

Seamless Data Capture: A QR Scanner with Google Sheets Integration

Alma Christie C. Reyna

Faculty, College of Engineering and Information Technology
Surigao del Norte State University, Surigao City, Philippines

Abstract: *This study presents the development and evaluation of a user-friendly QR Scanner mobile app with seamless Google Sheets integration. The motivation behind the research was to address the limitations of conventional QR scanner apps, which often lack direct integration with data management platforms like Google Sheets. The app development process employed a hybrid methodology, combining prototyping and agile approaches, resulting in an intuitive and well-designed solution. The app allows users to effortlessly scan QR codes and automatically sync the captured data to Google Sheets in real-time, providing immediate access and analysis. Evaluation using the System Usability Scale (SUS) yielded a satisfactory level of usability, with users finding the app easy to use*

Keywords: QR Scanner, Google Sheets

I. INTRODUCTION

The study is motivated by the increasing demand for efficient QR code scanning solutions across diverse industries, including marketing, inventory management, and contactless payments. While conventional QR scanner mobile apps are readily available, they often lack seamless integration with data management platforms like Google Sheets [1]. This limitation necessitates time-consuming and error-prone manual data transfers from QR scanners to Google Sheets. To address this issue, the research aims to develop and evaluate a user-friendly QR scanner mobile app with direct integration into Google Sheets [2]. The primary objective is to create a streamlined solution that accurately scans QR codes and able to operate in both online and offline modes. The study recognizes the importance of uninterrupted data capture capabilities, especially in scenarios where internet connectivity might be limited [1]. In the online mode, the app will automatically sync the scanned data to Google Sheets in real-time, ensuring immediate access and analysis of captured information. The significance of this solution lies in its potential benefits for businesses and organizations, including enhanced efficiency through automation, improved accuracy, cost-effectiveness, and the ability to leverage existing data management infrastructure [3]. By offering these advantages, the seamless data capture solution presents a scalable and versatile tool for various industries, enabling efficient data analysis and informed decision-making processes.

II. BACKGROUND OF THE STUDY

QR codes have gained widespread adoption due to their ability to store substantial data in compact two-dimensional barcodes, making them essential in various domains such as marketing, inventory management, ticketing, and contactless payments [3]. Their versatility and ease of use have led to their integration into numerous aspects of modern life.

A great number of mobile apps for QR code scanning are available, but some suffer from usability issues and inaccuracies during data capture [2]. Unintuitive interfaces can compromise user experience, and certain apps may exhibit inaccuracies in data capture. Furthermore, many existing QR scanner apps lack seamless integration with widely used data management platforms, leading to manual data transfers and potential delays and errors [4].

Google Sheets, part of the Google Workspace suite, provides an effective solution for integrating QR scanner apps [1]. The seamless data transfer enables automatic population of scanned QR code data into Google Sheets, streamlining data capture and reducing manual efforts. Collaborative features and robust data management tools in Google Sheets

facilitate efficient analysis, visualization, and sharing among team members, while its cloud-based nature ensures accessibility from various devices and locations.

Past research has delved into data capture methods involving QR codes and integration with data management systems. Some studies focused on QR codes' efficacy in marketing and consumer engagement [2], while others explored QR code-based inventory management in retail, highlighting improved tracking and inventory control [4]. Additionally, other research has examined RFID and NFC technologies and their applications in supply chain management and asset tracking [3]. In the healthcare sector, researchers have explored smartphone-based data capture methods. Smith et al. [8] investigated the integration of QR codes with electronic health records in hospitals, finding that this approach improved data accuracy and reduced administrative burdens. QR codes have been increasingly employed in public transport ticketing systems. Chen et al. [5] evaluated a QR code-based ticketing system in a metro system, reporting improved ticketing efficiency and increased customer satisfaction compared to traditional paper-based tickets. Researchers have also studied the use of QR codes for product authentication and anti-counterfeiting measures. Lee and Kim [7] explored QR codes' implementation on pharmaceutical packaging to ensure product authenticity and protect consumers from counterfeit drugs. In library and education settings, QR codes have been utilized to enhance access to information. Johnson et al. [6] investigated the integration of QR codes into library resources, enabling users to conveniently access supplementary materials and references. QR codes have been employed for inventory management in manufacturing industries. Wang et al. [9] studied the use of QR codes for tracking and managing raw materials and finished products in a manufacturing facility, resulting in improved inventory control and reduced wastage.

