Development of Estatix PMS: A Secure Edge-AI-Enabled Property Management System with Tenant Services, Role-Based Access Control, and Cryptographic Document Protection

A Capstone Project

Presented to the Faculty of the

College of Computing and Information Sciences

University of Makati

By:

ARAGON, KIVZ

BERONIA, EDRIAN

CAJEFE, JIMBOY

CULABAN, RHENZ

In Partial Fulfillment

of the Requirements for the Degree

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

(Information and Network Security Elective Track)

November 2026

APPROVAL SHEET

ACKNOWLEDGMENT

The proponents extend their heartfelt gratitude to their research adviser, instructors, and the faculty members of the College of Computing and Information Sciences at the University of Makati for their invaluable guidance, patience, and constructive recommendations throughout the conceptualization and development of Estatix PMS.

Sincere appreciation is likewise extended to the administration, staff, and residents of Torre De Manuel Residences 1 for their warm cooperation, permission, and operational insights during the requirements gathering phase. Their willingness to share daily property management challenges provided the foundation for this proposed system.

The proponents also express their warm gratitude to Jesally Mae G. Dela Pena for her generous assistance, encouragement, and invaluable help throughout the preparation and progress of this study.

Finally, the proponents convey their deepest love and gratitude to their parents, families, and friends for their unending prayers and moral support. The proponents also commend one another for their dedication, shared responsibilities, and teamwork in bringing this capstone research proposal to fruition.

DEDICATION

The proponents dedicate this capstone project to Almighty God, whose guidance, grace, and wisdom provided them with strength and perseverance throughout this journey. In moments of difficulty and uncertainty, their faith helped them remain hopeful, focused, and determined to complete this work.

This work is lovingly dedicated to their parents and families, whose sacrifices, patience, encouragement, and unwavering support have served as their inspiration. Their understanding and belief in the proponents' abilities made it possible to continue pursuing their education and accomplish this project. The proponents also dedicate this study to their teachers, research adviser, and mentors, whose knowledge, guidance, and encouragement contributed to their academic and professional growth. The lessons and values they shared helped shape the proponents' skills, discipline, and commitment to continuous learning. This capstone project is further dedicated to the administration, personnel, and tenants of Torre De Manuel Residences 1. May Estatix PMS contribute to more organized property management operations, more accessible tenant services, and improved protection of important records.

Lastly, the proponents dedicate this accomplishment to one another. The challenges encountered, responsibilities shared, and efforts made throughout the development of this project reflect their cooperation, perseverance, and collective commitment. May this work serve as a reminder that meaningful achievements are made possible through dedication, teamwork, and the support of others.

TABLE OF CONTENTS

Title Page..... i
Approval Sheet..... ii
Acknowledgement..... iii
Dedication..... iv
Table of Contents..... v
List of Tables..... vi
List of Figures..... vii

CHAPTER 1 - THE BACKGROUND AND ITS PROBLEM

Introduction..... 1
Background of the Study..... 5
Objective of the Study..... 9
Significance of the Study..... 12
Scope and Limitations of the Study..... 14
Operational Definition of Terms..... 16

CHAPTER 2 - **REVIEW OF RELATED LITERATURE AND STUDIES**

Small Residential Rental Systems in the Philippines & Property Management Operations..... 19
Local Large Language Models (LLM) for Tenant Support and Automated Ticket Triage..... 21
Filipino and Taglish Natural Language Processing (NLP) Performance in Conversational AI..... 22
Regulatory Compliance Under the Philippine Data Privacy Act of 2012 (RA 10173) and Privacy-by-Design..... 24
Progressive Web Applications (PWAs) & Responsive Frontend Engineering..... 25
Backend Architecture: Modular Monoliths and Relational Database Integrity (3NF)..... 26
Real-Time Communication: WebSockets vs. Traditional HTTP Polling.... 27
Identity Governance & Role-Based Access Control (RBAC)..... 27
Cryptographic Document Protection & Tamper Verification (AES-256-GCM & SHA-256)..... 28
Audit Logging, Non-Repudiation, and Application Security.....29
Software Quality and Security Evaluation Frameworks..... 30

Synthesis of Literature and Studies..... 30
Conceptual Model of the Study..... 36
CHAPTER 3 - RESEARCH METHODOLOGY
Research Design..... 41
Project Design..... 45
System Flowchart Diagram..... 48
Requirements Gathering..... 75
Project Development..... 85
Operational and Testing Procedure..... 93
Evaluation Procedure..... 97
Treatment of Data..... 103
Ethical Consideration..... 105
BIBLIOGRAPHY..... 107
APPENDICES..... 116
Appendix A..... 116
Appendix B..... 117
Appendix C..... 118
Appendix D..... 126
RESEARCHERS' PROFILE..... 228

LIST OF TABLES

No.	Title	Page
1	**Table 3.3.1** Functional Requirements of Estatix PMS	79
2	**Table 3.3.2** Non-Functional Requirements of Estatix PMS	81
3	**Table 3.3.3** Tools, Technologies, and Environment for Estatix PMS	83
4	**Table 3.4.1** Developmental Procedures Across Agile Sprints	91
5	**Table 3.5.1** Operational Procedure	93
6	**Table 3.5.2** Testing Procedure	94
7	**Table 3.6.1** Evaluation Frameworks and Standards Used	98
8	**Table 3.6.2** Evaluation Criteria and Focus Areas	99
9	**Table 3.6.3** TAM Evaluation Dimensions	100
10	**Table 3.6.4** Four-Point Rating Scale	101
11	**Table 3.7.1** Weighted Mean	104

