want to work.
Break Compliance Management	Ensures that legal break times are followed, eliminating early clock-ins or unauthorized overtime.
Task Management	Despite having limited customization options, users may create, assign, and monitor projects with real-time updates.
Payroll Integrations	Integrates with platforms like as QuickBooks, ADP, and Xero to provide smooth payroll processing.
Employee Communication	Includes a newsfeed for updates and private communication but lacks privacy in the message interface.
Leave and Availability Management	Uses automatic checks to keep tabs on employees' leave balances and handle conflicts with scheduling.
Multi-Device Support	Accessible through tablets, mobile apps, and the web
Compliance Tools	Incorporates labor law considerations into overtime tracking and scheduling.
Reporting and Analytics	Produces reports on attendance, scheduling efficiency, and compliance which have custom reporting currently in beta mode.

Deputy User Interface Dashboard Overview

Based on Figure 2.24, The Deputy dashboard provides a well-organized and easy-to-use interface for workforce management. It has a tidy interface with well-defined sections like "Needs Attention," "Needs Approval," and "What's Happening," giving users a quick overview of their priorities and activities. Personalized information, such as the user's name, role, and current shift status, is prominently presented to improve accessibility. The "Start Break" and "End Shift" buttons provide quick action, while the "Edit Shift" capability enables users to make immediate changes. The dashboard also includes a calendar view to help you manage shifts and leave requests more efficiently. Interactive aspects like as timesheet approval, request time off choices, and a newsfeed guarantee that tasks are managed and communicated

effectively. The dashboard is designed to be multi-device compatible, meaning it works on desktops, tablets, and mobile apps, ensuring accessibility for users on the go. This user-friendly interface streamlines workforce management, allowing users to navigate and complete activities more effectively.

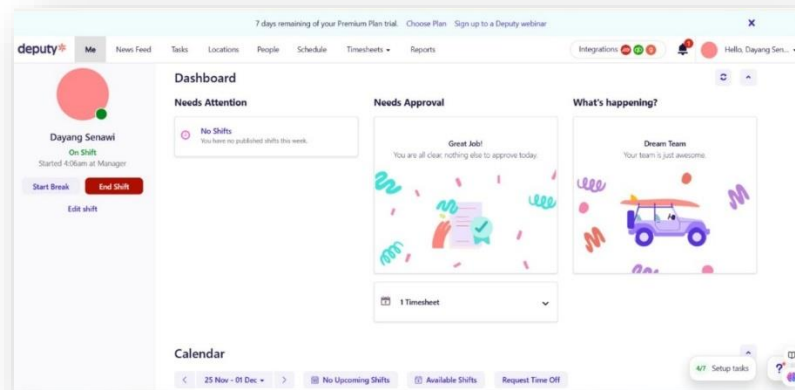


Figure 2.24: Deputy User Interface Dashboard.

Time Tracking

Deputy's time-tracking tool is a simple with a user-friendly interface that simplifies shift management and attendance tracking. Employees may simply start and terminate their shifts with a single swipe, as demonstrated in the Figure 2.25, to ensure proper time reporting.

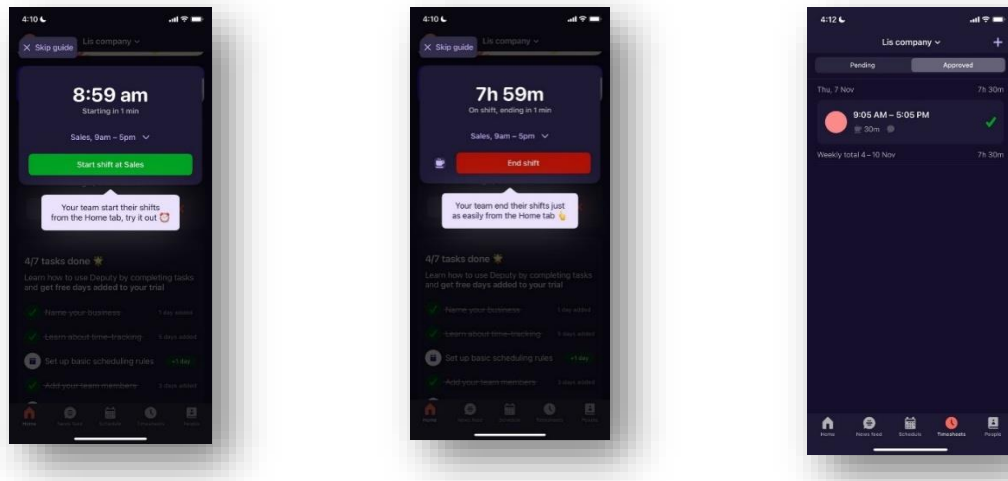


Figure 2.25: Shift Start, Ongoing Shift, and Weekly Shift Summary Interfaces.

The interface gives specific information about the current shift, such as the start time, scheduled duration, and role-specific responsibilities, which improves openness and accountability. The report also contains a current total of completed hours, allowing employees to stay up to date on their weekly totals and approvals. These features allow efficient shift monitoring while providing real-time information to both employees and managers, resulting in a powerful time management solution. Deputy turns all time entries into timesheets for employees. These are easy to look over, change, and accept from a web browser or a mobile app (Wahal, 2024). Figure 2.26 shows method to use Google Chrome to clock in and out of work via the Deputy website. Users only need to go to the website, click the "Start Shift" button, navigate to the "Me" tab, and input their email address and password to accomplish this. When users are finished with the day, they can easily return to the tab and click the end shift button. This will start the countdown.

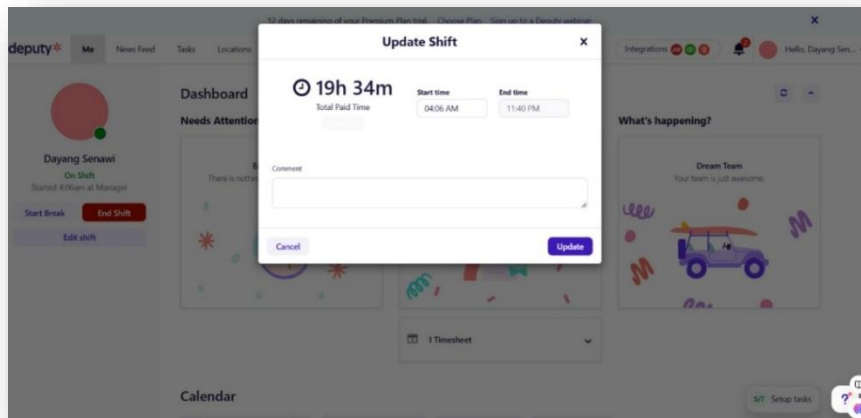


Figure 2.26: Clock In/Out using Website.

Geofencing

On the administrative side, attendance and accountability are ensured through the use of GPS location stamping to show that employees are clocking in from the correct location, or more significantly through geofencing, which ensures that only employees within the virtual perimeter around the job site are permitted to clock in. Figure 2.27 shows the process of enabling geofencing in Deputy, which restricts clock-ins to employees who are physically within the defined job site area. Managers who authorise timesheets can see the clock in and clock out locations of their team members using the Deputy mobile apps. These ensure that buddy punching and hours padding are eliminated from the ranks (Qureshi, 2024).

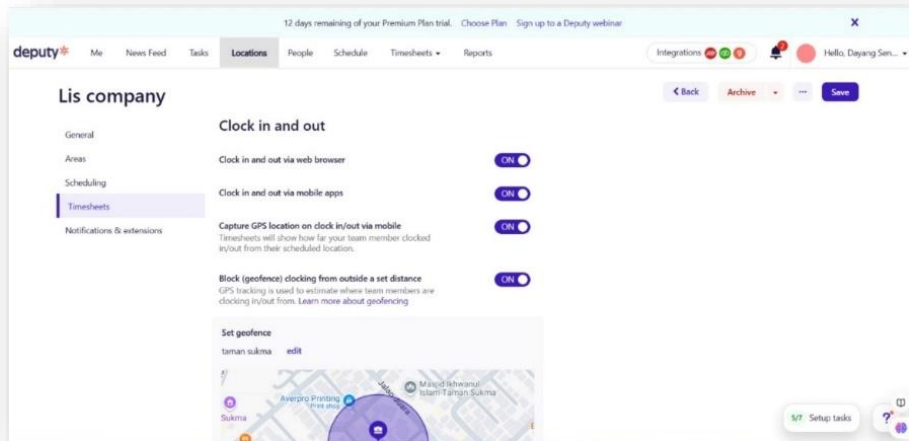


Figure 2.27: Enabling Geofence in location.

Shift scheduling, time off, and compliance

Figure 2.28 shows Deputy has an intuitive scheduling feature, allowing shifts to be made from start using a drag-and-drop tool, ready templates, or by duplicating prior shifts, with the opportunity to include notes for staff (Wahal, 2024).

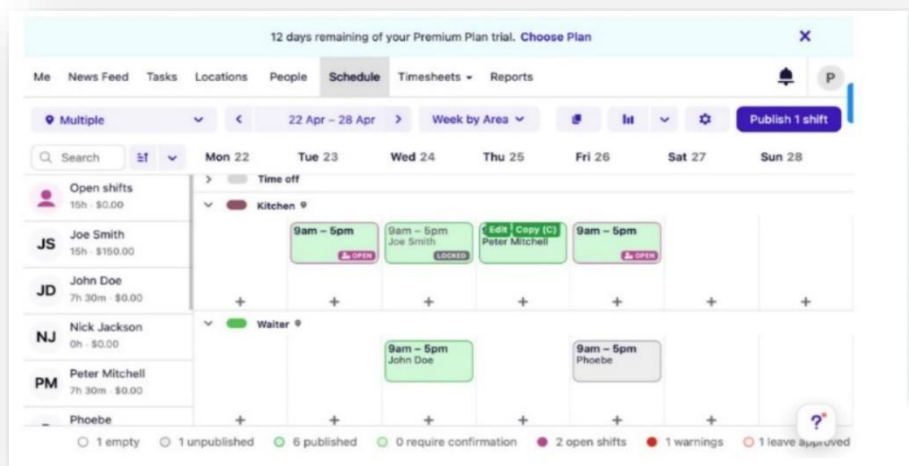


Figure 2.28: Deputy's scheduling interface showcasing flexible shift creation, time-off management, and compliance tracking (Wahal, 2024).

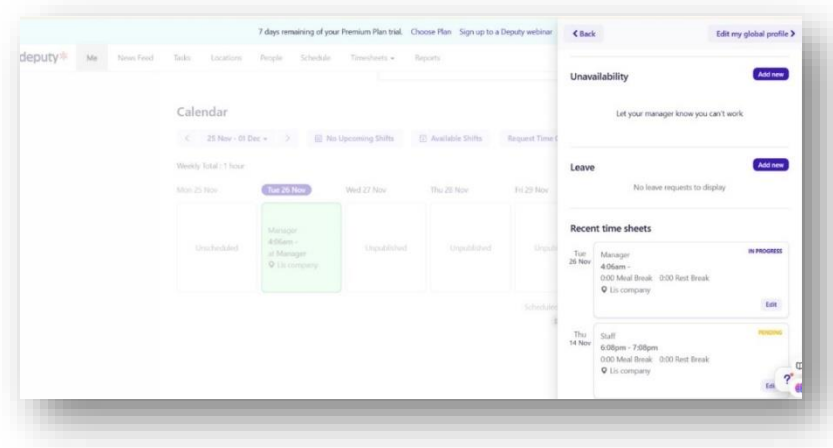


Figure 2.29: Employees can easily mark unavailable times and request leave.

Figure 2.29 shows how Deputy lets employees manage their availability easily by marking themselves as unavailable and requesting leave right in the platform. This feature lets managers check availability in real-time and steer clear of any scheduling mix-ups. Employees can easily let us know when they're not available, whether it's for a scheduled break, some personal time, or other things they've got going on. This feature makes it super easy to handle leave requests and keeps all the availability info in one place, which helps managers make better choices when it comes to assigning shifts. This method makes scheduling easier and helps avoid overbooking or giving shifts to employees when they're off.

Deputy Reporting & Analytics

The Attendance by Report function in Deputy focuses on the "Schedule vs. Timesheets vs. Sales" comparison, providing a full review of employee performance and staffing efficiency. Figure 2.30 illustrates Deputy's Attendance by Report feature, which compares scheduled hours, actual hours worked, and sales data to help managers evaluate employee efficiency and staffing accuracy. This report enables managers to assess the relationship between employees' scheduled hours, actual hours worked, and associated revenue figures. By

identifying gaps or overlaps, firms can adjust workforce levels based on operational needs and sales performance, assuring peak or low-demand efficiency. For example, the functionality tracks whether real work hours match scheduled hours and evaluates their influence on sales results. Filtering options for certain personnel, shifts, or time periods also gives users more flexibility and precision when making decisions. This functionality ultimately promotes better resource allocation and labor management.

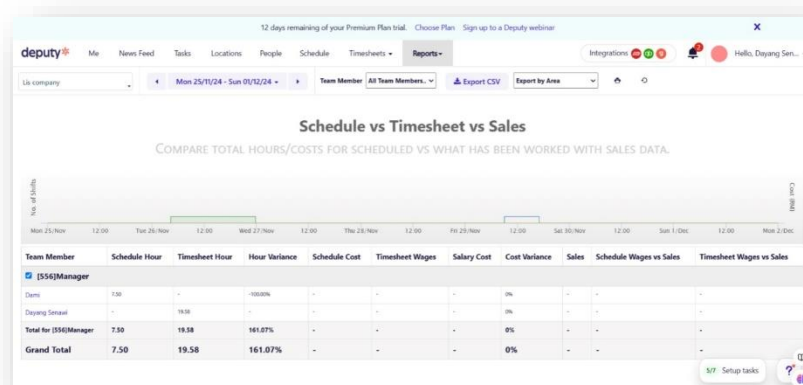


Figure 2.30: Analyse Employee Efficiency with Schedule, Timesheets, and Sales Insights.

Figure 2.31 shows the Deputy Report Dashboard serves as a consolidated platform for accessing and evaluating various workforce data. Managers can access a variety of pre-configured reports on this dashboard, including Schedule vs. Timesheet vs. Sales, Training Status, and Shift Engagement Feedback. Each report includes customizable date ranges and filtering options, allowing customers to tailor the data to their own operating needs. For example, the "Schedule vs. Timesheet vs. Sales" report combines sales data from platforms such as Shopify or manual uploads, allowing managers to compare scheduled hours against worked hours and the resulting sales performance. The dashboard also collects anonymous employee input via shift engagement surveys and monitors training compliance through linked

training records. These insights improve decision-making and workforce optimization by delivering real-time data and actionable intelligence.

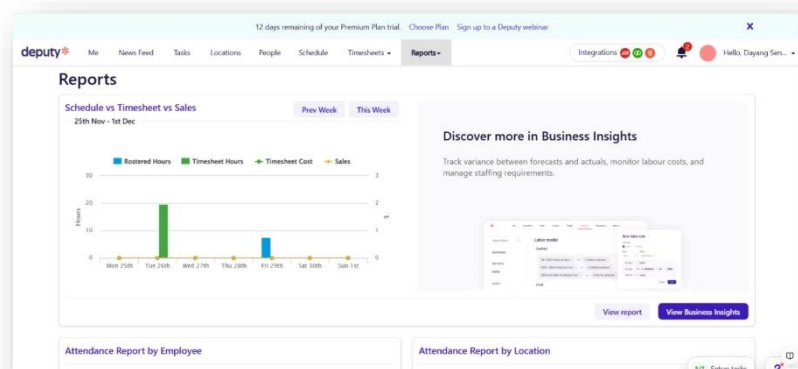


Figure 2.31: Deputy's Report Dashboard.

Benefits of Deputy

Table 2.8 summarises the main advantages of Deputy, a workforce management tool. It emphasises features such as adjustable pricing plans for enterprises of all sizes and biometric clock-in with geofencing, which improves accountability through modern technology. Its artificial intelligence-driven scheduling automates shift planning by taking into account demand projections and labour restrictions. Furthermore, Deputy provides a 31-day free trial for hands-on examination without obligation and includes compliance tools to assure legal conformance. The platform interacts effortlessly with common payroll/HR systems like as ADP and QuickBooks, and it has a user-friendly interface that is available on a variety of devices, including tablets, mobile apps, and websites.

Table 2.8: Benefits of Deputy.

Benefits	Explanation
Flexible pricing plans	Provides several pricing tiers, including alternatives for seasonal businesses, making it affordable to organizations of all sizes.
Geofencing	Improves accountability with modern technologies like location tracking to ensure precise attendance verification.
AI-driven scheduling	Automation of shift planning in response to demand predictions, staff availability, and labor regulations lessens the need for staff involvement
31-day free trial	Includes an ample trial period without a credit card, letting customers try services before committing.
Compliance tools	Ensures adherence to legal standards for scheduling, overtime monitoring, and breaks, reducing the risk of noncompliance.
Integrations with payroll/HR systems	Integrates easily with well-known programs like ADP, Xero, and QuickBooks to streamline staff administration and payroll processing.
User-friendly interface	Designed to be user-friendly, it is accessible across several platforms via tablets, web platforms, and mobile apps.

Limitations of Deputy

While Deputy provides a comprehensive workforce management solution, it has a number of restrictions as shown in Table 2.9 that might impact user experience and functionality. One significant disadvantage is its reliance on a consistent internet connection, which might impede real-time monitoring and communication in remote areas or during outages. Furthermore, the app's offline functionality is limited, making it less useful in locations with low network access. customisation options are very limited, with many corporate use cases hampered by a lack of significant customisation tools, particularly in work management and shift scheduling. Mobile and tablet interfaces might appear cluttered on tiny screens, making them difficult to navigate for employees. Furthermore, reporting features are limited to pre-loaded templates, and fully customizable reports are currently under beta testing, which may not fulfill all organizational requirements. Users have also noted minor technical concerns, such as poor performance and delayed loading times, which can impede usability

during critical tasks. Finally, errors and glitches further interrupt the experience, especially for non-desk employees who rely on real-time updates to successfully handle their activities, resulting in inefficiencies and delays.

Table 2.9: The limitations of Deputy.

Limitations	Explanation
Limited offline functionality	• Most features of Deputy need internet connectivity, making it less useful in locations where network access is limited or unavailable
Customization limitations	• Certain corporate use cases may be restricted by the absence of sophisticated customization capabilities in certain areas, such as task management and shift scheduling.
Cluttered mobile/tablet interfaces	• On smaller screens, the app UI can appear overwhelming, making it difficult for employees to use mobile devices.
Limited reporting customization	• Report generation is currently limited to templates, and fully customised reports are still in the beta stage, which may not satisfy all organisational requirements.
Dependent on stable internet	• Real-time monitoring and communication may be impeded by internet dependency in remote locations or during outages.
Occasional technical issues	• Users have reported problems or delayed loading times within the application, which can impair usability during crucial tasks.
Bugs and glitches	• These problems can have a significant impact on the user experience and functionality, particularly for non-desk staff who depend on real-time updates to effectively manage their duties. Delays and inefficiencies may result from such disruptions, which can complicate employees' ability to adhere to their schedules and obligations.

Deputy emerges as a vital personnel management tool, streamlining businesses with advanced scheduling, time tracking, and compliance capabilities. While it excels in automation and integration, its reliance on a stable internet connection and limits in certain functionalities may present challenges for some users. Overall, Deputy is a valuable tool for firms looking to improve their labor management.

2.3 Features Comparison of the Reviewed Systems

Table 2.10: Comparison of Features between Existing Systems and Proposed System.

Features	Existing System			Proposed Project
	Buddy Punch	Jibble	Deputy	
Free Subscription	✗	✗	✗	✓
Fingerprint Scanning	✗	✓	✗	✓
QR Code Scanning	✓	✓	✗	✓
Geofencing	✓	✓	✓	✓
Real-Time Location Tracking	✓	✓	✗	✓
Real-Time Attendance Tracking	✓	✓	✓	✓
Employee Dashboard	✓	✓	✓	✓
HR Dashboard for analytics	✓	✓	✓	✓
Buddy Punching Prevention (Biometric)	✓	✓	✓	✓

Table 2.10 compares the features of Buddy Punch, Jibble, Deputy, and the proposed project, focusing on major attendance management functionalities. The proposed project distinguishes itself by including advanced features such as fingerprint scanning, QR code scanning, geofencing, and real-time location monitoring, which are not all available simultaneously in other applications. Notably, Buddy Punch and Jibble offer typical functionality such as employee and HR dashboards for analytics, however they lack advanced biometric features such as fingerprint. Deputy, while providing real-time tracking, does not allow biometrics or QR code scanning, limiting its functionality for safe attendance verification.

While all applications are cross-platform compatible, the proposed project strives to improve the user experience by including real-time attendance monitoring, geolocation features, and a full HR dashboard with thorough analytics. These aspects are critical for assuring accuracy and preventing misuse, such as buddy punching, which is not fully addressed in Buddy Punch and Jibble despite their basic capabilities.

In summary, the proposed project improves existing programs by incorporating a wide range of sophisticated capabilities, making it more resilient and appropriate for modern attendance management requirements. Its flexibility to incorporate numerous authentication methods, together with improved analytics and reporting tools, makes it an excellent choice for enterprises looking to implement safe and efficient attendance systems.

2.4 Review on Tools and Technology

The purpose of this section is to provide a comprehensive analysis and discussion on the technological requirements and tools that are necessary for the development of the proposed system. The explication, and characteristics that apply to the requirements for both software and hardware are comprehensively covered in this section. Due to the fact that it elaborates on the reasoning for the system's specific software and hardware requirements in the presence of a number of different alternative choices, this section is fittingly referred to as the "justification" section.

2.4.1 Software

This section will detail the software requirements essential for the development of the proposed system. The software required for development can be classified into two main components: front-end software and back-end software.

2.4.1.1 React Native

A variety of programming languages can be used to develop mobile application development, such as Kotlin for Android, Swift for iOS, JavaScript with frameworks like React Native, and Dart via Flutter. This project uses React native which is an open-source framework for mobile app development that enables developers to create native Android and iOS applications using a single codebase based on JavaScript and React (PS Solutions, 2025). Figure 2.32 presents the architecture of React Native, showing how it enables cross-platform mobile app development using a single JavaScript-based codebase (EDUCBA, 2020).

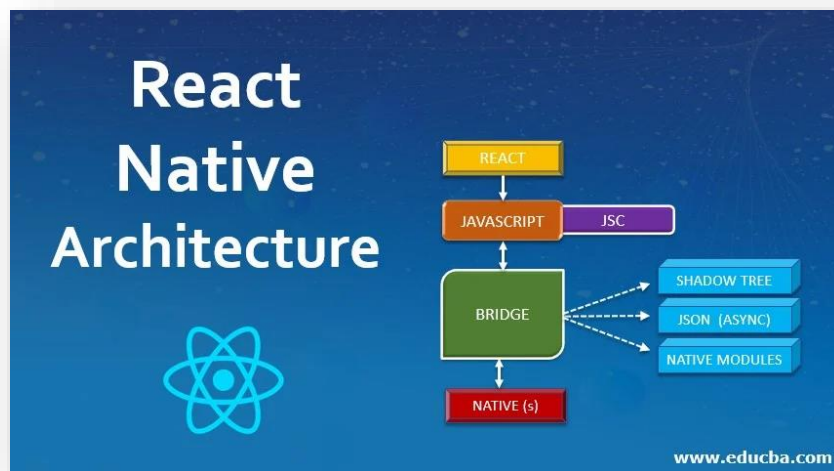


Figure 2.32: React Native Architecture Diagram (EDUCBA ,2020).

React Native was introduced by Meta in 2015 and has been actively maintained by the company since then (React Native, 2024). Figure 2.33 displays the official logo of React Native, representing the framework used for cross-platform mobile development. It fills the gap between web and mobile development by give authorize to developers to use React Native, A widely used JavaScript library for building mobile applications that use native components for supplying. Far from hybrid frameworks that use WebViews, React Native connects directly

with native APIs, ensuring high performance and smooth user experiences across platforms. A standout feature of React Native is its “bridge architecture”, which connects the JavaScript thread to the native thread. This allows developers to program the application logic in JavaScript while make use of native modules built in Java or Swift, enabling for both cross-platform development and platform-specific customization.

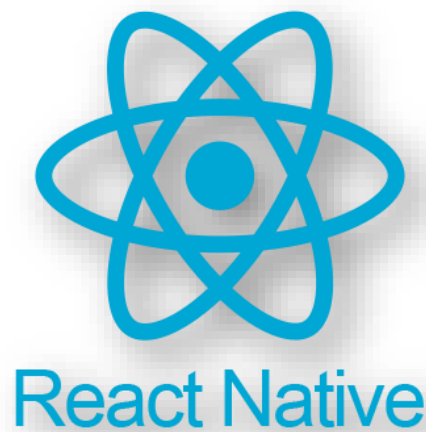


Figure 2.33: React Native Logo (PS Solutions, 2025).

React Native also has a powerful ecosystem with community-developed libraries and integrations. For instance, developers can smoothly develop features like QR code scanning, fingerprint authentication, or geolocation tracking using stable third-party packages. In this project, React Native is integrated with Expo, providing a smoother development and testing experience, especially during the early stages. Refer Table 2.11 for the justifications on choosing the React Native.

Table 2.11: Reasons and Justifications on Choosing React Native

Reasons	Justifications
Single Codebase	Provides native Android development using React and JavaScript without separate codebases.
Native performance	Utilizes native android components facilitates rendering that supports high-performance user interactions
Hot reloading	Allows real-time updates while coding, which helps speed up interface and logic testing.
Strong ecosystem	Provides many open-source libraries and community packages for features like authentication and QR scanning.
Expo integration	Makes testing and managing permissions easier in Android development through Expo

2.4.1.2 Android Studio

In this project, Android Studio is mainly used as a build tool after the Expo-based React Native app is ejected to support advanced Android configurations. After ejection, it plays a vital role in managing native Android files, setting the required permissions, and compiling the app into an Android App Bundle (AAB) for internal testing or Play Store distribution. Figure 2.34 shows the Android NDK version used during the build process, which is essential for managing native code in the React Native project (Mutinda, 2023). Despite not being used as the primary environment, Android Studio plays a huge role in the final phase of the project by generating the deployable build. Its Gradle based configuration capabilities and framework compatibility with React Native make it a reliable tool for handling the Android specific components of the mobile application.

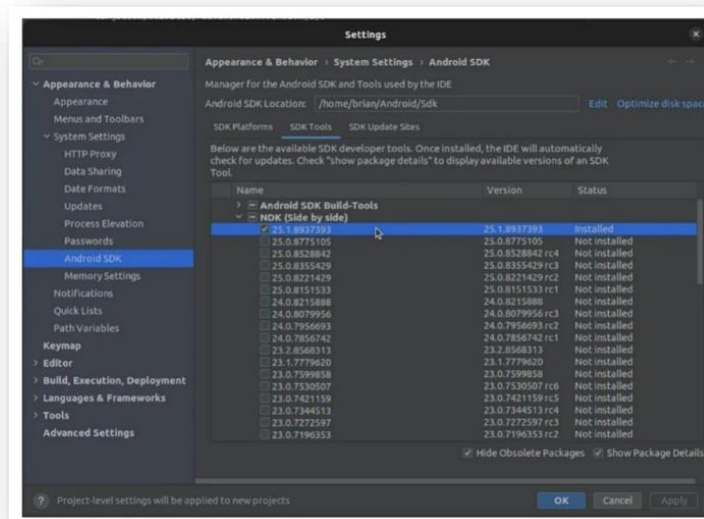


Figure 2.34: Android NDK Version (Mutinda, 2023).

Android Studio is a modified iteration of IntelliJ IDEA, a versatile integrated development environment (IDE) equipped with tools for Java, C/C++, and PHP development, as well as modeling, project management, testing, debugging, and additional functionalities, that is specifically designed for Android app development and is officially supported by Google. In this project, it is mainly used to assemble the deployable build through the Gradle build system, which simplifies dependency management and streamlines the packaging workflow.

Additionally, Android Studio provides functionality for generating Android App Bundles (AAB), which improve app distribution across multiple device types. It also facilitates collaboration and version tracking through integration with systems like Git. Even though it is not used for interface design or main development tasks in this project, Android Studio serves as a key component during the deployment phase of the Advanced Attendance and Presence Verification System by ensuring efficient packaging and distribution. Table 2.12 outlines the reasons and justifications for selecting Android Studio as the build and deployment tool for this project.

Table 2.12: Reasons and Justifications on Choosing Android Studio

Reasons	Justifications
Build Tool	It is used to build the ejected React Native project into an Android App Bundle (AAB) for app distribution
Gradle Build System	Enhances packaging by automating builds and handling dependencies.
Version Control Support	Integrates Git for managing code changes in the native Android part after Expo ejection
App Bundle Publishing	Builds efficient Android App Bundles (AAB) to improve installation performance and ensure compatibility with various devices.

To sum up, Android Studio is a powerful development tool designed for Android applications and crucial in the deployment phase of this project. It is equipped with essential tools to compile, package, and release a secure and efficient mobile app for attendance verification. Featuring Android App Bundles and Gradle build automation, Android Studio is ideal for projects that involve real-time tracking, biometric authentication, and geolocation capabilities.

2.4.1.3 Visual Studio Code

Visual Studio Code is a free and lightweight source-code editor developed by Microsoft. Visual Studio Code runs on macOS, Linux, and Windows, making it a highly accessible tool for developers across platforms. Figure 2.35 displays the Visual Studio Code interface with AI-powered coding assistance, highlighting its support for a streamlined and intelligent development experience (Code, 2025). Getting up and running with Visual Studio Code is fast and straightforward to set up (Code, 2025), providing a more user-friendly development process for individual and student use.

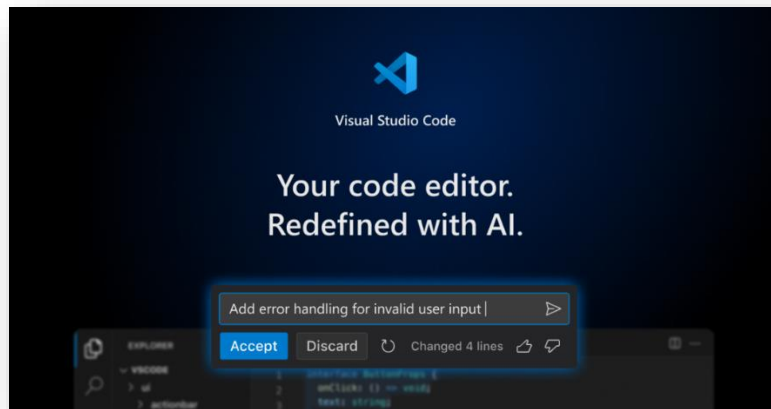


Figure 2.35: Visual Studio Code interface showcasing AI-powered coding assistance (Code, 2025).

In this project, Visual Studio Code is used as the main development environment for developing and maintaining the React Native codebase. It facilitates a range of development activities including programming JavaScript code, organizing project files, setting up Expo configurations, and implementing important libraries for geolocation, biometrics, and QR code scanning. Its responsive user interface and support for helpful extension like React Native Tools help facilitate faster development and maintain readable and error-free code.

Visual Studio Code's integrated terminal and smooth Built-in Git tools help handle version supports efficient change tracking and collaborative development control effectively. In addition, its compatibility with Expo facilitates easier testing and debugging, enabling the app to be developed and tested smoothly on Android devices. Table 2.13 outlines the reasons and justifications for choosing Visual Studio Code as the main code editor for this project.

Table 2.13: Reasons and Justifications on Visual Studio Code.

Reasons	Justifications
Free and Cross-platform	Runs on macOS, Linux, and Windows, with a quick and simple setup.
Lightweight IDE	Lightweight and responsive, making it suitable for JavaScript-based mobile development using React Native.
Built-in support for React Native	Offers support for extensions and plugins built specifically for React Native
Git Integration	Built-in version control simplifies managing and tracking code changes
Terminal & Build Tools	Built-in terminal facilitates the execution of Expo and npm commands in the development phase
Extension Marketplace	Includes a broad selection of productivity extensions like ESLint, Prettier, and React Native Tools.

2.4.1.4 Expo

Expo is an open-source framework that streamlines the development of Android and iOS applications. It provides functionalities like file-based routing and a set of built-in native modules, making mobile app development time saving and more streamlined. Figure 2.36 shows the Expo CLI connected with a mobile app preview, demonstrating live development and debugging capabilities (Expo, 2025). Expo is an open source and supported by a vibrant community on GitHub and Discord (Expo, 2025). Expo also provides Expo Application Services (EAS), a set of tools that support various phases of app development, such as building, testing, and deployment



Figure 2.36: Expo CLI and connected mobile app preview for live development and debugging. (Expo, 2025).

In this project, Expo was executed in the early stages of developing and testing the mobile attendance system. It facilitated rapid prototyping by allowing the app to run directly on physical Android devices using the Expo Go client, eliminating the need to generate APK files in the early development stage. Moreover, Expo's development server, Metro bundler, offered hot reloading, which applied real-time code changes instantly to the app, making it easier to accelerate the entire development process.

One key benefit of using Expo was the ease of configuring device features like the camera, location, and biometrics. These features were important for developing functionalities such as QR code scanning, location-based attendance, and fingerprint authentication in this project. As the project progressed and demanded native modules or advanced configurations, the app was exited from Expo environment for further customization generating the android/ directories to allow for additional customization in Android Studio. Table 2.14 outlines the reasons and justifications for using Expo during the initial development phase of the mobile application.

Table 2.14: Reasons and Justifications on Choosing Expo.

Reasons	Justifications
Cross-platform Framework	• Makes Android development easier by providing a shared set of tools and support for React Native module.
Hot Reloading	• Facilitates real-time code updates, enhancing the overall development workflow
Easy Device Integration	• Enables fast access to device features like geolocation, camera, and biometric authentication
Expo Go Support	• Supports live testing directly on Android devices, eliminating the need for APK generation during early development
Open-source Community	• Supported by a strong and active developer community on GitHub and Discord.
EAS (Expo Application Services)	• Supports extended tools for building, deploying, and testing apps beyond the framework itself.
Eject Capability	• Permits full access to native code when additional modules or advanced setup are required.

2.4.1.5 Supabase

Supabase is an open-source Backend-as-a-Service (BaaS) platform that offers comparable features to Firebase, but with the extra benefits of SQL support (Lee, 2025) . Whereas Firebase relies on NoSQL, Supabase is built on PostgreSQL, an open-source relational database recognized for its performance and flexibility (Lee, 2025). Supabase offers built-in backend tools like Authentication, Database, Storage, and Edge Functions, enabling developers to develop full-stack apps rapidly without handling server infrastructure. Figure 2.37 displays the Supabase interface with real-time database tables used for managing application data efficiently (John, 2024).

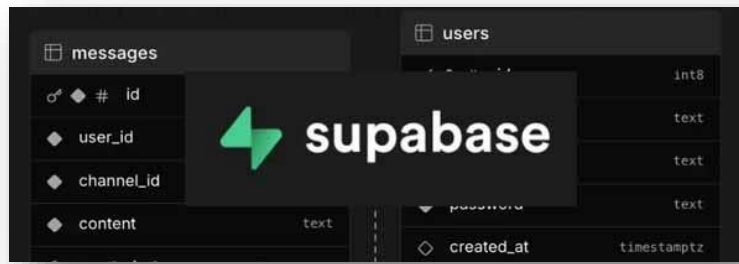


Figure 2.37: Supabase interface showing real-time database tables for application data management (John, 2024).

In this project, Supabase is used as the backend platform to handle user authentication, attendance tracking, and data storage. Supabase is used to store structured data such as user credentials, clock-in timestamps, geolocation information, and QR code scan history. Its integrated authentication service smooths out the login and registration process by providing secure email and password-based sign-in functionality. The Database API automatically offers RESTful and real-time endpoints for querying and managing the PostgreSQL database with the Storage API facilitating secure file storage and retrieval when required.

A key strength of Supabase is its developer-friendly interface along with support for row-level security (RLS) policies, which allow fine-grained control over user-specific data access. Moreover, it offers seamless integration with the React Native frontend via the official JavaScript SDK, allowing efficient interaction between the mobile application and backend services. Table 2.15 outlines the reasons and justifications for selecting Supabase as the backend solution for this project.