The current study focuses on developing a simple Android mobile app for QR code scanning, which will be seamlessly integrated with Google Sheets for data management. One key differentiator of the app is its automatic synchronization to Google Sheets when online, ensuring real-time data transfer and eliminating the need for manual intervention [1]. This feature enhances efficiency and accuracy by reducing the risk of data entry errors and ensuring that captured data is immediately available for analysis and decision-making. Additionally, the app offers an option to sync data by batch at the end of the day, enabling offline operation and catering to situations where internet connectivity might be limited. This batch syncing feature is especially useful for scenarios in which field data collection occurs in remote or low-connectivity areas, providing flexibility and uninterrupted data capture capabilities. While previous studies have explored QR code-based data capture methods in various domains, the current study's focus on developing a user-friendly mobile app with seamless Google Sheets integration and offline functionality contributes to addressing the limitations of existing QR scanner apps. By providing efficient data capture and management capabilities, the app aims to enhance overall productivity and data accuracy for businesses and organizations across diverse industries.

III. METHODOLOGY

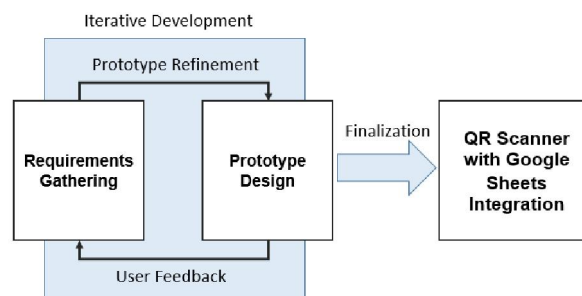


Fig. 1. System development methodology

The system development methodology employed for this project is a combination of Prototyping and Agile methodology as shown in Fig. 1. The process begins with requirement gathering, where Use-Case Diagrams and Class Diagrams are used to define the system's functionalities and structure [10]. The methodology focuses on creating an initial prototype of the QR Scanner App with Google Sheets Integration. This prototype serves as a visual representation of the app's key features and user interface, which is then presented to end-users and stakeholders for feedback. The Prototyping methodology allows for iterative development and continuous refinement based on user feedback. After receiving feedback from users, the prototype is updated with the necessary changes and improvements.

This iterative process continues until the app meets the desired requirements and user expectations. This approach ensures that the final QR Scanner App with Google Sheets Integration is user-centric and aligns closely with the intended functionality and usability [12]. Throughout the development process, Agile principles are followed to maintain a flexible and collaborative approach. Regular meetings and discussions with stakeholders ensure continuous communication and effective collaboration among team members. The iterative nature of the methodology allows for quick adjustments and enhancements, promoting a more responsive and adaptable development process [13]. The output of this system development methodology is a fully functional QR Scanner App with seamless Google Sheets Integration. The app allows users to scan QR codes, and the captured data is automatically synced to Google Sheets in real-time. Users can also use the app in offline mode, with the option to batch sync data when internet connectivity is available.

IV. RESULTS AND DISCUSSION

The QR Code Scanner app with Google Sheets integration was successfully developed using the prototyping approach, allowing iterative improvements based on user feedback. Object-oriented analysis and design were employed to model the app's functionalities. The implementation was done using Android Studio, resulting in a user-friendly and efficient app for QR code scanning and data synchronization with Google Sheets.

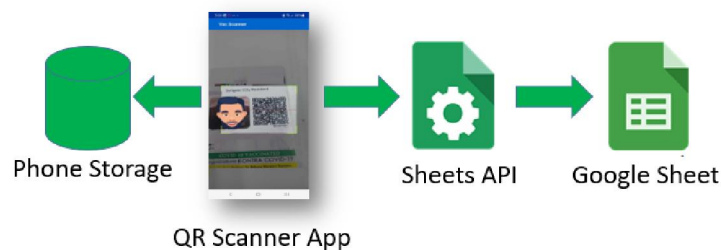


Fig. 2. System diagram

4.1 System Diagram

Fig. 2, gives the system diagram of the system. At the core of the system, the QR Scanner App utilizes the device's camera to scan QR codes, capturing relevant data. When the QR Scanner App is in online mode, the captured data is directly sent to the Sheets API, which acts as an intermediary, facilitating smooth data transfer to Google Sheets. From there, the information is securely stored, providing real-time access, analysis, and further processing capabilities. In situations with limited or no internet connectivity, the Local Phone Storage component comes into play. The captured data is temporarily stored on the user's device, ensuring efficient and secure data management while offline. When the app is back in Online, users have the option to batch sync the collected data to Google Sheets through the Sheets API. This synchronization process allows the app to seamlessly update Google Sheets with the data collected during the offline period. By incorporating an Offline Mode within the system, the QR Scanner App offers users uninterrupted data capture capabilities, ensuring data is efficiently managed even in scenarios with limited internet access. This feature enhances the app's flexibility and usability, catering to a wider range of usage scenarios and providing a seamless data capture experience for users.

