Technium Vol. 30, pp.12-18 (2025)
ISSN: 2668-778X
www.techniumscience.com

Automated Fixed Asset Management System with Predictive Analytics

Neil Vincent D. Alvior[1*], Jake R. Pomperada[2], and Dennis V. Madrigal[3]

[1]*Yusay Credit & Finance Corporation, Bacolod City, Philippines*
[2]*Technological University of the Philippines - Visayas, Talisay City, Negros Occidental, Philippines*
[3]*University of Negros Occidental-Recoletos, Bacolod City, Philippines*
[*]*Corresponding author:alviorneilvincent@gmail.com*

Abstract.

This study presents the design and development of an Automated Fixed Asset Management System with Predictive Analytics, which integrates QR code technology and machine learning to enhance asset tracking processes in organizations. The system aims to address common inefficiencies associated with traditional manual methods by automating asset registration, retrieval, and depreciation monitoring [1][2]. Through QR codes, assets can be easily identified and tracked, reducing human error and improving inventory accuracy [3]. The integration of predictive analytics, namely linear regression and random forest algorithms, enables the system to forecast lapsing schedules and asset lifecycles, supporting proactive decision-making and efficient resource allocation [4][5]. Developed on the PHP Laravel framework, the system ensures secure user access, real-time synchronization, and regular data backups [6]. It operates in both online and offline environments, making it adaptable to organizations with limited or inconsistent connectivity [7]. A case implementation at Yusay Credit & Finance Corporation demonstrates the system's capability to improve operational efficiency, reduce maintenance costs, and support scalable growth. This work contributes a robust and flexible solution for digital asset management tailored to the needs of mid-sized enterprises, particularly in developing regions where technology adoption is rapidly advancing [8][9].

Keywords: Fixed Asset Auditing, System Development, Software Testing, Asset Management, Automated Asset Management, QR Code, Predictive Analytics

1. Introduction

Efficient fixed asset management is essential for ensuring an organization's operational effectiveness, cost control, and financial accountability. In many companies, especially in developing countries like the Philippines, asset tracking remains reliant on manual and paper-based systems, which are prone to human error, data loss, and inefficiencies during audits [1][2]. This creates challenges in maintaining accurate inventories, ensuring timely maintenance, and making informed financial decisions about resource allocation and procurement [3].

The use of digital tools, including QR (Quick Response) code systems, has gained traction in inventory and asset management for its ability to simplify and streamline asset identification and data access [4]. QR code-based asset tagging provides real-time tracking, minimizes manual errors, and reduces the need for redundant records, making it particularly valuable in dynamic environments such as schools, hospitals, and corporate offices [5]. These tools enhance transparency and responsiveness while ensuring each asset's status is accessible instantly via smartphone or web interface.

However, existing QR-based systems still fall short in providing proactive, intelligent insights into asset conditions and lifecycle management. They often lack integration with predictive analytics capabilities, such as those offered by machine learning (ML), which can forecast asset depreciation,

predict maintenance needs, and suggest optimal replacement schedules [6][7]. Without these insights, organizations remain reactive, responding only after assets have failed or become inefficient, which leads to operational disruptions and unnecessary costs.

To address this gap, this study developed an Automated Fixed Asset Management System that integrates QR code functionality with machine learning algorithms, namely Linear Regression and Random Forest models. This hybrid approach allows for real-time data collection and predictive analysis, empowering organizations to reduce downtime, improve maintenance scheduling, and extend asset lifespans [8]. The system is designed for accessibility and security, using the Laravel PHP framework, and operates in both offline and online environments to accommodate varying infrastructure capacities [9].

A pilot implementation of the system at Yusay Credit & Finance Corporation was conducted to evaluate its effectiveness in a real-world setting. Initial findings revealed improved asset visibility, increased efficiency in inventory processes, and a reduction in maintenance-related disruptions. This paper discusses the design, implementation, and impact of the system, contributing a scalable, intelligent solution to fixed asset management tailored for mid-sized enterprises in the Philippines and similar economies undergoing digital transformation.

2. Product Descriptions

The Automated Fixed Asset Management System with Predictive Analytics was developed to streamline the process of tracking and maintaining fixed assets across organizational departments. It is designed to digitize asset records, automate repetitive tracking tasks, and introduce analytical capabilities that enhance strategic asset planning. The system's architecture is modular, scalable, and follows a user-centered design to ensure adaptability in diverse work environments.