Table 2.15 : Reasons and Justifications on Choosing Supabase.

Reasons	Justifications
SQL-Based BaaS	Provides functionality similar to Firebase, with the added advantage of SQL support through PostgreSQL, allowing for more complex queries and structured data handling.
Open-source and Transparent	Open-source and self-hostable, giving developers full control and flexibility.
Real-time Database API	Automatically generates RESTful and real-time APIs directly from the database schema.
Authentication API	Allows secure email/password login and user management.
Storage API	Supports uploading and retrieving files such as backups or scanned data.
React Native Integration	Provides a JavaScript client SDK designed for seamless integration with React Native.
Dashboard and RLS Policies	Delivers a visual UI and detailed access control via row-level security.

2.4.1.6 Firebase Cloud Messaging (FCM)

To meet the need of both data management and real-time communication requirements, this project implements a hybrid backend method. Among several backend technologies to Firebase such as Backendless, Google Cloud Firestore, and AWS Amplify, Supabase was selected as the core backend due to its SQL-based architecture, built-in authentication features, and smooth compatibility with React Native. Besides, Firebase Cloud Messaging (FCM) is implemented to support real-time push notifications. Figure 2.38 provides an overview of the Firebase Cloud Messaging (FCM) service, which enables real-time push notifications within the application (Firebase, 2025). Firebase Cloud Messaging (FCM) was chosen for its lightweight nature and proven reliability, providing timely alerts that smooth user engagement.



Figure 2.38: Overview of Firebase Cloud Messaging service (Firebase, 2025).

Firebase Cloud Messaging (FCM) is a free, cross-platform messaging service that enables reliable message delivery across different devices (FlutterFire, 2025). It supports cross-platform delivery of real-time push notifications to client applications. To support real-time communication, Firebase Cloud Messaging (FCM) is implemented into the app to notify users of attendance confirmations, updates, and system actions. While Supabase manages the core backend functions, Firebase Cloud Messaging (FCM) is implemented to offer efficient and cost-free push messaging. With the help of third-party libraries, it integrates effectively with React Native and delivers messages even when the app is running in background mode on Android devices. Table 2.16 outlines the reasons and justifications for choosing Firebase Cloud Messaging (FCM) to handle push notifications in the application.

Table 2.16: Reasons and Justifications on Choosing Firebase Cloud Messaging (FCM).

Reasons	Justifications
Real-time Notifications	Supports immediate push notifications for attendance updates and system messages on mobile devices.
Lightweight Integration	Easily integrates with React Native using libraries.
Background Delivery	Enables message delivery in both background and foreground modes to enhance user engagement.

2.4.1.7 PostgreSQL

PostgreSQL is a powerful open-source database solution that is fully free of licensing constraints, vendor lock-in, and the risk of over-deployment and it is administered using an object-relational database management system (ORDBMS), (IBM Cloud Education, 2024). Figure 2.39 illustrates the architecture of PostgreSQL, which serves as the underlying database engine powering Supabase in this project. PostgreSQL serves as the core database engine for this project and is accessed through Supabase, a backend-as-a-service platform built on top of it.

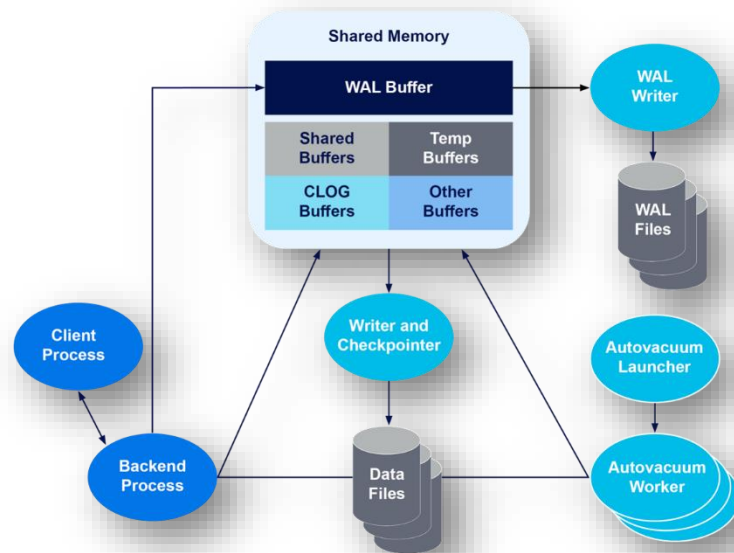


Figure 2.39: PostgreSQL Architecture.

PostgreSQL offers advanced querying, data integrity, and real-time access, allowing the application to efficiently store, retrieve, and manage records. PostgreSQL is support for RESTful and real-time APIs, generated by default by Supabase, provides smooth interaction between the mobile frontend and the backend database. Table 2.17 outlines the reasons and justifications for selecting PostgreSQL as the primary database engine for this project.

Table 2.17: Reasons and Justifications on Choosing PostgreSQL.

Reasons	Justifications
Open-source and Reliable	Established stability and an active community contribute to sustainable scalability and dependable performance.
SQL-based Structure	Enables relational data handling , suitable for managing user data and attendance logs.
Real-time API Support	Allows real-time data operations through the use of APIs provided by Supabase.
Integration with Supabase	Operates in the background to facilitate efficient functioning of Supabase's Authentication, Database, and Storage APIs.
Data Security	Provides secure role-based access control and row-level security (RLS) capabilities via Supabase.

2.4.2 Hardware

An overview of the hardware used in the development of the suggested system is given in this section.

2.4.2.1 HP Laptop 15-fc0xxx

The HP Laptop 15-fc0xxx is a laptop that fits into the middle of the price range and is meant to compromise between performance and affordability. As a result, it is an excellent option for a variety of computer jobs. The clock speed of this laptop is 2.30 GHz, and it is equipped with an AMD Ryzen 5 7430U CPU that is equipped with Radeon Graphics. This ensures that the laptop is capable of performing reasonable graphics performance and efficient multitasking. Because it has 16 gigabytes of random-access memory (RAM) installed, it has sufficient memory to handle apps that require a lot of resources and to multitask without any problems. It is driven by Windows 11 Home Single Language, which provides a user-friendly and up-to-date interface in addition to access to the most recent productivity tools. The 64-bit

operating system possesses these capabilities. In spite of the fact that it does not have a pen or touch input, the configuration of the laptop guarantees dependable performance for application development projects as well as day-to-day use. Because of its enormous memory capacity and efficient CPU, it is a reliable option for performing software engineering jobs and other computational activities. Table 2.18 outlines the reasons and justifications for selecting the HP Laptop 15-fc0xxx as the development device for this project.

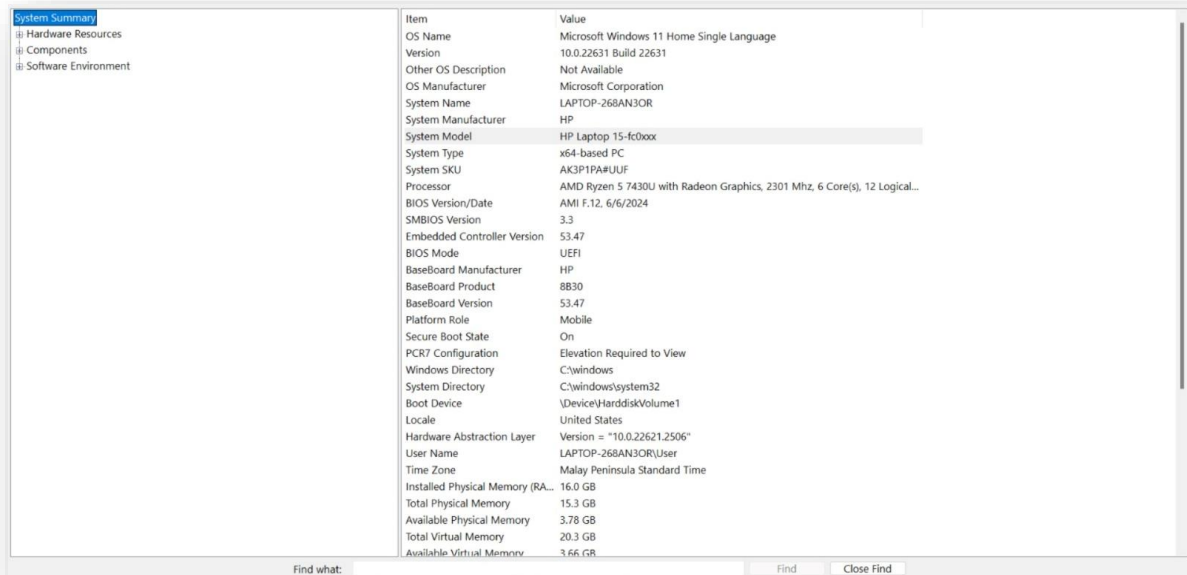
Table 2.18: Reasons and Justifications for Choosing HP Laptop 15-fc0xxx.

Reason	Justifications
Processor Power	• Providing considerable processing capabilities for development projects, including modest graphical activities, the AMD Ryzen 5 7430U with Radeon Graphics is a solid option.
Ample RAM	• 16 GB of RAM allows for smooth multitasking and the simultaneous execution of many apps, which is critical for software development and testing
Modern OS	• A user-friendly interface, effective task management, and access to the newest features and updates are all provided by Windows 11 Home Single Language.
Cost Efficiency	• Provides high performance at a competitive price, making it affordable for students and developers with limited financial resources.
Reliability	• Designed to handle common computing activities in an effective manner, ensuring consistent performance for extended use in situations that are considered to be development environments.

Why the HP Laptop 15-fc0xxx is Suitable for the Project

The HP Laptop 15-fc0xxx is a great choice for the proposed system because to its strong hardware specs and compatibility with recent development environments. The AMD Ryzen 5 processor guarantees that computational activities are processed efficiently, and the 16 GB RAM allows for the seamless handling of resource-intensive software and many tasks. Its Windows 11 operating system offers advanced development tools and is compatible with a wide range of apps. Figure 2.40 provides an overview of the HP Laptop 15-fc0xxx specifications, highlighting its suitability for development tasks in this project. These features

make it ideal for doing software simulations, debugging code, and thoroughly verifying system functionality. Furthermore, its cost means that great performance is available without exceeding budget restrictions, making it an invaluable tool in project development.



The image shows a Windows System Information window. On the left, there is a sidebar with a tree view containing 'System Summary', 'Hardware Resources', 'Components', and 'Software Environment'. The 'System Summary' tab is selected. The main area displays a list of system information items and their values. The 'System Model' entry is highlighted. At the bottom, there is a search bar labeled 'Find what:' and buttons for 'Find' and 'Close Find'.

Item	Value
OS Name	Microsoft Windows 11 Home Single Language
Version	10.0.22631 Build 22631
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	LAPTOP-268AN3OR
System Manufacturer	HP
System Model	HP Laptop 15-fc0xxx
System Type	x64-based PC
System SKU	AK3P1PA#UJUF
Processor	AMD Ryzen 5 7430U with Radeon Graphics, 2301 Mhz, 6 Core(s), 12 Logical...
BIOS Version/Date	AMI F.12, 6/6/2024
SMBIOS Version	3.3
Embedded Controller Version	53.47
BIOS Mode	UEFI
BaseBoard Manufacturer	HP
BaseBoard Product	8B30
BaseBoard Version	53.47
Platform Role	Mobile
Secure Boot State	On
PCR7 Configuration	Elevation Required to View
Windows Directory	C:\windows
System Directory	C:\windows\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "10.0.22621.2506"
User Name	LAPTOP-268AN3OR\User
Time Zone	Malay Peninsula Standard Time
Installed Physical Memory (RAM)	16.0 GB
Total Physical Memory	15.3 GB
Available Physical Memory	3.78 GB
Total Virtual Memory	20.3 GB
Available Virtual Memory	3.66 GB

Figure 2.40: An overview of the laptop specifications

2.4.2.2 Huawei P30 Pro

The Huawei P30 Pro, launched in 2019, is a powerful and flexible smartphone ideal for mobile app development and testing. It has a 6.47-inch OLED screen with a sharp 1080 x 2340 pixel resolution, equipping with clear visuals and a smooth interface, practical for checking UI and user experience during development. The P30 Pro operates on the Kirin 980 chipset, featuring an octa-core CPU and Mali-G76 MP10 GPU that offer smooth and fast performance for demanding apps. With 8GB of RAM and up to 256GB of storage It provides sufficient storage and performance for the user to handle large app files, media, and real-time tasks without issues.

The device supports Android 10 with EMUI 11 resulting in compatibility with current development libraries and SDKs commonly used in React Native and Expo-based projects. Its integrated under-display fingerprint scanner provides biometric features crucial for authentication, and the GPS features facilitate geolocation-based attendance tracking.

Moreover, the Huawei P30 Pro features a Leica Quad Camera system (40MP wide, 20MP ultra-wide, 8MP telephoto, and a TOF 3D sensor) that enables for high-resolution image capture, which is beneficial for testing camera-based functions like QR code scanning. Despite it doesn't support 5G, it provides 4G LTE, Wi-Fi 802.11ac, and Bluetooth 5.0, offering dependable connectivity for real-time push notification testing through Firebase Cloud Messaging (FCM). Figure 2.41 shows the device specifications of the Huawei P30 Pro used for testing various system functionalities, including QR code scanning and push notifications (Rahman, 2019).



Figure 2.41: Huawei P30 Pro device specifications screen used (Rahman, 2019).

Table 2.19 outlines the reasons and justifications for selecting the Huawei P30 Pro as the main testing device for this project.

Table 2.19: Reasons and Justifications for Choosing Huawei P30 Pro.

Reason	Justifications
Android Compatibility	Operates on Android, making it ideal for testing the React Native mobile app.
Biometric Authentication	Equipped with an under-display fingerprint scanner to support biometric login validation in the app.
Geolocation and QR Functionality	Includes GPS and a Leica camera system to enable geolocation tracking and QR code scanning.
Performance and Storage	Featuring a Kirin 980 processor alongside 8GB RAM and 256GB of storage, sufficient for performance testing.
Connectivity Support	Facilitates 4G LTE, Wi-Fi 802.11ac, and Bluetooth 5.0 for reliable communication during app testing.
Camera Capabilities	Featuring a quad-camera setup, it provides accurate QR scanning.

2.5 Summary

In Summary, Chapter 2 provides an extensive analysis of the various attendance and verification systems, specifically highlighting Jibble, Deputy, and Buddy Punch. The systems were analysed for their features, advantages, and drawbacks, yielding a comprehensive comparison with the proposed Multi-Modal Authentication and Geolocation Approach for Attendance Verification. The analysis emphasised key characteristics such time monitoring, geolocation and reporting, subsequently discussing the advantages and disadvantages of each system. This chapter also addressed the technological tools to be utilised in the creation of the proposed system, including React Native, Supabase (a backend-as-a-service built on PostgreSQL), Firebase Cloud Messaging (FCM), Android Studio, Visual Studio Code, Expo and PostgreSQL. Chapter 2 explains the selection of these technologies by analysing the parallels and contrasts with existing systems, arguing that the proposed system will provide superior performance and an improved user experience. The assessments indicate that incorporating multi-modal authentication and geolocation tracking in the suggested system

would enhance both security and efficiency, rendering it a superior solution for attendance verification.

CHAPTER 3: REQUIREMENT ANALYSIS AND DESIGN

3.1 Introduction

This chapter will dive into the requirements and architectural components of the proposed Multi-Modal Authentication and Geolocation Approach for Attendance Verification in depth. This encompasses an examination of the system's functional and non-functional needs, together with the methodologies employed in its development. The selected technique, Rapid Application Development (RAD), will be discussed as the fundamental structure for organizing and overseeing the project. This framework guarantees that the development process is both efficient and comprehensive, incorporating feedback from a user to improve the application's functionality (Pan et al., 2020; Qi et al., 2023; Rieg et al., 2024). The chapter highlights the necessity of incorporating user requirements and technology viability into the design, guaranteeing that the final system is efficient, scalable, and user centric. This research and design offer a complete roadmap for the system's implementation.

3.2 Methodology

Multi-Modal Authentication and Geolocation Approach for Attendance Verification is being developed using the Rapid Application Development (RAD) approach, which was chosen because of its emphasis on iterative development and user-centered design. RAD has four separate phases: requirements planning, user design, construction, and cutover. The RAD approach serves as a viable alternative to the Waterfall model, which is more rigid and emphasises extensive preliminary planning of the entire system (nCUBE, 2020). This methodology is especially well-suited to this project since it enables rapid prototyping and regular user feedback, guaranteeing that the system is created in accordance with user needs while reducing delays.

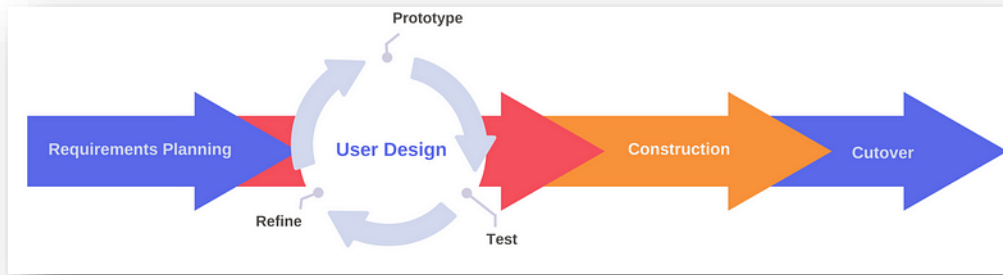


Figure 3.1: 4 phases of Rapid Application Development (Lomio & Mejidana, 2023).

As shown in Figure 3.1, during the requirements planning phase, project goals are set, including the integration of multi-modal authentication methods such as facial recognition, fingerprint scanning, and QR code capabilities, as well as geolocation tracking. In the RAD model, the requirements are less rigorous and comprehensive compared to the Waterfall model (nCube, 2020). Meetings with stakeholders, such as the Final Year Project (FYP) supervisor, are held at this stage to confirm all system requirements and identify any issues early on.

The user design phase focuses on iterative prototyping, in which basic concepts and functionality are swiftly built. Using rapidly produced prototypes fosters user involvement, testing, and feedback on a live system rather than trying to make abstract assessments of a design documentation (Chien, 2020). These prototypes enable users and stakeholders to visualize essential system components including the user interface and authentication routines. Feedback gathered during this stage aids in the refinement of prototypes, ensuring that they satisfy user expectations while maintaining functional.

In the construction phase, during RAD's rapid construction phase, the emphasis switches to creating the real application (Kolosky, 2024), that can perform its intended functions. Implementation of core features is accomplished through the utilization of the selected development tools, which include React Native for the development of cross-platform mobile applications, Supabase (a backend-as-a-service built on PostgreSQL) for the

management of databases, and utilising Firebase Cloud Messaging (FCM) for delivering real-time notifications to users. The stability and dependability of the system are ensured by doing regular testing and receiving feedback during this phase.

Finally, during the cutover phase, the full system is deployed, which includes user training, data migration, and system testing in real-life situations. This step verifies that the system is operationally ready and performing as expected. Maintenance procedures are also implemented to address future issues or upgrades.

Therefore, the proposed system is developed efficiently by utilizing RAD's iterative and adaptable methodology, ensuring excellent quality and conformity with user requirements. This methodology facilitates the project's objective of providing a dependable, scalable, and user-centric attendance verification system.

3.3 Requirements Planning

In the process of requirement planning, a variety of techniques are employed to elicit requirements, including ideation, task analysis, form analysis, user scenarios, and the FAST (Facilitated Application Development Technique) (GeeksforGeeks, 2024). It includes making a structured plan that lists important data, how to gather and handle it, and how to turn that plan into a complete model for system development (GeeksforGeeks, 2024).

During the requirements planning stage for the Multi-Modal Authentication and Geolocation Approach for Attendance Verification, the major business issues and objectives are determined. Activities such as brainstorming meetings with the FYP Supervisor are done to clarify project expectations. In addition, a questionnaire is given to a specific group of users to gather information on existing challenges and expectations regarding attendance systems. This stage guarantees that several views are addressed, highlighting issues such as inaccuracy,

fraud, and inefficiencies in manual processes and to obtain users' feedback on the current attendance system as well as their perspectives on the proposed system.

Following the collection of questionnaire responses, a comprehensive analysis is conducted to extract significant findings. These findings delineate user requirements, guaranteeing that the proposed solution corresponds with user needs and resolves highlighted obstacles. This phase also involves identifying and documenting the software and hardware requirements essential for system development. The result of this phase establishes the basis for other stages, guaranteeing that the project achieves its goals efficiently.

3.3.1 Analysis on Similar Existing Systems

Numerous existing attendance monitoring systems on the market provide powerful functionality, but they frequently have inherent drawbacks that limit their use for certain users, particularly small and medium-sized enterprises. These systems usually have sophisticated capabilities such as advanced analytics, project tracking, and integrations with third-party applications. While these are advantageous for larger firms, smaller organisations may find these systems overwhelming and difficult to use, resulting in inefficiencies in adoption. Understanding and deploying such complicated systems can be time-consuming and resource-intensive, limiting their practical utility for less tech-savvy teams.

Another shortcoming of current systems is their cost structure. Many attendance apps are subscription-based, with additional payments for sophisticated features such as geofencing, biometric verification, and real-time reporting. While free tiers exist, they are often limited to basic functionality, making them insufficient for enterprises with specific requirements. Small organisations may struggle to justify the cost of premium plans, which can vary dramatically

based on team size and feature requirements, limiting their ability to fully utilise such technologies.

These problems underscore the need for a simpler, more cost-effective solution that balances functionality and usability. The suggested solution intends to address these gaps by providing streamlined features such as multi-modal authentication, geolocation, and report generation adapted to the needs of various organisations.

3.3.2 Analysis on Proposed Application

The proposed system, Multi-Modal Authentication and Geolocation Approach for Attendance Verification, addresses the limitations of current solutions by providing a streamlined, user-friendly, cost-effective, and subscription-free approach to attendance management. Unlike many existing solutions, which are either overly complex, expensive, or locked behind paywalls, this application blends easy design with extensive functionality to appeal to modern businesses of all sizes.

To ensure safe and accurate attendance logging, the program uses a variety of authentication techniques such as fingerprint scanning and QR code scanning. When combined with geolocation tracking, the system assures that the employees' clock in and out from specific locations, essentially reducing concerns like "buddy punching" and attendance fraud. The absence of membership costs makes it more accessible to businesses with limited budgets, guaranteeing a fair solution for both small and large organisations.

The proposed application not only tracks attendance but also features reporting capabilities that enable HR teams to generate and export comprehensive attendance and productivity reports. This feature enhances workforce management and facilitates data-driven decision-making. The app is designed for accessibility and operating effectively on Android

devices. It is supported by a strong backend infrastructure utilising Supabase and Firebase Cloud Messaging (FCM) for real-time notifications and secure data management.

Key functionalities include:

1. No subscription fees are required, promoting accessibility for all businesses.
2. Implement secure login through biometric authentication methods, such as fingerprint.
3. Geolocation verification for clock-in and clock-out procedures.
4. QR code scanning facilitates rapid attendance logging.
5. Real-time tracking and alignment of data regarding attendance.
6. Accessible reports for human resource management.
7. User profile administration for both employees and administrators.

3.3.3 Questionnaire

36 respondents from administrative departments and corporate offices in Kuching and Peninsular Malaysia were given a questionnaire disseminated via Google Forms to better understand their needs and expectations for the proposed attendance and presence verification system. Employees, and management representatives were among the responses, guaranteeing a wide variety of viewpoints from different positions inside the companies. This endeavour was made to better understand the requirements of possible customers for the proposed system. Furthermore, the survey attempted to determine whether the proposed system is consistent with the preferences and expectations of its potential customers, ensuring that it fits their needs successfully. A sample of the questionnaire used in the study is provided in Appendix A.

3.3.3.1 Demographics

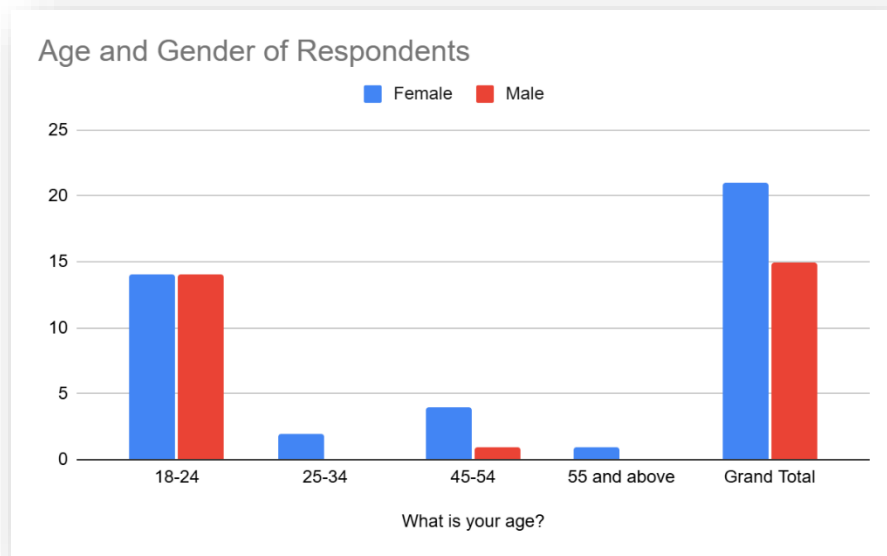


Figure 3.2: Age and Gender of Respondents.

As shown in Figure 3.2, in the first chart, the respondents are 36 people working in diverse office environments, ranging in age and gender. The bulk of responses between the ages of 18 and 24 are females and males, with each accounting for around 10 people. A considerable proportion of responders aged 55 and up are female, indicating a higher participation rate within this demographic. There are fewer responses in the middle age ranges (25-34 and 45-54 years), with low representation of males and females.

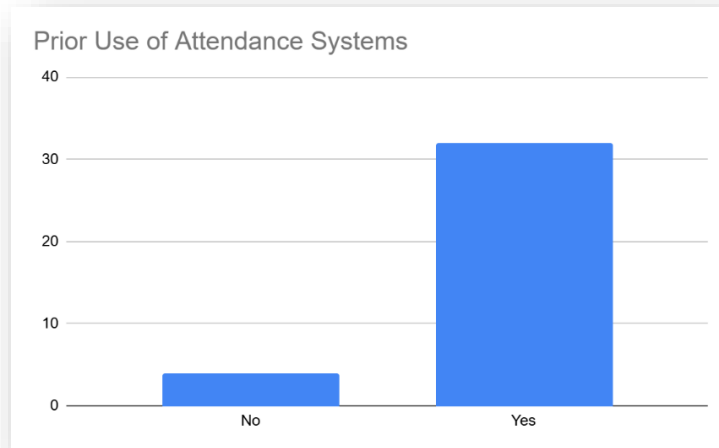


Figure 3.3: Respondent's prior experience with Attendance Systems.

For the Figure 3.3, the second chart shows the respondents' prior experience with attendance systems. A significant majority (nearly 30 respondents) stated that they have previously used attendance systems at their business. In contrast, a smaller sample of roughly 6 people claimed no prior experience with any type of advanced attendance system, meaning that they are employing traditional attendance methods that lack a structured framework. This demonstrates that, despite the prevalence of modern systems, certain workplaces continue to use basic or manual methods.

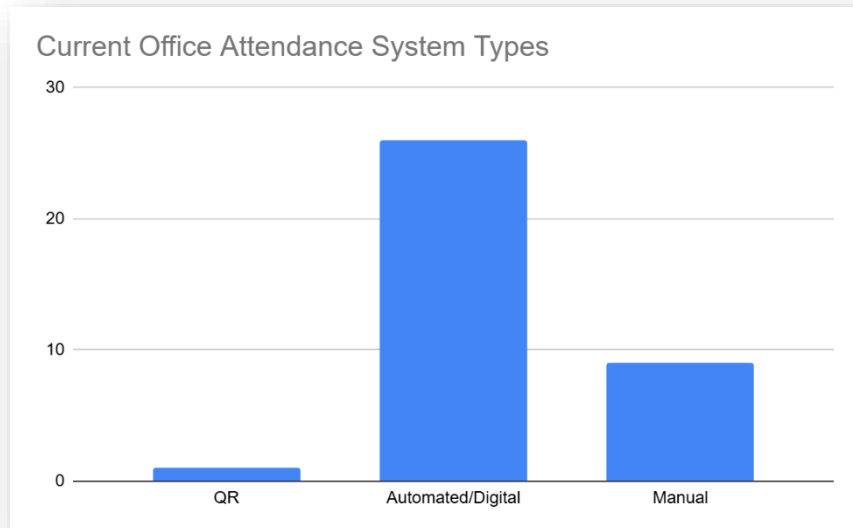


Figure 3.4: Different kinds of attendance systems used in respondents' workplaces.

From the Figure 3.4, the third graph depicts different kinds of attendance systems used in respondents' workplaces. The vast majority (more than 20 respondents) reported employing automatic or digital attendance methods. Less than ten respondents used manual systems, while the QR-based solution received little traction. This distribution shows a significant move toward technology-driven options for managing attendance in modern office settings.

3.3.3.2 Questions for Those Working in an Office

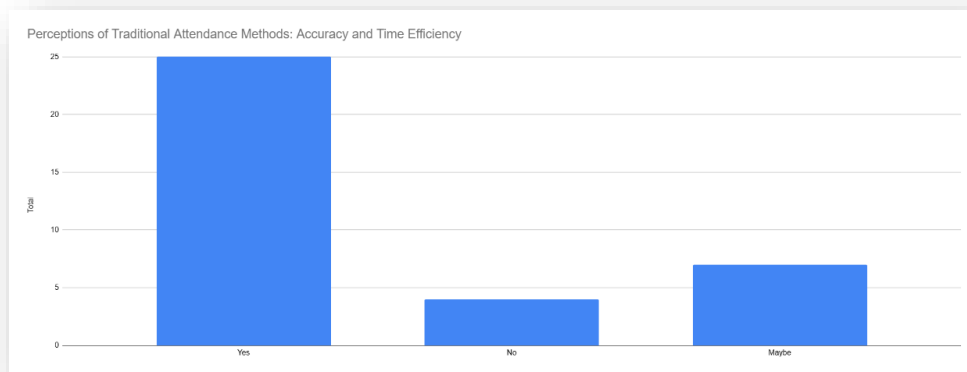


Figure 3.5: Opinions of respondents toward traditional attendance methods.

Figure 3.5 illustrates the chart depicts the opinions of respondents toward traditional attendance methods, such as punch cards or paper-based systems. The majority of respondents (over 25) evaluated these approaches to be inaccurate or time-consuming, with a smaller percentage disagreeing or being unclear. These findings point to widespread unhappiness with old systems, underlining the need for more efficient and dependable attendance tracking solutions. Figure 3.6 illustrates a graphic highlighting the prevalence of buddy punching in workplaces. A large number of responders (more than 20) claimed having witnessed buddy punching, with fewer responding "No" or "Unsure." This highlights a critical issue with traditional attendance systems and emphasises the importance of improved measures to avoid fraudulent attendance marking.

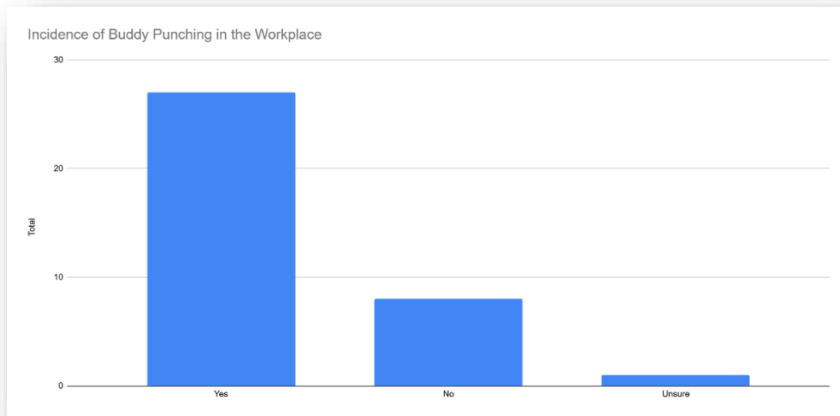


Figure 3.6: Prevalence of buddy punching in workplaces.

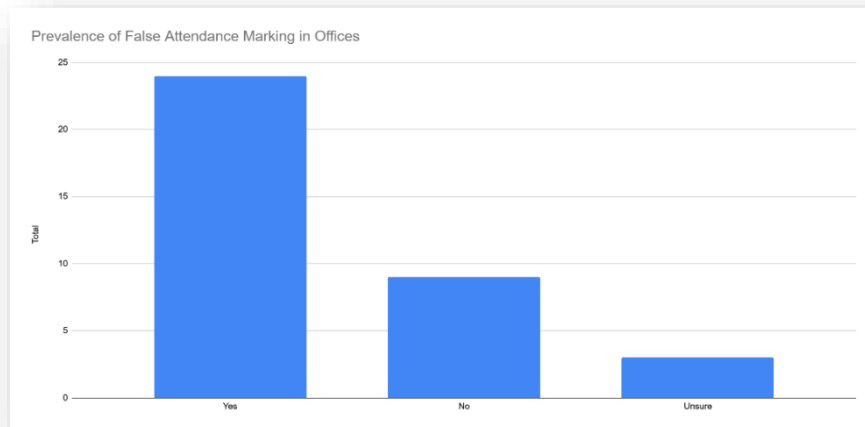


Figure 3.7: Prevalence of False Attendance Marking in Offices.

Figure 3.7 indicates that most respondents (about 25 people) acknowledged the presence of fraudulent attendance markings in their offices. A smaller percentage said otherwise, with some unclear. These findings highlight a reoccurring problem with erroneous attendance reporting and the necessity for rigorous verification procedures.

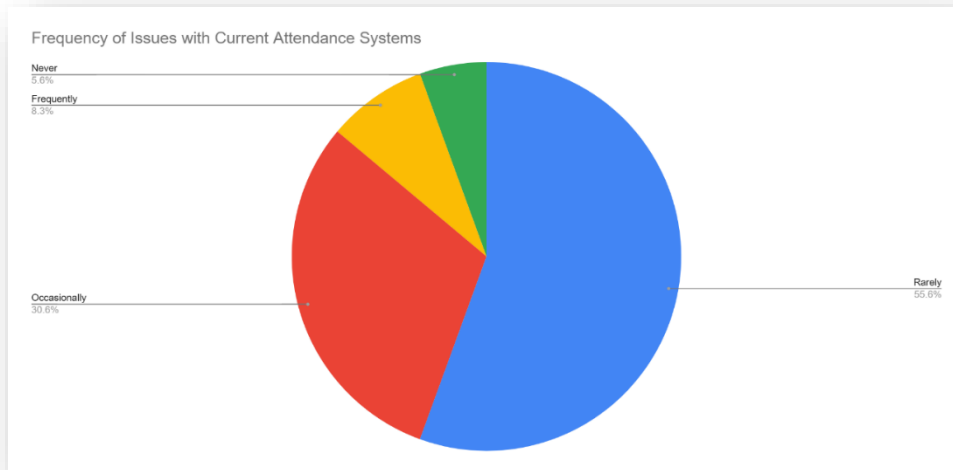


Figure 3.8: Frequency of Issues with Current Attendance Systems.

Figure 3.8 shows the pie charts that indicates how frequently issues happen with current attendance systems. The majority of respondents reported encountering problems rarely (50%), while others encountered them occasionally (30.6%) or frequently (8.3%). Only a small percentage said they had never faced problems. This shows that, while current systems function to some extent, they are not completely trustworthy and may be improved.

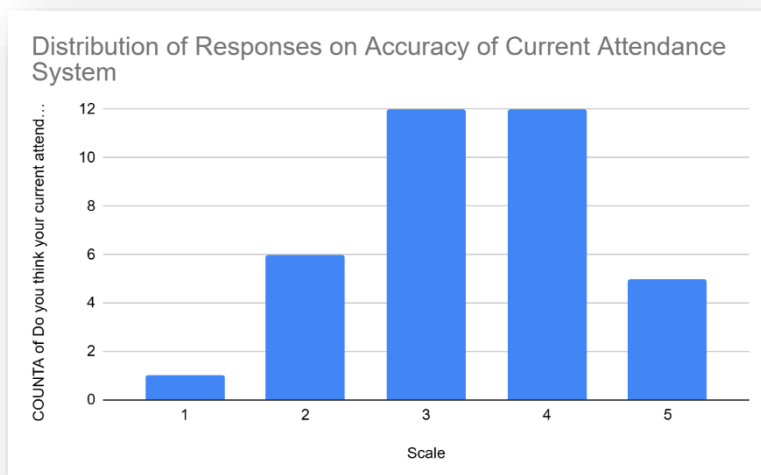


Figure 3.9: Various levels of concurrence on whether the existing attendance system accurately tracks attendance and presence.