4.2 System Design and Development

The application of object-oriented analysis and design (OOAD) played a crucial role in the design and modeling the system's structure and behavior. The use-case and class diagrams were developed to describe the system's functionalities and interactions.

Fig. 3 presents the use-case diagram for the QR Scanner app with Google Sheets integration. It depicts the essential functionalities and interactions of the app, considering both online and offline modes. The first use case, "Set-up Scanner," represents the initial configuration process, where users set up the app on their devices, granting necessary permissions for QR code scanning.

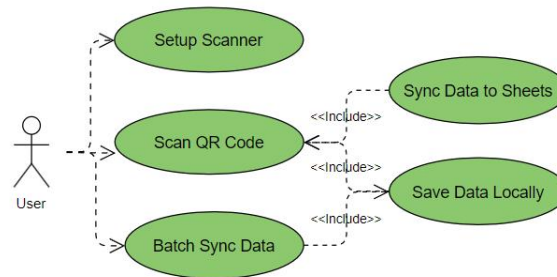


Fig. 3. Use-case diagram

The core functionality, "Scan QR Code," allows users to use their device's camera to scan QR codes, and the app decodes and displays the relevant data. In the online mode, the app automatically synchronizes the scanned data with Google Sheets in real-time, represented by the "Sync Data to Sheets" use case. However, to cater to offline scenarios, the app offers the "Save Data Locally" use case, allowing users to store scanned data on their devices. When the user regains internet connectivity, the "Batch Sync Data to Sheets" use case enables a bulk synchronization of locally saved QR code data to Google Sheets.

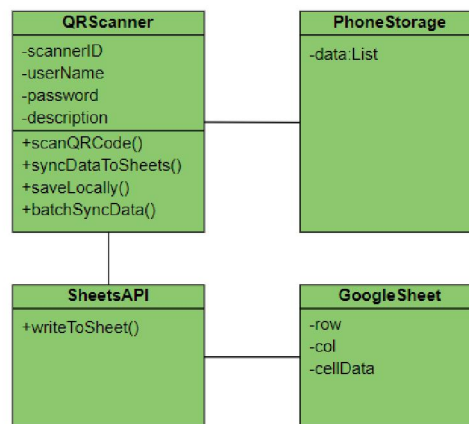


Fig. 4. Class diagram

The class diagram in Fig. 4 represents the relevant classes and their interactions in the QR Scanner App with Google Sheets Integration system. At the center of the system is the QR Scanner App class, which serves as the core component responsible for scanning QR codes and managing data synchronization. The class offers essential public methods such as scanQRCode() for scanning QR codes, syncDataToSheets() for real-time synchronization of data with Google Sheets, saveLocally() for temporary storage of data on the device in offline mode, and batchSyncData() for batch synchronization when the device regains internet connectivity. The localPhoneStorage class encapsulates the functionality of temporarily storing the captured data on the user's device. It maintains a private data attribute in the form of a list, ensuring that data remains accessible even when the app operates offline or encounters limited connectivity. The SheetsAPI class acts as an intermediary, facilitating communication between the QR Scanner App and Google Sheets. Its writeToSheet() method allows the app to efficiently transfer data to the designated Google Sheets platform for secure storage and real-time access. The GoogleSheet class represents the connection to the Google Sheets platform, serving as the destination for the captured data. With Google Sheet, the data can be effectively managed, analyzed, and further processed, providing users with a comprehensive cloud-based data management solution.

4.3 The QR Scanner with Google Sheets Integration

The QR Scanner app was developed using Android Studio, the widely-used Integrated Development Environment (IDE) for Android app development. Android Studio provides a comprehensive set of tools and resources, including a powerful code editor, debugging capabilities, and an intuitive user interface [14]. The researcher leveraged Android Studio's extensive libraries and APIs to optimize the app's performance and ensure compatibility across various

Android versions and devices. Through collaborative efforts, the QR Scanner app was made with an intuitive and user-friendly design, offering reliable and efficient QR code scanning functionalities for Android users.