At its core, the system features a centralized asset database that stores detailed records including asset name, tag number, classification, condition, assigned personnel, location, and depreciation value. Each asset is assigned a unique QR code that facilitates easy identification and updates through scanning. The system supports both one-time and recurring asset updates, ensuring real-time data accuracy across departments.

The system also incorporates predictive analytics functionality using two key machine learning models: Linear Regression and Random Forest. These models are trained on historical depreciation and usage data to forecast future asset conditions and replacement timelines. Users can view graphical reports showing projected maintenance schedules and depreciation trends, helping management to make informed decisions regarding asset lifecycle management.

Developed using the Laravel PHP framework, the system benefits from MVC architecture, built-in authentication, and robust backend support. It is hosted on an Apache server with a MySQL database, and its front end employs HTML, CSS, and JavaScript with responsive design for mobile and desktop access. The QR scanning feature utilizes web APIs compatible with modern mobile browsers, reducing the need for separate scanning devices.

In terms of data protection, the system ensures user authentication through hashed passwords using Bcrypt, and database backups are performed automatically on a scheduled basis. Different user roles—such as administrator, asset manager, and auditor—are defined with varying levels of access to system functionalities. This role-based access control enhances accountability and operational security, particularly in multi-departmental settings.

Additionally, the system is built with flexibility in mind: it operates effectively in both offline and online modes, allowing data to be temporarily stored locally and synced with the central server once internet access is restored. This feature makes it particularly useful for field work or institutions in areas with intermittent connectivity. Together, these components form a powerful, integrated tool for organizations aiming to modernize their asset management practices.

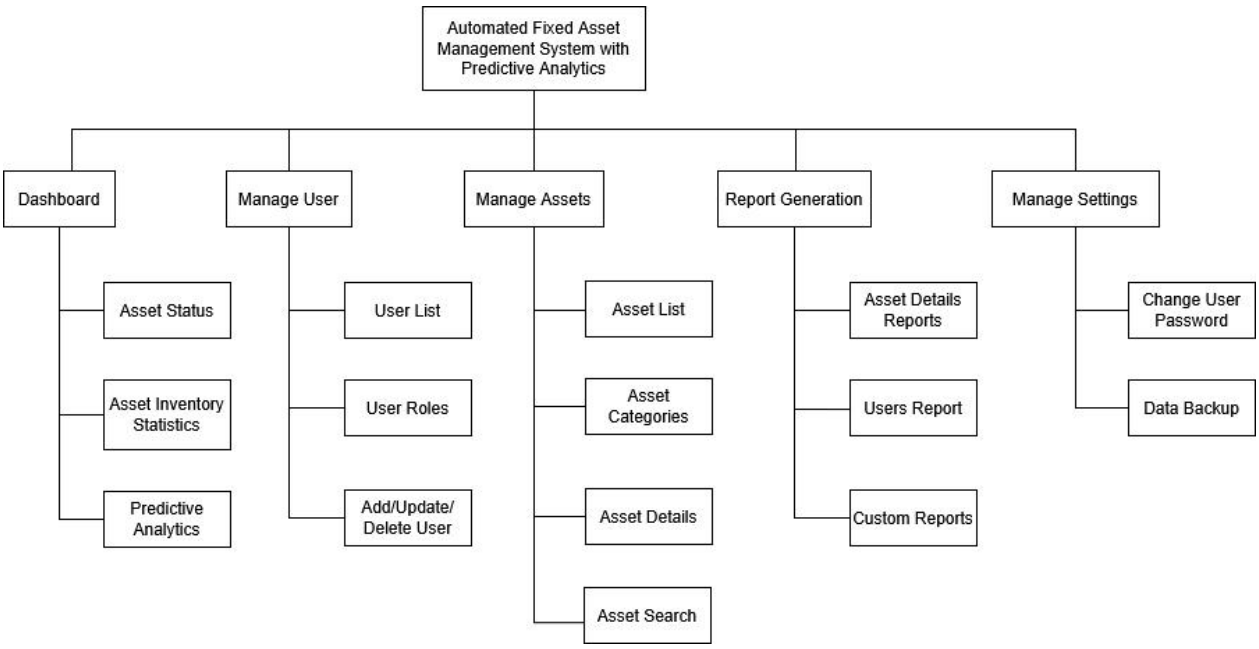
3. System Features

The system is designed to provide a comprehensive solution to the challenges of asset tracking, maintenance, and lifecycle planning. It is modular in nature, which allows organizations to adopt or scale

functionalities according to their specific operational needs. Each module performs distinct yet interconnected roles, ensuring seamless workflows and data synchronization across users and departments.