As seen in the Figure 3.9, the responses reveal various levels of concurrence on whether the existing attendance system accurately tracks attendance and presence. A total of 12 respondents (24%) gave the maximum possible rating of 5, indicating strong agreement. Similarly, another 12 respondents (24%) gave it a rating of 4, indicating agreement. Meanwhile, 8 respondents (16%) awarded it a 3, indicating a neutral view. Fewer respondents, four (8%), rated it two, and only one (2%) gave the lowest grade of one, indicating a significant disagreement. This distribution shows that, while the majority of respondents have a positive perception of the system, there are some noticeable neutral and critical opinions that should be addressed.

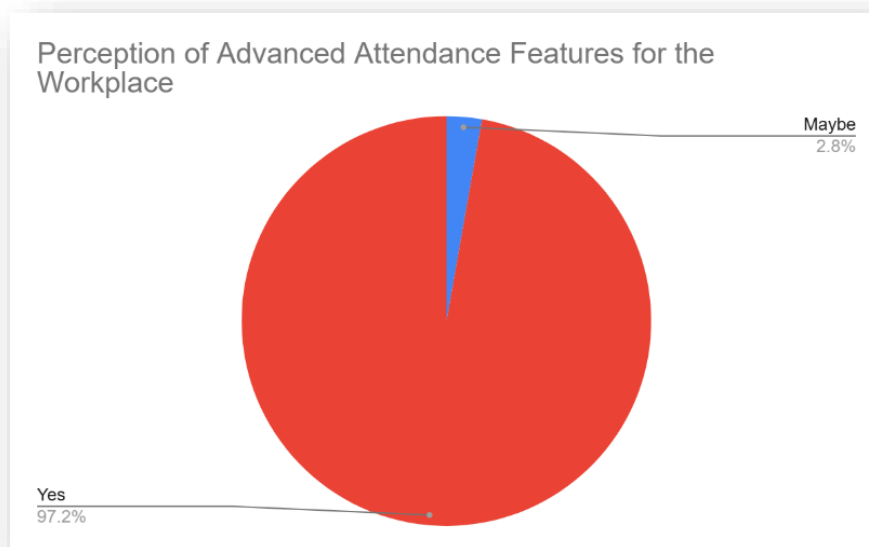


Figure 3.10: Perception of Advance Attendance Features for the workplace.

Based on Figure 3.10, the chart shows that a large percentage of respondents (97.2%) support advanced attendance features at work. Only a small number of respondents indicated uncertainty ("Maybe"), and none opposed these aspects. This reflects a strong desire to upgrade workplace attendance systems using new technologies.

3.3.3.3 Usability Design and Preference

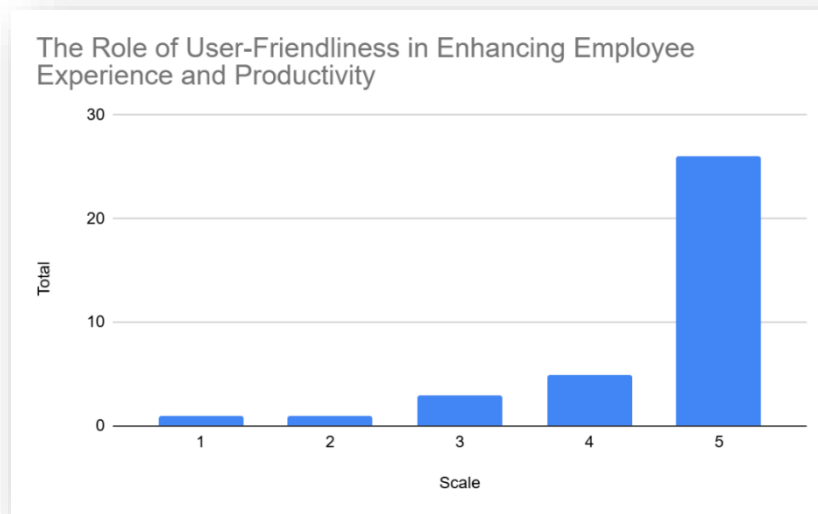


Figure 3.11: The Role of User-Friendliness in the proposed Attendance System.

Figure 3.11 illustrates the chart depicts respondents' evaluations on the relevance of user-friendliness in attendance systems, from 1 to 5. The majority of participants evaluated it as highly important (5), with a minority giving lower scores. These findings highlight the importance of intuitive and user-friendly interfaces in promoting positive employee experiences and productivity.

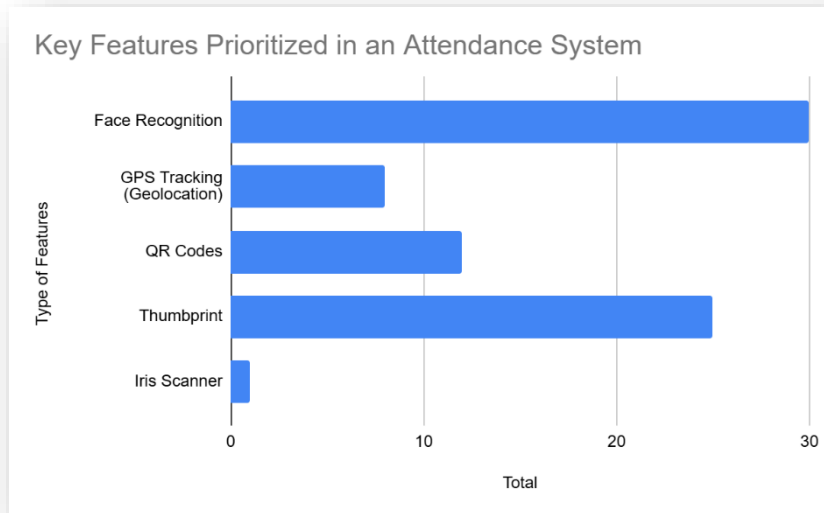


Figure 3.12: Key Features Prioritized in an Attendance System.

Figure 3.12 displays the bar chart shows the aspects that respondents value most in attendance systems. Face recognition and thumbprint scanning were the most preferred options, followed by geolocation (GPS tracking) and QR codes. GPS tracking may be less prevalent than biometric methods because of privacy concerns, battery concerns, and accuracy issues in indoor settings. Employees may reluctant to having their whereabouts tracked all the time out of concern that their information may be misused. Additionally, due to ethical and regulatory concerns, certain companies may not support GPS-based tracking, especially those with strict privacy policies. Additionally, GPS tracking may not be as accurate indoors, particularly in high-rise office buildings where signal quality may be compromised. GPS is still essential for confirming attendance in spite of this, especially for employees who work remotely or in the field where biometric methods might not be enough. Concerns regarding fraudulent clock-ins and attendance manipulation can be allayed by combining GPS with biometric verification to improve security and guarantee that workers are physically present at the designated location. The least popular option was iris scanning. These findings emphasise the need for biometric and geolocation-based features to improve accuracy and security.

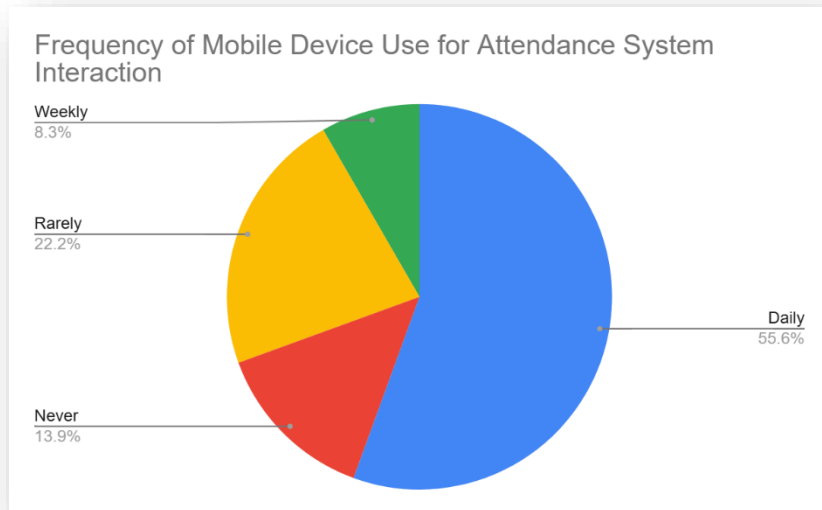


Figure 3.13: Frequency of Mobile Device Use for Attendance System Interaction.

Figure 3.13 shows how frequently respondents engage with the attendance system using mobile devices. The majority, 55.6%, indicated everyday usage, indicating a rising reliance on mobile technologies for attendance tracking. Meanwhile, 22.2% use it infrequently, and 13.9% never use it, indicating potential limitations in accessibility or adaption to mobile technologies.

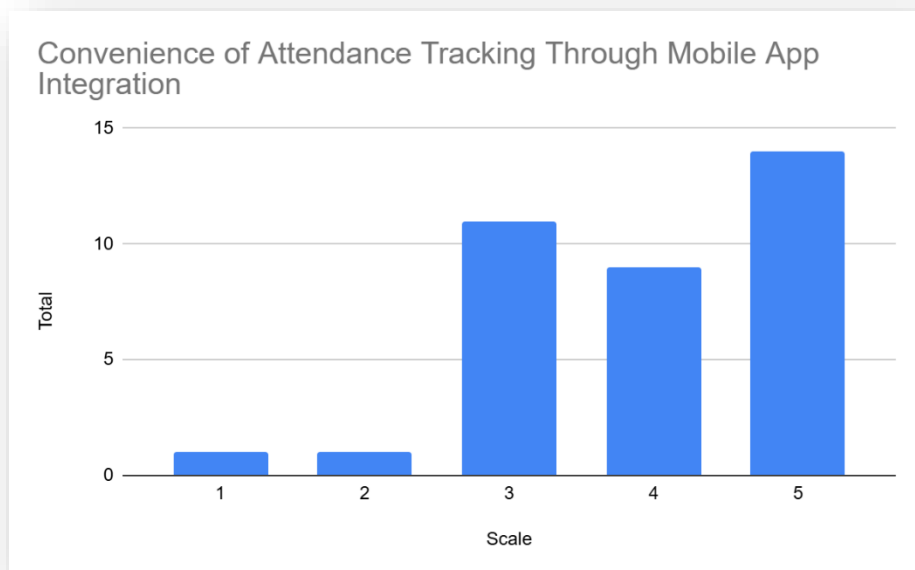


Figure 3.14: Convenience of Attendance Tracking via Mobile App.

Figure 3.14 shows the bar chart that evaluates the perceived ease of attendance tracking using mobile apps. A significant percentage of respondents rated the experience positively, with 5 being the most common rating. This suggests that mobile integration is largely viewed as a convenient method for managing attendance, however a small percentage assessed it less favourably, indicating that the user experience may be improved further.

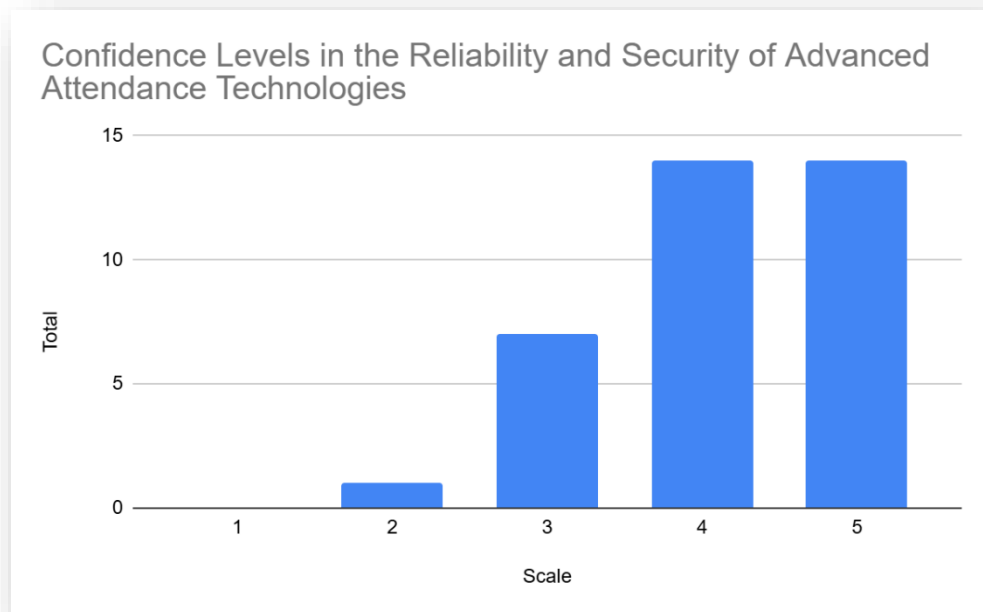


Figure 3.15: Confidence Levels in the Reliability and Security of Advanced Attendance Technologies.

Figure 3.15 depicts a chart that represents respondents' trust in the dependability and security of advanced attendance systems. Most people ranked their confidence at 4 or 5, indicating a high belief in these systems. Lower scores were less prevalent, showing that while security and reliability are not big concerns for the majority of users, they are still important issues to solve for some.

3.3.3.4 Additional Feedback

The final section of the survey included an open-ended feedback section with two questions, the first asked respondents to share any concerns they had about the proposed attendance system, and the second asked them to make suggestions for improving the system's features. Regarding the first question, The majority of respondents stated no concerns, suggesting a positive attitude towards the system. However, other respondents voiced serious concerns, including privacy concerns, such as fears of data breaches and personal information fraud. Technical issues were also raised, such as biometric detection failures (e.g., thumbprint not detected), system problems, and possible server overload during peak usage. Additionally, concerns concerning geolocation accuracy and ethical consequences, as well as reliability in bad network conditions, were raised. Several respondents expressed concern about system downtime and asked for information about backup strategies for dealing with failures. Fairness in addressing punctuality and tardiness was also emphasised. These observations, summarised in Table 3.1, are useful for enhancing the proposed system.

Table 3.1: Respondents' Concerns with Concluded Concerns.

Respondents' Concerns	Concluded Concerns
"Positive feedback with no concerns"	Most responders stated no worries, indicating a positive response to the system.
"Privacy concerns / Personal information gets leaked"	Some responders express serious concerns about privacy, including fears about data leaks.
"Sometimes my thumbprint fails to be detected"	Concerns regarding biometric detection failures (e.g., thumbprint is not identified).
"System errors"	General concerns regarding system dependability and errors.
"Quite unsure with the geolocation feature"	Concerns regarding the precision of geolocation tracking.

“Accuracy and Reliability: Biometric or GPS systems might fail in certain conditions”	Technical concerns with system dependability and accuracy in real-world scenarios.
“Server overload. Might cause server to slow down when multiple users are using it.”	Concerns regarding system performance under high loads (e.g., server overload).
“The system down”	Concerns about interruptions in systems and their impact on operations.
“What is the backup plan if the proposed attendance system is malfunctioning?”	Questions about contingency and backup strategies in the case of system failures.
“Unfair people coming late while others are on time”	Concerns about justice in addressing timeliness and punctuality.

As for the second question, A large majority of respondents highlighted the importance of user-friendly features, stating that the system should be intuitive and simple to use, possibly with guides or assistance choices. Some responses suggested combining fallback authentication mechanisms, such as face recognition, fingerprint, QR codes, or access cards, to mitigate probable failures in any one approach. Other suggestions included providing offline attendance recording with data syncing, adding notifications, multilingual support, and a customisable interface to improve accessibility and usability. Suggestions also emphasised the importance of system dependability, such as minimising downtime and facilitating a faster login process. A few responders suggested including unique features to set the system apart from ones already in use in the workplace. Overall, while many respondents were pleased with the proposed system, the ideas summarised in Table 3.2 offer useful insights for improving its features and functionalities.

Table 3.2: Respondents' Suggestions with Concluded Suggestions.

Respondents' Suggestions	Concluded Suggestions
“User-friendly features / Attendance system that is easy to use / Providing guides or help options”	Emphasise user-friendly design and give guides or assistance choices to make things easier to use.

“Faster login process”	Optimise the system to provide a quick . and easy login process.
“Include notifications, offline mode, multi-language support, and a customizable interface”	Add notifications, turn on offline attendance recording with data syncing, multilingual support, and customisation.
“Making sure the system never goes down “	Ensure superb system dependability and minimal downtime.
“Using an access card every day every hour”	Explore additional authentication methods such as access cards.
“Since it's already being used in most workplaces, try to add new unique features”	To set the system apart from existing solutions, innovate and provide unique features.
“Enable offline attendance recording, with data syncing once the system reconnects to the network”	Support offline functionality to improve system resiliency in low-connect scenarios.

3.3.4 Software Requirements

The proposed mobile application's software requirements are thoroughly described in Section 2.4 - Review of Tools and Technology (Chapter 2, 2.4.1 - Software). Table 3.3 gives a quick summary of the software components that will be used in the creation of the proposed mobile application.

Table 3.3: Software Requirements for the proposed system.

Software	Justifications
React Native	A JavaScript framework that facilitates development of the Android mobile app using a single codebase. It provides native visualization and enables flexible design of both the user interface and app functionality.
Android Studio	An integrated development environment (IDE) used after Expo ejection to set up native Android files and create the Android App Bundle (AAB) for deployment. It acts a crucial role in the packaging and release phases.
Expo	A development platform applied in the early phases for testing and previewing the app. It

	enhances integration with device features like geolocation and biometrics before switching to native code.
Visual Studio Code	The principal code editor applied throughout development. It simplifies JavaScript and React Native programming with beneficial extensions, integrated Git support, and built-in terminal connection.
Supabase	A backend-as-a-service platform utilized to manage authentication, database operations, and file storage. It offers real-time APIs and SQL-based management, equipped with reliable integration for React Native.
PostgreSQL	An open-source relational database that operates as the core of Supabase. It provides structured, SQL-based data storage and enables real-time functionality, advanced queries, and reliable data integrity processes.
Firebase Cloud Messaging (FCM)	A cloud messaging platform used to send real-time push notifications to mobile users. It supports background message sending and integrates with React Native through third-party libraries.
Google Chrome	The Google Chrome web browser is extensively employed during the development of the mobile application to conduct review on essential information and resources. In addition, it functions as a testing platform for assessing the application's functionality and user interface on devices with varying screen resolutions.

3.3.5 Hardware Requirements

The hardware requirements for the mobile application are detailed in Section 2.4, under 2.4.2 - Hardware. Table 3.4 summarises the hardware components for developing the backend system while Table 3.5 summaries the hardware components for the Proposed Android Application. Hardware Components for developing the backend system:

Table 3.4: Hardware Requirements for the Back-end System.

Reason	Justifications
Processor Power	Providing considerable processing capabilities for development projects,

	including modest graphical activities, the AMD Ryzen 5 7430U with Radeon Graphics is a solid option.
Ample RAM	16 GB of RAM allows for smooth multitasking and the simultaneous execution of many apps, which is critical for software development and testing
Modern OS	A user-friendly interface, effective task management, and access to the newest features and updates are all provided by Windows 11 Home Single Language.
Cost Efficiency	Provides high performance at a competitive price, making it affordable for students and developers with limited financial resources.
Reliability	Designed to handle common computing activities in an effective manner, ensuring consistent performance for extended use in situations that are considered to be development environments.

Hardware components for the Proposed Android Application:

Table 3.5: Hardware Requirements for the Proposed Android Application.

Reason	Justifications
Powerful Processor	Using the Kirin 980 chipset with an octa-core CPU, the device executes real-time location tracking and QR scanning smoothly
Biometric Support	Equipped with an under-display fingerprint scanner, supporting efficient testing of biometric features.
Display Quality	Offering a 6.47-inch OLED screen, it offers the detail and area necessary for user interface testing and verification
Android Compatibility	Built on Android 10 with EMUI 11, ensuring compatibility with React Native, Expo, and the up-to-date development SDKs.
Camera Capabilities	Provided with a Leica Quad Camera setup, ideal for evaluating camera-based functions like QR code scanning.
Portability	Its lightweight yet sturdy design enables for smooth testing in office and remote settings
Battery Life	With a 4200mAh battery, the device supports long testing periods and ongoing engagement with the app.

3.4 User Design

The user design phase of the RAD methodology emphasises a collaborative and iterative process between users and developers to generate prototypes and mock-ups of the system's user interface (UI) and functionality. Once the requirements have been identified, the development process begins. During this phase, developers quickly design a number of prototypes with varied features and functionality. Iterative design process, a critical component of RAD, has been proved to improve the usability of a product (Bailey, 1993). This phase guarantees that the system meets user expectations through constant input and improvement, offering a clear picture of the end result and building the groundwork for succeeding development stages.

A clear illustration of this iterative approach is reflected in the development of the GPS (Geolocation Tracking) interface within CorePresence. This component progressed through three major stages which are low-fidelity prototyping, functional implementation, and visual refinement with each phase shaped by supervisor feedback and informal user testing:

1. Initial Prototype (Low-Fidelity Wireframe):

Created using Figma, this version provided a basic map layout with a simple direction button. Early feedback highlighted that the map lacked clarity and failed to indicate employee proximity effectively.

2. Intermediate Version (Post-Feedback Testing Stage):

Rebuilt in React Native with Google Maps integration, this version introduced live user location tracking. However, after receiving feedback from the supervisor, it became clear that the interface still lacked visual differentiation, making it difficult for users to identify who was within or outside the designated location range.

3. Final Interface (After Iteration and Validation):

The final version introduced significant improvements:

- Categorised employees as either “Within 1km” or “Outside 1km”
- Integrated color-coded visual cues (green and red)
- Displayed exact distances, names, faculty, and profile pictures for easy identification

This three-stage design process demonstrates RAD's strength in managing continuous feedback loops and iterative refinement, particularly in user-centric modules like the GPS Geolocation page. Supporting artefacts from each stage of the iterative design process including the low-fidelity wireframe, intermediate prototype, and final validated interface are provided in Appendix B.

3.4.1 Overall System Architectural Design

System architecture defines the general framework of a system, outlining the organisational structure of its components or modules and their interactions to fulfil the system's objectives. It functions as a framework for the design and development of the system, guaranteeing the seamless integration and operation of all components. This architecture delineates the interconnections among the different layers, components, and data flows, while also promoting a thorough comprehension of the system's behaviour and its interaction with external entities. Figure 3.16 illustrates the overall system architectural design, highlighting the three-tier structure and the flow of data between system components.

The Multi-Modal Authentication and Geolocation Approach for Attendance Verification comprises three tiers: the User Interface Layer, the Application Layer, and the Data

Layer. Every layer comprises distinctive features and tasks, enhancing the system's overall usefulness and efficiency.

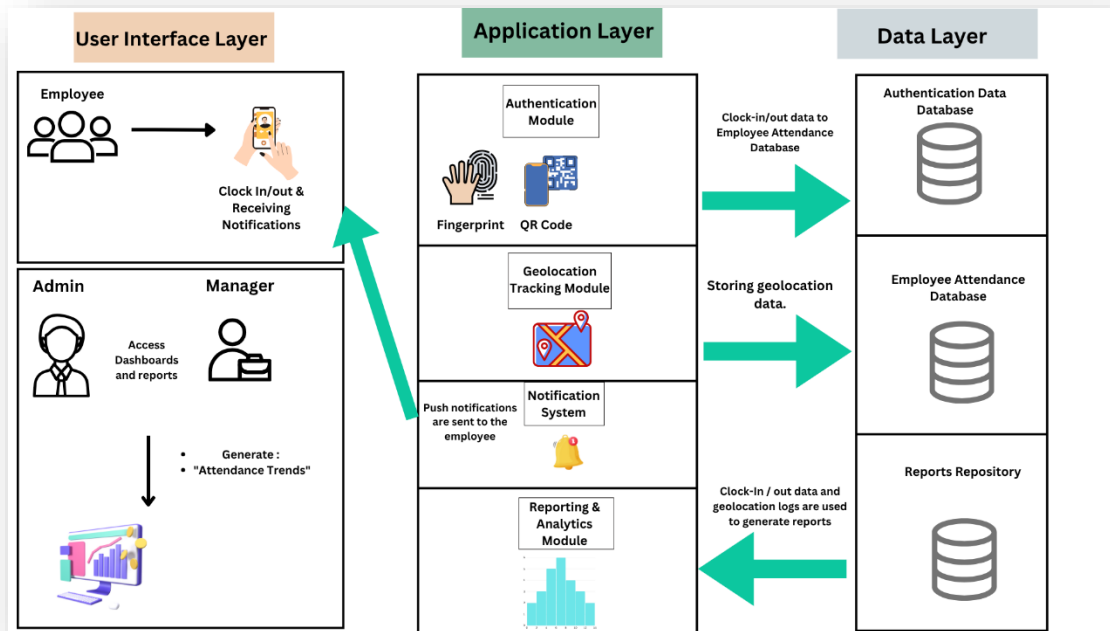


Figure 3.16: Overall System Architectural Design.

User Interface Layer:

This layer facilitates user interaction with the system. It encompasses:

1. **Employee Module:** Employees utilise mobile devices to record their attendance and receive notifications or prompts for attendance verification, such as location requests.
2. **Admin and Manager Modules:** Administrators and managers utilise dashboards and reports to oversee attendance patterns and evaluate staff productivity.

Application Layer:

This is the fundamental layer of the system where essential processes and operations transpire. It comprises:

1. Authentication Module: Facilitates multi-modal authentication techniques, including fingerprint scanning and QR codes, to verify employee attendance.
2. Geolocation Tracking Module: Monitors the employee's location inside a specified geofenced zone to verify workplace attendance.

Notification System: Delivers push notifications to employees for regular attendance verification and other alerts concerning attendance status.

3. Reporting & Analytics Module: Evaluates clock-in/out data and geolocation records to produce detailed statistics on attendance patterns.

Data Layer:

This layer oversees the storage and administration of all system data. It encompasses:

1. Authentication Data Repository: Maintains logs of authentication events, encompassing information on both successful and unsuccessful authentication attempts.
2. Employee Attendance Database: Preserves records of clock-in and clock-out timestamps together with geolocation information.
3. Reports Repository: Preserves the reports produced by the Reporting & Analytics Module for subsequent access by administrators and managers.

Data Flow

The data flows easily between layers:

1. The application layer processes authentication data, which is then saved in the Authentication Data Database.
2. Clock-in/out and geolocation data are routed to the Employee Attendance Database for real-time monitoring and storage.
3. The Reporting & Analytics Module collects attendance data to provide insights and trends, which are then saved in the Reports Repository for admin and manager access.

3.4.2 Flowchart

A flowchart is a diagram that illustrates an algorithm, workflow, or process, depicting the steps as variously shaped boxes and indicating their sequence through connecting arrows (Okafor & Odiliobi, 2021). Flowcharts improve clarity and understanding by visually depicting the process flow, making them an effective tool for system analysis and design.

Figure 3.17 delineates the workflow for the proposed system, detailing the process from the initiation of user interaction to the final output. The steps outlined in the flowchart are as follows:

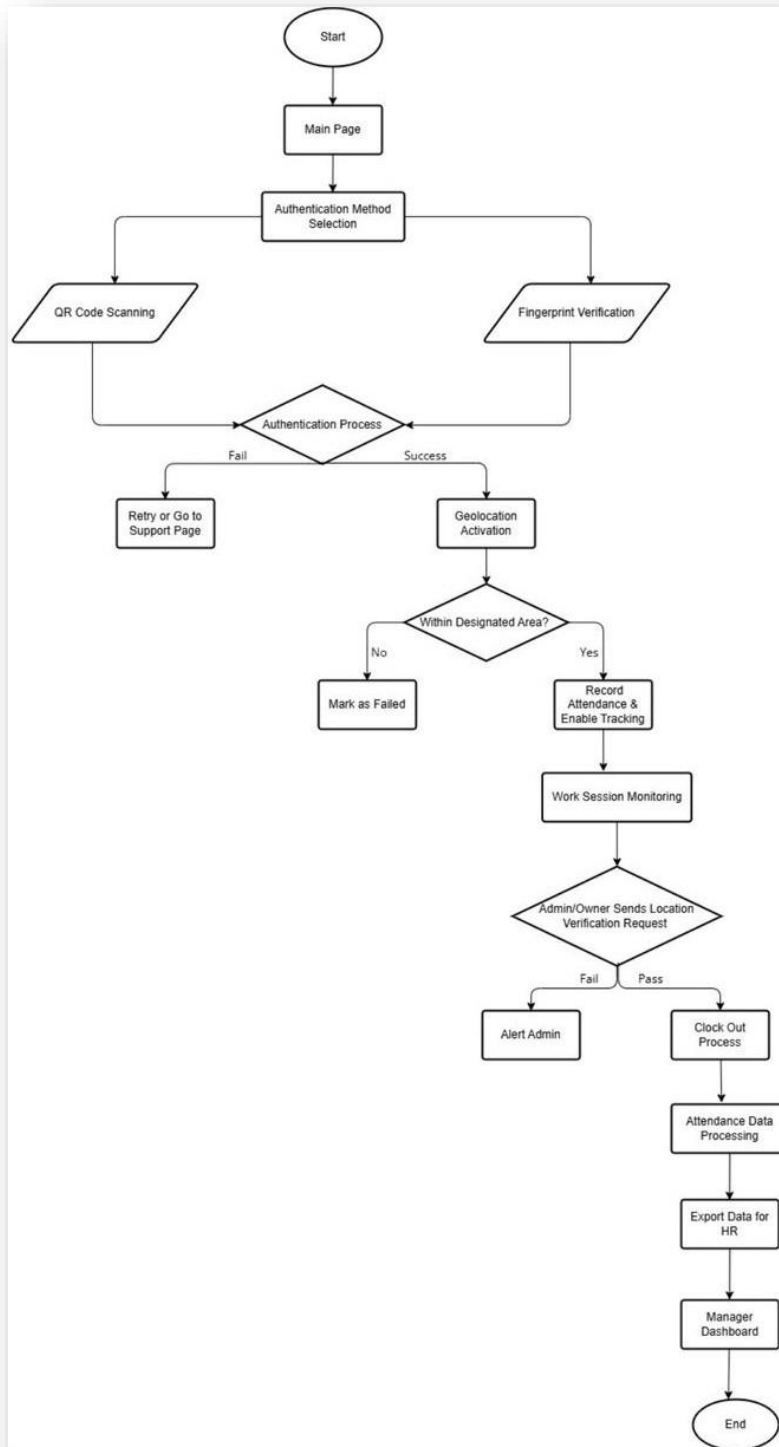


Figure 3.17: Flowchart of the proposed system

1. **Start:**

The system begins its operation, initializing the necessary features and interfaces to guarantee seamless functionality for the user.

2. **Home Page:**

The user is presented with the home page, which acts as the starting point for navigating the authentication system. It offers alternatives for implementing different authentication methods.

3. **Selection of Authentication Method:**

The user must select one of the two available authentication methods:

- i. The user scans a QR code for identity validation.
- ii. Fingerprint Verification: The individual submits a fingerprint scan to authenticate their identity.

4. **Authentication Process:**

The system executes the selected authentication method. Upon successful completion of the process, the user proceeds to geolocation verification. In the event of a failure, the system offers the user a chance to retry or notifies the system administrator for subsequent action.

5. **Geolocation Activation:**

The system initiates the geolocation feature to confirm the user's presence within the specified area. This step is essential to verify that attendance or activity is documented from an authorized location.

6. **Designated Area Verification:**

The system verifies whether the user's geolocation corresponds to the designated area:

- i. In the absence of the user's presence in the authorized location, the system categorizes the attempt as unsuccessful, thereby restricting access or further advancement.

- ii. Upon the user's presence in the designated area, the system logs their attendance and initiates tracking to oversee their engagement throughout the work session

7. **Work Session Monitoring:**

The system provides continuous monitoring of user activity throughout the work session. Near the session's end, the system prompts an Admin or Manager to trigger a location verification request to verify the user's continued presence and participation.

8. **Facial Recognition Prompt:**

The system features a manual verification process where the admin or manager sends a location request to the user. This prompt validates the user's current location prior to allowing them to clock out:

- i. If the user's location does not match the specified area or they fail to respond, the system notifies the admin for further action.
- ii. If the user passes, they can proceed with the clock-out process.

9. **Clock Out Process:**

The system marks the user as clocking out once their session has officially finished. This allows for precise reporting of their work hours.

10. **Attendance Data Processing:**

The system converts the acquired attendance data into a structured format for administrative usage.

11. **Export Data for HR:**

The processed attendance data is exported in a format that the HR department may analyse and integrate into payroll or performance evaluation systems.

12. **Manager Dashboard:**

The processed data, including attendance details and employee's attendance records, is shown on the manager's dashboard for convenient monitoring and decision-making.

13. **End:**

The procedure is completed successfully, indicating that all needed steps were completed and that the data was properly captured and transferred.

3.4.3 Context Diagram

The context diagram, as depicted in the Figure 3.18, which is also referred to as a Level 0 DFD, is the most prominent representation in a data flow diagram. It emphasises a singular process that serves as a representation of the entire system and defines its boundaries. The initial phase in the process of creating a collection of Data Flow Diagrams (DFDs) is the development of a context diagram. The exchange of information between the system and its external entities is delineated in this diagram. It provides a clear and concise representation of the system's interactions with its environment.

The context diagram for the Multi-Modal Authentication and Geolocation Approach for Attendance Verification is depicted in the diagram below. The proposed mobile application, denoted as Process 0, enables the monitoring and verification of attendance through the use of advanced features, including geolocation, QR codes, and biometric data. Employees, administrators, and managers comprise the external entities that interact with the system. These entities and the central process are connected by data flows.

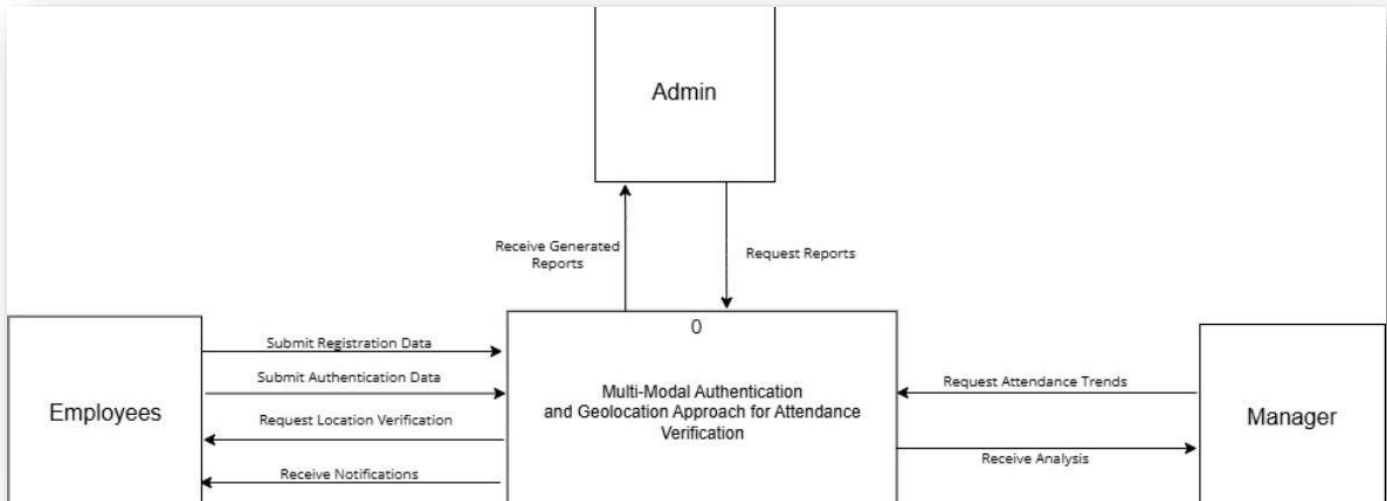


Figure 3.18: Context Diagram.