LIST OF FIGURES

No.	Title	Page
1	**Figure 2.1.1** Conceptual Framework of the Study	37
2	**Figure 3.1.1** System Architecture of Estatix PMS	46
3	**Figure 3.2.1** Flow 1: Login Flowchart	50
4	**Figure 3.2.2** Flow 2: Register Flowchart	52
5	**Figure 3.2.3** Flow 6: Apply Viewing Schedule Flowchart	54
6	**Figure 3.2.4** Flow 7: Lease Application Flowchart	56
7	**Figure 3.2.5** Flow 8: Maintenance Request Flowchart	58
8	**Figure 3.2.6** Flow 9: Staff Tasks Flowchart	60
9	**Figure 3.2.7** Flow 10: Payments Flowchart	62
10	**Figure 3.2.8** Flow 11: Lease Termination Flowchart	64
11	**Figure 3.2.9** Flow 12: Security Deposit Refund Flowchart	66
12	**Figure 3.2.10** Flow 14: Property Management Flowchart	68
13	**Figure 3.2.11** Flow 15: Announcement Flowchart	70
14	**Figure 3.2.12** Flow 16: Chat Support Flowchart	72
15	**Figure 3.2.13** Flow 17: Community Forum Flowchart	74
16	**Figure 3.4.1** Agile Software Development Framework	88

Chapter I

THE BACKGROUND AND ITS PROBLEM

This chapter presents the background of the study and the problem it addresses at Torre De Manuel Residences 1, where property operations still rely on manual, spreadsheet-based records. It introduces Estatix PMS, a secure Edge-AI-enabled Progressive Web Application with Role-Based Access Control, audit logging, AES-256-GCM encryption, and SHA-256 file-integrity verification. It also states the objectives, significance, scope and limitations of the study, and the operational definition of terms.

INTRODUCTION

Torre De Manuel Residences 1 (TDMR1) is a medium-height residential building situated along E. Rodriguez, Cubao, Quezon City, comprising 96 rental studio units. Each day, operations for the property involve a variety of administrative tasks such as: maintaining tenant personal data, managing lease contracts, monitoring rent and utility payments, processing maintenance service requests, and sharing general updates. This operational baseline was validated during an interview conducted upon the signing of the Letter of Intent dated August 13, 2026, where the building administration confirmed that their current setup relies strictly on manual operations, with physical property records and lease documents stored simply in a metal document box. There is currently no centralized record-keeping system, only a variety of spreadsheets, paper-based

processes, and disconnected messaging channels. Many unlinked records contain information about tenants and daily operations. With increasing tenant turnover and administrative tasks, this dependence on disjointed communication and manual recording has become a heavy administrative burden. The manual and disjointed system reflects the urgent need to streamline and modernize the current property management process.

The practical implications of having a disjointed and disorganized record system represent a serious administrative challenge every day. Dispersing records across paper documents and individual spreadsheet files makes it difficult to keep tenant information consistent, synchronized, and immediately accessible. Chauhan (2024) emphasized that when property records are not properly maintained or centralized, administrative coordination between operational staff deteriorates and residents become frustrated while awaiting assistance. The practical results of these manual workflows are evident in delayed follow-ups on delinquent accounts and the heightened risk of critical lease terms being lost across informal communication channels. Furthermore, without a unified platform, property managers repeatedly encounter difficulties in maintaining reliable maintenance request histories and establishing structured payment schedules, making it substantially more time-consuming to address tenant concerns.

In addition to operational delays, this manual setup poses significant information security and privacy risks regarding critical tenant personal data and lease agreements. Villasin and Encarnacion (2024) established that manual data

handling and the absence of a centralized digital platform lead to administrative inefficiencies, data entry errors, and limited accessibility for property managers and clients in the Philippines. Their empirical survey revealed that 45.5% of real estate professionals experience frequent operational difficulties when relying on manual workflows, with 90.9% expressing a decisive preference for web-based property management systems. Because physical files at TDMR1 are stored in a basic metal box and electronic spreadsheet files remain unencrypted, records are vulnerable to unauthorized access, accidental disclosure, physical loss, or data corruption. Property records store personally identifiable information (PII) and legally binding contractual terms; without enforced digital boundaries, confidentiality is compromised and accountability cannot be proven. To mitigate these operational and security vulnerabilities, a standardized role-based access control framework and cryptographic document protection mechanism are essential.

To address these challenges, the proponents propose the development of Estatix PMS, a centralized and secure Progressive Web Application (PWA) that unifies property administration into a single platform. Estatix PMS encompasses tenant profiling, lease lifecycle management, digital payment tracking, maintenance request ticketing, and automated notification dissemination. Consolidating these operational functions into one system eliminates duplicate records and removes administrative bottlenecks. Property administrators gain an authoritative control hub, while tenants receive private, structured access to their accounts and an organized communication channel.

Estatix PMS will incorporate modern technical modules engineered to support administrative workflows while enforcing strict data security. Edge-AI-assisted tenant services will process common inquiries and automatically categorize and route maintenance service requests to appropriate personnel. Role-Based Access Control (RBAC) enforces the principle of least privilege across user tiers. Comprehensive audit logging records critical system transactions to ensure traceability and non-repudiation. Authenticated AES-256-GCM encryption secures sensitive stored files such as lease agreements and billing statements, while SHA-256 cryptographic hashing provides continuous file-integrity verification.

Existing literature and commercial software address individual components of property administration, but they consistently leave a critical operational gap for small-to-medium residential rental properties in the Philippines. While enterprise property management platforms offer comprehensive automation, their high subscription pricing and per-unit licensing models exceed the operational budgets of small facilities like Torre De Manuel Residences 1. Conversely, informal local workarounds relying on spreadsheets, paper files, and public messaging apps expose highly sensitive personally identifiable information, including government identification cards, employment records, financial statements, and tenancy contracts, to severe confidentiality and legal risks under the Data Privacy Act of 2012 (Republic Act No. 10173).

This operational challenge is further compounded by the complete absence of a dedicated in-house IT staff or cybersecurity team at the property

level. The administration cannot support complex distributed cloud architectures, third-party cloud AI subscriptions that risk external data leakage, or complicated database management systems. Consequently, the practical research problem is not simply assembling disconnected software features, but rather engineering a unified system that resolves this tension: delivering automated, responsive tenant services and strong cryptographic document protection (AES-256-GCM authenticated encryption and SHA-256 integrity verification) alongside privacy-preserving, localized Edge-AI support, while remaining lightweight and maintainable on a modest budget without specialized technical personnel. To bridge this gap, the proponents propose to develop Estatix PMS for Torre De Manuel Residences 1 to provide a secure, cost-effective, and operationally viable management platform tailored to the realities of local residential housing.