Figure 1 shows the decomposition structure of the Automated Fixed Asset Management System with Predictive Analytics, which shows that the system is divided into five modules: Dashboard, Manage User, Manage Assets, Report Generation, Manage Settings.

Figure 1.*System Decomposition*



The Dashboard serves as the central access point for users. It displays high-level system analytics including total assets, asset conditions, upcoming maintenance, and depreciation summaries. Graphical reports generated by predictive analytics models are presented here for immediate managerial insight. Widgets and charts are dynamically updated to reflect the current status of all assets, supporting real-time decision-making.

The Manage User module provides administrators with the tools to define roles, assign permissions, and manage user access across the platform. There are three primary roles: administrator, asset manager, and auditor. Each role is granted different levels of access to features such as data entry, editing, viewing reports, and managing system settings. This modular access control promotes accountability and minimizes the risk of unauthorized changes.

The Manage Assets module serves as the system's central component, providing core functionality for organizing and maintaining fixed asset records. Users can register and categorize assets by entering key details, including the asset's name, description, acquisition date, current condition, location, and assigned custodian. Each asset is linked to a system-generated QR code, enabling quick and efficient identification through mobile scanning. This feature streamlines access and updating of asset data, eliminating the need to navigate multiple system menus manually.

A key innovation is the Predictive Analytics Engine, which employs Linear Regression and Random Forest algorithms to forecast asset depreciation and anticipate maintenance needs. These models are trained on historical data, including usage frequency, condition updates, and cost records. Forecasts are displayed in user-friendly visualizations that help organizations preemptively budget for repairs or replacements, reducing unplanned downtime and expenditures.

The Report Generation Module allows for the generation of customized reports based on filters such as asset type, location, condition, or assigned user. Reports can be exported as PDFs and include charts generated by the analytics engine. This supports both internal reviews and compliance audits, providing a documented trail of asset activity, value changes, and system usage.

The QR Code Integration simplifies asset interaction. When a code is scanned via mobile device, the user is immediately directed to that asset's profile, which includes condition history, usage logs, and predictive depreciation insights. The system supports real-time updates following QR scans, and offline scanning is also enabled with local cache storage, which syncs once connectivity is restored.

The Manage Settings Module allows administrators to define global preferences such as depreciation methods, alert thresholds for maintenance, and backup scheduling. Notifications can also be configured to alert users via email or in-system messages when critical events occur—such as overdue maintenance, declining asset value, or system anomalies.

Lastly, the system supports Audit Trails, where all user activity is logged and timestamped for transparency. This feature enables administrators to review actions taken by individual users, ensuring integrity and traceability within the asset management process. The audit logs are immutable, serving as an internal security and compliance measure in case of discrepancies or investigations.

4. External Interface Requirements

The Automated Fixed Asset Management System was designed with user accessibility and system interoperability in mind. Its primary interface is web-based and responsive, allowing users to access the system via desktops, laptops, tablets, and mobile phones. The front end is optimized for compatibility with modern browsers such as Google Chrome, Mozilla Firefox, and Microsoft Edge. The interface was built with responsiveness to ensure a consistent user experience across different screen sizes and devices, especially during mobile-based QR scanning operations.

The system requires minimal client-side setup. Users do not need to install any software; instead, they simply access the application through a secure browser. Mobile users interact with the system primarily via QR code scanning, which is facilitated by the device's camera and web API access. No proprietary scanning app is needed, as the platform leverages native browser permissions and device capabilities to launch the asset retrieval function after scanning.