Six data flow movements occur between the system and employees:

1. Employees are required to register and establish their accounts within the system.
2. Requests for clock-in/clock-out have been submitted by employees.
3. Employees provide authentication data, including fingerprints or QR codes.
4. The system sends prompts to employees when admins or managers initiate location verification.
5. The system sends notifications to employees to request periodic verification via Firebase Cloud Messaging (FCM).
6. Employees receive location-based notifications from the system.

There are two data transit movements between the system and the administrator:

1. Attendance reports are exported by administrators.
2. Admins/manager manually trigger location verification requests to confirm user presence ahead of clock-out.

The system and Manager undergo two data transmission movements:

1. Real-time attendance data is requested by managers.
2. Attendance trends and geolocation data are transmitted to managers.

3.4.4 Data Flow Diagram (DFD)

A data flow diagram depicts the processes, data repositories, and external entities within a business or system, along with the interconnecting data flows (Aleryani, 2016). It offers a visual representation of data flow inside a system, facilitating the identification and comprehension of the various components and their interconnections. Data Flow Diagrams (DFDs) model information movement and are essential for the design and analysis of information systems, facilitating clear communication of system needs and logical structure to both technical and non-technical stakeholders.

3.4.4.1 Data Flow Diagram Level 0

The Data Flow Diagram (DFD) Level 1 provides a comprehensive depiction of the primary functions of the proposed Multi-Modal Authentication and Geolocation Approach for Attendance Verification System, which utilises multi-modal authentication and geolocation in office settings. In contrast to a context diagram, a Level 0 DFD disaggregates the main process into sub-processes, emphasising the interactions among users, data flows, and data stores. This level of detail illustrates the operation and interconnection of each function, providing a clearer understanding of the system's internal processes.

The Level 0 DFD for the proposed system includes five key processes: Register Account, Authentication Handling, Attendance Recording, Data Analysis, and Report Generation. Furthermore, the system interacts with five data stores, namely User Data File, Authentication Logs, Attendance Logs, Geolocation Data, and Reports.

Employees, administrators, and managers are the primary external entities that are involved in the utilisation of the system's features. Each of these entities has a distinct role to perform.

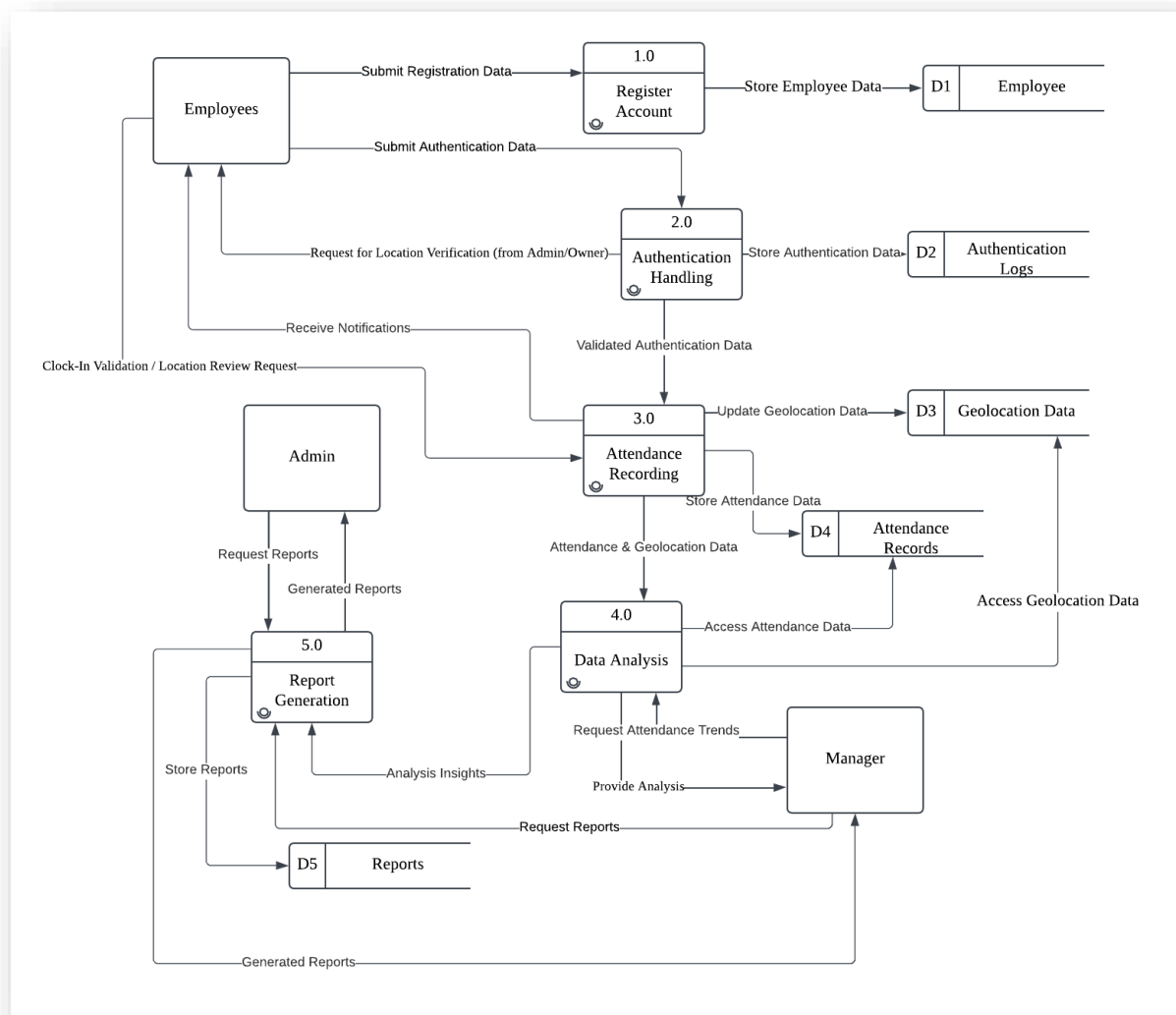


Figure 3.19: Data Flow Diagram (DFD) Level 0 of the Multi-Modal Authentication and Geolocation Approach for Attendance Verification System.

Based on the Figure 3.19, The processes are explained in detail as follows:

Process 1.0: Register User

Account creation is mandatory for employees to gain access to the mobile application. In this process, registration data including name, employee ID, and contact information is submitted. The system validates this data to ensure its accuracy. Upon validation, the information is recorded in the D1 Employee for subsequent reference. Upon successful registration, the account setup is done.

Process 2.0: Authentication Handling

This process enables employees to verify their authentication through QR code scanning or fingerprint verification. Upon successful verification, system access is granted, and all authentication attempts regardless of success are logged with timestamps in the D2 Authentication Logs for ongoing tracking and security purposes.

Process 3.0: Attendance Recording

Employees record attendance by clocking in and out via the mobile application. In addition to clock-in/out data, geolocation information is collected to verify the employee's presence at the designated workplace. The data is validated and stored in the D4 Attendance Records and D3 Geolocation Data for precise tracking. Furthermore, the system enables admin or owner-triggered location verification during ongoing sessions. If the verification is unsuccessful, the system can alert the admin for further review.

Process 4.0: Data Analysis

The system evaluates attendance and geolocation data to produce insights that assist administrators and managers in monitoring workplace trends. Data from D4 Attendance Records and D3 Geolocation Data is analysed to identify patterns including absenteeism, and other behavioural trends. These insights are stored for further processing and utilised to improve decision-making by identifying areas for enhancement in workforce management.

Process 5.0: Report Generation

The report generation process compiles attendance data from Attendance Records (D4) and Employee (D1) to generate structured attendance reports. These reports include fields such as full name, department, and attendance status. The system exports this data as CSV files, enabling managers and administrators to evaluate employee attendance in a clear and straightforward format. Finalized reports are saved in the Reports File (D5) and can be generated from time to time or when needed to assist with HR documentation, evaluations, and decision-making.

3.4.4.2 Data Flow Diagram Level 1

The Data Flow Diagram (DFD) Level 1 provides a more extensive analysis of the processes described in the context diagram and DFD Level 0, allowing for a better understanding of the system's internal activities. It divides the central system into five basic processes: registration, authentication, attendance data management, report analysis and trend generation, and report generation. Each procedure is intended to solve a certain system functionality, such as user registration and secure authentication, attendance management, and

thorough report generating. These procedures interact with a variety of data stores, including User Data Files, Authentication Logs, Attendance Records, and Reports, to ensure smooth data flow and storage. DFD Level 1 illustrates these interactions, providing a clear visual representation of how the system processes information to meet its functional requirements.

DFD Level 1 for Process 1.0

Figure 3.20 depicts the Level 1 Data Flow Diagram for Process 1.0, which refers to Register User.

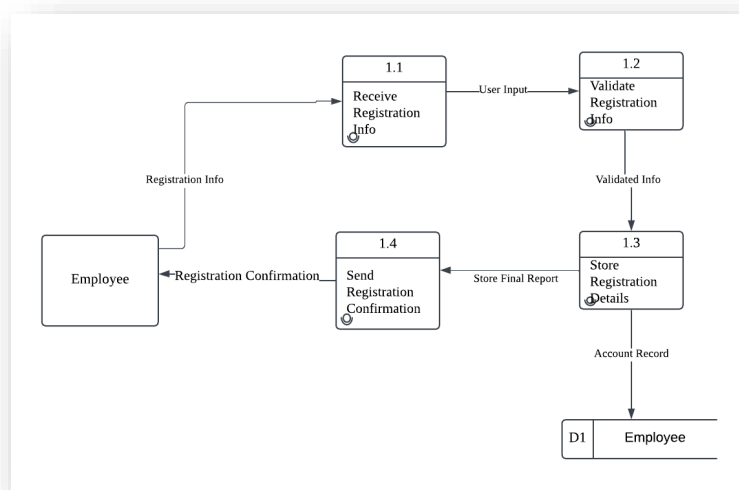


Figure 3.20: Data Flow Diagram Level 1 for Process 1.0.

The registration process allows employees to create accounts in the system. Employees begin by entering their personal information, such as name, email, and contact information, which is processed and temporarily saved for validation. The system then notifies the administrator, who can analyse the given data and make a decision about the registration. Based on the choice, the system updates the employee's new information in the Employee (D1).

DFD Level 1 for Process 2.0

Figure 3.21 depicts the Level 1 Data Flow Diagram for Process 2.0, which refers to Authentication Handling.

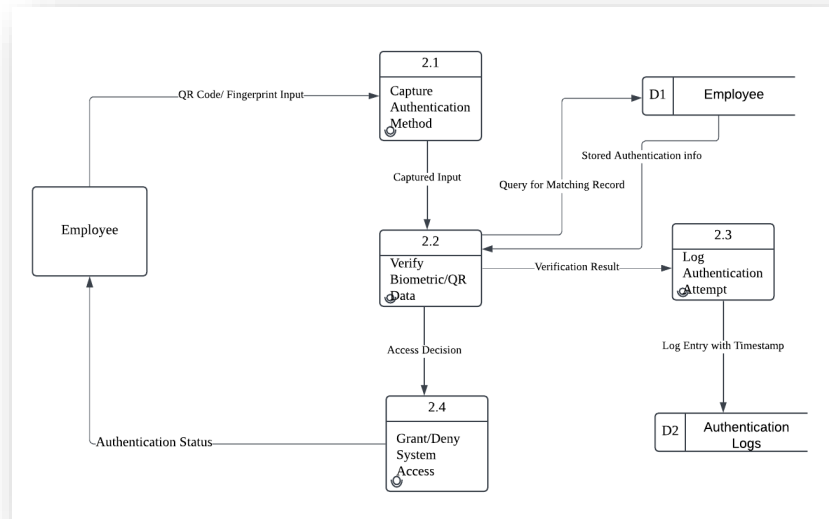


Figure 3.21: Data Flow Diagram Level 1 for Process 2.0.

The authentication process supports secure system access via multi-modal verification methods, including QR code scanning and fingerprint authentication. Employees initiate authentication through the mobile application by submitting either a scanned QR code or biometric fingerprint. The system captures this input and validates it by comparing the submitted data against existing records in the Employee (D1). All authentication attempts whether successful or failed are logged in the Authentication Logs (D2), complete with timestamps for audit and security tracking. Upon successful verification, the system grants access and communicates the result back to the employee via the mobile interface. If authentication fails, the employee is notified accordingly.

DFD Level 1 for Process 3.0

Figure 3.22 depicts the Level 1 Data Flow Diagram for Process 3.0, which refers to Attendance Recording.

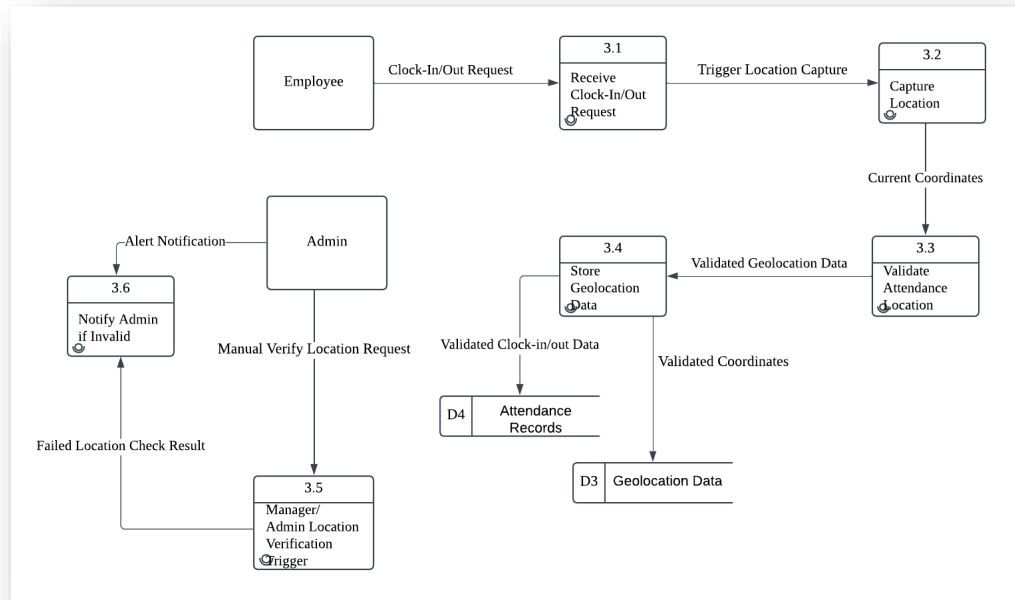


Figure 3.22: Data Flow Diagram Level 1 for Process 3.0

Process 3.0 facilitates employees clocking in and out via the mobile app. The system captures GPS coordinates to validate the user's presence at the correct location. Once verified, the data is stored in both D3: Geolocation Data and D4: Attendance Records. Additionally, admins can trigger manual location checks during active sessions. If the user's location is invalid, the system notifies the admin for further action. This approach enhances accuracy and ensures accountability in the attendance process.

DFD Level 1 for Process 4.0

Figure 3.23 depicts the Level 1 Data Flow Diagram for Process 4.0, which refers to Data Analysis.

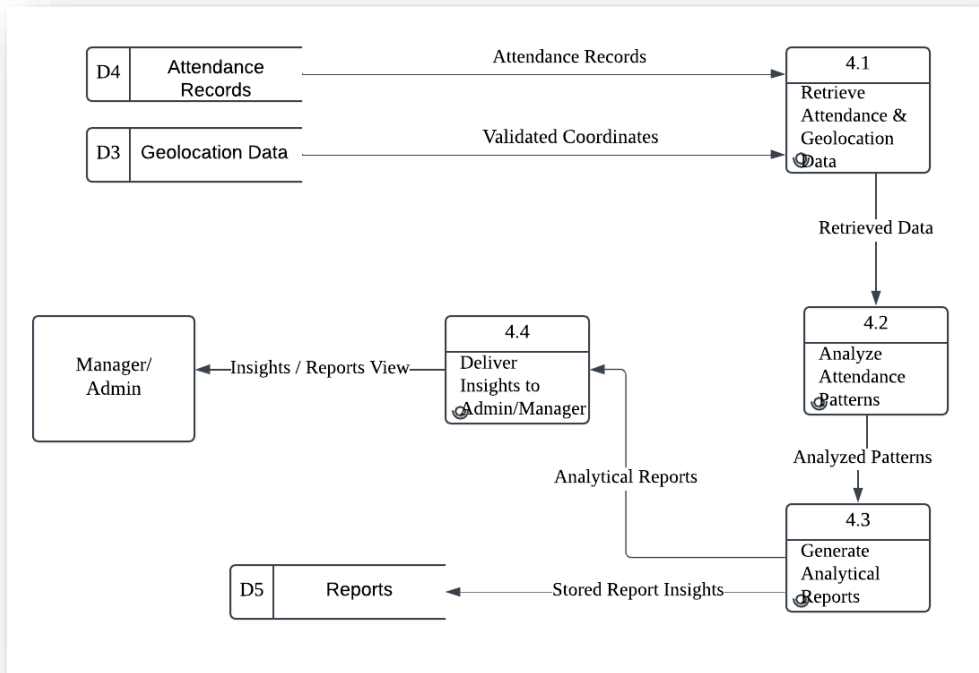


Figure 3.23: Data Flow Diagram Level 1 for Process 4.0

This process involves retrieving attendance and geolocation records from the system's database to uncover trends such as absenteeism and punctuality. The collected data is analysed to generate insightful reports that help identify behavioural patterns among employees. These reports are then provided to administrators and managers to support informed decision-making and improve overall workforce management.

DFD Level 1 for Process 5.0

Figure 3.24 below depicts the Level 1 Data Flow Diagram for Process 5.0, which refers to Report Generation.

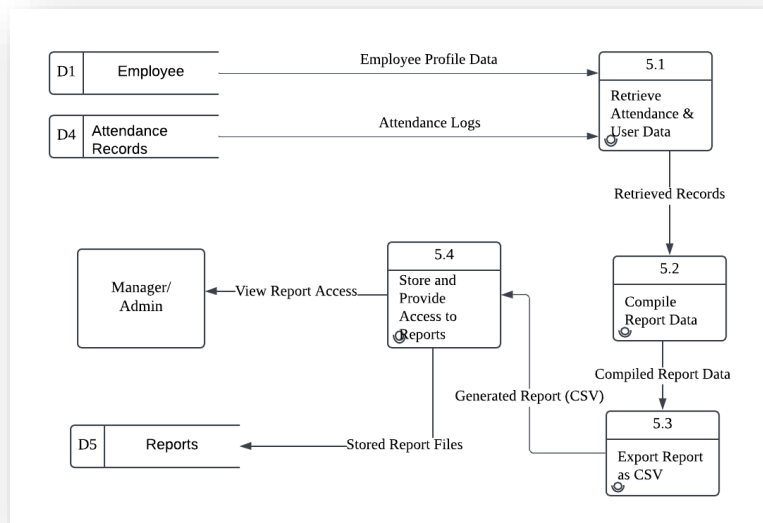


Figure 3.24: Data Flow Diagram Level 1 for Process 5.0

This process is broken down into four key activities. First, the system retrieves necessary attendance and user information (5.1). Then, it compiles this data into a structured report format containing details such as name, department, and attendance status (5.2). After compilation, the report is exported in CSV format (5.3) and stored for future access (5.4). Admins and managers can access or download the final reports from the system as needed for evaluations and documentation.

3.4.5 Entity Relationship Diagram (ERD)

A vital component in database modelling, the Entity Relationship Diagram (ERD) offers a graphical depiction of a system's entities and their interconnections. This tool is fundamental in crafting relational databases by showcasing the logical framework and linkages among tables. The ERD technique produces a detailed model that includes relationship names, cardinality, and optionality, as well as attributes and permitted values for each entity (Brady & Loonam, 2010). In an ERD, entities correspond to tables in a relational database, with each

entity signifying a unique dataset. Attributes, on the other hand, define the characteristics or properties of these entities, functioning as table columns. The relationships between entities demonstrate how datasets are connected, providing insights into the system's relational dynamics. This ensures that the database design aligns with functional requirements and serves as a foundation for effective implementation.

The ERD for the proposed System is displayed in Figure 3.25. Crow's Foot notation is used in the ERD to succinctly and clearly represent entities, properties, and relationships. The fundamental database structure needed to implement the system is shown in this diagram.

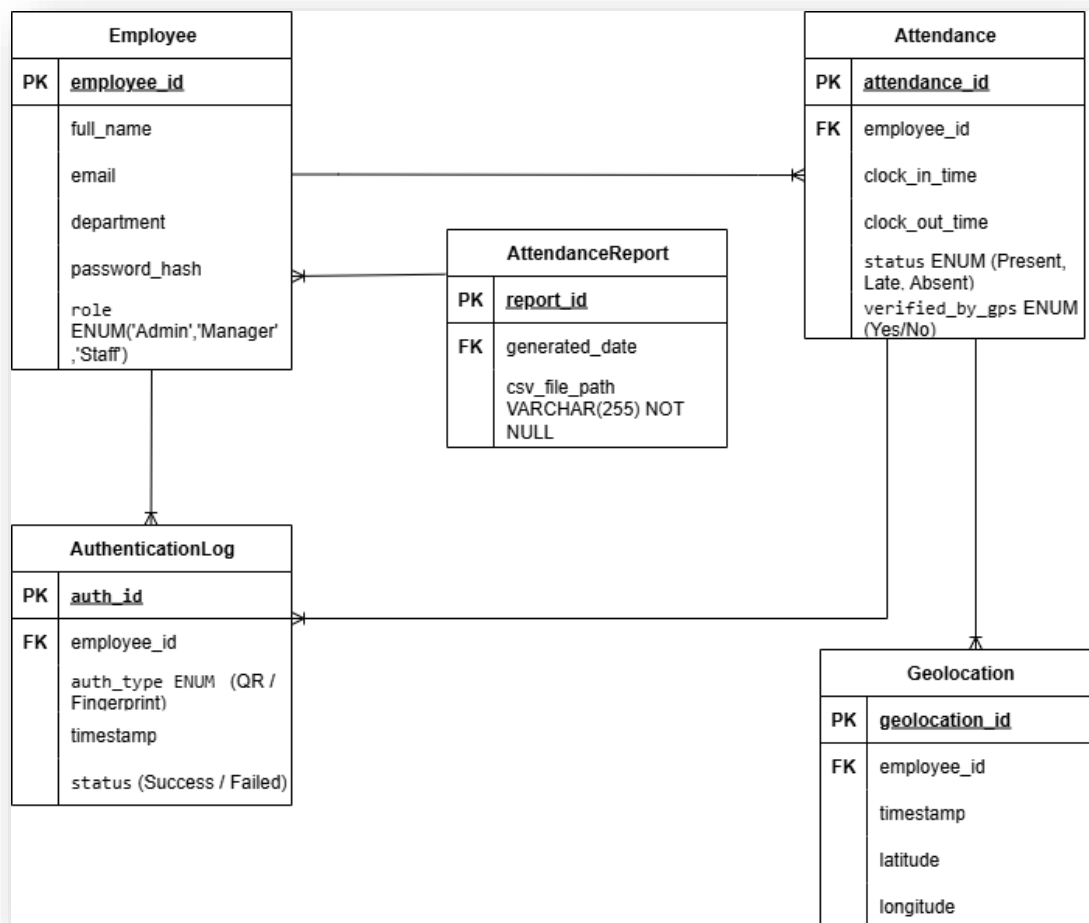


Figure 3.25: Entity Relationship Diagram (ERD).

Key Entities in the Database Design:

1. Employees
2. Attendance
3. AuthenticationLog
4. Geolocation
5. Report

Database Relationships:

The relationship between the entities specifies how the data in the system is related to one another:

1. Each employee can have multiple authentication attempts, such as QR code scans or fingerprint entries.
2. Each employee can generate multiple attendance records, corresponding to their clock-in and clock-out sessions.
3. Each attendance record may be linked to geolocation data, verifying that the employee was physically within the designated radius during clock-in/out.
4. Reports may be generated by employees with admin or manager roles. The system supports tracking the generator of each report through a generated_by field.
5. Employees with the role of Manager may be logically associated with one or more employees through supervisory responsibilities, though this relationship may be implicit (e.g., via department matching) unless modeled with a separate junction table.

Business Rules:

1. There are several ways for an employee to authenticate (e.g., Thumbprint, QR code).

2. Role-Based Access Control (RBAC) is implemented through the role field in the Employee table, which defines user privileges such as report generation and access to filtered attendance data for Admins and Managers.
3. Reports can be generated in CSV format by either Admins or Managers, and each report stores metadata including generation date, associated department (if applicable).

3.4.6 Data Dictionary

A comprehensive document that specifies and explains the components, relationships, and structure of a database is called a data dictionary. It acts as a reference manual, offering crucial details regarding fields, tables, data kinds, sizes, and limitations. To assist end users in understanding the significance of the data, providers usually provide text descriptions for every column in a dataset (Rashid et al., 2020). By providing a clear understanding of the database's architecture and functioning, this tool guarantees consistency in data usage and promotes communication among developers, analysts, and stakeholders.

Table 3.6: Employee Table of the proposed system.

No Field	Name	Type	Size	Description	Not Null	Primary Key (PK)	Foreign Key (FK)
1.	EmployeeID	INT	11	Unique identifier for each employee.	Yes	Yes	-
2.	full_name	VARCHAR	100	Full name of the employee.	Yes	No	-
3.	department	VARCHAR	100	Department the employee belongs to	Yes	No	-
4.	email	VARCHAR	100	Email used for login and notifications	Yes	No	-
5.	password	VARCHAR	255	Encrypted login password	Yes	No	-

6.	role	ENUM	-	Role of employee: 'admin', 'manager', 'staff'	Yes	No	-
----	------	------	---	---	-----	----	---

Explanation for Data Dictionary Table 3.6: Employee Table

Table 3.6 outlines the database schema used to manage all registered users within the system, including employees, administrators, and managers. It defines critical attributes required for authentication, role differentiation, and departmental organization.

The employee_id field, specified as VARCHAR, serves as the primary key and is marked NOT NULL to ensure each user is uniquely identifiable within the system. The full_name, email, and password fields are also designated as NOT NULL, storing essential user credentials for secure login and authentication.

The department field, defined as VARCHAR, categorizes employees according to their assigned departments, supporting clearer organizational structure. Meanwhile, the role field, implemented as an ENUM, establishes role-based distinctions employee, manager, or admin, enabling Role-Based Access Control (RBAC) throughout the application.

Table 3.7: AuthenticationLog Table of the proposed system.

No.	Field Name	Type	Size	Description	Not Null	Primary Key (PK)	Foreign Key (FK)
1.	auth_id	INT	11	Unique ID for authentication attempt (Primary Key)	Yes	Yes	-
2.	employee_id	INT	11	References employee who attempted authentication	Yes	No	Yes
3.	auth_type	ENUM	-	Method used: 'QR', 'fingerprint'	Yes	No	-

4.	status	ENUM	-	Result of authentication: 'success', 'failed'	Yes	No	-
5.	timestamp	DATETIME	YYYY-MM-DD HH:MM:SS	Date and time of authentication attempt	Yes	No	-

Explanation for Data Dictionary Table 3.7: AuthenticationLog

Table 3.7 defines the schema for logging employee authentication activities, functioning as both a security mechanism and an audit trail to track system access. The `auth_id` field, defined as INT, serves as the primary key and guarantees the uniqueness of each authentication event. The `employee_id` field acts as a foreign key referencing the Employee table, thereby establishing a direct relationship between authentication records and their respective users.

The `auth_type` field, implemented as an ENUM, specifies the authentication method used such as QR code or fingerprint while the `timestamp` field records the exact time of each authentication attempt. All fields are set as NOT NULL to ensure complete traceability, accountability, and the integrity of authentication data.

Table 3.8: Geolocation Table of the proposed system.

No Field	Name	Type	Size	Description	Not Null	Primary Key (PK)	Foreign Key (FK)
1.	geolocation_id	INT	11	Unique ID for each geolocation record	Yes	Yes	-
2.	attendance_id	INT	11	References Attendance(attendance_id)	Yes	No	Yes
3.	latitude	DECIMAL	10,8	Latitude captured during clock-in/out	Yes	No	-
4.	longitude	DECIMAL	11,8	Longitude captured during clock-in/out	Yes	No	-
5.	timestamp	DATETIME	-	Date and time when the geolocation was recorded.	Yes	-	-

Explanation for Table 3.8: Geolocation Table

Table 3.8 shows the schema for managing GPS-based location data, which plays a critical role in verifying employee presence at designated work sites.

The `geo_id` field, defined as INT serves as the primary key and is marked NOT NULL to ensure the uniqueness of each geolocation entry. The latitude and longitude fields, both defined as DECIMAL(10,8) and (11,8), store the geolocation coordinates captured during attendance events.

The `captured_time` field, of type TIMESTAMP, logs the precise moment the location data was recorded, enabling accurate correlation with clock-in and clock-out activities. This table is designed to integrate with the Attendance table, thereby enhancing the system's ability to validate user presence based on real-time location data.

Table 3.9: Attendance Table of the proposed system.

No Field	Name	Type	Size	Description	Not Null	Primary Key (PK)	Foreign Key (FK)
1.	attendance_id	INT	11	Unique ID for each attendance record	Yes	Yes	-
2.	employee_id	INT	11	References Employee (employee_id)	Yes	No	Yes (Employees.EmployeeID)
3.	clock_in_time	DATETIME	-	Timestamp of Employee clock-in.	Yes	No	-
4.	clock_out_time	DATETIME	-	Timestamp of Employee clock-out	Yes	No	-
5.	status	ENUM	-	'Present' and 'Absent'	Yes	No	-

Explanation for Table 3.9: Attendance Table

Table 3.9 captures the attendance activity for all employees, including clock-in/out times and geolocation verification.

The `record_id` field is of type INT, functions as the primary key, and is marked Not Null. It uniquely identifies each attendance entry. The `employee_id` is of type INT, a foreign key referencing the Employee table, establishing a one-to-many relationship (one employee, many records). The `clock_in_time` and `clock_out_time` are of type TIMESTAMP, both Not Null, capturing the time events for working hours. The `status` field is an ENUM (e.g., Present, Late, Absent) that reflects the attendance category for the record and is Not Null. The `geo_id` is of type INT, acting as a foreign key referencing the Geolocation table. It links each attendance to a specific GPS data log, ensuring traceability. This structure ensures reliable attendance tracking and auditability.

Table 3.10: Attendance Report Table of the proposed system.

No Field	Name	Type	Size	Description	Not Null	Primary Key (PK)	Foreign Key (FK)
1.	report_id	INT	11	Unique identifier for each report.	Yes	Yes	-
2.	generated_by	INT	11	References Employee(employee_id) (only for admin or manager role)	Yes	No	Yes
3.	generated_date	TIMESTAMP	-	Date and time the report was generated.	Yes	No	-
4.	report_type	ENUM	-	Report type, e.g., daily, monthly, department-wise	Yes	No	-
5.	export_format	ENUM	'CSV'	Only allowed value: 'CSV' (fixed format used in this project)	Yes	No	-

Explanation for Table 3.10: Attendance Report Table

Table 3.10 outlines the schema for the generation and storage of reports within the system, facilitating structured and efficient report management. The system comprises five fields that enable the creation and monitoring of diverse reports.

The `report_id` field is an INT, marked as the primary key and Not Null, ensuring uniqueness for every report generated. The `generated_by` field is an INT, a foreign key linking to the Employee table (only Admin or Manager roles are permitted), and is Not Null.

The `report_type` field is an ENUM (Daily, Monthly, etc.) and Not Null, allowing classification of reports for filtering and organization. The `generated_date` field is a TIMESTAMP, Not Null, storing the date and time the report was created. The `export_format` field is an ENUM restricted to only one value: CSV, and Not Null, since all exports are standardized as .csv files.

3.4.7 Wireframes

A wireframe is a fundamental blueprint that delineates the essential structure and functionality present on a single screen of a website or application (Hamm, 2014). It functions as a visual reference, delineating the arrangement of components such as buttons, icons, text, and images, while emphasizing usability and user navigation. Wireframes are generally low-fidelity designs that establish a basis for more intricate prototypes and guarantee consensus among stakeholders prior to the commencement of development. The mobile application for the suggested system will be illustrated as part of the design using wireframe.

3.4.7.1 Wireframe for Register Page

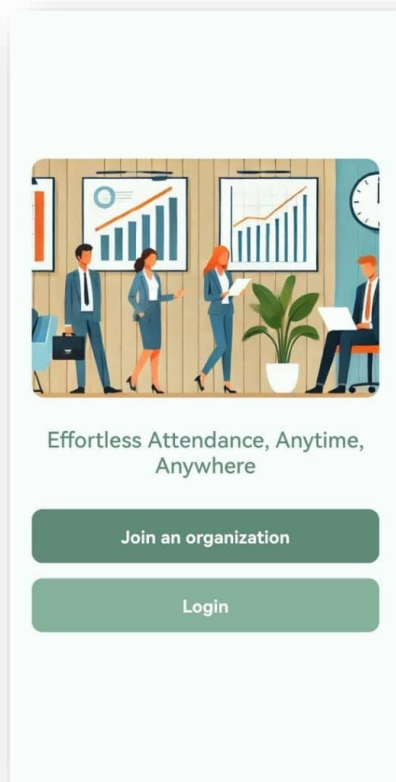


Figure 3.26 depicts the first page that appears when a user enters the application. If they are a member of a company, new users can go to the sign-up page by selecting "Join an organization". Existing users can access their accounts by clicking the "Log in" button.

Figure 3.26: Register Page.

3.4.7.2 Wireframe for Sign in Page

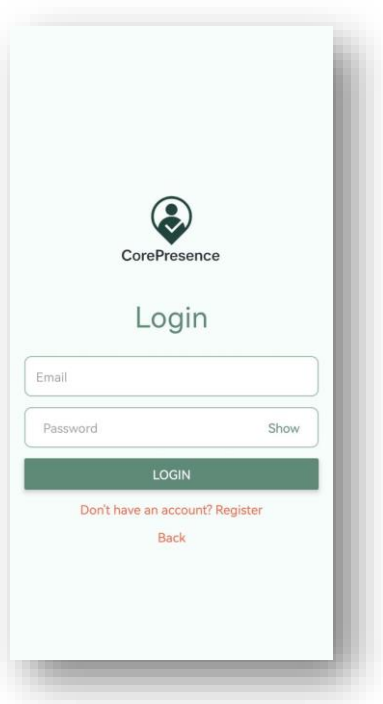


Figure 3.27 depicts the system's login page. To access the system's homepage, users must enter their email and password and click the login button.

Figure 3.27: Sign in Page.

3.4.7.3 Wireframe for User Profile Setting Page

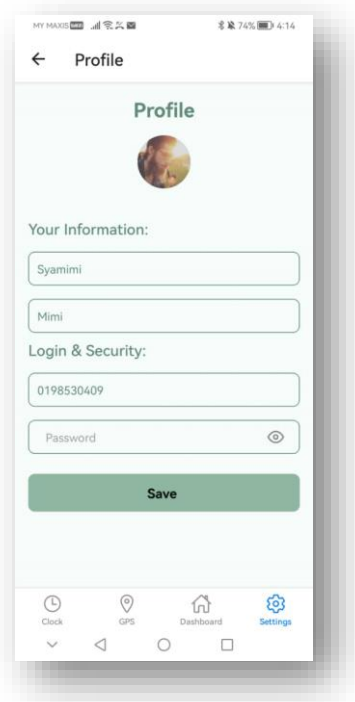


Figure 3.28: Setting Page.

Figure 3.28 displays the user profile settings page. Here, users can view and edit their personal details, which comprise their name and phone number. The page also shows a password field and a Save button to confirm any changes. The navigation bar at the bottom provides fast access to key system functions, including Clock, GPS, Dashboard, and Settings.

3.4.7.4 Wireframe for Employee Clock In/Out Page

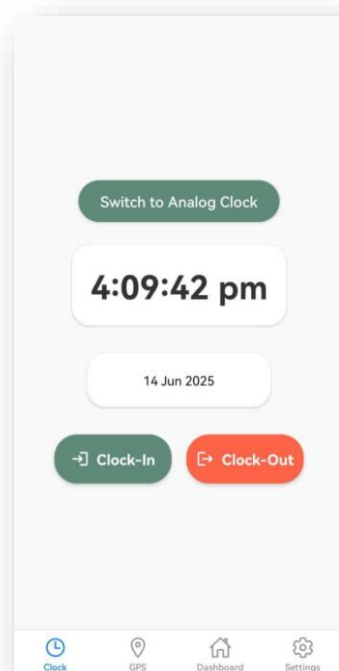


Figure 3.29: Employee Clock In/Out Page.