BACKGROUND OF THE STUDY

For modern property administration, centralizing digital records is critical to ensure operational transparency, workflow automation, and information security. Contemporary residential facilities require integrated platforms that combine lease administration, payment tracking, tenant interaction, and administrative oversight within a secure computing environment. Oyetunji et al. (2025) demonstrated that implementing structured digital solutions for property management operations streamlines routine workflows, reduces administrative overhead, and significantly enhances communication channels between property managers and residents, particularly regarding lease monitoring and rent collection. In the absence of contemporary property management software,

multi-tenant residential facilities face operational disruption, financial leakage, and increased risks of data exposure. When records are kept across disparate physical and electronic media, keeping them accurate and synchronized becomes increasingly difficult as unit occupancy expands.

Torre De Manuel Residences 1 (TDMR1) was previously affiliated with RedDoorz, during which its units were operated under a hybrid model of short-term hotel bookings and private condominium rentals. In 2025, the building transitioned to a 100% long-term residential rental model. Its units are studio-type residences with floor areas ranging between 17 and 19 square meters. With this operational transformation, the property administration became responsible for long-term lease agreements, recurring monthly rent and utility billing, and ongoing tenant relations rather than temporary guest hospitality.

According to the Building Administration, the property's 96 studio units are currently managed using physical paper forms, disconnected spreadsheets, and informal messaging channels for tenant onboarding, lease tracking, maintenance request handling, and utility bill verification. The leasing workflow begins when a prospective tenant inspects an available unit and deposits a reservation fee. Upon approval, the tenant pays two months of security deposit along with one month of advance rent, followed by the physical signing of a printed lease contract. Throughout the tenancy, communication is handled individually by a single building administrator who contacts tenants via personal messaging platforms such as Viber. This arrangement relies heavily on a single individual to address inquiries, dispatch repairs, and manually track ongoing conversations.

Utility billing and collection present acute operational friction points. Water and electricity (Meralco) bills are physically distributed by lobby security guards, while general building notices are taped to elevator walls. Water bill collections and monthly rental payments are settled strictly in person at the administration office, where the administrator manually calculates balances and writes paper receipts. This siloed workflow delays follow-ups on delinquent payments, leaves room for calculation and transcription errors, and makes it difficult to maintain a reliable payment history. Because paper notices and bills can be overlooked or misplaced, tenants frequently miss payment deadlines without the administration having proof of receipt.

The practical consequences of this centralized, person-dependent payment setup were directly documented during an on-site field observation conducted by the researchers in September 26, 2026. When a resident visited the building administration office during normal operating hours to settle a due utility and rental bill, the available on-site staff informed the tenant that payments could not be accepted because the building administrator was away from the office. Upon returning later that same afternoon, the administrator was still unavailable. Because the subordinate on-site staff lacked authorization, receipt books, and access to the manual payment ledger, the transaction could not be processed, forcing the tenant to delay payment until the following Monday. This documented scenario exposed a critical operational bottleneck: the entire financial collection process relies entirely on the physical presence of a single administrator. Such dependency not only causes inconvenience and potential

false delinquencies for tenants, but also delays revenue collection and disrupts administrative cash flow reconciliation.

Similarly, the operational vulnerability of relying on physical notices was documented during an on-site observation on October 28, 2026. On this occasion, residents on multiple floors experienced an unexpected drop in water pressure throughout their units. Because the building lacks an automated, real-time alert distribution mechanism, the administration's sole method of informing residents was to draft, print, and tape a physical paper advisory onto the elevator walls announcing a temporary low water supply. Consequently, residents only learned about the ongoing utility interruption reactively, after already encountering water disruption in their units and physically descending the elevator to the ground floor. This incident highlights the operational latency of manual communication channels, which fail to reach residents who remain in their units or are away from the building, while leaving the administration with no verifiable proof of announcement receipt.

From an information security and legal compliance standpoint, the current manual procedure exposes the property to substantial risks. Lease agreements and tenant profiles containing sensitive personally identifiable information (PII), such as government IDs, contact numbers, and employment details, are kept in an unlocked metal container or unencrypted local drives, lacking role-based authorization or encryption. Elevator notices expose private resident unit numbers to visitors, while transaction discussions conducted over informal chat applications fail to generate a tamper-evident, auditable record. Furthermore, the

absence of system access controls prevents the establishment of accountability. Without tamper-evident audit logs, it is impossible to determine who accessed, updated, or deleted tenant records.

Given these operational, financial, and security challenges, Torre De Manuel Residences 1 requires an integrated, secure web platform to consolidate records, billing, maintenance, and communications. This study addresses this need through the development of Estatix PMS, providing administrators with automated invoicing and online payment tracking tools while affording tenants a private, structured portal for lease monitoring, payment verification, and service assistance.

OBJECTIVE OF THE STUDY

The primary goal of this study is to design, develop, test, and evaluate a secure Edge-AI-enabled Property Management System (Estatix PMS) for Torre De Manuel Residences 1, incorporating centralized tenant services, Role-Based Access Control (RBAC), append-only audit logging, and cryptographic document protection to enhance administrative efficiency and data confidentiality.

Specifically, the study aims to:

1. **Design**

   Design the system architecture, relational database schema (normalized to Third Normal Form), and user interaction workflows of Estatix PMS to integrate property and tenant record management, lease monitoring, maintenance ticketing, digital payment tracking, and real-time

communication. The design maps out the integration of an on-premise Edge-AI inference pipeline for tenant support alongside core security mechanisms, including RBAC, audit logging, AES-256-GCM authenticated encryption, and SHA-256 file-integrity hashing.