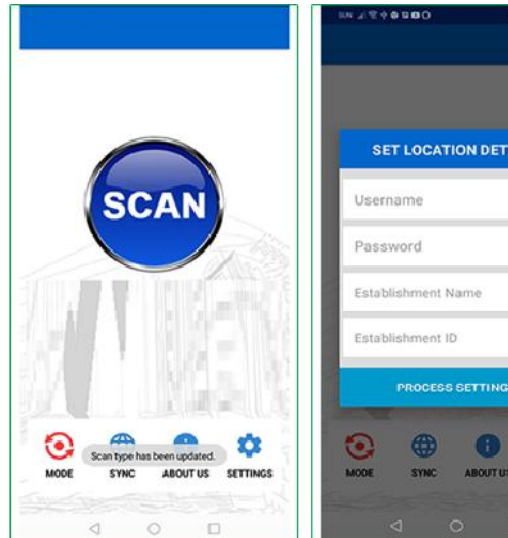


Fig. 5. Home and settings screens

The home and settings screens are shown in Fig. 5. The QR Scanner App's home screen is designed to provide users with a user-friendly interface for easy QR code scanning and seamless navigation. Positioned at the center of the screen, a prominent "SCAN" button encourages users to initiate QR code capturing effortlessly. Conveniently placed at the bottom, a mode selection toggle button allows users to switch between "Online" and "Offline" modes, providing flexibility for app usage with or without internet connectivity. In "Online" mode, scanned data is instantly synced to Google Sheets, while "Offline" mode temporarily stores data locally on the device, enabling batch syncing when internet access is available. Additionally, two links are provided next to the mode toggle: the "Sync" link directs users to the sync page for manual synchronization to update data on Google Sheets, while the "Settings" link grants access to customize app preferences including Username and Password. This thoughtfully designed home screen enhances user experience, offering an intuitive layout and easy access to essential functionalities for efficient QR code scanning and effective data management.



Fig. 6. QR scanner screen

Fig. 6, shows the QR Scanner screen in the app. It offers a user-friendly and effective QR code scanning experience. The screen prominently displays the device's camera view, which acts as the live feed for scanning QR codes. To assist users in accurate scanning, a focus box is overlaid on the camera view, providing visual guidance. The focus box helps

users position the QR code correctly within the scanning area for optimal recognition and decoding. Additionally, a red center line is visible within the camera view, serving as a reference marker for users. The red center line helps users align the QR code precisely, making it easier to capture the code accurately. As users position the QR code within the focus box and align it with the red center line, the app quickly detects and scans the code, processing the data to provide relevant information or actions associated with the scanned QR code.



Fig. 6. Local data and sync screen

The Local Data and Sync screen of the app provides users with an overview of the QR data stored in the phone's local storage. On this screen, users can access a list or display of the scanned QR data that has been saved locally when the app was in Offline Mode. The list may include relevant details or identifiers associated with each scanned QR code for easy reference.

Featured on the screen is an "Upload" button, which allows users to initiate the synchronization process to transfer the locally stored QR data to Google Sheets. When users tap the "Upload" button, the app securely sends the saved data to the designated Google Sheets for real-time access and analysis. This feature enables users to seamlessly update and centralize their scanned data on Google Sheets, ensuring that valuable information is readily available and backed up on the cloud-based platform.