Figure 3.29 shows the main page for managing work hours. Employees can clock in and out, and the system will record the overall time worked throughout the session.

3.4.7.5 Clock In Method Selection Page

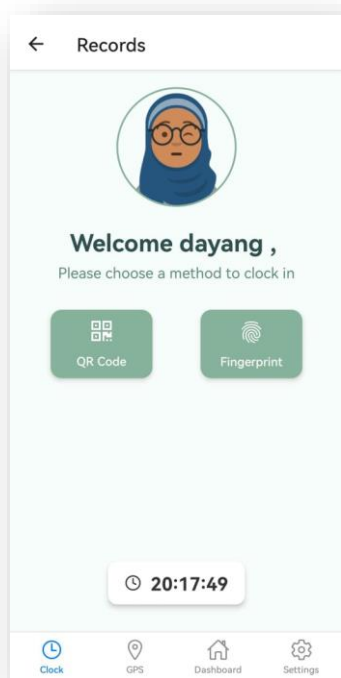


Figure 3.30 shows after the employees choosing the "Clock In" button on the main page, employees are directed here to select their chosen clock-in method (QR Code or Fingerprint).

Figure 3.30: Clock in Method Selection Page.

3.4.7.6 Fingerprint Method Page



Figure 3.31 indicates where employees can clock in using their right thumbprint. The system uses the fingerprint as part of the attendance technique.

Figure 3.31: Fingerprint Method Page.

3.4.7.7 QR Code Clock-In Page

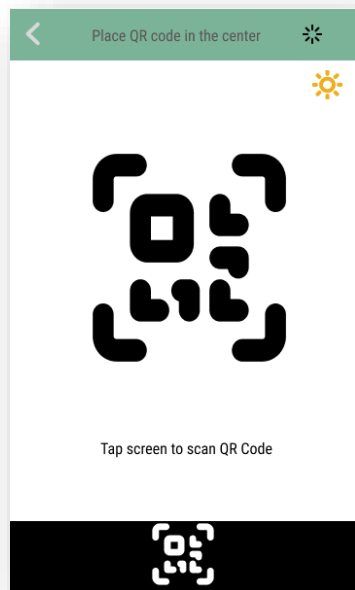


Figure 3.32 depicts a page where employees scan a workplace-specific QR code to clock in, ensuring location-based authentication. The system employs a QR Code-based attendance method.

Figure 3.32: QR Code Clock in Page.

3.4.7.9 Account Settings Page

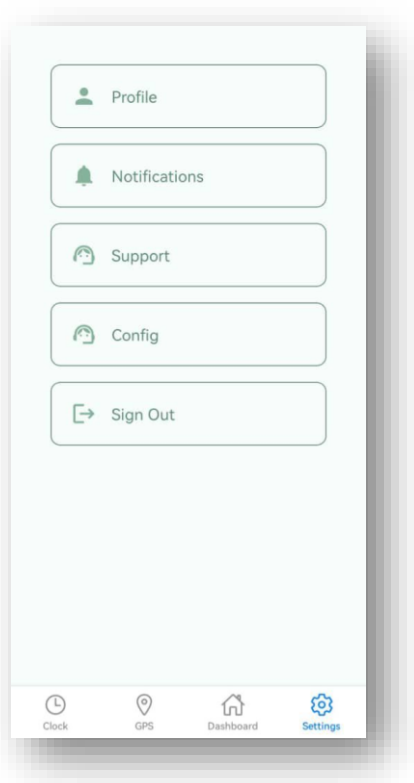


Figure 3.33: Account Settings Page.

Figure 3.33 shows the account settings page, accessible to employees, managers, and administrators. This page acts as a central hub for managing user preferences. Users can view their profile details, notifications, request support, adjust settings, and sign out of the app. The straightforward layout ensures essential actions are easy to locate, helping to streamline user interactions.

3.4.7.10 Geolocation Tracking Page

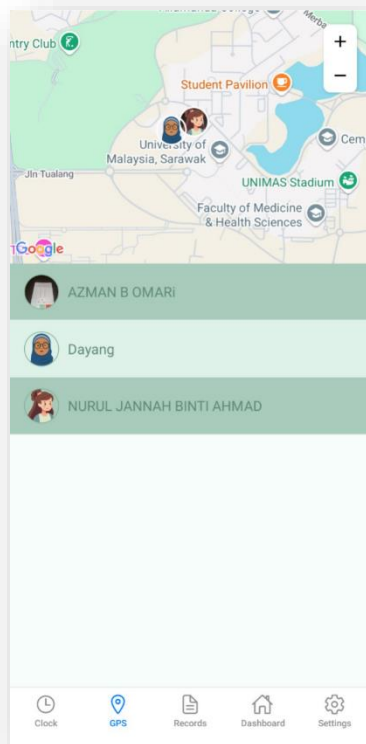


Figure 3.34: Geolocation Tracking Page.

Figure 3.34 shows where the page enables managers to track staff attendance within a chosen geofenced area. The authorised area includes both the workplace and a buffer zone just outside of it. Managers can monitor employee movements during working hours to verify they are in the designated zone. This feature improves accountability and keeps staff inside their permitted work radius. This feature keeps track of an employee's location throughout working hours after they clock in. However, in order to balance efficiency and privacy, location updates take place at predetermined intervals rather than instantly.

3.4.7.11 Report Page

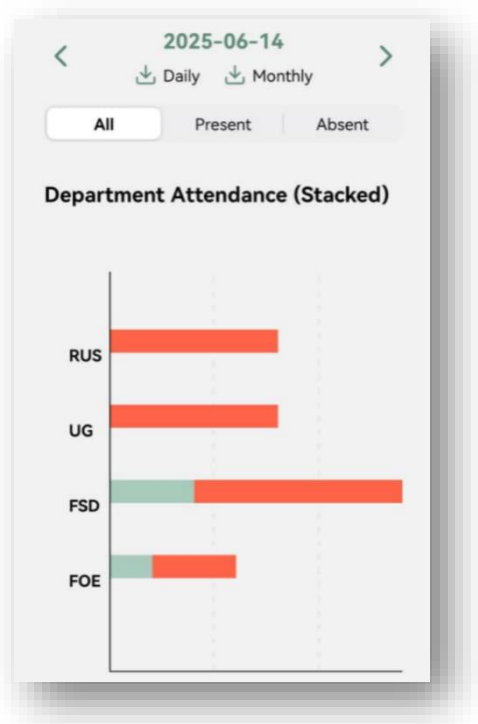


Figure 3.35: Report Page.

Figure 3.35 displays the report page, offering an overview of staff attendance patterns and key metrics. The stacked bar chart allows managers to visualize attendance trends by department, including daily and monthly summaries of present and absent employees. Both admin and manager users can export attendance reports in CSV format, simplifying data analysis and sharing as needed.

3.4.7.14 Employee Dashboard Page

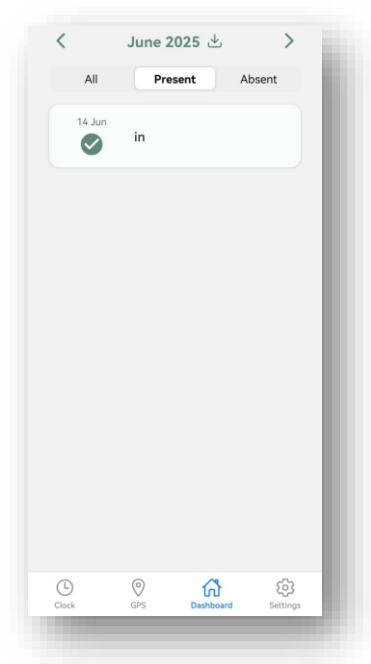


Figure 3.36 shows the employee dashboard page, where the employee can monitor their daily attendance status. The dashboard highlights core information such as the attendance date and the employee's presence status. This allows employees to conveniently review their attendance history in a clear and accessible format.

Figure 3.36: Employee Dashboard Page.

3.4.7.15 Manager Dashboard Page



Figure 3.37: Manager Dashboard Page.

Figure 3.37 provides a summary of attendance data across all departments. Managers can access the attendance status of each employee including administrators and filter the display by daily or monthly reports, as well as by attendance status which are present and absent. The dashboard includes department legends and summaries for each attendance category, allowing managers to quickly evaluate team presence. From this page, managers can generate CSV attendance reports for data management or further analysis.

3.4.7.16 Admin Dashboard Page

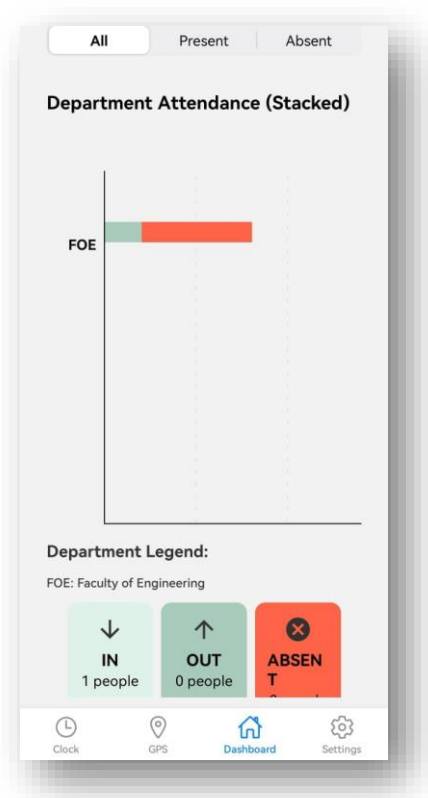


Figure 3.38 shows the admin dashboard page, where attendance statistics are limited to the admin's designated department. The stacked bar chart is paired with a department legend and shows the total number of employees marked as present or absent. This view allows for department-focused monitoring while upholding role-based data access. Admins can filter the data by daily or monthly intervals and produce attendance summaries customized for their team.

Figure 3.38: Admin Dashboard Page.

3.5 Summary

This chapter provides a thorough overview of the methodology and design concerns essential to the development of the Multi-Modal Authentication and Geolocation Approach for Attendance Verification. The chapter utilizes the Rapid Application Development (RAD) approach to guarantee an iterative, user-centric, and efficient development process. The methodology section explicates the four phases of the RAD model Requirements Planning, User Design, Construction, and Cutover clarifying their use in system development. In the requirements planning phase, user needs are discerned and assessed via brainstorming sessions, surveys, and reviews of existing systems. This phase underscores difficulties such

inefficiencies, fraud, and elevated costs linked to conventional attendance systems, thereby facilitating a solution aligned with user expectations.

The design phase establishes the groundwork for system implementation by showcasing several architectural and conceptual designs, such as the Overall System Architectural Design, Flowcharts, Context Diagrams, Data Flow Diagrams (DFDs), Entity-Relationship Diagrams (ERD), and a data dictionary. These artifacts are essential for illustrating the system's architecture, elements, and processes. High-fidelity wireframes depict essential user interface concepts for functionalities including clock-in techniques, geolocation tracking, and reporting dashboards.

The chapter subsequently delineates the construction phase, highlighting iterative development, user feedback, and refining. Essential functionalities are executed utilizing technologies such as React Native Firebase, and Supabase (a backend-as-a-service built on PostgreSQL), guaranteeing operational efficiency, scalability, and real-time interactivity. The concluding two phases system testing and deployment are examined, highlighting the necessity of thorough assessment and seamless execution to get a dependable, user-centric system.

This chapter offers an extensive structure for the development of the Multi-Modal Authentication and Geolocation Approach for Attendance Verification, assuring its conformity with user requirements and technology viability while following the iterative and collaborative concepts of Rapid Application Development (RAD).

CHAPTER 4: IMPLEMENTATION

4.1 Introduction

This chapter outlines the development of the mobile attendance verification system. It explains the construction process, the tools and platforms applied, and the features implemented in line with the project's design and requirements. The system uses a multi-modal authentication method, combining face recognition, QR code scanning, fingerprint input, and geolocation tracking to improve attendance verification precision and security. Development followed the Rapid Application Development (RAD) methodology, focusing on rapid prototyping, user feedback, and gradual improvements. Each module was developed step-by-step and refined through testing and feedback. Additionally, this chapter covers setup procedures, example user interfaces, and key configuration steps necessary to make the application operational and ready for testing.

4.2 Installation and Configuration of Multi-Modal Authentication and Geolocation Approach for Attendance Verification

4.2.1 Visual Studio Code

Visual Studio Code (VS Code) is a free and lightweight source code editor created by Microsoft. It supports a wide variety of programming languages and frameworks and is popular among developers for its extensive extension library, intuitive interface, and solid built-in features like debugging, version control, and an integrated terminal. This section outlines the procedure for installing and configuring Visual Studio Code on a Windows operating system.

Step 1: Download Visual Studio Code Installer

The installation begins by downloading the official installer from Visual Studio Code's website. Navigate to: <https://code.visualstudio.com/>. Click the “Download for Windows” button to obtain the latest stable version of the installer, as shown in Figure 4.1.

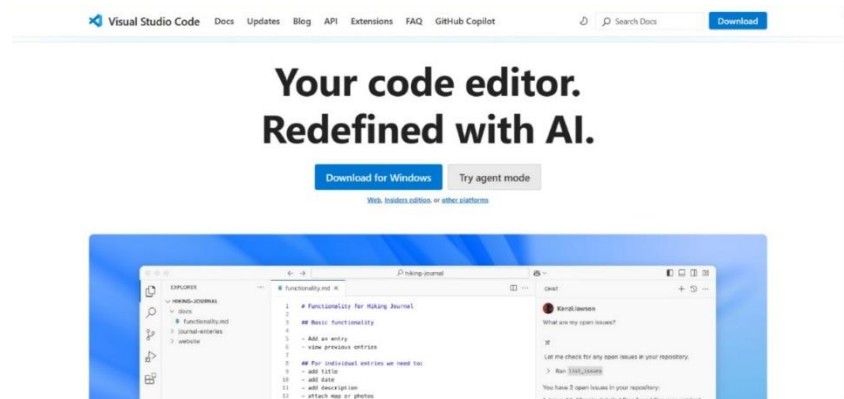


Figure 4.1: Visual Studio Code Official Website.

Step 2: Run the Installer

Once the .exe file is downloaded, double-click it to launch the installation wizard. Follow the instructions as outlined below:

1. Accept the License Agreement

Read and accept the license terms to proceed to the next step as shown in Figure 4.2.

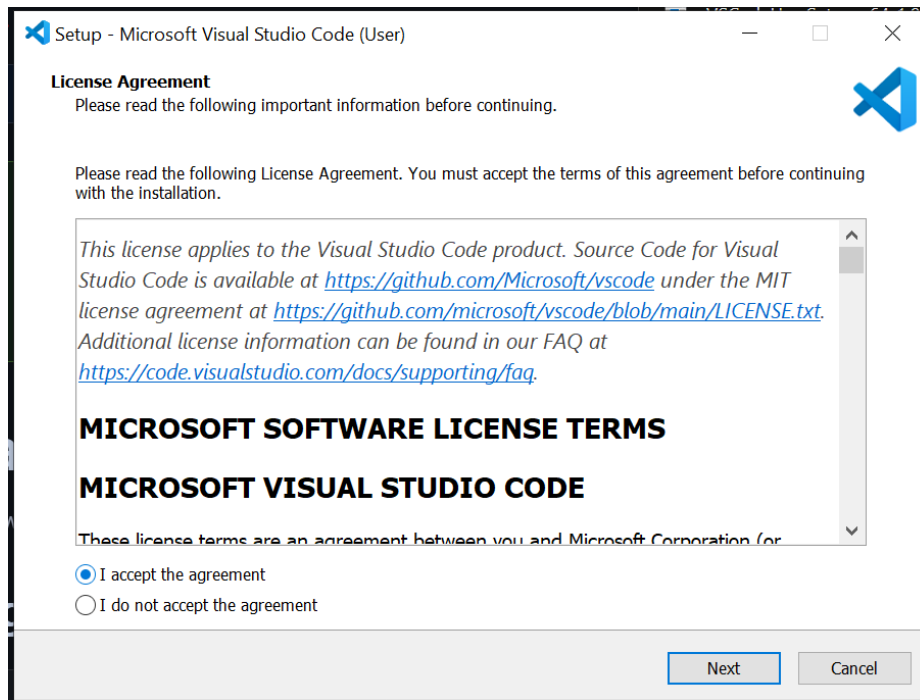


Figure 4.2: Visual Studio Code Setup.

2. Choose Installation Location

The default installation path for Visual Studio Code is C:\Program Files\Microsoft VS Code, which can be adjusted if necessary, as shown in Figure 4.3.

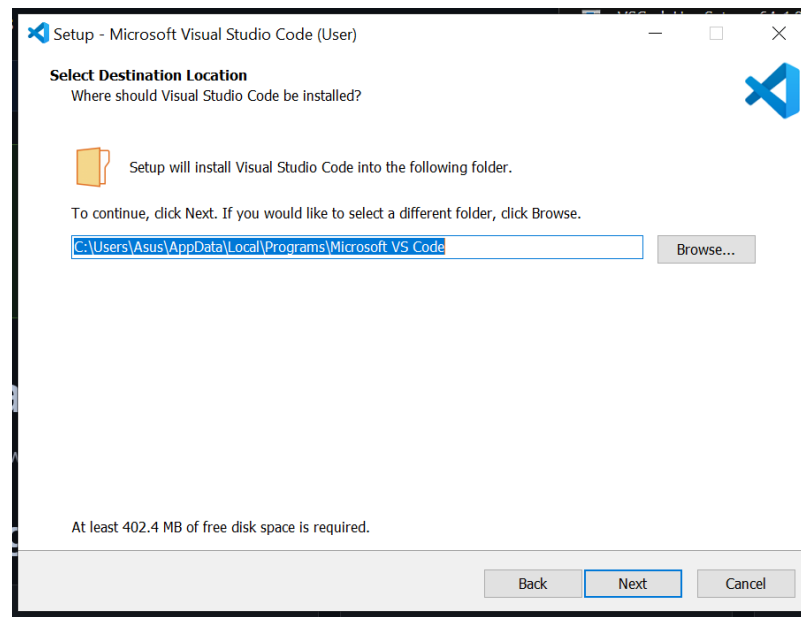


Figure 4.3 Visual Studio Code Installer displays the default installation directory, which can be customized as needed before proceeding to the next step.

3. Select Additional Tasks

Tick the following checkboxes for better usability (see Figure 4.4):

- Create a desktop icon
- Add “Open with Code” action to Windows Explorer file context menu
- Add “Open with Code” action to Windows Explorer directory context menu
- Register code as an editor for supported file types
- Add to PATH (important for using VS Code in command prompt)

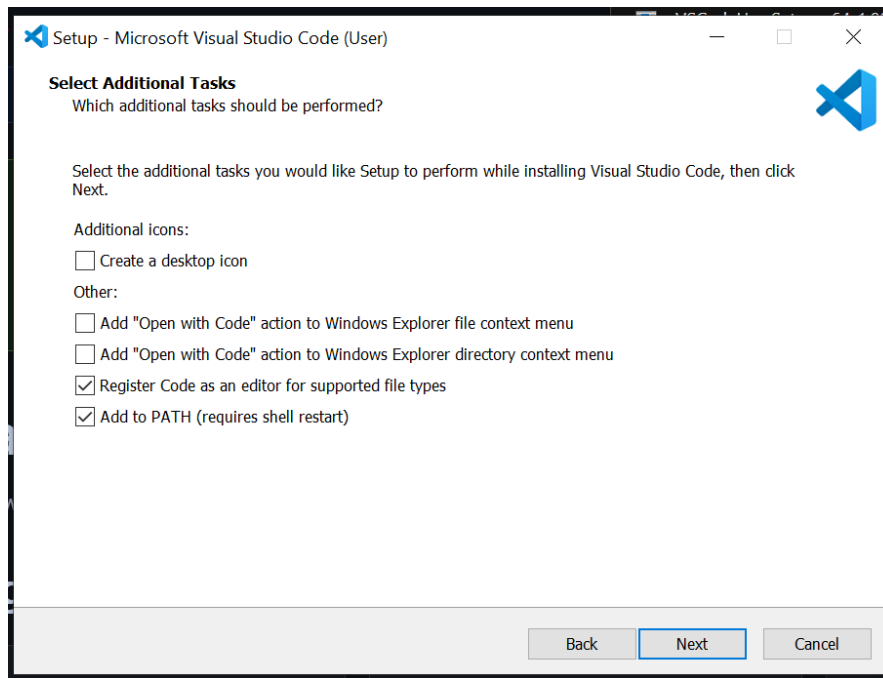


Figure 4.4: Additional setup options in Visual Studio Code.

4. Click Next, then Install to begin the installation process as shown in Figure 4.5.

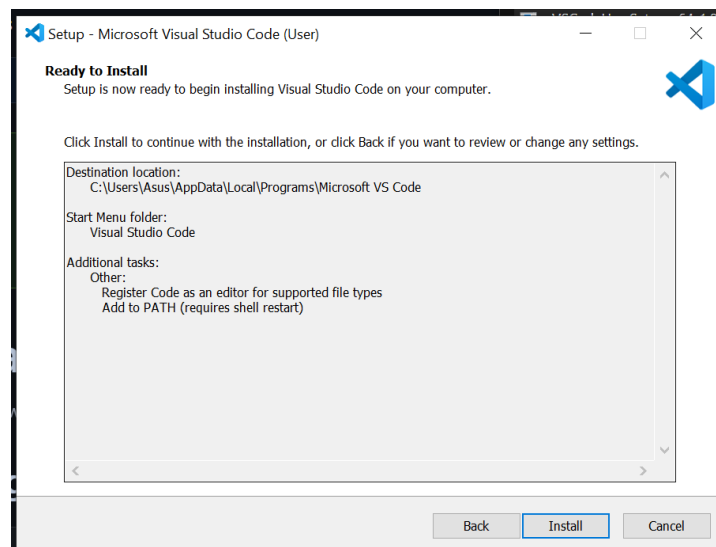


Figure 4.5: Installation Ready.

Step 3: Launch Visual Studio Code

After the installation finishes, click Finish and ensure the “Launch Visual Studio Code” checkbox is selected. When the application opens, users will see a Welcome screen where they can start creating new projects, open current files, or install extensions, as illustrated in Figure 4.6.

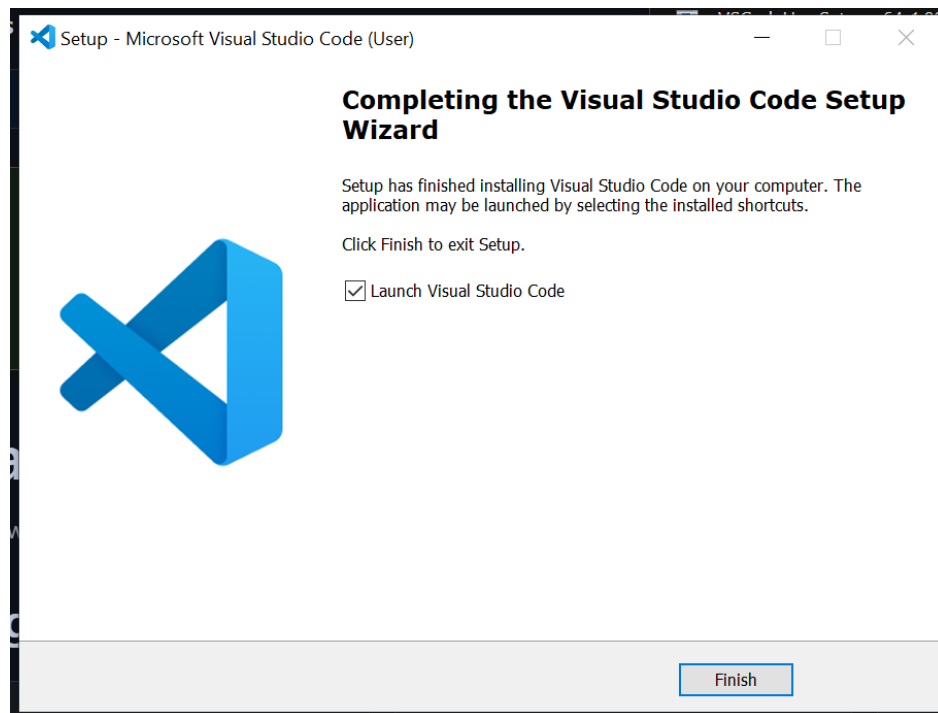


Figure 4.6: Visual Studio Code Setup Wizard.

Step 4: Configure Visual Studio Code (Optional)

While VS Code is ready for use instantly after installation, users can enhance their productivity by modifying settings and installing extensions designed for their specific development needs. This can be achieved through the following steps (as shown in Figure 4.7):

1. Search for and install any essential extensions, such as:

- Programming language support (e.g., Python, Java, PHP)
 - Themes and icons
 - Code formatters and linters
2. Open the Settings by clicking the gear icon at the bottom left, where you can adjust configurations such as font size, auto-save, tab spacing, and other options.

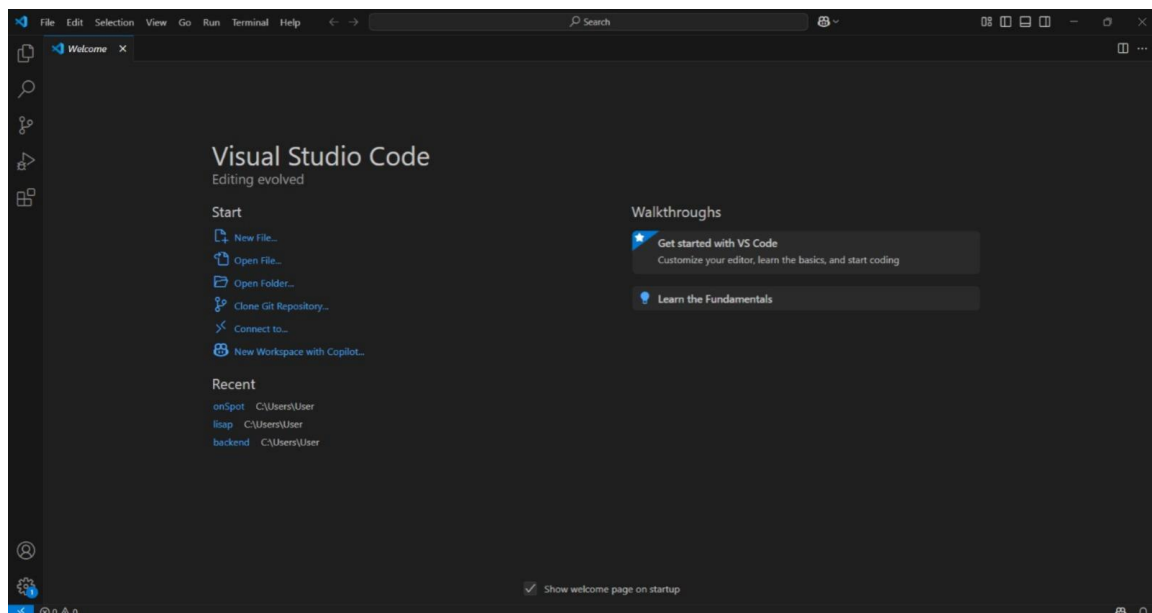


Figure 4.7: Visual Studio Code Interfaces.

Step 5: Verify the Installation

To confirm a successful installation, users can open the built-in terminal and run standard commands related to their programming language or framework. To open the terminal:

- Navigate to Terminal > New Terminal, or press Ctrl + (backtick).

This confirms that the environment is ready for coding, as shown in Figure 4.8.

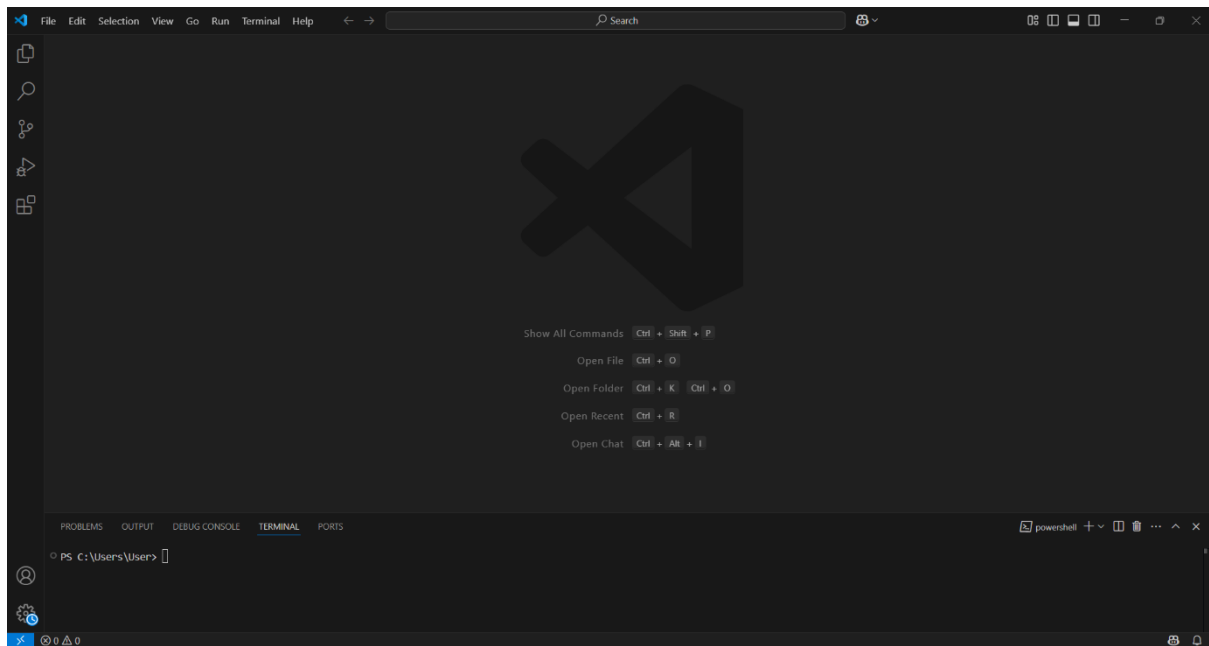


Figure 4.8: Visual Studio Code terminal confirming a successful setup.

4.2.2 Expo

Expo is an open-source framework and platform built for cross-platform React applications. It enables developers to develop mobile apps using JavaScript and React Native with basic setup. The Expo CLI (Command Line Interface) facilitates the creation, building, and management of Expo projects. This section details the installation process of Expo CLI on a Windows operating system.

Step 1: Install Node.js

Expo CLI needs Node.js to be installed on the system. Node.js is a JavaScript Execution platform that lets the execution of JavaScript code external to a web browser.

1. Go to the official Node.js website: <https://nodejs.org/>
2. Download the LTS (Long Term Support) version for Windows as shown in Figure 4.9.

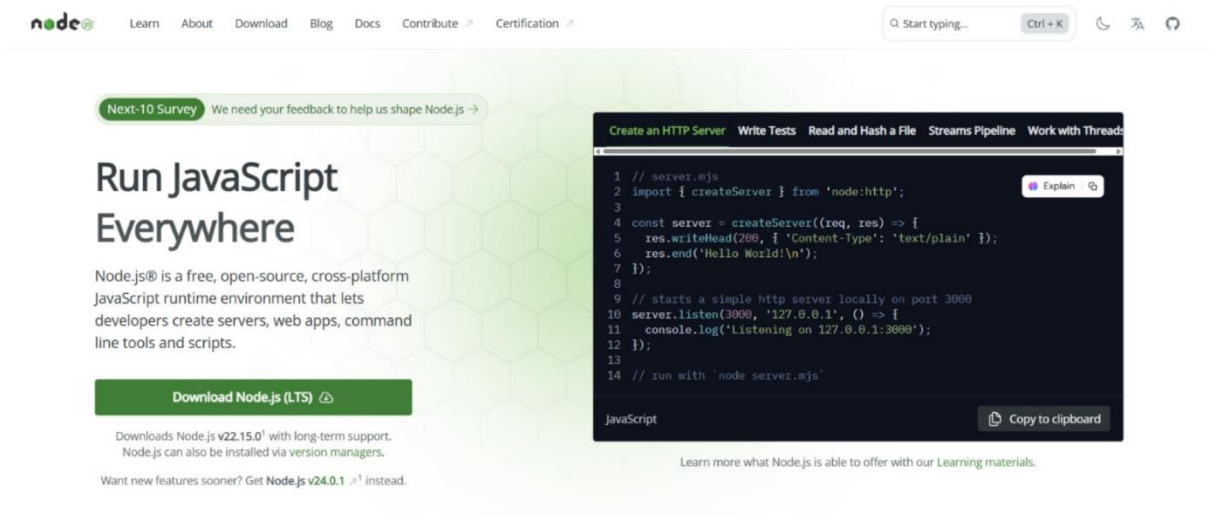


Figure 4.9: Official Node.js website showing the LTS version download option.

3. Run the installer and follow the default installation steps:
 - Accept the license agreement and proceed with the default install settings
 - Choose installation path
 - Install Node.js and npm (Node Package Manager)

As shown in Figure 4.10. To verify the installation, open Command Prompt and type:

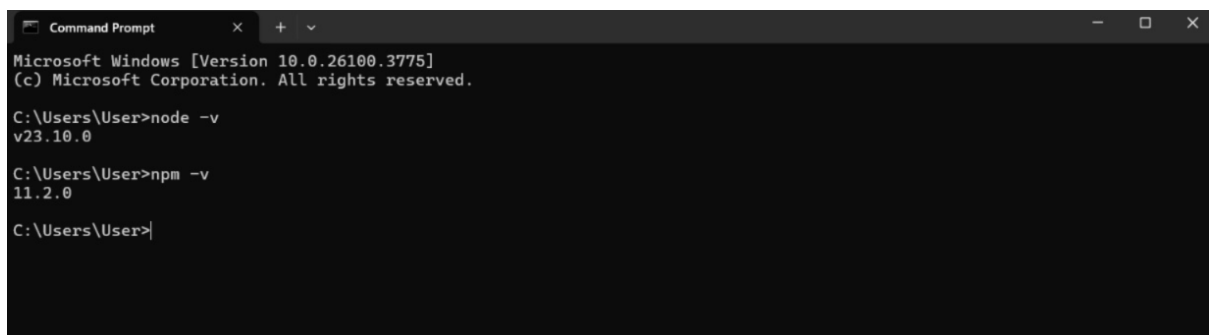
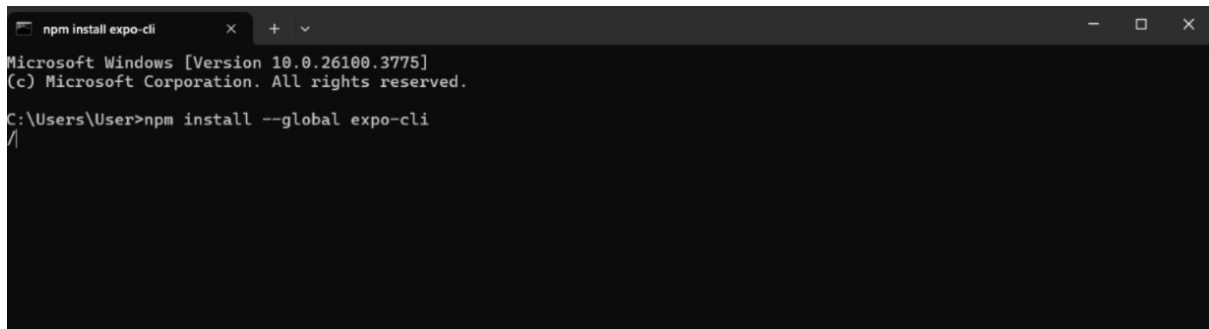


Figure 4.10: Command Prompt showing successful verification of Node.js and npm installation.

If installed correctly, it will display the version numbers of Node.js and npm.