2. **Develop**

Develop Estatix PMS as a responsive Progressive Web Application (PWA) using a modular monolithic PHP architecture with MySQL, Soketi WebSockets, and Tailwind CSS. The system integrates an on-premise Large Language Model (quantized Qwen2.5 7B deployed via Ollama on a dedicated edge server) to handle routine tenant support and automate maintenance ticket triage without relying on external cloud AI endpoints, while implementing AES-256-GCM encryption for stored lease documents and SHA-256 hashing for tamper detection.

3. **Test**

Test Estatix PMS through systematic unit, integration, security, and performance testing to validate that its operational and cryptographic modules function reliably. Testing encompasses validating the accuracy of Edge-AI classification and intent recognition (including code-switched Taglish inputs), verifying the enforcement of RBAC boundaries and authentication gates, confirming the correctness of AES-256-GCM file encryption/decryption, verifying SHA-256 tamper-detection, and validating the completeness of append-only audit logging.

4. **Evaluate:**

Evaluate the proposed system of Estatix PMS through a multi-tier assessment in terms of:

4.1 Software Product Quality: The software product quality characteristics of the ISO/IEC 25010:2023 standard, specifically functional suitability, performance efficiency, interaction capability, reliability, security, and maintainability.

4.2 Information Security Controls: The technical information security controls adapted from ISO/IEC 27001:2022 Annex A, covering access control, identity management, secure authentication, audit logging, and cryptographic protection.

4.3 User Acceptance and Adoption: The level of user acceptance and practical adoption among tenants, building administrators, and staff personnel using the Technology Acceptance Model (TAM), measuring perceived usefulness, perceived ease of use, user trust, and behavioral intention to use the system.

SIGNIFICANCE OF THE STUDY

This study consolidates property administration, localized AI assistance, and cryptographic document security into a centralized platform tailored for TDMR1. By transitioning manual paper records and informal chats into a unified digital ecosystem, the study provides direct benefits to the following stakeholders:

Torre De Manuel Residences 1 (TDMR1): Provides the property with a modern operational infrastructure that eliminates data fragmentation, streamlines lease administration for its 96 rental units, protects tenant records, and ensures systematic billing and maintenance coordination.

Property Owner / Landlord: Grants the property owner real-time executive visibility into building performance, rental income, cash flow projections, unit occupancy rates, and maintenance turnaround times. Append-only audit logs ensure operational transparency and financial integrity.

Building Administrator: Dramatically reduces administrative workload by automating monthly rent and utility billing, centralizing maintenance dispatching, providing digital payment receipts, and filtering repetitive tenant inquiries through the Edge-AI assistant.

Building Maintenance Staff: Establishes a structured digital work order pipeline. Staff receive categorized repair tickets with priority ratings and attached

photos directly on their mobile portal, enabling efficient field execution, status updating, and photo-resolution submission.

Tenants / Residents: Delivers a secure, private self-service portal where residents can review lease terms, download itemized billing statements, submit maintenance tickets with photos, make digital payments, receive building advisories, and obtain 24/7 AI-assisted support.

Prospective Tenants / Applicants: Streamlines the onboarding process by enabling prospective clients to browse vacant units, schedule viewings, submit lease applications, complete identity verification, and sign digital contracts remotely.

The Researchers: Enables the proponents to apply software engineering principles, full-stack web development, edge AI deployment, cryptographic data protection, and international software evaluation standards (ISO/IEC 25010 and ISO/IEC 27001) in solving an authentic operational problem.

Future Researchers: Serves as an empirical reference and architectural blueprint for subsequent investigations exploring privacy-preserving edge AI, code-switched NLP in administrative systems, and cryptographic document protection in localized enterprise software.

SCOPE AND LIMITATIONS OF THE STUDY

This study focuses on the development and evaluation of Estatix PMS, a secure Edge-AI-assisted property management system for TDMR1 in E. Rodriguez, Cubao, Quezon City. The system covers property and tenant record management, lease creation and tracking, online payment through the Xendit payment gateway, payment monitoring, maintenance requests, tenant-administrator communication, announcements, and report generation. Its intended primary users are the property owner/landlord, the building administrator, operational staff, and tenants, with external IT specialists participating during system evaluation. The Edge-AI-assisted tenant services will run on a dedicated local edge server using Ollama and a quantized Qwen2.5 7B model, which will answer routine inquiries and classify and route maintenance requests to the appropriate personnel. The security scope includes user authentication, RBAC, audit logging, AES-256-GCM encryption of documents, and SHA-256 hashing for file-integrity verification. Estatix PMS will be developed as a PWA using a Modular Monolithic architecture, with Vanilla PHP, Vanilla Javascript, Tailwind CSS and DaisyUI, Vue.js, MySQL, and Soketi for real-time communication. The system will be evaluated using six ISO/IEC 25010:2023 quality characteristics: functional suitability, performance efficiency, interaction capability, reliability, security, and maintainability. The security evaluation will use selected control areas from ISO/IEC 27001:2022 Annex A: access control, identity management, authentication, logging, cryptography, application security, secure architecture, and security testing. In addition, user acceptance and

practical adoption will be assessed among end users using the Technology Acceptance Model (TAM), examining perceived usefulness, perceived ease of use, perceived security and trust, and behavioral intention to adopt the platform.

The study covers only the operational requirements of TDMR1 and does not cover workflows of other commercial or residential property types. The Edge-AI component only supports routine tenant inquiries and the classification and routing of maintenance requests, so it does not make administrative decisions. Complex or policy-sensitive issues are referred to authorized personnel, and AI responses may not always be correct. Online payments are processed by Xendit, so the study covers only how Estatix PMS records and monitors these payments and does not cover the security of the gateway itself. Cryptographic document protection applies only to sensitive documents handled inside Estatix PMS, such as lease contracts and billing statements, and not to external storage platforms, third parties, or systems outside the application. The security evaluation covers only the implemented controls and is not an ISO/IEC 27001:2022 certification or a full Information Security Management System (ISMS) assessment. Similarly, the TAM evaluation is delimited to measuring the user perceptions and adoption intentions of the target respondents at TDMR1 and does not generalize behavioral acceptance patterns across broader commercial real estate markets.