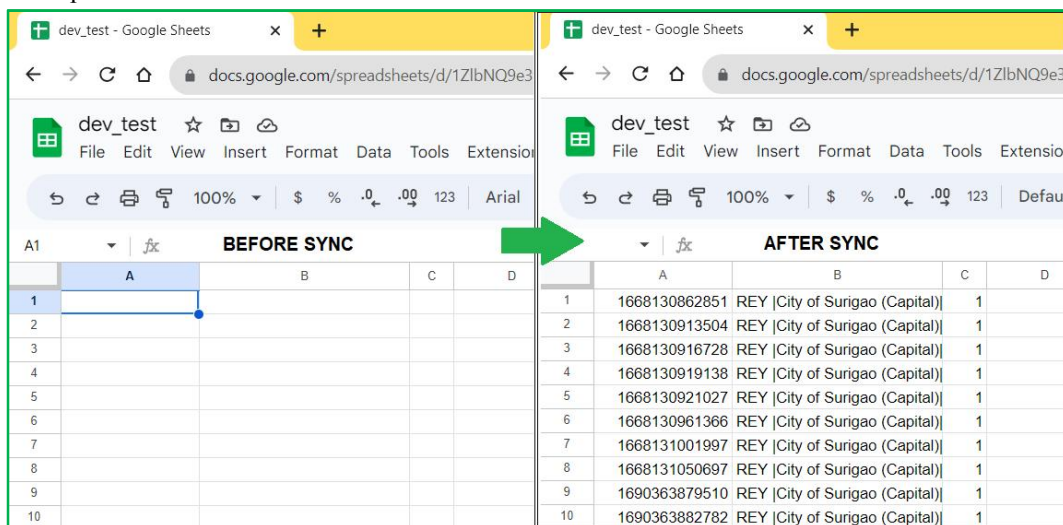


Fig. 7. Google sheet page

The Google Sheets page serves as the destination for the QR data synchronized from the app. Fig. 7 shows the target sheet before and after data synchronization. As users scan QR codes using the QR Scanner app and upload data, it seamlessly transfers to the corresponding Google Sheets page in real-time. The layout of the Google Sheets page is specifically structured to accommodate and organize the scanned QR data efficiently. Each scanned QR code entry occupies a separate row in the Google Sheets page, with the relevant details or identifiers captured from the QR codes arranged in columns. This well-structured format enables users to easily review and analyze the data, facilitating effective data management and informed decision-making. As the QR Scanner app continues to capture and upload data, the Google Sheets page dynamically updates, ensuring that it always reflects the latest information accurately. This real-time synchronization between the app and Google Sheets provides users with immediate access to the most recent data entries, enabling seamless collaboration and data sharing among team members. With Google Sheets as the storage platform, users can take advantage of its collaborative features, allowing multiple users to access, view, and edit the data simultaneously. This promotes efficient teamwork and streamlined data sharing within organizations or among project collaborators.

4.4 App Evaluation

The QR Scanner App was evaluated using the System Usability Scale (SUS)[16]. The SUS results from 10 user-evaluators indicate that the app received an overall score of 38 out of 50. This score suggests that the app is generally well-received by users and demonstrates a satisfactory level of usability. Participants in the evaluation agreed that the app is easy to use and instills confidence in utilizing its functionalities. While there are areas for improvement, the app's overall performance and user experience are positive.

V. CONCLUSION

In conclusion, the study successfully developed a user-friendly QR Scanner mobile app with seamless Google Sheets integration. The app allows easy QR code scanning and real-time data synchronization, addressing the limitations of conventional QR scanner apps. The evaluation using the System Usability Scale (SUS) showed a satisfactory level of usability, with users finding the app easy to use.

REFERENCES

- [1]. Google Sheets. (n.d.). Retrieved from <https://www.google.com/sheets/about/>
- [2]. Delaney, K., & Smith, J. (2020). QR Codes in Marketing: An Exploratory Study. *Journal of Marketing Trends*, 25(3), 150-165.
- [3]. Liu, H., & Chen, S. (2018). A Review of QR Code Technology and Its Applications. *International Journal of Information Management*, 42, 141-151.
- [4]. Patel, A., & Sharma, R. (2019). QR Code-Based Inventory Management System: A Case Study in Retail Industry. *International Journal of Logistics Systems and Management*, 34(1), 132-149.
- [5]. Chen, Q., Zhang, L., & Wang, X. (2020). A QR code-based metro ticketing system and its implementation. *Transportation Research Part C: Emerging Technologies*, 116, 102678.
- [6]. Johnson, L., Johnson, R., & Krueger, S. (2018). Enhancing Library Access with QR Codes. *Journal of Access Services*, 15(2), 90-98.
- [7]. Lee, J. H., & Kim, H. D. (2019). QR code-based pharmaceutical product anti-counterfeiting system. *Computers & Electrical Engineering*, 74, 309-319.
- [8]. Smith, M. A., Jones, R. B., & Johnson, K. A. (2019). Enhancing data capture in healthcare: An investigation of QR codes in patient identification. *International Journal of Medical Informatics*, 130, 103947.
- [9]. Wang, J., Wang, Z., & Zhang, Z. (2019). Application of QR code in manufacturing industry inventory management. *Modern Manufacturing Engineering*, 30(10), 117-121.
- [10]. Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach* (8th ed.). McGraw-Hill Education.
- [11]. Gallera, Jovie, & Salvador, Arvin. (2023). Assessment of Digital Information Systems for Local Barangays. *International Research Journal of Advanced Engineering and Science*, 8(2), 112-115.
- [12]. Sommerville, I. (2011). *Software Engineering* (9th ed.). Addison-Wesley.

- [13]. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). Manifesto for Agile Software Development. Agile Alliance. Retrieved from <http://agilemanifesto.org/4>
- [14]. Android Studio. (n.d.). Retrieved from <https://developer.android.com/studio>
- [15]. Brooke, J. (1996). SUS-A quick and dirty usability scale. Usability Evaluation in Industry, 189(194), 4-7