Step 2: Install Expo CLI Using npm

After Node.js and npm are installed, Expo CLI can be installed system-wide using the following command in Command Prompt or PowerShell as shown in Figure 4.11:



```
Microsoft Windows [Version 10.0.26100.3775]
(c) Microsoft Corporation. All rights reserved.

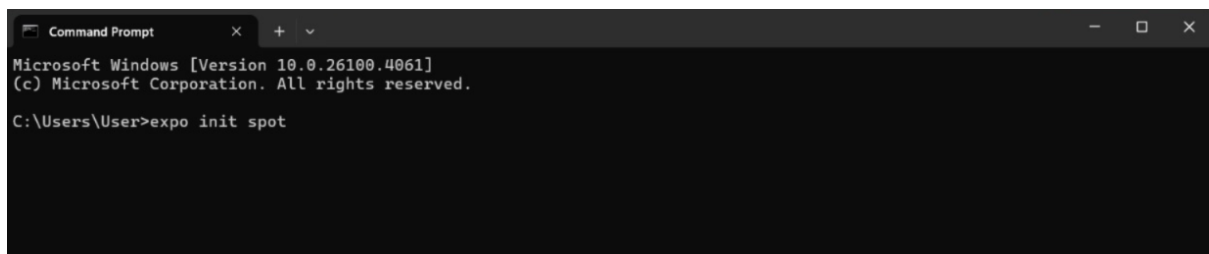
C:\Users\User>npm install --global expo-cli
```

Figure 4.11: Installing Expo CLI globally via npm using the command prompt.

This command installs Expo CLI system-wide on the system, making the expo command available from any directory.

Step 3: Create a New Project Using Expo

To create a new project using Expo, run the following command as shown in Figure 4.12 in the terminal (eg., Command Prompt):



```
Microsoft Windows [Version 10.0.26100.4061]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User>expo init spot
```

Figure 4.12: Initializing a new Expo project named spot using the expo init command.

This command sets up a new Expo project called spot and prompts for a template selection (e.g., blank (JavaScript)). Next, Figure 4.13 shows in order to navigate into the project directory:



```
Command Prompt
Microsoft Windows [Version 10.0.26100.4061]
(c) Microsoft Corporation. All rights reserved.
C:\Users\User>cd spot
```

Figure 4.13: Navigating into the newly created project directory with cd spot.

As shown in Figure 4.14 and 4.15, to start the development server, run:



```
Command Prompt
Microsoft Windows [Version 10.0.26100.3775]
(c) Microsoft Corporation. All rights reserved.
C:\Users\User>npm start
```

Figure 4.14: Running npm start to launch the Expo development server.



```
Command Prompt
Microsoft Windows [Version 10.0.26100.3775]
(c) Microsoft Corporation. All rights reserved.
C:\Users\User>expo start
```

Figure 4.15: Expo start command window.

Expo Developer Tools will open in the default browser and provide a QR code to allow app preview on a mobile device.

Step 4: Install Expo Go on Mobile Device

To preview and test the app on a physical device, download Expo Go from the Google Play Store or Apple App Store.

1. Open the Expo Go app.
2. Scan the QR code displayed in the browser or terminal.

This allows real-time previewing of the app without compiling a native build.

Step 5: Verify Installation

The installation and configuration of Expo CLI can be verified by effectively launching a project and viewing the app using Expo Go. If the terminal shows a proper build process and the app loads on the mobile device, the setup is validated as shown in Figure 4.16.



Figure 4.16: QR code from Expo CLI displayed in the terminal.

4.2.3 Android Studio

Android Studio is the official integrated development environment (IDE) for Android app development. It offers a comprehensive set of tools, including a code editor, emulator, and

debugging features, designed to build and test Android applications effectively. In this project, Android Studio is primarily used to run the Android Emulator, allowing thorough testing of the mobile app covering features like geolocation tracking, QR code scanning, and fingerprint authentication without the requirement for a physical device.

Step 1: Download and Installation

Install the latest version of Android Studio from the official website. Make sure to incorporate the Android SDK, Platform-Tools, and AVD Manager throughout the setup process as shown in Figure 4.17.

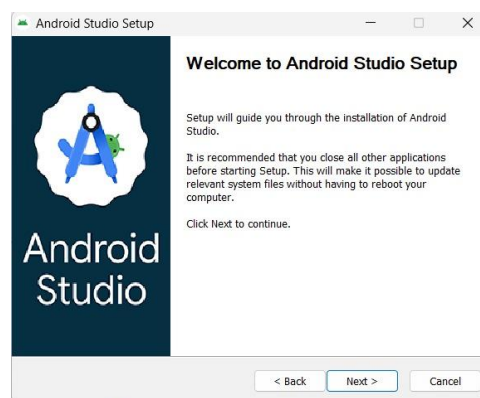


Figure 4.17: Welcome to Android Studio Setup.

Step 2: Select Components to Install

Installing AVD is crucial for emulator-based testing, refers to Figure 4.18.

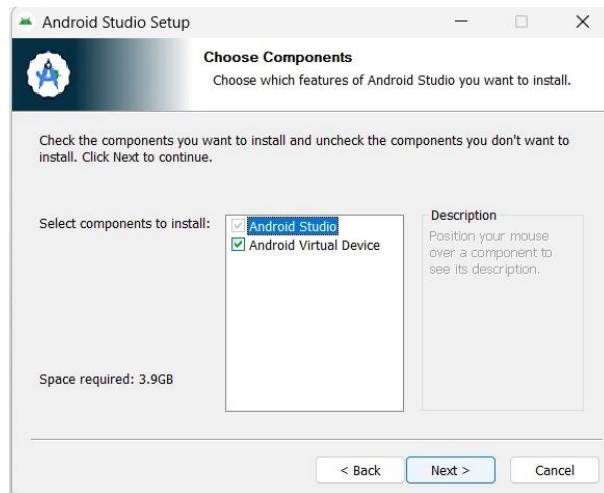


Figure 4.18: Choose Components.

Step 3: Set Installation Paths

Outline the installation folder (e.g. C:\Program Files\Android\Android Studio), as illustrated in Figure 4.19.

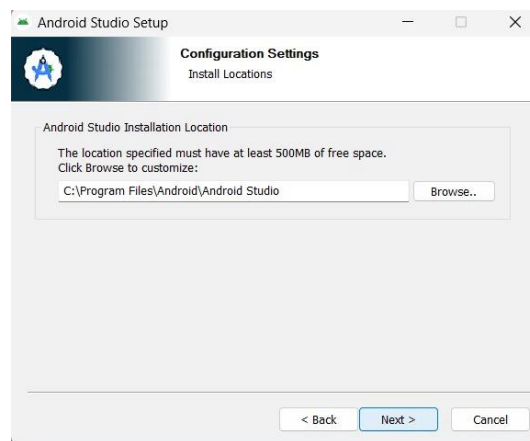


Figure 4.19: Configuration Settings.

Step 4: Pick Installation Type

As shown in Figure 4.20, Select the Standard setup option to install all recommended tools including the emulator and SDK for standard development requirements.

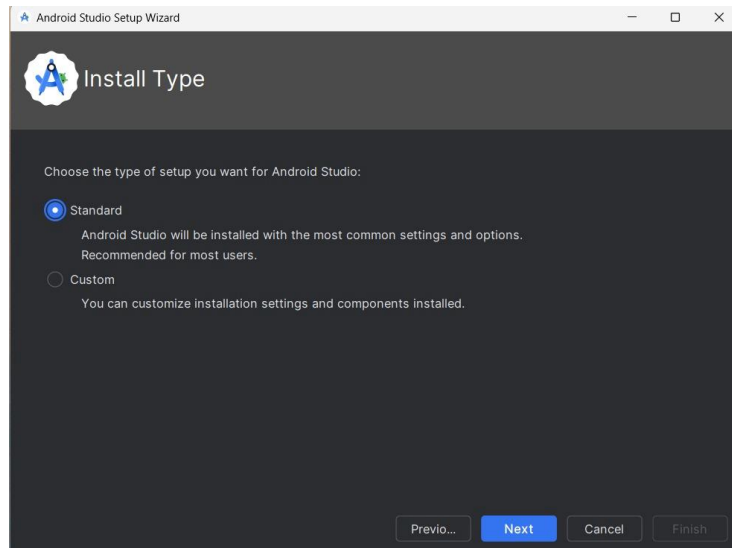


Figure 4.20: Install Type (Standard).

Step 5: Verify Settings & Download SDK Components

Verify the SDK folder and components (emulator, SDK Build Tools) are correct, then click Next, refers to Figure 4.21.

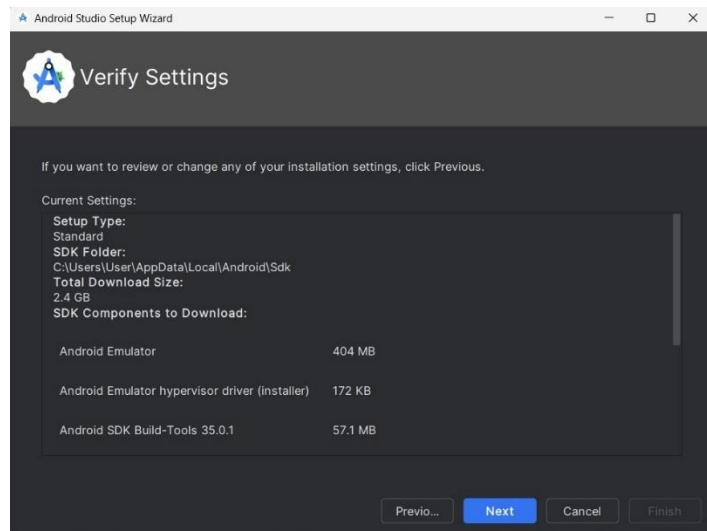


Figure 4.21: Verify Settings.

As shown in Figure 4.22, Accept Android SDK license agreement and Finish.

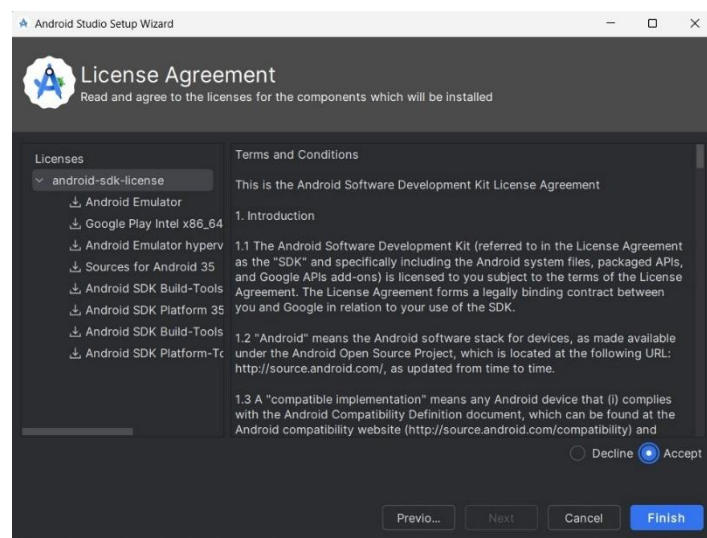


Figure 4.22: SDK's License Agreement.

Next, based on Figure 4.23, the download process for the SDK, emulator, and build tools will commence in Android Studio.

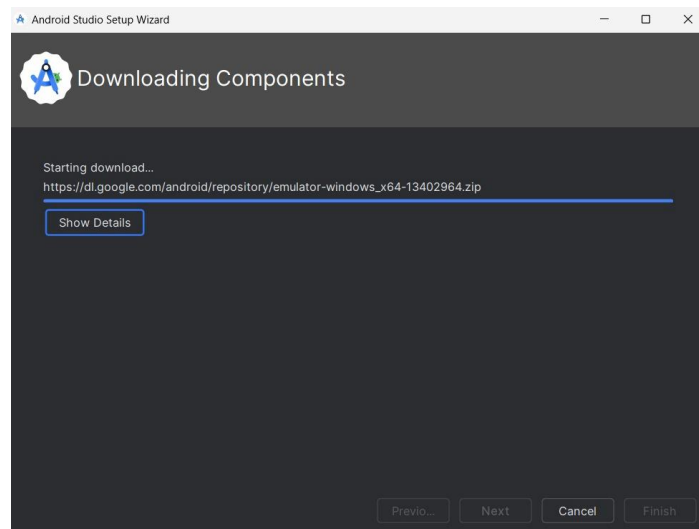


Figure 4.23: Downloading Components.

Step 6: SDK Platforms Setup

Launch the SDK Manager by navigating to Configure → SDK Manager. Install the required API level (e.g., Android 15 / API 35) together with the necessary SDK tools. Additionally, ensure that platform-tools is added to your system's PATH to enable ADB operation when debugging through the terminal or Expo as depicted in Figure 4.24.

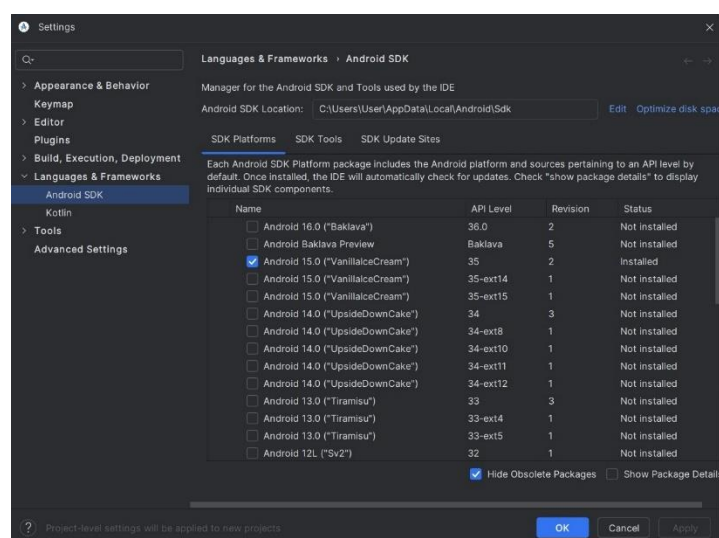


Figure 4.24: Languages & Frameworks Android SDK.

Step 7: Create and Verify AVD Configuration

In the AVD Manager, click “Add Device,” select a hardware profile such as Pixel 5 or “Medium Phone,” and choose a system image like API 35 or 36. Android Studio will seamlessly download and install the selected image. After installation, confirm that the new AVD appears in the Device Manager list (see screenshot), signifying that the emulator is set up and ready to start for use with your Expo app, as demonstrated in Figure 4.25.

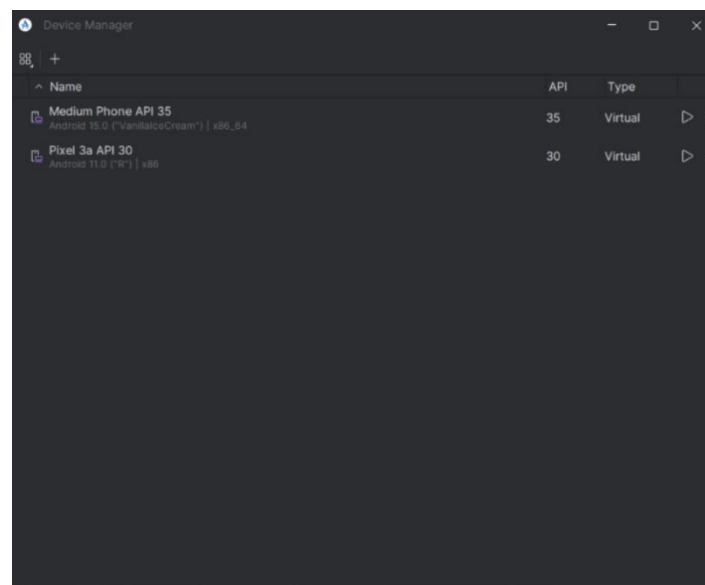


Figure 4.25: AVD Manager showing configured virtual devices (APIs 35 & 30).

4.2.4 Integration Overview

This section highlights the main software libraries and frameworks incorporated into the Multi-Modal Authentication and Geolocation Approach for Attendance Verification system. These elements serve as the foundation for both the frontend interface and backend services, supporting essential features such as user interactions, data management, and push notifications. Importantly, these integrations happen at the code level, installed through

package managers, thus no separate software installations are necessary on the development machine.

4.2.4.1 React Native

React Native is the core framework used for building cross-platform mobile user interfaces with JavaScript and React. Utilizing the Expo environment, it smoothly manages native device features such as camera access, location tracking, and authentication, eliminating the need to manage separate codebases for Android and iOS. React Native can be installed using Expo CLI and npm, as demonstrated in Figure 4.26.

A screenshot of a Git Bash terminal window. The title bar says "Git Bash". The terminal text shows the Windows version and copyright information, followed by the command `C:\Windows\System32>expo install react-native` being entered at the prompt.

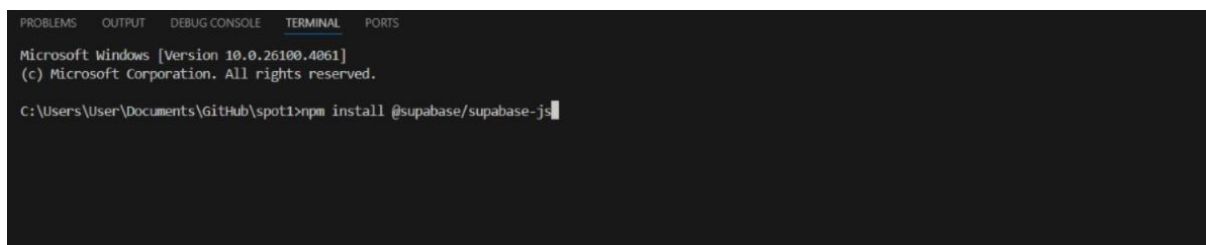
```
Microsoft Windows [Version 10.0.26100.4061]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>expo install react-native
```

Figure 4.26: Terminal command to install React Native using Expo CLI.

4.2.4.2 Supabase

Install via npm install as depicted in Figure 4.27:

A screenshot of a VS Code terminal window. The title bar shows tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The terminal text shows the Windows version and copyright information, followed by the command `C:\Users\User\Documents\GitHub\spot1>npm install @supabase/supabase-js` being entered at the prompt.

```
Microsoft Windows [Version 10.0.26100.4061]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User\Documents\GitHub\spot1>npm install @supabase/supabase-js
```

Figure 4.27: Terminal command showing installation of Supabase JavaScript client via npm.

Supabase powers the app's authentication, database, and storage APIs. It is integrated into the application by initializing a Supabase client using the project's credentials. This enables secure data management without the need to build a custom backend solution, as illustrated in Figure 4.28.

```
lib > JS supabase.js > ...
1 // lib/supabase.js
2 import { createClient } from '@supabase/supabase-js';
3
4 const supabaseUrl =
5 const supabaseAnonKey =
6
7 export const supabase = createClient(supabaseUrl, supabaseAnonKey);
8 |
```

Figure 4.28: Supabase client setup example in JavaScript.