OPERATIONAL DEFINITION OF TERMS

The following technical terms and operational concepts are defined according to how they are specifically used, implemented, and interpreted within the context of this study:

AES-256-GCM (Advanced Encryption Standard in Galois/Counter Mode): A symmetric authenticated encryption algorithm used in Estatix PMS to encrypt sensitive stored documents (lease contracts, identification files, and billing receipts), ensuring data confidentiality while producing an authentication tag to detect unauthorized tampering.

Audit Logging: The automated, append-only recording of user and system events within Estatix PMS, capturing user IDs, action names, timestamps, client IP addresses, User-Agents, and before-and-after JSON state differentials to ensure non-repudiation and forensic accountability.

Edge-AI-Assisted Tenant Services: A localized artificial intelligence module running a quantized Qwen2.5 7B language model on an on-premise server via Ollama, securely connected to the application via an encrypted tunnel. It processes routine tenant inquiries and classifies maintenance tickets locally, ensuring personal tenant data is never transmitted to external cloud AI vendors.

Estatix PMS: The secure, web-based Property Management System being developed specifically for Torre De Manuel Residences 1, integrating tenant services, property workflows, RBAC, edge AI, and cryptographic document protection.

File-Integrity Verification: The security process of generating a cryptographic SHA-256 hash digest of an uploaded file, storing the hash in a secure database, and re-computing the digest upon document retrieval to verify that the file has not been altered or corrupted.

ISO/IEC 25010:2023: The international standard for Systems and Software Quality Requirements and Evaluation (SQuaRE), utilized in this study to evaluate functional suitability, performance efficiency, interaction capability, reliability, security, and maintainability.

ISO/IEC 27001:2022 Annex A: The international information security standard whose control categories (access control, identity management, authentication, logging, and cryptography) serve as the benchmark for evaluating the technical security posture of Estatix PMS.

Maintenance Request: A structured digital service order submitted by a tenant detailing a physical unit defect (including title, description, category, and photo attachments), tracked through open, in-progress, resolved, and closed states.

Modular Monolithic Architecture: The software architectural pattern utilized to develop Estatix PMS, organizing application logic into distinct, self-contained functional modules (Authentication, Leases, Billing, Maintenance, AI Support) within a single unified codebase and relational database.

Progressive Web Application (PWA): A web application design paradigm utilizing modern browser APIs, web app manifests, and service

workers to provide installable, cross-platform, app-like experiences on mobile and desktop devices without app store distribution.

Role-Based Access Control (RBAC): An authorization mechanism that restricts system features, data records, and administrative actions based on assigned user roles (Superadmin, Building Admin, Landlord, Staff, Tenant, and Applicant), enforcing the principle of least privilege.

SHA-256 (Secure Hash Algorithm 256-bit): A cryptographic hash function that produces a unique 256-bit signature representing a file's content, used in Estatix PMS for tamper detection and document integrity validation.

Soketi: An open-source, lightweight WebSocket server compatible with the Pusher protocol, used to provide real-time, bidirectional event broadcasting for instant messaging and operational alerts.

Technology Acceptance Model (TAM): An information systems framework originally formulated by Davis (1989), adapted in this study to evaluate user acceptance and adoption intention of Estatix PMS across perceived usefulness, perceived ease of use, user trust, and behavioral intention.

Torre De Manuel Residences 1 (TDMR1): The target research and deployment site, a mid-rise residential building in Cubao, Quezon City, comprising 96 studio rental units.

Chapter II

REVIEW OF RELATED LITERATURE AND STUDIES

This chapter examines the relevant literature and studies related to the development and evaluation of a secure, web-based property management system for Torre De Manuel Residences 1. It encompasses local and foreign literature, related studies, the conceptual framework, and the technical background of the project. The aim is to contextualize the current study within the broader academic and practical landscape, highlighting foundational principles, existing technological advancements, and pertinent findings that inform and support the development of the proposed system.

Small Residential Rental Systems in the Philippines & Property Management Operations

In the Philippine real estate and residential rental sector, property managers frequently grapple with administrative friction arising from fragmented, manual record-keeping practices. Villasin and Encarnacion (2024) conducted an empirical assessment among Philippine realtors evaluating property information processing workflows. Their investigation revealed that 45.5% of real estate professionals experience routine operational difficulties when relying on manual workflows, with 90.9% expressing a decisive preference for web-based property systems. They documented that manual data handling causes recurring data entry errors, billing discrepancies, and significant delays in retrieving tenant

records and lease contracts. Consequently, they emphasized that transitioning to centralized, web-based property management platforms is essential for operational transparency and client satisfaction in local property environments.

Complementing these local observations, Caparos et al. (2025) analyzed cloud-based property management systems designed to enhance residential operations and tenant experiences. Their study demonstrated that centralized digital portals significantly improve tenant-landlord communication, simplify monthly rental billing, and reduce operational bottlenecks in property facilities. Similarly, Magno et al. (2024) evaluated an apartment rental management system designed for real-time transaction processing and task organization. They established that automating billing calculations, digital payment receipt generation, and maintenance dispatching substantially reduces administrative friction and human accounting mistakes.

On a broader scale, Oyetunji et al. (2025) examined the adoption of digital solutions for streamlining property management operations within affordable housing communities. Their research established that implementing structured digital workflows simplifies lease tracking, standardizes tenant communications, and mitigates the recurring operational overhead associated with disparate physical files. In parallel, Chauhan (2024) emphasized that poor data and record-keeping in residential housing directly harms residents by delaying maintenance responses and creating severe accountability deficits. Salunkhe (2026) and Buradkar (2022) further affirmed that integrated web frameworks combining lease lifecycle management, digital invoicing, and maintenance

tracking provide landlords with dependable operational visibility while preventing payment disputes.

Local Large Language Models (LLM) for Tenant Support and Automated Ticket Triage

The integration of artificial intelligence in customer service and property maintenance has transitioned from static rule-based chatbots to advanced Large Language Models. However, deploying public cloud-hosted LLM APIs poses severe data privacy, operational cost, and internet dependency challenges. Qu et al. (2024) conducted an extensive survey on mobile edge intelligence for large language models, demonstrating that quantized open-source models deployed on edge servers or local infrastructure drastically reduce network latency, eliminate recurring third-party API subscription costs, and prevent sensitive enterprise and user data from leaving internal boundaries.