Server-side, the system requires a hosting environment capable of running PHP 8.1 or higher and a MySQL/MariaDB database. It is compatible with Apache and Nginx web servers and can be deployed on Windows-based infrastructures. The minimum recommended server specification includes an Intel i5 processor, 4GB RAM, and SSD storage to ensure stable performance for concurrent users and real-time analytics rendering. The system also includes backup utilities for scheduled local or cloud-based data preservation.

5. Other Nonfunctional Requirements

The system is engineered for high performance and responsiveness, ensuring that user requests such as asset lookups, QR scans, and report generation are processed within a maximum of two seconds under normal load. The backend logic is optimized through indexed queries and cached operations, allowing simultaneous access by multiple users without significant delays. The database schema follows normalization standards to minimize redundancy while supporting complex search operations across asset fields.

For a system to be functional, it must be available and reliable. The system targets a 99% uptime rate, supported by automated backups, error logging, and system health monitoring. Scheduled backups are configured daily, with cloud-based redundancy options to prevent data loss. System logs track server errors and user interactions, enabling developers or administrators to respond quickly to failures or anomalies.

Security measures are present in the system's application and server levels. User authentication is enforced via hashed passwords using the Bcrypt algorithm, and role-based access controls are strictly applied to minimize data exposure. Input validation is integrated across all forms to prevent SQL injection

and cross-site scripting (XSS) attacks. Server configurations also include HTTPS enforcement and firewall protection to safeguard data transmissions.

From a usability standpoint, the interface follows intuitive design principles and employs consistent iconography, tooltips, and form validation to guide users. Minimal training is required for end users due to the system's logical layout and straightforward task flows. Built-in user guides and help messages are included to support first-time users and reduce the need for external documentation.

The system is also scalable and maintainable. Its modular architecture allows for the easy addition of new features or modules without disrupting current functionalities. Code is written using Laravel's MVC pattern, which separates data logic from presentation layers for simplified debugging and upgrades. In anticipation of growing asset inventories or expanding user bases, the system supports horizontal scaling via load-balanced deployment or cloud-based hosting when necessary.

To assess user satisfaction, usability, and nonfunctional performance, a post-implementation survey was conducted using a five-point Pomel scale. This survey covered the system's six major features and their respective software quality attributes. The computed overall mean score was 4.60, which falls under the interpretation of "Excellent", indicating strong user approval of the system's design and performance.

For User-Friendliness, the system received mean scores of 3.85 for functional completeness, 4.85 for functional correctness, and 4.18 for functional appropriateness, reflecting its overall effectiveness in supporting user needs. In terms of Compatibility, both operability and consistency scored 4.87, showing that the system performed reliably across different devices and platforms with consistent interface behavior.

Reliability was one of the highest-rated dimensions, with accuracy receiving a mean score of 4.93, consistency at 4.89, and simplicity at 4.91. These ratings demonstrate the system's ability to maintain stable and precise asset data across various transactions. In the category of Interactivity, the system was rated 4.25 for simplicity and 4.26 for communicativeness, suggesting users found the interface intuitive and easy to navigate.

The Learning Reinforcement dimension yielded high scores as well: 4.15 for consistency, 4.85 for conciseness, and 4.86 for modularity. These indicate that the system's structure supports user comprehension and recall of processes. Lastly, in the Dynamic Environment category, the system earned 4.78 for operability, 4.70 for expandability, and 4.35 for modularity—confirming the platform's potential for future scalability and integration with other systems.

The consistently high scores validate its readiness for deployment in production environments and suggest strong potential for replication in similar enterprise settings.

6. Project Management

The development of the Automated Fixed Asset Management System followed a structured project management approach to ensure systematic planning, execution, and evaluation. At the outset, a feasibility study was conducted to assess the technical, economic, and operational viability of the project. Stakeholder interviews, environmental scanning, and SWOT analysis helped clarify the project's scope and objectives, resulting in a proposal that was both realistic and aligned with the needs of Yusay Credit & Finance Corporation.

A defined team structure supported project delivery. The project leader served as the systems analyst and lead developer, while support roles included testers, UI designers, and client representatives. Communication between stakeholders and the technical team was facilitated through regular meetings and progress reviews, which ensured that the project timeline and requirements remained synchronized with the client's expectations. Role assignments were clearly documented to promote accountability and workflow efficiency.