4.2.4.3 Firebase Cloud Messaging (FCM)

Firebase Cloud Messaging enables real-time push notifications for alerts such as clock-in confirmations and geo-based prompts. It is fully configurable through both code using device token registration and message handlers and the Firebase Console settings. The necessary packages can be installed via Expo CLI, as shown in Figure 4.29.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\User\Documents\Github\spot1> expo install expo-firebase-messaging firebase
```

Figure 4.29: Terminal command showing installation of Firebase Cloud Messaging via Expo CLI.

4.3 Using the Multi-Modal Authentication and Geolocation Approach for Attendance Verification

4.3.1 Welcome Page (Join or Login)

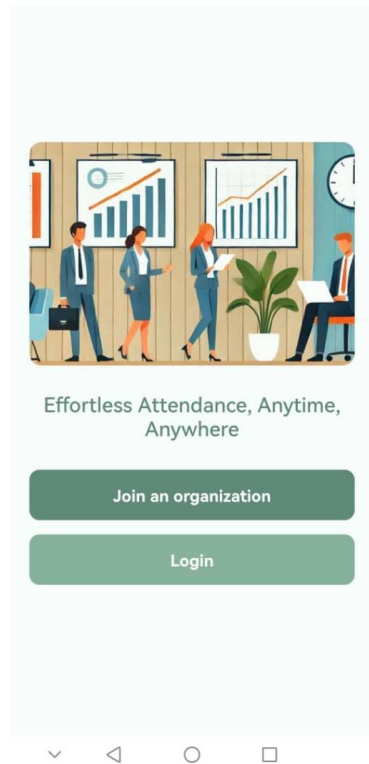
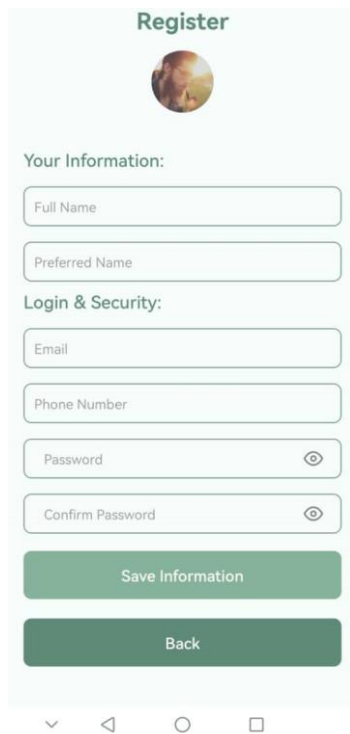


Figure 4.30: Welcome Page.

This is the initial screen users view when launching the CorePresence app. They can choose to either join an existing organization or log into their account. The welcoming slogan, "Effortless Attendance, Anytime, Anywhere," emphasizes the app's goal of simplifying attendance tracking, as depicted in Figure 4.30.

4.3.2 Registration Page



The image shows a mobile application registration screen. At the top, the word "Register" is displayed in green, next to a circular profile picture placeholder. Below this, the form is organized into two sections. The "Your Information:" section contains two input fields: "Full Name" and "Preferred Name". The "Login & Security:" section contains four input fields: "Email", "Phone Number", "Password", and "Confirm Password". The "Password" and "Confirm Password" fields include eye icons for toggling visibility. At the bottom of the form are two large, rounded buttons: a green "Save Information" button and a dark green "Back" button. The entire form is set against a light green background with a subtle pattern. At the very bottom of the screen, there are four small, faint icons: a downward arrow, a left-pointing triangle, a circle, and a square.

Figure 4.31: Registration Page.

Figure 4.31 shows that the Users registering for the first time as a new user must provide personal details, including full name, display name, email, phone number, and a secure password. This information is vital for authentication and personalized attendance tracking within the system.

4.3.3 Login Page

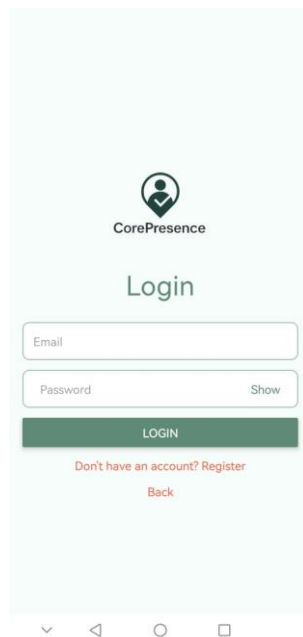


Figure 4.32: Login Page.

Figure 4.32 shows an existing users can log in with their registered email and password. New users without an account can quickly access the registration page by clicking the "Don't have an account? Register" link.

4.3.4 Login with Credentials

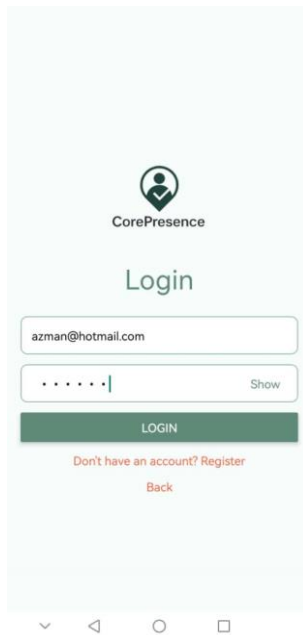


Figure 4.33 Login with Credentials Page.

Figure 4.33 displays the login form pre-filled with the user's email and password. After entering valid credentials, the user taps the 'LOGIN' button to access the system. This process prompts Firebase Authentication to verify the user's identity prior to granting access to the dashboard.

4.3.5 Clock-In/Clock-Out Process

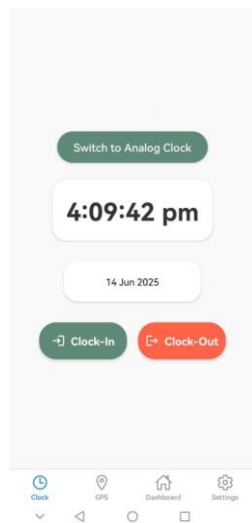


Figure 4.34: Main Clock Page.

Figure 4.34 illustrates when the user has successfully logged in, users are taken to the main page featuring a real-time clock, current date, and buttons for Clock-In and Clock-Out. This interface is consistent across all roles, including Employees, Admins, and Managers.

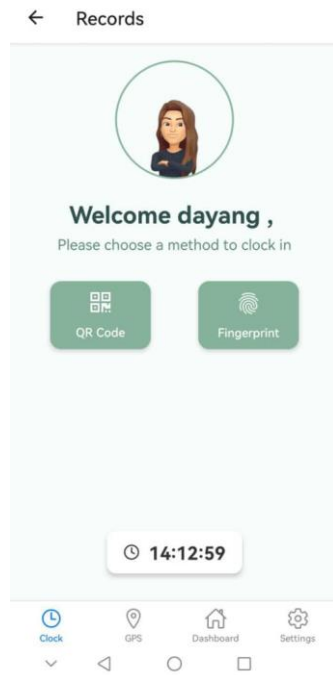


Figure 4.35 displays when the user initiates clock-in or clock-out, a screen appears prompting them to select an authentication method: QR Code or Fingerprint.

Figure 4.35: Authentication Method Selection Page.

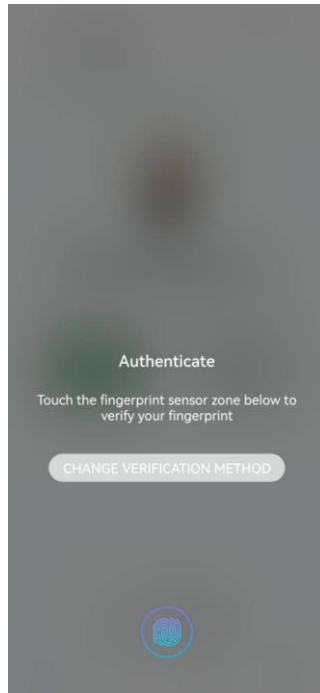
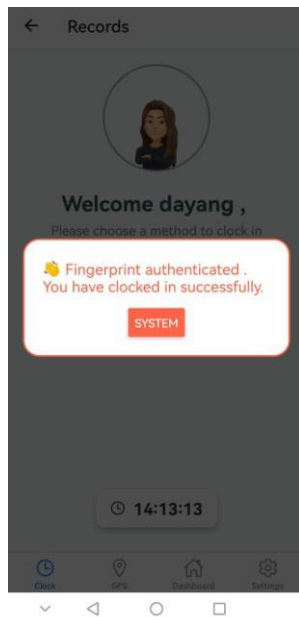


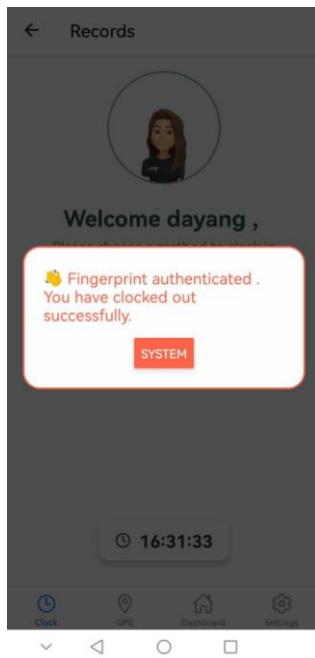
Figure 4.36 shows if the fingerprint option is chosen, the system prompts the user to scan their fingerprint using the device's sensor.

Figure 4.36: Fingerprint Authentication Prompt.



Upon successful fingerprint verification, a confirmation message appears indicating that the user has clocked in, as shown in Figure 4.37.

Figure 4.37: Fingerprint Clock-In Success Notification.



Likewise, a successful fingerprint-based clock-out triggers a distinct confirmation popup, as displays in Figure 4.38.

Figure 4.38: Fingerprint Clock-Out Success Notification.



Figure 4.39 shows when the users choosing QR authentication scan a dynamically generated QR code to record their attendance.

Figure 4.39: Fingerprint Clock-Out Success Notification.

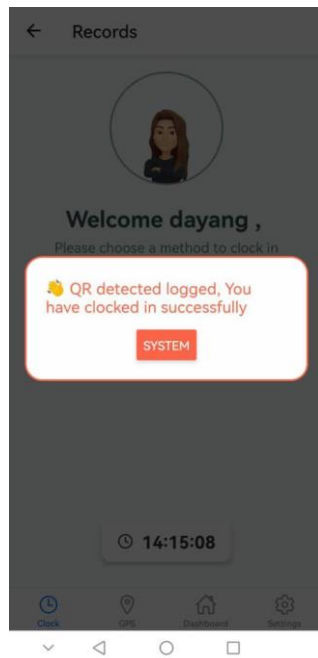


Figure 4.40 displays a popup appears to confirm a successful clock-in via the QR method.

Figure 4.40: Fingerprint Clock-Out Success Notification.

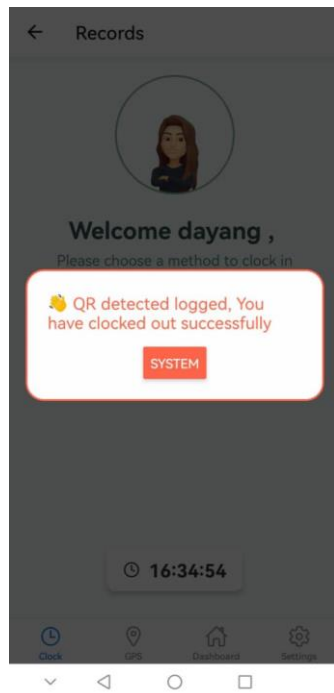


Figure 4.41 shows a popup appears to confirm a successful clock-out via the QR method.

Figure 4.41: Fingerprint Clock-Out Success

Notification.

4.3.6 Geolocation Feature Overview (After Successful Clock-In)

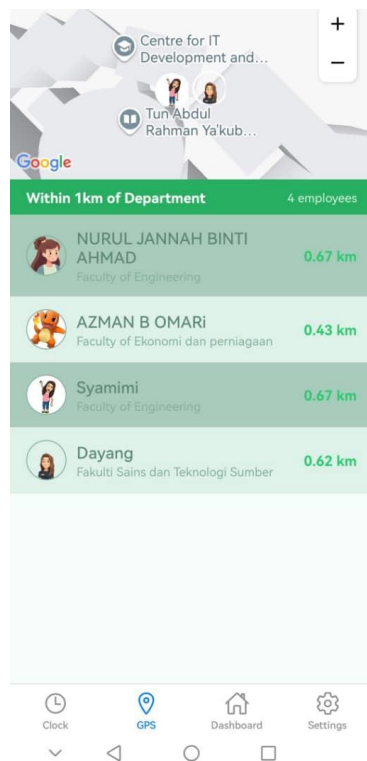


Figure 4.42: Manager Overview.

Managers can monitor the real-time locations of all employees comprising administrators via the map interface. The system filters the displayed locations based on a specified radius (e.g., within 1 km of the department's registered location). Each employee's profile picture, name, department, and distance from the reference point are presented, as shown in Figure 4.42.



Figure 4.43: Admin View.

Figure 4.43 shows that Admins can only view the geolocation of users within their own department. The map interface also allows filtering based on distance (e.g., a 1 km radius) to display department members near the designated location.

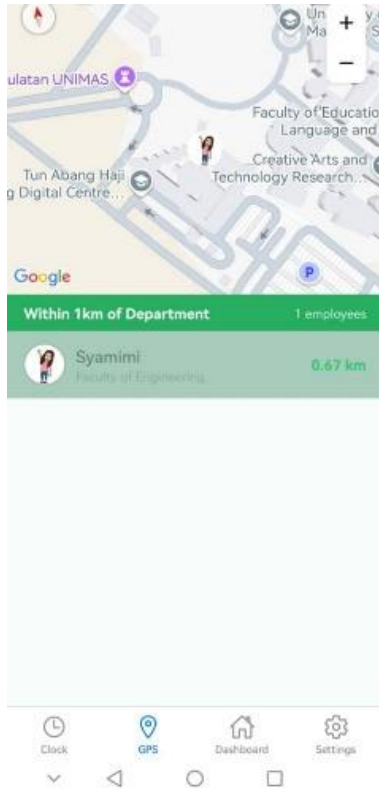


Figure 4.44 shows employees are restricted to viewing only their own location on the map. This measure upholds privacy and ensures that geolocation data is used solely for attendance verification.

Figure 4.44: Employee View.

4.3.7 Dashboard and Reporting



Figure 4.45: Dashboard- Manager View.

Figure 4.45 shows dashboard of the attendance summaries in bar chart form by department. Managers can filter the data by day or month and view attendance status (All, Present, Absent). Reports can be exported as CSV files.

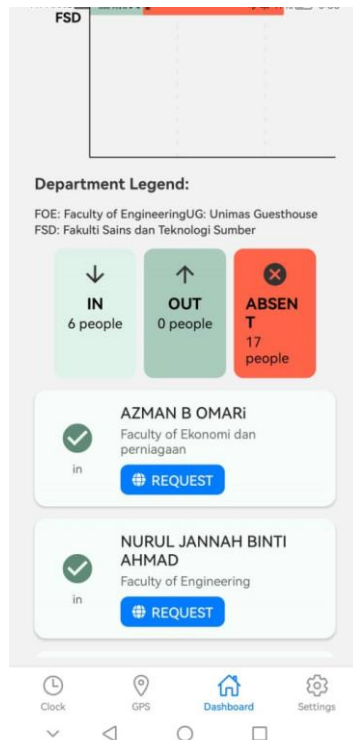
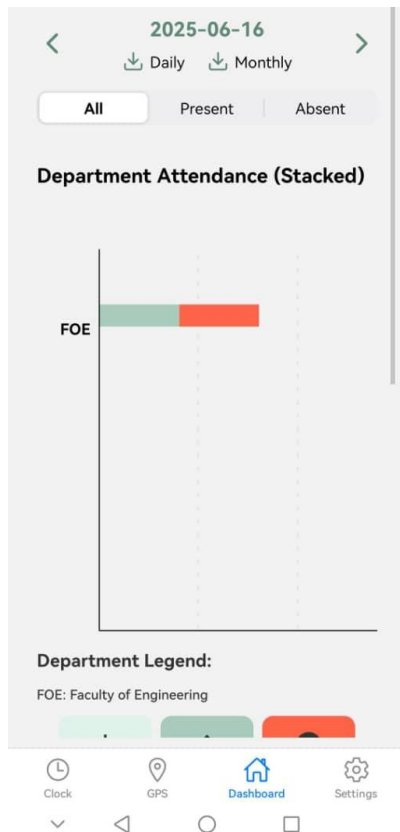


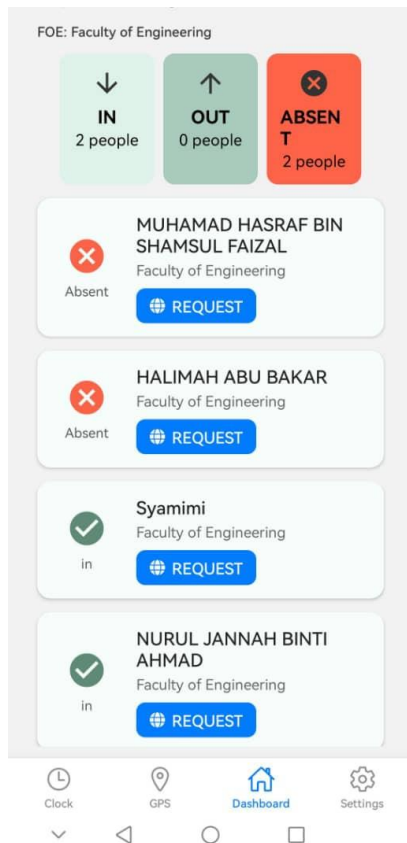
Figure 4.46 shows After applying filters, such as 'Present' filter. Managers can access detailed lists of individual attendance statuses and have the option to send requests.

Figure 4.46: Attendance List - Manager View.



Presents attendance data and charts accessible only to employees within the admin's assigned department, as shown in Figure 4.47.

Figure 4.47: Dashboard- Admin View.



Admins can see the daily attendance status (present/absent) of employees within their assigned department only as illustrates in figure 4.48.

Figure 4.48: Attendance List – Admin.

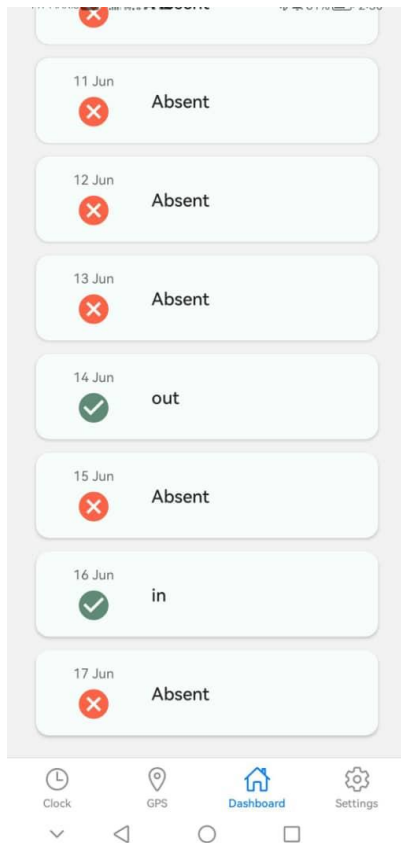


Figure 4.49 shows that the Employees can review their own attendance records in a simple list format, without charts or information about other users.

Figure 4.49: Dashboard – Employee View.

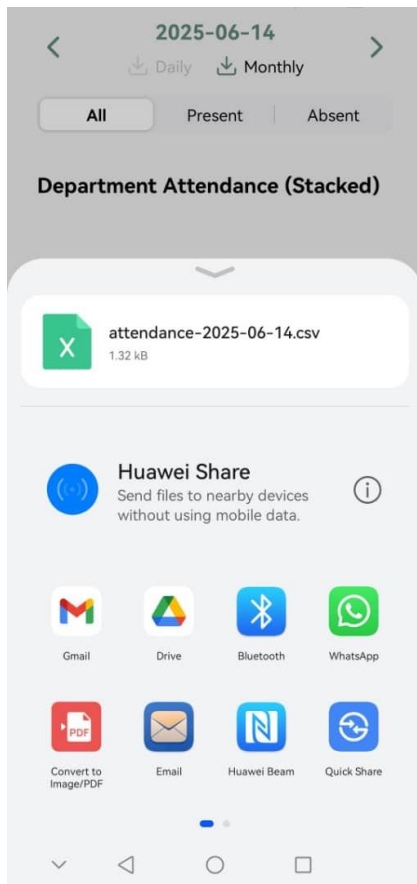


Figure 4.50 illustrates the process of exporting daily or monthly attendance reports in CSV format directly from the mobile app. After choosing the desired time range, the system generates a file (e.g., attendance-2025-06-14.csv) that users can share through various methods such as WhatsApp, Gmail, Google Drive, Bluetooth, and more.

Figure 4.50: Exported CSV attendance report ready for sharing via mobile.

	A	B	C	D
	Employee No	Department	Jun-01	Jun-02
1	2	MUHAMAD ROOM2STAY-	Absent	At
3	3	KHAIRUNSHUUnimas Gue-	Absent	At
4	4	Hhh Fakulti Sains-	Absent	At
5	5	Hanafi Fakulti Sains-	Absent	At
6	6	kamariah Fakulti Sains-	Absent	At
7	7	kucing Fakulti Sains-	Absent	At
8	8	Nur Syazrin Fakulti Sains-	Absent	At
9	9	NURUL JANIROOM2STAY-	Absent	At
10	10	Dayang Fakulti Sains-	Absent	At
11	11	NUR ADILA Faculty of El-	Absent	At
12	12	arnab Fakulti Sains-	Absent	At
13	13	NORSUHAD,Faculty of El-	Absent	At
14	14	AHMAD AZIUnimas Gue-	Absent	At
15	15	AMIR HAMZROOM2STAY-	Absent	At
16	16	HALIMAH AIROOM2STAY-	Absent	At
17	17	MOHAMMAIUnimas Gue-	Absent	At
18	18	MOHD FARIIUnimas Gue-	Absent	At
19	19	Syamimi Fakulti Sains-	Absent	At
20	20	Abg Senawi Fakulti Sains-	Absent	At
21	21	AZMAN B OIFaculty of El-	Absent	At
22	22	Absent		
23	23			
24	24			
25	25			
26	26			
27	27			
28	28			

Figure 4.51 displays a preview of the generated CSV attendance file opened on a mobile device. The file contains employee names, departments, and daily statuses (IN/OUT/Absent) over a specific time frame. It enables quick data review without needing to open external applications like Excel.

Figure 4.51: Attendance report previewed directly on a mobile device.

4.3.8 Notification Feature for Presence Verification

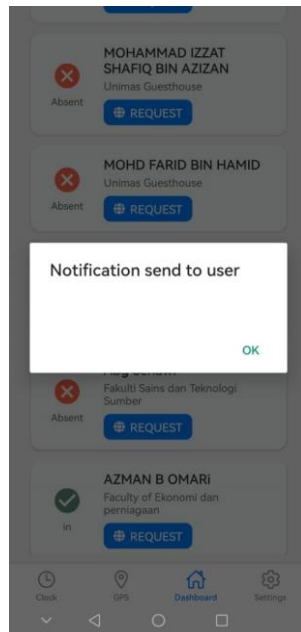


Figure 4.52: Notification confirmation shown after an Admin/Manager sends a location request to the user.

Figure 4.52 illustrates after employees have clocked in for a few hours, managers and admins can click the “Request” button to send a location update prompt. Managers can send this request to all employees, whereas admins are restricted to sending it only to those within their department. This feature enables managers and admins to manually trigger a notification that asks employees to confirm their presence post clock-in, ensuring ongoing attendance verification.

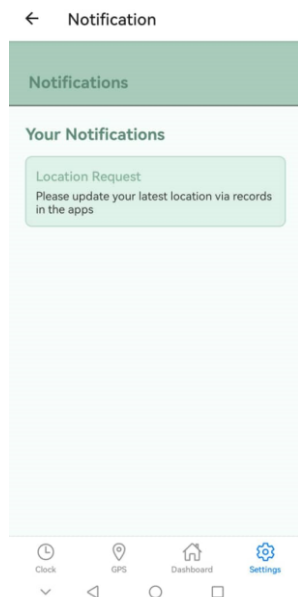


Figure 4.53 displays the employee's in-app notification panel, where request from managers or admins are received. The notification prompts the employee to update their location to support ongoing attendance verification.

Figure 4.53: In-app notification screen showing the employee's received location verification request.

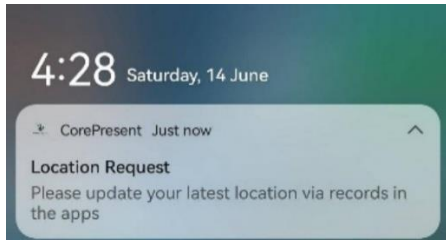


Figure 4.54 displays the targeted employee gets a real-time push notification on their phone via Firebase Cloud Messaging, prompting them to update their current location.

Figure 4.54: Push notification received on the user's device prompting location update.

4.3.9 Setting Page

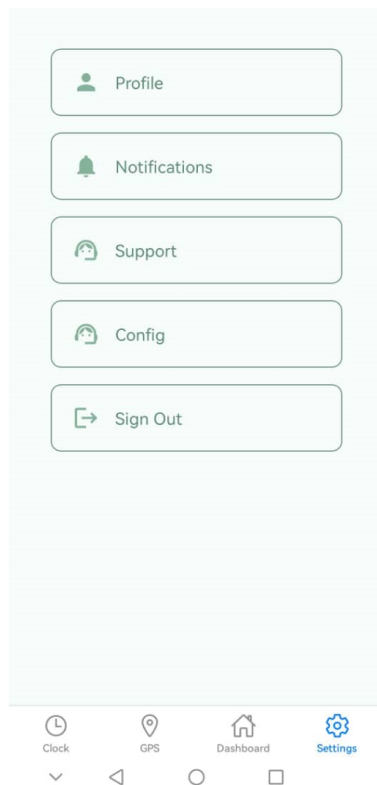


Figure 4.55 shows the Settings page acts as the main hub for users to manage their account preferences and configurations. Through this page, users can view profile information, track notifications, request support, customize app settings, or log out of the system. It guarantees that all users, regardless of role, have convenient access to essential account functions.

Figure 4.55: In-app notification screen showing the employee's received location verification request.

4.3.10 User Profile Page

The screenshot displays a mobile application interface for a user profile. At the top, there is a back arrow and the text 'Profile'. Below this, the word 'Profile' is centered above a circular profile picture of a woman. The form is divided into two sections: 'Your Information:' and 'Login & Security:'. Under 'Your Information:', there are two text input fields; the first contains 'Dayang' and the second contains 'dayang'. Under 'Login & Security:', there is a text input field containing '012-3723424' and a password field with six dots and a toggle eye icon. A green 'Save' button is positioned below the password field. At the bottom of the screen, there is a navigation bar with four icons: a clock labeled 'Clock', a location pin labeled 'GPS', a house labeled 'Dashboard', and a gear labeled 'Settings' (which is highlighted in blue). Below the navigation bar are four standard Android navigation icons: a downward arrow, a left-pointing triangle, a circle, and a square.

Figure 4.56: Profile Page.

When users access the User Profile page, they see editable fields showing their personal information such as full name, display name, phone number, and password. This enables them to keep their contact and login details current. The system also allows updating the profile photo for enhanced personalization, as shown in Figure 4.56.

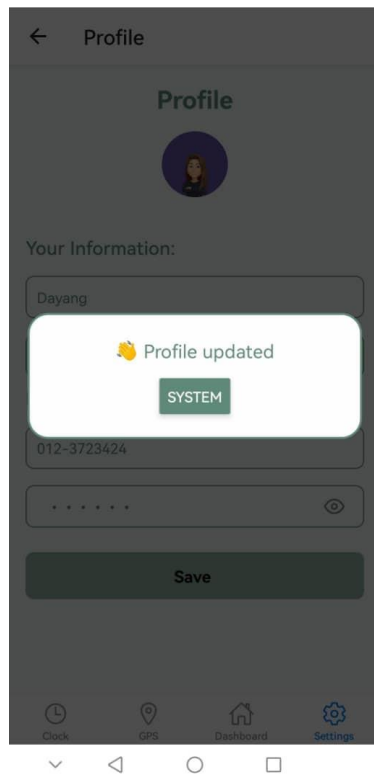


Figure 4.57 shows After any changes are made, a confirmation popup promptly notifies the user that their profile has been successfully updated.

Figure 4.57: Profile Update Confirmation.

4.3.11 Config Page

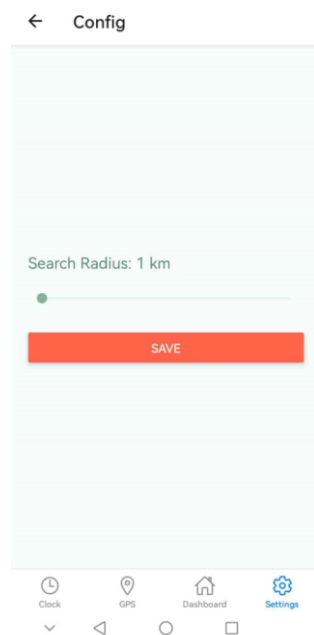


Figure 4.58: Config Setting Page.

Figure 4.58 shows the Config page includes a location tracking feature where managers and admins can set a GPS radius for attendance validation. Managers can apply the radius system-wide, while admins can adjust it only for their own department.

4.3.12 Support Page

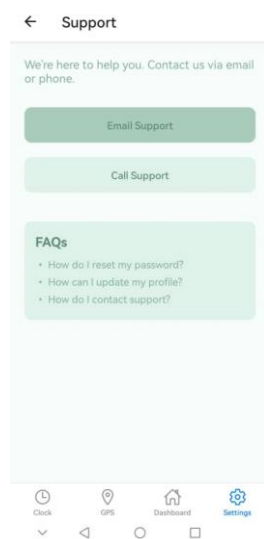


Figure 4.59: Support Page.

As illustrates in Figure 4.59 to support users requiring assistance or seeking clarification, the Support page provides direct contact options. Users can get in touch via email or phone, and a brief FAQ section addresses common questions such as password resets and account updates. This ensures assistance is readily available, improving user experience and trust in the system.

4.4 Summary

This chapter explains the implementation of the CorePresence mobile application, focusing on the setup of essential software, integration of key services, and demonstration of core system functionalities. The development environment was configured using tools such as Node.js, Expo CLI, Visual Studio Code, and Android Studio. Critical backend features including authentication, data storage, and notification services were enabled through Supabase and Firebase Cloud Messaging (FCM). These services required no separate installations, as they were integrated directly within the codebase using JavaScript libraries. The application overview highlights various user interfaces and principal features available to Employees, Admins, and Managers. Key functionalities include user registration and login, multi-modal clock-in/out options for QR and fingerprint, geolocation verification, role-specific dashboards, real-time push notifications, CSV attendance report generation, and administrative configuration options like setting location radius preferences. Each interface and feature are

depicted with actual implementation screenshots, demonstrating the system workflow from login through to report export. In summary, this chapter highlights how frontend technologies, cloud-based backend services, and intuitive UI design combine to create a dependable and efficient attendance tracking system.

CHAPTER 5: TESTING

5.1 Introduction

This chapter outlines the testing process undertaken to verify the functionality and performance of the Multi-Modal Authentication and Geolocation Approach for Attendance Verification, CorePresence mobile application. Following the Rapid Application Development (RAD) methodology, testing was performed iteratively throughout the development cycle, enabling early feedback and ongoing enhancements. Testing is crucial to confirm that the system fulfils its intended requirements and operates correctly across various scenarios and user roles.

Both functional and non-functional testing were conducted. Functional testing focused on validating key features such as user authentication, multi-modal clock-in/out, geolocation validation, real-time notifications, and role-based dashboards. Unit and system testing evaluated the accuracy of individual components together with complete workflows.

Furthermore, non-functional testing was conducted to evaluate elements such as usability, reliability, and user satisfaction. User Acceptance Testing (UAT) was incorporated to validate the system's preparedness from the end-users' perspective, ensuring that employees, admins, and managers can navigate the application seamlessly in a real working environment.

5.2 Functional Testing

This section describes the process of validating that the CorePresence mobile application performs its required functionalities as expected. Functional testing is essential for confirming that each system component works correctly and meets the specified requirements.

Following to the Rapid Application Development (RAD) methodology, functional testing was performed iteratively alongside the development of new features. This approach enabled early detection and resolution of bugs, ensuring reliable functionality across modules including multi-modal clock-in/out, geolocation tracking, real-time notifications, and report generation. Functional testing consisted of unit testing and system testing, each with its specific scope and purpose.

5.2.1 Component Testing

Component testing requires assessing specific features of the CorePresence mobile application to verify that each feature operates independently and produces the expected results. Conducted throughout the iterative development cycles of the RAD methodology, this testing validated core functions such as registration, login, multi-modal clock-in (QR/fingerprint), geolocation tracking, and notification triggers. The purpose of component testing is to verify that each separated module whether button logic, authentication, or data submission executes successfully before integration with other parts of the system. These tests were carried out on a Huawei P30 Pro using a development build.

5.2.1.1 Results Table for Component Testing

Table 5.1: Component Testing Result for Registration Module

Module Name: Registration Test Case Description: To ensure that the system correctly handles various user registration inputs. Testing Objective: Ensure that valid users can register successfully and invalid or incomplete inputs are properly handled. Remarks: Applies to all user roles (Employee, Admin, Manager).								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity Failure	Notes
TC01	Registration with valid input	1. Open Register Page. 2. Enter valid name, email, phone, and password. 3. Click Save.	Name: Muhammad Email: muhammad@gmail.com Phone: 0123456789 Password: 123456	Account created, user redirected to login	Account created, user redirected to login	Pass	Medium	Profile picture is optional
TC02	Missing required name field	1. Open Register Page. 2. Leave name	Name: [Blank] Email: muhammad@gmail.com Phone: 0123456789 Password: 123456	"Name required" message displayed	Registration prevented, but no message displayed	Pass	Low	Field validation functioning

		field blank.						
		3. Fill other fields and click Save						
TC03	Invalid email format	1. Open Register Page. 2. Enter invalid email format. 3. Submit form.	Email: muhammadgmail.com	"Invalid format" email error message displayed	Registration prevented, but no error message displayed	Pass	Low	Basic input validation triggered
TC04	Duplicate email registration	1. Open Register Page. 2. Enter email already in use. 3. Submit form	Email: dayang@gmail.com	"Email already exists" error message displayed	Registration prevented, but no error message displayed	Pass	Medium	Lacks user feedback; error message not shown

Table 5.2: Component Testing Result for Login Module

Module Name: Login Test Case Description: To validate that the login functionality correctly authenticates valid users and handles invalid login attempts appropriately. Testing Objective: Ensure the system allows access for users with correct credentials and deny attempts with incorrect or incomplete inputs. Remarks: Applies to all roles (Employee, Admin, Manager).								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity Failure	Notes
TC05	Login with valid credentials	1. Open Login Page. 2. Enter valid email and password. 3. Click Login.	Email: dayang@gmail.com Password: 123456	User successfully logged in and redirected to clock-in Page.	User logged in and redirected to clock-in Page.	Pass	Medium	Role-based redirection verified.
TC06	Invalid password	1. Open Login Page. 2. Enter valid email, wrong password. 3. Click Login.	Email: dayang@gmail.com Password: Wrong password input	"Invalid email or password" error message displayed.	"Invalid email or password" error message displayed.	Pass	Low	Password validation working.
TC07	Invalid email	1. Open Login Page. 2. Enter an unregistered email. 3. Click Login.	Email: notfound@gmail.com Password: 123456	"Invalid email or password" error message displayed.	"Invalid email or password" error message displayed.	Pass	Low	Unregistered email correctly handled.
TC08	Missing email field	1. Open Login Page. 2. Leave email empty. 3. Click Login.	Email: [Blank] Password: 123456	"Invalid email or password" error	"Invalid email or password" error	Pass	Low	Basic field validation triggered.

				message displayed.	message displayed.			
TC09	Missing password field	1. Open Login Page. 2. Leave password empty. 3. Click Login.	Email: dayang@gmail.com Password: [Blank]	"Invalid email or password" error message displayed.	"Invalid email or password" error message displayed.	Pass	Low	Basic field validation triggered.
TC10	SQL injection attempt (logic scenario)	1. Open Login Page. 2. Enter SQL injection string in email or password. 3. Click Login.	Email: ' OR 1=1-- Password: anything	Login blocked, error message shown.	"Invalid email or password" error message displayed.	Pass	High	Input sanitization confirmed.

Table 5.3: Component Testing Result for Attendance Module (Clock-In / Clock-Out)

Module Name: Attendance (Clock-In / Clock-Out) Test Case Description: To verify that all users (Employee, Admin, Manager) can execute clock-in and clock-out using QR code or fingerprint, and that incorrect attempts are handled appropriately. Testing Objective: Ensure clock-in/out logic works with geolocation and triggers appropriate feedback or restriction. Remarks: Applies to all roles (Employee, Admin, Manager). Role-based dashboard access follows successful clock-in.								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity Failure	Notes
TC11	Successful clock-in (QR scan)	1. Log in as user. 2. Tap clock-in. 3. Scan valid QR. 4. System verifies GPS within radius.	QR: Valid GPS: Within 1KM	Clock-in recorded, user notified of success	Clock-in recorded, confirmation shown	Pass	Medium	Applies to Employee, Admin, and Manager

TC12	Clock-in with invalid QR	1. Log in. 2. Tap clock-in. 3. Scan invalid QR.	QR: Expired/incorrect	"Invalid QR code" error message shown	Clock-in blocked with error message	Pass	Low	Error handling verified
TC13	Successful clock-in (fingerprint)	1. Log in. 2. Tap clock-in. 3. Tap fingerprint option. 4. Scan registered thumbprint.	Fingerprint: Registered	Clock-in successful, data saved	Clock-in recorded with success message	Pass	Medium	-
TC14	Clock-in outside allowed radius	1. Log in. 2. Tap clock-in. 3. Attempt clock-in with GPS outside radius.	GPS: >1KM from office	The map should indicate "Outside 1km Radius" with red highlighting.	"Outside 1km Radius" in red on the GPS tracking interface. No tracking inside the allowed radius occurs.	Pass	Medium	Radius configurable by Admin/ Manager
TC15	Clock-out without prior clock-in	1. Open app. 2. Tap Clock-out directly.	—	"No active clock-in session" error	Clock-out blocked with message	Pass	Low	Enforces correct clock-in/out sequence
TC16	Attempt without internet	1. Log in. 2. Turn off internet. 3. Tap clock-in.	No network/location	"Location access/internet required" message shown	Operation blocked with error	Pass	High	Prevents spoofing or incomplete records

Table 5.4: Component Testing Result for Notification Module

Module Name: Notification (Manual Presence Prompt) Test Case Description: To ensure that notification prompts for verifying continued presence are successfully sent by Admins/Managers and received by Employees. Testing Objective: Verify push notification logic works correctly across roles and that FCM triggers behave as expected. Remarks: Notifications are triggered after a set time post clock-in or manually by Admins/Managers.								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity Failure	Notes
TC17	Manager sends presence prompt notification	1. Manager logs in. 2. Selects employee. 3. Triggers location prompt manually	User: Employee clocked in Role: Manager	Push notification sent to employee's device	Notification triggered and sent via FCM	Pass	Medium	Trigger available after clock-in
TC18	Admin sends notification to own department	1. Admin logs in. 2. Views employee list. 3. Sends presence prompt.	Employee: Under same department Role: Admin	Push notification delivered to targeted employee.	Message appears on employee's phone.	Pass	Medium	Access limited to same department
TC19	Employee receives notification prompt	1. Clock in. 2. Wait for manager/admin to send presence prompt.	—	Notification received by employee	Employee sees alert on their phone	Pass	Medium	Device must allow push notifications
TC20	Trigger fails if employee has not clocked in	1. Admin/Manager selects an employee. who	Employee status: Not clocked in	Error message: "User has not clocked in yet"	Notification not sent; no error message shown	Pass	Low	Silent failure; system should

		2. has not clocked in. Try to send presence prompt.						ideally display feedback to inform the sender
TC21	Internet disabled on employee phone	1. Admin sends notification. 2. Employee's phone is offline.	—	Notification not received immediately; queued or dropped by FCM	Delay in delivery or missed notification	Pass	High	Behavior depends on FCM & device state

Table 5.5: Component Testing Result for Geolocation View Module

Module Name: Geolocation View Test Case Description: To ensure that users can view real-time location data based on their role and that access is limited according to their permissions. Testing Objective: Guarantee that Admins and Managers monitor users within specified radius limits, whereas employees are limited to their own location view. Remarks: Location visibility is triggered after successful clock-in.								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity Failure	Notes
TC22	Manager views all users' locations	1. Log in as Manager. 2. Navigate to Geolocation page. 3. Select full view.	Role: Manager Employees: Multiple	Manager sees all Admins and Employees on the map, filtered by radius	All users displayed within radius, others marked "Outside"	Pass	Medium	Works across departments
TC23	Admin views only own department	1. Log in as Admin. 2. Open map page. 3. Verify only department staff shown.	Admin Dept: eg., Faculty Of Engineering	Admin sees only users from "Faculty Of Engineering" department	Users filtered correctly by department	Pass	Medium	Scope limited to one department

TC24	Employee views own location only	1. Log in as Employee. 2. Go to Geolocation.	–	Map displays only the employee's own location	Own pin displayed on map	Pass	Low	No access to others' location
TC25	User shown as "Outside Radius"	1. Clock in. 2. Move beyond defined radius. 3. Reload map view.	Location: 5KM from site	User icon appears in red with label "Outside 1km Radius"	Red icon + message shown correctly	Pass	Medium	Applies to Employee, Admin, Manager
TC26	Change radius preference	1. Manager/Admin goes to Settings. 2. Adjust radius value to 2KM. 3. View updated map.	Radius: Changed from 1KM to 2KM	Map updates to reflect new radius boundaries	New locations included based on 2KM radius	Pass	Medium	Preference stored per admin/manager
TC27	GPS off or denied	1. Log in. 2. Deny location permission. 3. Attempt to access map view.	GPS: Off / denied	Location unavailable warning displayed	Warning message shown, no map updates	Pass	High	Critical for GPS-dependent features

Table 5.6: Component Testing Result for Dashboard & Report Module

Module Name: Dashboard & Report Test Case Description: To make sure that users can view role-based dashboard summaries and generate daily/monthly attendance reports accurately. Testing Objective: Ensure correct data display, filtering, and CSV export per user role. Remarks: Applies to all roles, but with different access levels.								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity Failure	Notes
TC28	Manager views overall attendance	1. Log in as Manager. 2. View dashboard.	Role: Manager	Graphs and list display all users'	All employees and admins	Pass	Medium	Manager has full visibility

		3.	Filter by date, department, and status.		attendance by filters.	visible; filters work correctly			
TC29	Admin views department dashboard	1. 2.	Log in as Admin. View dashboard for assigned department.	Department: Faculty of Engineering	Only employees from Faculty of Engineering are visible.	Only department-specific data displayed	Pass	Medium	Limited to one department
TC30	Employee views personal attendance	1. 2.	Log in as Employee. Open dashboard.	Role: Employee	Only user's own records displayed, no filters.	Correct self-data shown, no other users	Pass	Low	No graph, only self-summary
TC31	Filter by Present/Absent	1. 2.	Manager/Admin selects date. Filters Present or Absent.	Status: Absent	List updates to reflect filtered attendance.	Status filter reflects correct count	Pass	Medium	Important for monitoring performance
TC32	Generate CSV report (Daily)	1. 2.	Manager/Admin clicks "Generate Report". Select Daily.	Date: 2025-06-09	Report downloads in CSV format	File downloaded to device storage	Pass	Medium	Filename: attendance-2025-06-09.csv
TC33	Open CSV on laptop	1. 2.	Transfer file to laptop. Open using Excel.	File: attendance-2025-06-09.csv	Displays list of users with name, department, status	Data appears as expected in Excel	Pass	Low	Fields shown: Name, Department, Status only

5.2.2 Integration Testing

Integration testing evaluates the CorePresence application as a fully integrated system. It verifies if various modules such as registration, login, attendance tracking, geolocation, notifications, and reporting function correctly when assembled into complete user workflows. This phase confirms that each user role (Employee, Admin, and Manager) can effectively carry out their designated tasks in realistic scenarios. These include processes like logging in, clocking in/out, sending and receiving presence prompts, and generating reports.

Integration testing was performed following unit testing and follows the Rapid Application Development (RAD) methodology, which emphasizes iterative integration and continuous validation throughout development.

5.2.2.1 Results Table for Integration Testing

Table 5.7: Integration Testing Result for CorePresence Application

Purpose: To ensure that all integrated components of the CorePresence application operate as expected in real-world situations. Methodology: Test entire workflows from login to attendance recording, notification triggering, and reporting for each user role.								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity	Notes
ST01	Full attendance flow (Employee)	1. Login as Employee. 2. Clock-in via QR. 3. Confirm location. 4. Dashboard shows present status.	Valid credentials + QR and Location within radius	Clock-in recorded, status updated on dashboard	Successfully clocked-in and dashboard updated	Pass	Medium	Role: Employee
ST02	Clock-in + notification + dashboard (Admin)	1. Login as Admin. 2. Clock-in. 3. Open dashboard to view employees. 4. Send presence request notification to employee.	Admin with department access Employee clocked in	Admin clocked-in, views employee list, sends notification	Notification sent, admin clock-in recorded	Pass	Medium	Notification logic verified
ST03	Manager dashboard + map view + CSV generation	1. Login as Manager. 2. View dashboard summary. 3. Open geolocation map. 4. Export CSV report (daily)	Manager account + multiple users clocked in	All data visible in dashboard, map pins accurate, CSV downloaded	CSV correct, map accurate, filtering works	Pass	High	Combined flow, reporting and visual data

ST04	Clock-out and data refresh	1. Login. 2. Clock-in. 3. Clock-out. 4. Check status on dashboard and reports.	Role: Employee	Clock-out registered, final status “Present”, dashboard and report updated	Clock-out successful, data synced to dashboard and CSV	Pass	Medium	Confirms flow closure
ST05	Access denied when not clocked-in (notification attempt)	1. Login as Admin. 2. Try to send presence prompt to an employee who has not clocked in	Admin + employee not clocked-in	Notification not sent, system does not respond or shows error	Notification not sent, no visible alert	Pass	Low	Silent handling of invalid logic
ST06	Employee clocks in outside allowed geofence radius	1. Employee logs in. 2. Attempts clock-in while located >1KM from workplace. 3. Open map as Admin/Manager. 4. Check if shown as “Outside Radius”	Location: 5KM from designated site	Clock-in prevented OR recorded but marked in red on map with alert	Marked as “Outside 1KM Radius” in red; visible to Admin/Manager	Pass	Medium	Visually shown on map but flagged for admin follow-up

5.3 Non-Functional Testing

This section describes the non-functional testing activities carried out for evaluating the quality attributes of the CorePresence mobile application beyond its basic functionality. Non-functional testing is essential for analyzing user experience, system reliability, and practical usability. Following the Rapid Application Development (RAD) methodology, these non-functional aspects were tested iteratively throughout development to ensure continuous enhancement. The goal was to ensure the application is user-friendly, consistently delivers attendance features (like notifications and geolocation), and satisfies acceptance criteria for all user roles, including employees, administrators, and managers. Non-functional testing for this project includes Usability Testing, Reliability Testing, and User Acceptance Testing (UAT).

5.3.1 Usability Testing

Usability testing in this project aimed to assess the intuitiveness, accessibility, and overall efficiency of the CorePresence mobile application from the developer's standpoint. As a form of non-functional testing, it was carried out internally by the programmer during development to ensure that key functionalities such as registration, clock-in/out, and dashboard navigation could be executed seamlessly without confusion or errors.

In line with the Rapid Application Development (RAD) methodology, usability feedback and enhancements were applied iteratively throughout the development cycle. While no external users were involved at this stage, the testing thoroughly evaluated aspects like visual layout clarity, interaction responsiveness, onboarding flow, and variations across role-based interfaces. This internal testing phase was essential for identifying and resolving potential design or usability issues early on, thereby improving the user experience ahead of broader evaluations like User Acceptance Testing (UAT).

5.3.1.1 Results Table for Usability Testing

Table 5.8: Usability Testing Result for Core Function Modules

Module Name: Core Function Modules (GPS, Notification, Clock-In/Out, Session) Test Case Description: To verify that key features such as geolocation, notification delivery, session management, and clock-in/out behave consistently across different usage conditions. Testing Objective: Ensure that the CorePresence mobile app's critical functions stay reliable and perform accurately, even after frequent use and under different user interactions or network conditions. Remarks: Tested using Huawei P30 Pro under both Wi-Fi and mobile data conditions.								