Investigating practical customer service automation, Deng and Chen (2024) introduced TickIt, an automated system demonstrating that localized open-source LLMs can effectively classify support tickets, determine issue severity, and extract structured diagnostic metadata from conversational descriptions without relying on external cloud endpoints. In parallel, Sharma and Gupta (2023) demonstrated that deploying quantized open-source LLMs via lightweight containerized execution engines such as Ollama achieves reliable inference throughput on consumer-grade hardware while preserving strict data residency.

Pandya and Holia (2023) demonstrated that deploying open-source, on-premise language models enables automated customer assistance without exposing proprietary or personally identifiable information to external third-party cloud providers. By confining language model inference to local hardware, organizations can process user inquiries instantaneously while preventing external cloud data leakage. Complementing this, Truss et al. (2024) established that machine learning classification of support and trouble tickets accurately routes incoming requests to appropriate service personnel, preventing maintenance bottlenecks. Furthermore, Adam et al. (2022) revealed that user trust and willingness to comply with automated chatbot recommendations depend fundamentally on the conversational agent's transparency, response speed, and perceived security.

Filipino and Taglish Natural Language Processing (NLP) Performance in Conversational AI

In multi-tenant residential properties across the Philippines, tenants routinely communicate maintenance concerns, lease inquiries, and emergency reports using code-switched Filipino-English (Taglish) rather than formal English. Evaluating language models in this linguistic context is critical. Ramos et al. (2026) developed BATAYAN, an extensive Filipino NLP benchmark evaluating large language models across multiple tasks. Their findings revealed that while state-of-the-art models exhibit strong capabilities in formal English, their zero-shot accuracy degrades on colloquial Filipino and informal Taglish syntactic

constructs, emphasizing the necessity of targeted few-shot prompting, localized system instructions, and domain-adapted vocabularies for Philippine service applications.

Addressing code-switching dynamics, Cinco and Devaraj (2026) developed LexCAT, demonstrating that lexicon-enhanced transformer architectures significantly enhance intent identification and sentiment extraction in Tagalog–English code-switched communications. Earlier foundational work by Cruz and Cheng (2022) highlighted the persistent performance disparities between high-resource languages and low-resource Austronesian languages, establishing open-source language models tailored for Filipino linguistic subtleties.

Cosme and De Leon (2024) evaluated transformer-based language models on code-switched Filipino-English consumer and service reviews, demonstrating that transformer architectures effectively capture informal slang, abbreviations, and mixed syntax commonly found in local customer interactions. Furthermore, Nguyen and Le (2023) assessed multilingual LLMs across Southeast Asian languages, verifying that contextually grounded prompt structures and localized inference pipelines mitigate syntactic misunderstanding in multilingual user interfaces.

Regulatory Compliance Under the Philippine Data Privacy Act of 2012 (RA 10173) and Privacy-by-Design

Handling tenant government identification documents, contact details, payment transaction proofs, and lease contracts imposes strict legal obligations under the Philippine Data Privacy Act of 2012 (Republic Act No. 10173). The National Privacy Commission (NPC, 2022), through NPC Circular No. 2022-01, established stringent administrative fines and compliance guidelines, holding personal information controllers (PICs) and processors (PIPs) accountable for failing to enforce appropriate organizational, physical, and technical security safeguards.

To operationalize these mandates, the National Privacy Commission (2023) issued NPC Circular No. 2023-06, establishing comprehensive technical standards for personal data protection across both public and private entities. Under this regulatory issuance, systems processing sensitive personal records must enforce mandatory encryption for data in transit and at rest, institute non-bypassable role-based access restrictions, implement tamper-evident event logging, and conduct periodic vulnerability assessments. Operating a property management system without these verified safeguards exposes administrators to substantial statutory penalties and civil liability.

At an architectural level, Gurses and Del Alamo (2022) articulated the paradigms of privacy engineering and privacy-by-design, demonstrating that privacy protections cannot be retrofitted post-development. Instead, privacy

safeguards must be embedded directly into database schemas, API boundaries, session handlers, and access control matrices from inception.

Progressive Web Applications (PWAs) & Responsive Frontend Engineering

Modern property management systems require universal accessibility across smartphones, tablets, and desktop workstations without imposing separate, platform-specific native application downloads. Roumeliotis and Tselikas (2022) evaluated progressive web app accessibility and cross-platform mobile standardization, showing that service-worker-enabled web applications offer responsive interface rendering and accessible navigation across diverse screen resolutions while eliminating app store deployment fragmentation and reducing ongoing maintenance costs.

Muawwal (2024) demonstrated that service workers and asset caching strategies significantly decrease page load latency and network bandwidth consumption across mobile devices. Complementing this, Wagh et al. (2023) compared user experiences between PWAs and native mobile apps, confirming that PWAs deliver comparable task completion speed and user satisfaction. Furthermore, Silva & Santos (2023) evaluated utility-first CSS frameworks (such as Tailwind CSS), finding that utility classes streamline responsive UI styling, reduce stylesheet bundle overhead, and maintain visual consistency across complex multi-role dashboards.

Backend Architecture: Modular Monoliths and Relational Database Integrity (3NF)

Choosing an appropriate backend architecture is critical for long-term maintainability, operational efficiency, and deployment simplicity. Coda et al. (2024) investigated software architectural trends, demonstrating that for small-to-medium enterprise platforms, a modular monolithic architecture provides superior code navigability, simpler transaction boundaries, and lower deployment complexity compared to distributed microservices, which introduce substantial operational overhead and inter-service network latency.