The development lifecycle adopted an Agile methodology, allowing iterative updates and continuous user feedback. The project was divided into sprints, each with specific deliverables: database design, user interface development, QR code integration, analytics module training, and deployment testing. This approach allowed for flexibility, especially when implementing machine learning components that required multiple rounds of tuning and validation.

Tools and technologies used included Visual Studio Code for development, MySQL Workbench for database management, and Git for version control. The system was built using PHP (Laravel framework), with front-end scripting in HTML, CSS, and JavaScript. For machine learning, Python was used to train the models, which were later integrated via REST APIs into the Laravel backend. Testing was conducted across various browsers and devices to ensure cross-platform functionality.

To prepare for deployment, training and documentation were provided to the organization's staff. This included system orientation sessions, user role demonstrations, and hands-on QR scanning exercises. Feedback from these sessions informed final refinements. Upon deployment, a support and maintenance plan was also created, including documentation for troubleshooting, system updates, and contact protocols in the event of technical issues.

7. Summary

The Automated Fixed Asset Management System with Predictive Analytics was developed to address the limitations of traditional manual asset tracking methods, which often result in inefficiencies, data loss, and delayed maintenance. By digitizing the process and incorporating modern technologies such as QR codes and machine learning algorithms, the system provides a more intelligent and scalable solution for fixed asset management. It facilitates real-time tracking, predictive depreciation analysis, and role-based access, all within a secure, responsive platform.

Through the use of QR code scanning, asset information can be accessed and updated swiftly, reducing human error and improving audit accuracy. Meanwhile, the integration of predictive analytics empowers organizations to anticipate maintenance needs and plan replacements before assets fail. These predictive features are supported by robust reporting tools and data visualization components, which provide valuable insights into asset conditions and lifecycle trends.

The system's architecture was designed for versatility and scalability, capable of operating in both connected and offline environments. Its modular design allows for easy integration of future enhancements such as API connections to ERP platforms or additional analytics models. Built with accessibility and user experience in mind, the system proved to be practical and effective during its implementation at Yusay Credit & Finance Corporation, improving both operational efficiency and decision-making.

In conclusion, the project successfully delivered a forward-thinking solution tailored for mid-sized enterprises in emerging markets. The results demonstrate that technology-driven asset management not only reduces administrative workload but also contributes to better financial planning and organizational resilience. This system lays a strong foundation for further innovation and may serve as a model for similar institutions seeking digital transformation in asset-intensive operations.

8. Acknowledgement

The author extends sincere thanks to Dr. Dennis V. Madrigal, Dr. Jake Pomperada, and the panel members, Dr. Reymund L. Sabay, PhD, Dr. Ruby Mae M. Bermejo, PhD, Dr. Wayne Custer G. Alegata, PhD, and Mariano D. Antenor, Jr., MIT, for their support. To Dean Dennis V. Madrigal, PhD for his guidance and support throughout this study. Special appreciation is also given to family, classmates, and colleagues for their encouragement throughout the completion of this research project. Thank you.

9. References

[1] Infraon. (2023). QR-Based Asset Management Systems. Retrieved from https://infraon.io

[2] World Bank. (2022). ASEAN Digital Transformation Insights. World Bank Publications.

[3] Gavrikova, E., Vorontsova, A., & Kapustina, I. (2020). Modern Challenges in Asset Management. International Journal of Economics and Business Administration, 8(4), 63–75.

[4] Ganti, S. (2024). QR Code Integration in Local Systems. Asian Journal of Emerging

Technologies, 6(2), 22–30.

[5] Murshida, H. (2023). Technology Adoption in Emerging Markets. Journal of Information Systems & Operations, 11(1), 18–27.

[6] Huang, Y., Qian, L., & Zhang, M. (2022). Predictive Maintenance Algorithms in Asset Management. Journal of Machine Learning and Systems, 4(3), 102–116.

[7] Rapach, D. E., & Zhou, G. (2020). Machine Learning in Forecasting: Algorithms and Evaluation. Journal of Financial Econometrics, 18(1), 1–20.

[8] Kartal, M., Duru, M., & Yilmaz, B. (2023). Lifecycle Analysis in IT Systems. Information Technology and Decision Making, 21(4), 597–612.

[9] Bezzina, F. (2024). Bridging Gaps in AI-Driven Systems. Journal of Emerging Digital Solutions, 9(2), 45–53.