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Result	Actual Result	Status	Severity	Notes
RT01	GPS consistently detects valid locations	1. Clock in at office. 2. Refresh location on map multiple times.	Device location ON	Location updates accurately every time	GPS consistently tracked and mapped	Pass	Medium	Radius logic works as expected
RT02	Notification delivery via FCM	1. Admin/Manager sends presence prompt. 2. User receives it within expected delay.	Active network connection	Notification delivered within few seconds	Notification received promptly	Pass	High	Delays if network is weak
RT03	Clock-in/out data stored reliably	1. Clock in → Restart app → Clock out.	Same user session	Data is saved and reflected on dashboard	Clock-in/out persisted correctly	Pass	Medium	Data saved in Supabase backend

		2. Check if attendance data persists.						
RT04	Offline behavior (no GPS/internet)	1. Turn off internet and GPS. 2. Attempt clock-in.	No network / GPS	System blocks action and displays proper error	Clock-in blocked with location/network error	Pass	High	Prevents incomplete or spoofed data
RT05	Exported report reflects current data	1. Manager generates report after employee clocks in. 2. Open downloaded CSV.	Filter: Daily	CSV includes updated user attendance info	File accurate and synced with live system	Pass	Medium	Confirms real-time data connection to report generation
RT06	Repeated clock-in attempt handling	1. Clock in. 2. Tap clock-in again on same day	Same user, same date	The app permits a second attempt but retains only the initial clock-in timestamp.	Multiple taps allowed, first only time recorded.	Pass	Low	Timestamp logic confirmed
RT07	Consistency during role switching sessions	1. Log in as Admin. 2. Logout. 3. Log in as Manager. 4. View dashboard and map	Different roles, but use the same device	Dashboard and permissions match role accurately	New role and data loaded correctly	Pass	Medium	Prevents session leak between roles

5.3.2 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) was conducted to evaluate if the CorePresence mobile application meets its functional and usability requirements in real-world scenarios. This phase of non-functional testing involved 12 participants who obtained the system for testing and provided feedback. Participants were recruited from WhatsApp contact lists and included both working professionals and university students, representing the target user demographic. All testers used Android smartphones during the evaluation. Each participant received the CorePresence APK installation file along with a straightforward user manual to guide them through the app's features and navigation. After using the system, they completed a Google Form survey designed to evaluate usability, attendance functionality, interface design, and overall user experience. The UAT evaluation form used to gather participant feedback is included in Appendix C.

5.3.2.1 Analysis on User Acceptance Survey

The questionnaire comprises four sections, specifically General Usability, Attendance Features, Interface Design, and Overall Experience. Each section was crafted to evaluate particular aspects of the CorePresence mobile application from the viewpoint of real users. Respondents were asked to rate their agreement with various items on a 5-point Likert scale ranging from “Strongly Disagree” to “Strongly Agree.”

As shown in the Figure 5.1, most respondents evaluated the application highly for usability. Seven participants strongly agreed that the CorePresence app is easy to navigate and understand, while five agreed. A comparable pattern was observed for system responsiveness, with seven strongly agreeing that the app reacts promptly to user actions, followed by five who

agreed. These findings suggest that the app delivers a smooth and user-friendly experience for the majority of users.

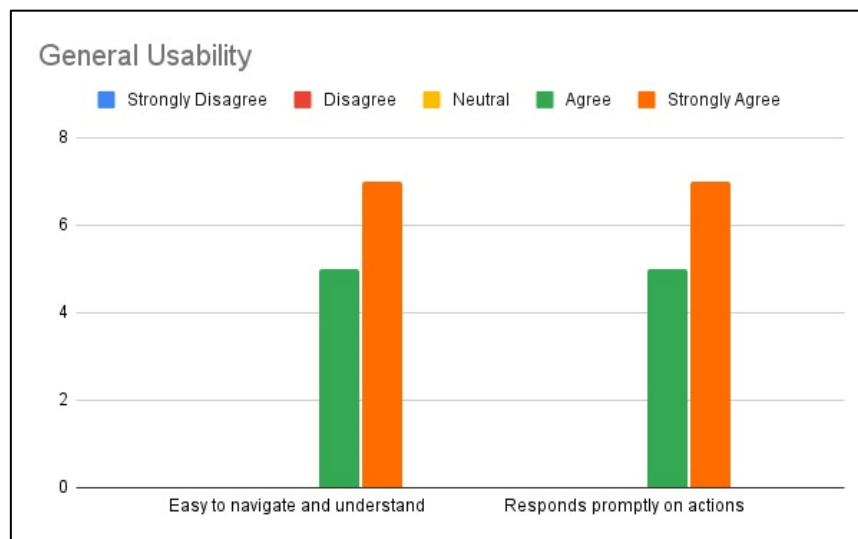


Figure 5.1: General Usability of CorePresence.

Based on responses regarding Figure 5.2, attendance features, eight participants strongly agreed that the clock-in/out functionality operates smoothly, while four concurred. For geolocation accuracy, seven strongly agreed and five affirmed, indicating positive feedback on the app's real-time location tracking. Similarly, seven respondents strongly agreed they could clearly view their attendance records, with five agreeing. Notably, nine users strongly agreed that the app effectively displays attendance status after clock-in, validating successful actions. Overall, most respondents considered the attendance module reliable and functional.

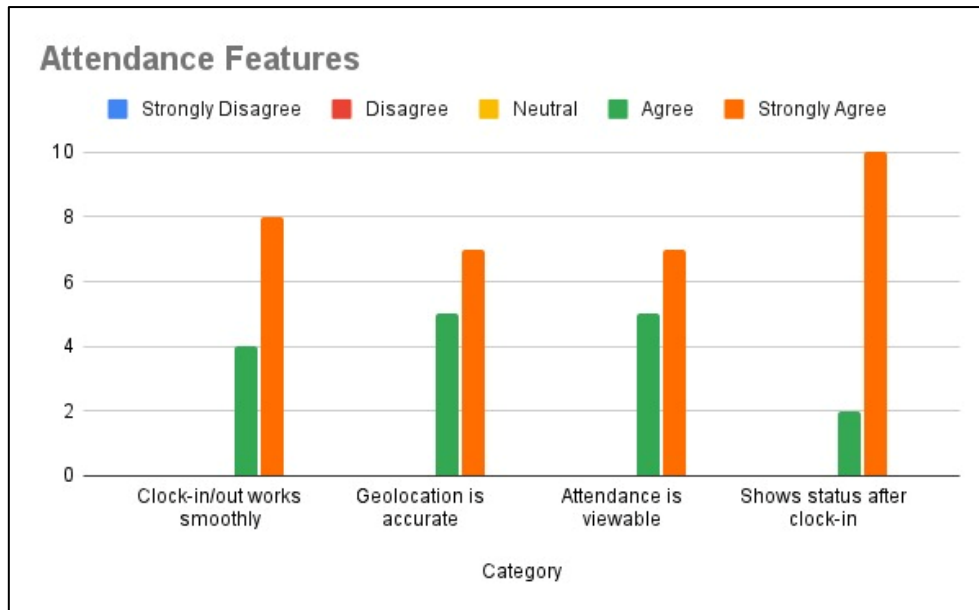


Figure 5.2: Attendance Features for CorePresence.

Regarding Figure 5.3, interface design, eight respondents strongly agreed that the app's features are well-organized, with four agreeing. When questioned about the font style which including size, type, and readability, nine users strongly agreed and three agreed. These findings indicate that the design layout and visual presentation of the CorePresence app were well received, positively impacting overall usability.

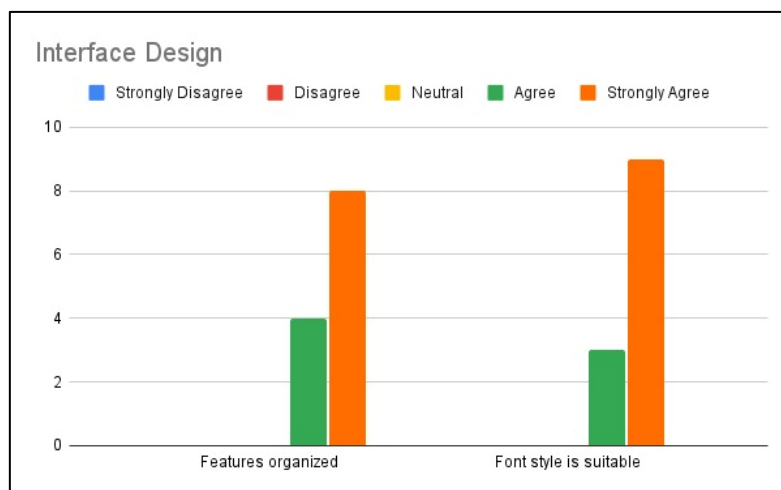


Figure 5.3: Interface Design of CorePresence.

Figure 5.4 illustrates the responses from the final section, where 11 respondents strongly agreed that the app is useful, and one agreed. Additionally, eight users strongly

expressed confidence in using CorePresence in a real work environment, with four more agreeing. These results clearly demonstrate that the application not only fulfils its intended functionality but also instils confidence among users for professional use.

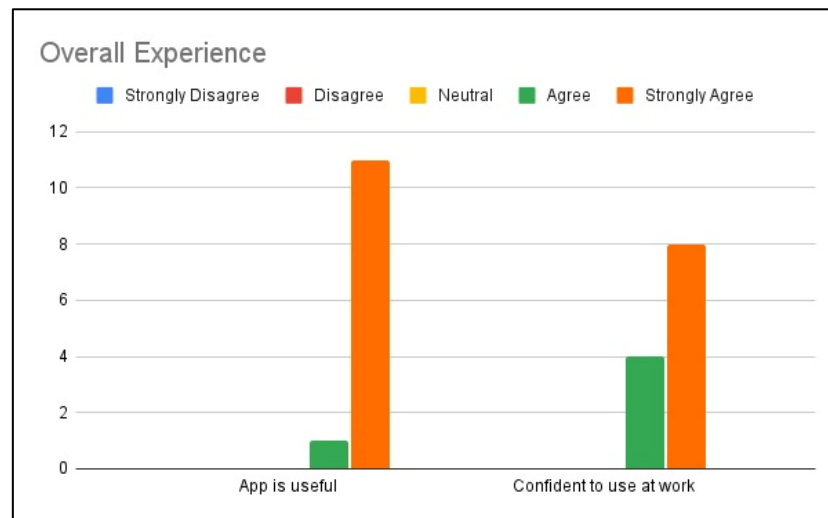


Figure 5.4: Overall Experience of CorePresence.

5.4 Discussion on Results

Even though the CorePresence mobile application met its intended goals and performed well during testing, it is not yet ready for real-world deployment or publication. This is primarily due to realistic limitations associated with the tools and technologies used.

The app was developed using React Native, with Supabase (PostgreSQL) as the backend and Firebase Cloud Messaging (FCM) for real-time notifications. While these tools enable fast development, they may encounter scalability challenges, such as limited user capacity and delayed data synchronization particularly within Supabase's free-tier platform.

The geolocation feature, reliant on device GPS, can yield unreliable results influenced by signal strength and location permissions. Additionally, the manual presence verification prompt, while functional, lacks automatic processing and real-time logging capabilities typically required in enterprise systems.

As of now, the app is available only on Android, with iOS deployment excluded due to technical constraints, limiting its cross-platform usability. Furthermore, while QR code and fingerprint authentication were implemented, more advanced verification methods like facial recognition were left out, restricting its applicability in high-security environments.

Overall, CorePresence shows potential as an attendance tracking solution for smaller-scale deployments. However, improvements in automation, scalability, and platform compatibility are necessary before it can be effectively used in real-world professional settings.

5.5 Summary

To sum up, testing plays a crucial role in the system development lifecycle to ensure the CorePresence mobile application functions as intended across various user roles and scenarios. This chapter addressed both functional and non-functional testing approaches, performed in alignment with the Rapid Application Development (RAD) methodology.

Functional testing included unit and system testing, focusing on key features such as user registration, login, clock-in/clock-out, geolocation tracking, and notification triggers. Each module was evaluated to verify both individual components and their integration within the overall system.

Non-functional testing focused on assessing the application's qualities beyond core functionality. This included reliability testing, which evaluated the consistency of geolocation and attendance processes, as well as User Acceptance Testing (UAT), which collected feedback from real users regarding usability, interface design, and overall satisfaction.

Together, these testing efforts offered valuable insights into system stability, user experience, and readiness for future enhancements or deployment.

CHAPTER 6: CONCLUSION AND FUTURE WORKS

6.1 Introduction

This chapter concludes the development of the CorePresence mobile application by summarizing the accomplishments achieved throughout the project and acknowledging the limitations encountered during implementation. It also highlights potential future enhancements aimed at improving the system's overall performance and usability. The final evaluation confirms that the system fulfils its primary objectives outlined in Chapter 1, particularly in providing accurate, real-time attendance tracking through geolocation and multi-modal clock-in methods.

6.2 Objectives and Achievements

The CorePresence project was developed to enhance employee attendance systems through the incorporation of geolocation, role-based access, and multi-modal authentication. The Table 6.1 below aligns the original objectives with the outcomes achieved:

Table 6.1: Details of Objectives and Achievements.

Objective	Achievement
To develop a mobile application that supports clock-in and clock-out attendance functionality with geolocation tracking	The system logs attendance with live GPS coordinates and verifies clock-ins within a user-specified radius.
To implement multi-modal attendance methods, such as QR code and fingerprint scanning	Users can verify their presence using fingerprint or QR code authentication, both initiated within the app.
To create administrative and managerial roles for managing attendance data and generating reports	Admins and managers have the ability to view staff attendance, apply filters by department or date, and export reports in CSV format.
To incorporate a real-time notification system for presence verification	The app utilizes Firebase Cloud Messaging (FCM) to deliver push notifications for presence verification.
To design a user interface that is mobile-friendly and tailored to different user roles	The app provides distinct interfaces for employees, admins, and managers to

6.3 Project Limitations and Constraints

Although the CorePresence mobile application meets its intended objectives and provides a functional attendance system, it still faces several limitations that hinder its readiness for large-scale or enterprise deployment. These challenges mainly stem from technical dependencies, time constraints, and the project scope defined during development.

Platform Limitation:

The current version of CorePresence is only available on Android. Due to time limitations, limited testing resources, and the lack of Apple development tools such as Xcode and an iOS testing device, the project could not extend support to iOS users. This restricts usage to Android-based workplaces and excludes users from mixed-platform environments, constraining the application's flexibility and reach.

No Biometric Authentication Engine:

Although the app incorporates fingerprint and QR code as attendance triggers, it lacks a unified biometric authentication system. In particular, facial recognition was not implemented, so the system does not verify the user's identity against stored biometric profiles during clock-in. While fingerprint authentication is confined to the user's device and provides basic identity verification, the absence of facial recognition diminishes the app's suitability for high-security settings that demand more stringent verification methods.

Lack of Notification Feedback Loop:

A key feature of the system allows managers and admins to manually send presence verification notifications after an employee clocks in. However, the application currently lacks a mechanism for employees to respond to or acknowledge these prompts. This missing feedback loop reduces the system's ability to validate continuous presence and prevents automated recording of interaction data.

Limited Access Control Logic:

While the system incorporates role-based interfaces for employees, admins, and managers, the access control remains fairly static. Permissions are predefined and do not adjust dynamically based on user behavior or real-time data like location anomalies or repeated failed login attempts. More sophisticated control mechanisms such as time-based restrictions or automatic lockouts were not implemented due to time and complexity limitations.

No Internet-Free Mode:

The application depends on a stable internet connection to operate. Because it relies heavily on Supabase for database access and Firebase Cloud Messaging (FCM) for real-time notifications, core features such as clock-in, attendance tracking, and user authentication are unavailable offline. Consequently, the system is not suitable for deployment in remote or network-constrained environments.

6.4 Future Work

As development of the CorePresence mobile application continues, several areas have been identified for future enhancement to boost its reliability, scalability, and user experience. Implementing these improvements will strengthen the system's robustness and usability, while broadening its potential adoption in real-world organizational environments. The following highlights the key recommendations for future work:

1. Cross-Platform Support (iOS Implementation):

Future versions of CorePresence should incorporate support for iOS devices to expand usability across a wider range of work environments. This will involve configuring the app using Xcode and conducting comprehensive testing to ensure compatibility with Apple's development ecosystem and App Store requirements.

2. Offline Functionality and Synchronization:

Currently, all app functions rely on a stable internet connection. Adding an offline mode with local data storage and background sync would improve the system's robustness in low-connectivity areas, allowing users to record attendance data that syncs automatically when the connection is reestablished.

3. Real-Time Confirmation for Notifications:

The presence verification feature can be improved by allowing users to respond immediately to push notifications. Adding an interactive confirmation or "Check-In Acknowledged" button within the notification could help supervisors monitor continuous employee presence more effectively.

4. Enhanced Biometric Verification:

Although fingerprint triggers are confined to individual devices and provide basic identity assurance, they do not verify users against a centralized biometric profile. Future improvements could include integrating facial recognition, adding an extra layer of security by allowing real-time user identity confirmation. This enhancement would reinforce authentication and help prevent proxy attendance, particularly in organizations with stringent verification requirements.

5. Attendance Analytics and Visualization Tools:

Implementing a visual dashboard to showcase attendance patterns, daily summaries, and punctuality metrics could support HR decision-making and enhance system usability. Features like graphs, heatmaps, and exportable charts would be valuable for management and auditing purposes alike.

6. Automated Reminder Notifications

Another possible improvement is the addition of automated clock-in and clock-out reminder notifications. The system could be set to detect when an employee fails to record attendance within the designated time frame and send an automatic prompt. This feature would help reduce missed entries, improve the accuracy of attendance records, and lessen the administrative workload associated with following up on forgotten clock-ins. It would be particularly valuable for remote or hybrid teams where direct supervision is limited

7. Payroll System Integration

To enhance the system's functionality beyond attendance tracking, future developments could aim to integrate the app with payroll systems. This might involve exporting attendance data in formats compatible with existing payroll software or establishing direct API

integrations. By calculating total work hours from attendance logs, the system could support automated salary processing, overtime monitoring, and leave deductions streamlining HR workflows and boosting administrative efficiency.

6.5 Summary

In summary, this chapter has highlighted the achievements, limitations, and future enhancement opportunities in the development of the CorePresence mobile attendance system. CorePresence delivers an effective mobile attendance tracking solution by utilizing geolocation validation and multi-modal input methods, including QR scanning and fingerprint authentication. Its key features such as real-time notifications through Firebase Cloud Messaging (FCM), role-specific dashboards for employees, admins, and managers, and efficient CSV report generation enhance both functionality and user experience.

While the current system faces limitations like Android-only availability, absence of facial recognition, and lack of notification feedback mechanisms, it builds a secure and scalable platform for modern workplace attendance monitoring. Recommended future improvements include adding iOS support, enabling offline functionality, integrating interactive notification responses, and incorporating payroll systems to strengthen its stability and readiness for enterprise deployment.

REFERENCES

- Aleryani, A. (2016). Comparative Study between Data Flow Diagram and Use Case Diagram. *International Journal of Scientific and Research Publications*, 6(3).
<https://www.ijsrp.org/research-paper-0316/ijsrp-p5122.pdf>
- Ali, M., Usman Zahoor, H., Ali, A., & Ali Qureshi, M. (2020, November 1). *Smart Multiple Attendance System through Single Image*. IEEE Xplore.
<https://doi.org/10.1109/INMIC50486.2020.9318103>
- Bailey, G. "Skip. (1993). Iterative methodology and designer training in human-computer interface design. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '93*. <https://doi.org/10.1145/169059.169163>
- Basloom, H., Bosaeed, S., & Mehmood, R. (2020, April 1). *Hudhour: A Fuzzy Logic based Smart Fingerprint Attendance System*. IEEE Xplore.
<https://doi.org/10.1109/FMEC49853.2020.9144948>
- Beres, J. (2022, September 6). *What Is Rapid Application Development (RAD Model)*. App Builder. <https://www.appbuilder.dev/blog/rapid-application-development>
- Berger, H., Beynon-Davies, P., & Cleary, P. (2004). The Utility of a Rapid Application Development (RAD) Approach for a Large Complex Information Systems Development. *ECIS 2004 Proceedings*. <https://aisel.aisnet.org/ecis2004/7/>
- Beynon-Davies, P., Carne, C., Mackay, H., & Tudhope, D. (2015). Rapid application development (RAD): an empirical review. *European Journal of Information Systems*, 8(3), 211–223. <https://doi.org/10.1057/palgrave.ejis.3000325>
- Brady, M., & Loonam, J. (2010). Exploring the use of entity-relationship diagramming as a technique to support grounded theory inquiry. *Qualitative Research in Organizations and Management: An International Journal*, 5(3), 224–237.
<https://doi.org/10.1108/17465641011089854>

- Chauhan, V., Singh, H., Kinshuk Dewari, & Kumar, I. (2024). *Efficient Employee Tracking with Smart Attendance System Using Advanced Face Recognition and Geofencing*. 471, 750–755. <https://doi.org/10.1109/icssc560660.2024.10625038>
- Chien, C. (2020, February 4). *What Is Rapid Application Development (RAD)?* Codebots. <https://codebots.com/app-development/what-is-rapid-application-development-rad>
- Code, V. S. (2025). Setting up Visual Studio Code. Code.visualstudio.com. <https://code.visualstudio.com/docs/setup/setup-overview>
- Czerwonka, E. (2024, September 24). *The Top 5 Time Clocks for Cannabis Businesses in 2025*. Buddy Punch. <https://buddypunch.com/blog/the-top-5-time-clocks-for-cannabis-businesses/>
- Dutta, R., Tamang, T., Paul, P., Kumar, N., Chetri, C., & Dutta, P. K. (2020, December 1). *Smart and Secure Fingerprint Attendance System using Arduino UNO with GSM Alert*. IEEE Xplore. <https://doi.org/10.1109/ICISS49785.2020.9316127>
- EDUCBA . (2020). React Native Architecture | Realms | Architecture | Features. EDUCBA. <https://www.educba.com/react-native-architecture/>
- Elmasri, R., & Sham Navathe. (2004). *Fundamentals of Database Systems*. Addison Wesley Longman.
- Expo. (2025). Expo Go vs Development Builds: Which should you use? Expo Blog. <https://expo.dev/blog/expo-go-vs-development-builds>
- Firebase. (2025). Firebase Cloud Messaging | Send notifications across platforms for free. Firebase. <https://firebase.google.com/products/cloud-messaging>

- Floyd, C., Fanny-Michaela Reisin, & Schmidt, G. (1989). STEPS to software development with users. *Lecture Notes in Computer Science*, 48–64. https://doi.org/10.1007/3-540-51635-2_32
- FlutterFire. (2025). Firebase Cloud Messaging | FlutterFire. <https://firebase.flutter.dev/docs/messaging/overview/>
- geeksforgeeks. (2024, September 22). *Software Engineering | Rapid application development model (RAD) - GeeksforGeeks*. GeeksforGeeks. <https://www.geeksforgeeks.org/software-engineering-rapid-application-development-model-rad/>
- GoGet. (2023, October 20). *From Manual Punch Cards to Digital Timesheets*. Goget.myblog. <https://goget.my/blog/from-manual-punch-cards-to-digital-timesheets>
- Hamm, M. J. (2014). *Wireframing Essentials*. Packt Publishing Ltd.
- IBM Cloud Education. (2024, August). *PostgreSQL vs. MySQL: What's the Difference?* | IBM. [Ibm.com. https://www.ibm.com/think/topics/postgresql-vs-mysql](https://www.ibm.com/think/topics/postgresql-vs-mysql)
- Indeed Editorial Team. (2021, August 18). *Punctuality and Attendance at Work: Definition and Tips*. Indeed Career Guide. <https://www.indeed.com/career-advice/career-development/punctuality-and-attendance>
- Jain, P. (2024, September 17). *Real-Time Attendance Tracking System for Optimal Workforce Management - Tankhwa Patra*. Tankhwa Patra. <https://www.tankhwapatra.com/real-time-attendance-tracking-system-for-optimal-workforce-management/>
- Jibble. (2024, October 23). *100% FREE Face Recognition Attendance* | JibbleTM. Jibble. <https://www.jibble.io/face-recognition-attendance-system>
- John, C. S. (2024). AI Dev Tips 3.5: Review of AI Supabase “Postgres.new.” Medium; AI Dev Tips. <https://medium.com/ai-dev-tips/ai-dev-tips-3-5-review-of-ai-supabase-postgres-new-efaa969c0f6e>

- Kiarie, S. (2024a, July 14). *Honest Buddy Punch Review 2024: Pros, Cons, Features & Pricing*. Connecteam. <https://connecteam.com/reviews/buddy-punch/>
- Kiarie, S. (2024b, December 4). *Honest Jibble Review 2025: Pros, Cons, Features & Pricing*. Connecteam. <https://connecteam.com/reviews/jibble/>
- Kolosky, C. (2024, July 31). *4 Phases of Rapid Application Development*. Knack: No-Code Application Development Platform. <https://www.knack.com/blog/rapid-application-development-phases/>
- Lee, D. E. (2025). What is Supabase? The Free Open-Source Firebase Alternative You've Been Looking For. Medium. <https://ernestodotnet.medium.com/what-is-supabase-the-open-source-firebase-alternative-youve-been-looking-for-b1e412197084>
- Maina, E. (2024, May 1). *Timeero*. Timeero.com. <https://timeero.com/reviews/jibble-review#toc--what-is-jibble->
- Manikam, K. (2024). Biometrics and Beyond: Innovating Against Buddy Punching Losses. *Human Resource and Leadership Journal*, 9(1), 85–100. <https://doi.org/10.47941/hrlj.1657>
- Maurizfa, & Trio Adiono. (2021). *Smart Attendance Recording Device Based on Fingerprint Identification*. <https://doi.org/10.1109/isesd53023.2021.9501823>
- Memane, R., Jadhav, P., Patil, J., Mathapati, S., & Pawar, A. (2022, August 1). *Attendance Monitoring System Using Fingerprint Authentication*. IEEE Xplore. <https://doi.org/10.1109/ICCUBEA54992.2022.10010791>
- Mutinda, B. (2023, January 8). *How to fix - "One or more plugins require a higher Android NDK version" error on Android studio*. Medium. <https://medium.com/@brianmutinda49/how-to-fix-one-or-more-plugins-require-a-higher-android-ndk-version-error-on-android-studio-4abaf3fcad00>

- nCube. (2020, May 26). *What is Rapid Application Development?* - nCube. <https://Ncube.com/What-Is-Rapid-Application-Development>. <https://ncube.com/what-is-rapid-application-development>
- Okafor, F., & Ogechukwu Justina Odiliobi. (2021). APPLICATION OF FLOWCHART IN TEACHING SIMPLE INTEREST TO SECONDARY SCHOOL STUDENTS; A PANACEA FOR SUSTAINABLE DEVELOPMENT. *COOU Journal of Educational Research*, 6(1). <https://www.cooujer.com/index.php/COOUJER/article/view/25/24>
- Pan, G., Seow, P.-S., Shankararaman, V., & Koh, K. (2020). Essence of partnership management in project-based learning: insights from a university's global project programme. *Journal of International Education in Business*, ahead-of-print(ahead-of-print). <https://doi.org/10.1108/jieb-04-2020-0031>
- PS Solutions. (2025). React Native – PS Solutions. Pssolutions.me. <https://pssolutions.me/react-native/>
- Purnapatre, H., Shukla, S., Gangishetti, S., & Bohra, H. (2020). Improved Facial Recognition for Attendance Systems. *International Journal of Scientific Research in Science and Technology*, 456–467. <https://doi.org/10.32628/ijrst207678>
- Qi, C., Lai, M. Y., Liu, L., Zuo, S., Liang, H., & Li, R. (2022). Examining teachers' learning through a project-based learning lesson study: a case study in China. *International Journal for Lesson & Learning Studies*. <https://doi.org/10.1108/ijlls-09-2022-0127>
- Qureshi, A. (2024a, October 2). *How to Deal With Employee Absenteeism* | 2024. Jibble. <https://www.jibble.io/article/employee-absenteeism>
- Qureshi, A. (2024b, December 1). *Honest Review of Deputy by a Competitor* | 2023. Jibble. <https://www.jibble.io/reviews/deputy>

- Rahman, M. (2019). EMUI 10 beta on the Huawei P30 Pro - Hands-on with Huawei's Android Q update. XDA. <https://www.xda-developers.com/emui-10-huawei-p30-pro-android-q-hands-on/>
- Rahman, M. (2021). Study on Introducing Biometric Fingerprint Authentication in Automated Student Attendance System. *New Visions in Science and Technology*, 4, 121–131. <https://doi.org/10.9734/bpi/nvst/v4/4580f>
- Rashid, S. M., McCusker, J. P., Pinheiro, P., Bax, M. P., Santos, H. O., Stingone, J. A., Das, A. K., & McGuinness, D. L. (2020). The Semantic Data Dictionary – An Approach for Describing and Annotating Data. *Data Intelligence*, 2(4), 443–486. https://doi.org/10.1162/dint_a_00058
- React Native. (2024). React Native · A framework for building native apps using React. Reactnative.dev. <https://reactnative.dev/>
- Rieg, D., MacLennan, M. L., Scramim, F., Huertas, M., & Augusto, E. (2024). Project-based learning through the lens of SEEM: enhancing implementation in the Brazilian context. *Journal of International Education in Business*, 17(2), 210–227. <https://doi.org/10.1108/jieb-06-2023-0039>
- Rissell, Matt. “The Day I Stopped Paying an Imaginary Employee.” *Forbes*, 7 Mar. 2016, www.forbes.com/sites/mattrissell/2016/03/07/the-day-i-stopped-paying-an-imaginary-employee/#6553efb41b37.
- Semmy Wellem Taju, Yonatan Putra Mamahit, & Pongantung, J. A. (2024). Implementing QR code and Geolocation Technologies for the Student Attendance System. *CogITO Smart Journal*, 10(1), 642–653. <https://doi.org/10.31154/cogito.v10i1.636.642-653>

- Surve, M., Joshi, P., Jamadar, S., & Vharkate, M. (2020). *A2644059120 - International Journal of Recent Technology and Engineering (IJRTE)*. International Journal of Recent Technology and Engineering (IJRTE). <https://www.ijrte.org/portfolio-item/A2644059120/>
- Team Kissflow. (2024, December 13). *Rapid Application Development (RAD) | Definition, Steps & Full Guide*. Kissflow.com. <https://kissflow.com/application-development/rad/rapid-application-development/>
- Teye, J. (2022). Workplace attendance management system. *Air.ashesi.edu.gh*. <https://air.ashesi.edu.gh/items/ab1ca119-e507-43f9-9a86-4755ae6754e5>
- TheKnowledgeAcademy. (2024). *Firebase Authentication: A Comprehensive Guide for Developers*. Theknowledgeacademy.com. <https://www.theknowledgeacademy.com/blog/firebase-authentication/>
- Vinod, V. M., Murugesan, G., Mekala, V., Thokaiandal, S., Vishnudevi, M., & Siddharth, S. M. (2021). A Low-Cost Portable Smart Card Based Attendance System. *IOP Conference Series: Materials Science and Engineering*, 1012, 012046. <https://doi.org/10.1088/1757-899x/1012/1/012046>
- Wahal, Y. (2024, October 7). *Deputy Honest Review*. Connecteam. <https://connecteam.com/reviews/deputy/>

APPENDICES

Appendix A

Sample Questionnaire

Project Survey : Verify+ - Advanced Attendance and Presence Verification System Using Multi Modal Authentication and Geolocation for Office Environments

Dear respected respondents,

My name is Dayang Nur Alisa Binti Abang Senawi, a final-year undergraduate student specializing in Software Engineering at the Faculty of Computer Science and Information Technology (FCSIT) at Universiti Malaysia Sarawak (UNIMAS). As part of my Final Year Project, I am conducting a survey to gather crucial insights and feedback on the features of the proposed Advanced Attendance and Presence Verification System.

The aim of this system is to improve attendance accuracy by using multi-modal authentication methods, including QR code scanning, fingerprint recognition, facial recognition, and geolocation-based attendance tracking. It also offers continuous presence verification to enhance its reliability for workplace environments.

Your valuable responses will contribute significantly to understanding user needs and refining the proposed system. All responses will be kept strictly confidential and will only be used for academic purposes related to this project.

If you have any questions or need further clarification, please feel free to contact me via WhatsApp at 013-5558783 or email me at 77919@siswa.unimas.my.

Thank you for your time and participation. Your support is greatly appreciated and plays a vital role in the success of my project.

Warm regards,
Dayang Nur Alisa Binti Abang Senawi
Final Year Software Engineering Student
Universiti Malaysia Sarawak (UNIMAS)

Section A: Demographic Questions

What is your age? *

- ☐ 18-24
- ☐ 25-34
- ☐ 45-54
- ☐ 55 and above

What is your gender? *

- ☐ Male
- ☐ Female

Have you used an attendance system at your workplace before? *

- ☐ Yes
- ☐ No

How would you describe your office's current attendance system? *

- ☐ Manual
- ☐ Automated/Digital
- ☐ Other: _____

Section B: Questions for Those Working in an Office

Do you find the traditional attendance method (e.g., punch cards or paper-based) to be inaccurate or time-consuming? *

- ☐ Yes
- ☐ No
- ☐ Maybe

Have you witnessed instances of buddy punching (one employee clocking in for another)? *

- ☐ Yes
- ☐ No
- ☐ Unsure

Have you noticed employees marking their attendance but not being present at the office to work? *

- ☐ Yes
- ☐ No
- ☐ Unsure

How often do you experience issues with your current attendance system? *

- ☐ Frequently
- ☐ Occasionally
- ☐ Rarely
- ☐ Never

Do you think your current attendance system ensures accurate tracking of attendance and presence? *

- | | | | | | | |
|-------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|----------------|
| | 1 | 2 | 3 | 4 | 5 | |
| Strongly Disagree | ☐ | ☐ | ☐ | ☐ | ☐ | Strongly Agree |

Would you consider a system with advanced features (e.g., geolocation, face recognition) beneficial for your workplace? *

- ☐ Yes
- ☐ No
- ☐ Maybe

Section C: Usability Design and Preference

How important is user-friendliness in an attendance system for improving employee experience and productivity? *

1 2 3 4 5
Not Important ○ ○ ○ ○ ○ Very Important

Which features would you prioritize in an attendance system? *

- ☐ Face recognition
- ☐ QR code scanning
- ☐ Thumbprint recognition
- ☐ GPS tracking (Geolocation)
- ☐ Other: _____

How frequently do you use a mobile device to access or interact with attendance systems? *

- ☐ Daily
- ☐ Weekly
- ☐ Rarely
- ☐ Never

Do you find integrating attendance tracking with mobile apps enhances convenience and accessibility? *

1 2 3 4 5
Strongly Disagree ○ ○ ○ ○ ○ Strongly Agree

On a scale of 1 to 5, how confident are you in the reliability and security of advanced technology (e.g., biometrics) in attendance systems? *

1 2 3 4 5
Not Confident ○ ○ ○ ○ ○ Very Confident

Section D: Additional Feedback

Do you have any concerns regarding the proposed attendance system? *

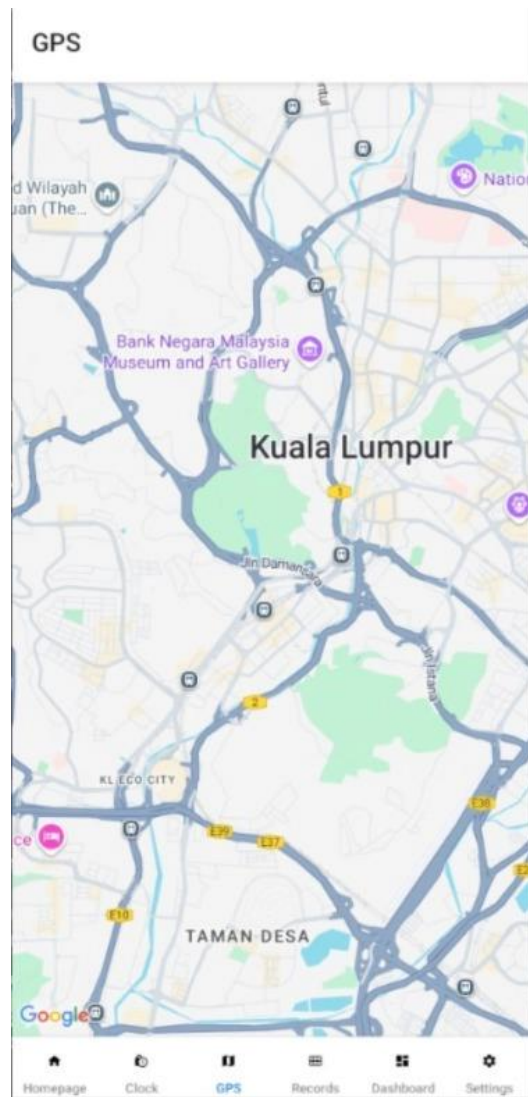
Your answer _____

Do you have any suggestions for improving the features of the proposed attendance system? *

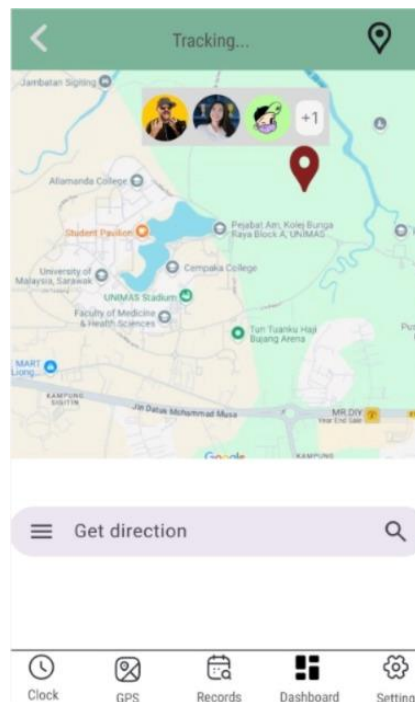
Your answer _____

Appendix B

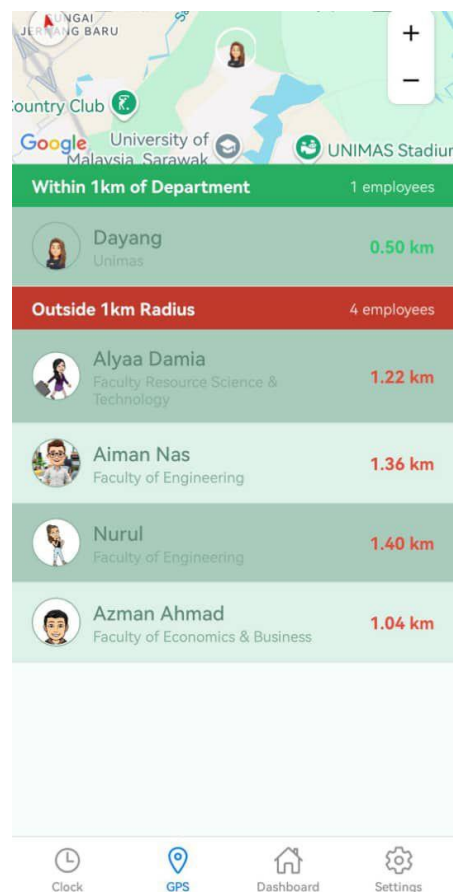
Initial Prototype (Low-Fidelity Wireframe):



Intermediate Version (Post-Feedback Testing Stage):



Final Interface (After Iteration and Validation):





CorePresence

Smart Attendance. Smarter Teams.

User Acceptance Survey for CorePresence

Dear respected respondents, Thank you for using CorePresence!

My name is Dayang Nur Alisa Binti Abang Senawi, and I am currently in my final year as an undergraduate student specializing in Software Engineering at Universiti Malaysia Sarawak (UNIMAS). As part of my Final Year Project, I am conducting a survey to gather valuable feedback for the evaluation and improvement of CorePresence, a mobile-based attendance system that leverages multi-modal authentication and geolocation technology.

This system aims to enhance workplace attendance tracking by offering features such as QR scan, fingerprint verification, geolocation-based validation, and real-time notifications. Your input is crucial in assessing the usability, functionality, and reliability of the app before final deployment.

Please be assured that all responses will remain confidential and are used solely for academic purposes. Should you have any questions, feel free to contact me at 77919@siswa.unimas.my or 0135558783.

Thank you very much for your time and cooperation!

General Usability

The app is easy to navigate and understand. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

The system responds promptly when actions like clocking in or clocking out are performed *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Attendance Features

The clock-in and clock-out process (QR or fingerprint) works smoothly. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

The geolocation feature correctly detects and records my location. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

I can easily view my attendance status (e.g., Present/Absent) in the dashboard.

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

The system shows confirmation or status updates after I clock in/out. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Interface Design

All features inside the application are neatly organized. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

The font type, color, and size used in the app are appropriate. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Overall Experience

The application is useful for managing attendance and presence. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

I would feel confident using this system in a real work environment. *

	1	2	3	4	5	
Strongly Disagree	☐	☐	☐	☐	☐	Strongly Agree

Additional Feedback / Suggestions *

Your answer