Regarding data persistence, relational database normalization remains the cornerstone of enterprise data integrity. Akadal and Satman (2022) investigated relational schema optimization, proving that algorithmic decomposition up to Third Normal Form (3NF) systematically eliminates transitive dependencies, avoids update anomalies, and optimizes storage efficiency in multi-table transactional architectures. In a property management context, 3NF schema design prevents insertion, update, and deletion anomalies across tenant accounts, utility billings, and maintenance tracking tables while facilitating rapid indexed query execution.

Real-Time Communication: WebSockets vs. Traditional HTTP Polling

Real-time capabilities are critical for interactive chat interfaces, instant maintenance status notifications, and live operational alerts. Łasocha and Badurowicz (2022) conducted a comparative performance benchmark between WebSocket and HTTP protocols. Their findings proved that full-duplex WebSocket connections drastically minimize communication latency and network overhead by eliminating the redundant HTTP header payloads and TCP connection teardowns characteristic of periodic HTTP polling.

Obonguko (2024) analyzed high-frequency data streaming protocols for web applications, demonstrating that WebSocket connections maintain persistent bidirectional communication channels, reducing server CPU utilization and bandwidth consumption in concurrent multi-user environments compared to long-polling techniques.

Identity Governance & Role-Based Access Control (RBAC)

Multi-tenant management systems require strict separation of operational duties between administrators, property managers, maintenance personnel, and tenants. Ri et al. (2022) evaluated role-based access control (RBAC) models with separation of duties constraints, establishing that defining explicit user-role mappings and verifying permissions across data layers prevents unauthorized privilege escalation and maintains strict boundaries between administrative and client operations.

Addressing web application and API exposure, the OWASP Foundation (2023) highlighted broken object level authorization (BOLA) and broken access control as the most prevalent vulnerabilities facing modern web services. OWASP established that client-side role checks are inherently insufficient; backend controllers must validate session privileges, enforce route middleware, and check explicit tenant resource ownership on every transaction before executing create, read, update, or delete commands.

Cryptographic Document Protection & Tamper Verification (AES-256-GCM & SHA-256)

Property management platforms handle highly confidential digital assets, including government IDs, tenancy agreements, and payment transaction receipts. Saha and Das (2023) investigated authenticated encryption with AES-256 in Galois/Counter Mode (AES-GCM), demonstrating that AES-256-GCM provides both strong confidentiality and built-in cryptographic integrity verification via authentication tags, effectively protecting sensitive files at rest against unauthorized tampering and chosen-ciphertext attacks.

Chen (2024) evaluated the comparative security and performance of Advanced Encryption Standard operational modes, confirming that Galois/Counter Mode (GCM) combines counter-mode high throughput with authenticated tag validation, delivering superior security margins over legacy cipher modes. In parallel, Bhattacharya and Roy (2023) examined cryptographic hashing using SHA-256 for document integrity verification, proving that storing

SHA-256 digests enables instantaneous, tamper-proof verification of stored digital records, ensuring that unauthorized alterations are reliably detectable.

Audit Logging, Non-Repudiation, and Application Security

Establishing undeniable accountability for operational actions is essential for fraud prevention and legal compliance. Thazhath et al. (2022) evaluated privacy-preserving and append-only audit log architectures for sensitive operations, showing that recording comprehensive system events with strict append-only constraints and cryptographic binding ensures robust non-repudiation and forensic reliability. Schulz and Weber (2022) analyzed forensic readiness in web architectures, concluding that capturing detailed contextual telemetry (including user identifiers, client IP addresses, timestamps, modified fields, and action types) is indispensable for investigating security anomalies and verifying regulatory compliance.

From a defensive coding standpoint, Ngo and Nguyen (2023) evaluated web application vulnerabilities, demonstrating that utilizing PHP Data Objects (PDO) with parameterized queries and strict input sanitization completely neutralizes SQL injection vulnerabilities. Furthermore, enforcing robust session management and data sanitization mitigates cross-site scripting (XSS) and ensures that transactional databases remain shielded from malicious injection attempts.

Software Quality and Security Evaluation Frameworks

Finally, evaluating the resulting software application requires recognized, multi-dimensional assessment frameworks that encompass both technical performance and end-user adoption. The International Organization for Standardization (2023) provides the ISO/IEC 25010:2023 software product quality model, offering structured criteria to evaluate functional suitability, performance efficiency, interaction capability, reliability, security, and maintainability (Anisah et al., 2023). In addition, selected technical controls from ISO/IEC 27001:2022 Annex A provide an authoritative benchmark for reviewing access governance, cryptographic protection, and audit logging integrity (International Organization for Standardization, 2022). Complementing these formal engineering standards, the TAM serves as an established psychometric framework to evaluate practical user adoption across property administrators, on-site personnel, and residents (Alyoussef, 2022). Measuring perceived usefulness, perceived ease of use, user trust, and behavioral intention to adopt ensures that technical rigor translates directly into human usability and operational success in daily property management.

SYNTHESIS OF LITERATURE AND STUDIES

The reviewed literature and empirical studies establish both the conceptual basis and practical considerations for digitalizing and securing residential property operations. The literature suggests that organizing property workflows through a centralized digital platform provides a structured data source

for tenant profiles, lease contracts, utility sub-metering records, and payment schedules, helping reduce duplicate records and administrative delays in multi-unit properties. These administrative challenges were highlighted by Chauhan (2024), who observed that fragmented record-keeping and uncoordinated files can weaken staff communication and lead to resident dissatisfaction. In the Philippine context, Villasin and Encarnacion (2024) documented that 45.5% of surveyed real estate practitioners encounter operational difficulties when relying on manual procedures, with 90.9% indicating a preference for web-based property management systems. These findings are complemented by Caparos et al. (2025) and Magno et al. (2024), who noted that automated billing calculations, digital receipts, and centralized communication interfaces can help streamline recurring rental transactions. Broadening this perspective, Oyetunji et al. (2025) reported that structured digital tools in community housing support more consistent lease tracking and administrative coordination, while Salunkhe (2026) and Buradkar (2022) observed that web-based management platforms provide clearer transaction visibility and assist in mitigating billing misunderstandings.

In selecting appropriate software architectures for multi-user property systems, existing studies emphasize balancing system availability, maintainability, and deployment feasibility. Progressive Web Application (PWA) architecture offers an accessible approach to cross-platform usability, leveraging service workers and client-side caching to provide responsive, app-like interactions across standard desktop and mobile browsers (Roumeliotis &

Tselikas, 2022; Muawwal, 2024). Studies by Wagh et al. (2023) indicate that PWAs achieve usability and task-flow efficiency comparable to native applications while avoiding the maintenance overhead associated with separate mobile store releases. On the backend, Coda et al. (2024) observed that for small-to-medium systems, a modular monolithic architecture offers manageable deployment, clearer transaction boundaries, and lower architectural overhead compared to distributed microservice setups. To maintain data consistency within this structure, relational database normalization to Third Normal Form (3NF) serves as an established standard; as demonstrated by Akadal and Satman (2022), 3NF schema design systematically eliminates update anomalies, enforces referential integrity across relational entities, and supports reliable data retrieval. Complementing the backend, utility-first CSS frameworks like Tailwind CSS assist in maintaining visual layout consistency and responsive behavior across multi-role interfaces (Lehtonen, 2022).

To support tenant assistance without significantly expanding administrative burdens, studies indicate growing interest in localized artificial intelligence and event-driven communication protocols. Deploying quantized, open-source Large Language Models on local or edge computing environments offers a practical alternative to cloud-hosted AI APIs, helping reduce network latency, eliminate recurring third-party API costs, and keep sensitive operational data within local boundaries (Qu et al., 2024; Sharma & Gupta, 2023). Deng and Chen (2024) observed that localized language models can assist in preliminary ticket routing by analyzing conversational descriptions and extracting basic issue categories. In

maintenance operations, Truss et al. (2024) noted that automated classification assists staff in triaging requests more systematically, while Pandya and Holia (2023) demonstrated that on-premise language models provide responsive, privacy-preserving customer support. Adam et al. (2022) pointed out that user acceptance of conversational interfaces largely depends on response clarity, consistency, and perceived privacy. Within the Philippine setting, tenant interactions often involve linguistic code-switching. Recent benchmark evaluations by Ramos et al. (2026) and Cinco and Devaraj (2026) indicate that standard LLMs tend to exhibit lower accuracy when parsing colloquial Filipino and mixed Taglish phrasing compared to formal English. Their observations, along with insights from Cruz and Cheng (2022) and Cosme and De Leon (2024), suggest that localized prompt design and domain context are useful in improving language model comprehension for informal service requests. For synchronous communication, Łasocha and Badurowicz (2022) and Obonguko (2024) showed that WebSocket protocols consume less network and server overhead than continuous HTTP polling, providing a suitable foundation for real-time messaging and operational alerts.

Processing personal tenant records, payment proofs, and tenancy agreements also requires careful alignment with information security standards and data privacy regulations. In the Philippines, the National Privacy Commission (2022), through NPC Circular No. 2022-01, sets forth compliance standards and administrative penalties under Republic Act No. 10173 (Data Privacy Act of 2012), obligating entities handling personal information to enforce reasonable

technical and organizational safeguards. These requirements were reinforced by the National Privacy Commission (2023) in NPC Circular No. 2023-06, which mandates robust data encryption, role-governed access boundaries, and periodic security evaluations. Gurses and Del Alamo (2022) emphasized that incorporating privacy-by-design principles early in system architecture helps establish clearer data boundaries. At the application layer, Ri et al. (2022) demonstrated that fine-grained Role-Based Access Control (RBAC) based on least privilege helps restrict unauthorized resource access across distinct user tiers, directly addressing critical vulnerabilities highlighted in the OWASP Top 10 API Security Risks (OWASP Foundation, 2023). For document protection, Saha and Das (2023) and Chen (2024) noted that authenticated AES-256-GCM encryption provides data confidentiality alongside built-in integrity tags, while Bhattacharya and Roy (2023) demonstrated that SHA-256 cryptographic hashing serves as a dependable method for detecting file modifications. Furthermore, maintaining append-only audit logs with contextual user telemetry supports operational traceability and assists in administrative reviews (Thazhath et al., 2022; Schulz & Weber, 2022), while parameterized database queries through PDO help protect against common SQL injection vulnerabilities (Ngo & Nguyen, 2023).

Finally, evaluating the resulting software application requires recognized evaluation frameworks. The International Organization for Standardization (2023) provides the ISO/IEC 25010:2023 software product quality model, offering structured criteria to assess functional suitability, performance efficiency,

interaction capability, reliability, security, and maintainability (Anisah et al., 2023). In addition, selected technical controls from ISO/IEC 27001:2022 Annex A offer a recognized reference for reviewing access control, authentication, audit logging, and cryptographic practices (International Organization for Standardization, 2022). Complementing these formal engineering standards, the TAM serves as an established psychometric framework to evaluate practical user adoption across property administrators, on-site personnel, and residents, measuring perceived usefulness, perceived ease of use, user trust, and behavioral intention to adopt (Alyoussef, 2022).

While the reviewed literature discusses individual components, such as basic property billing tools, progressive web apps, or cloud-based customer support, there remains a notable research gap in addressing the specific operational trilemma of local mid-rise residential facilities in the Philippines: handling sensitive personally identifiable information (PII) under strict RA 10173 regulations, operating within a constrained small-building budget, and functioning with no dedicated in-house IT or cybersecurity staff. Existing commercial platforms that provide advanced automation and security are priced for large corporations and rely on public cloud AI APIs that introduce third-party data residency risks, while local rental workarounds remain manual, unencrypted, and vulnerable to data exposure. Estatix PMS directly addresses this practical gap. By integrating a modular monolithic PHP/MySQL backend, a responsive PWA interface, an on-premise quantized Qwen2.5 7B model capable of interpreting colloquial Taglish service inquiries, RBAC authorization, AES-256-GCM

document encryption, SHA-256 integrity verification, and append-only audit logging, the proposed system provides an accessible, privacy-conscious, and maintainable platform suited to the operational and budgetary realities of Torre De Manuel Residences 1.

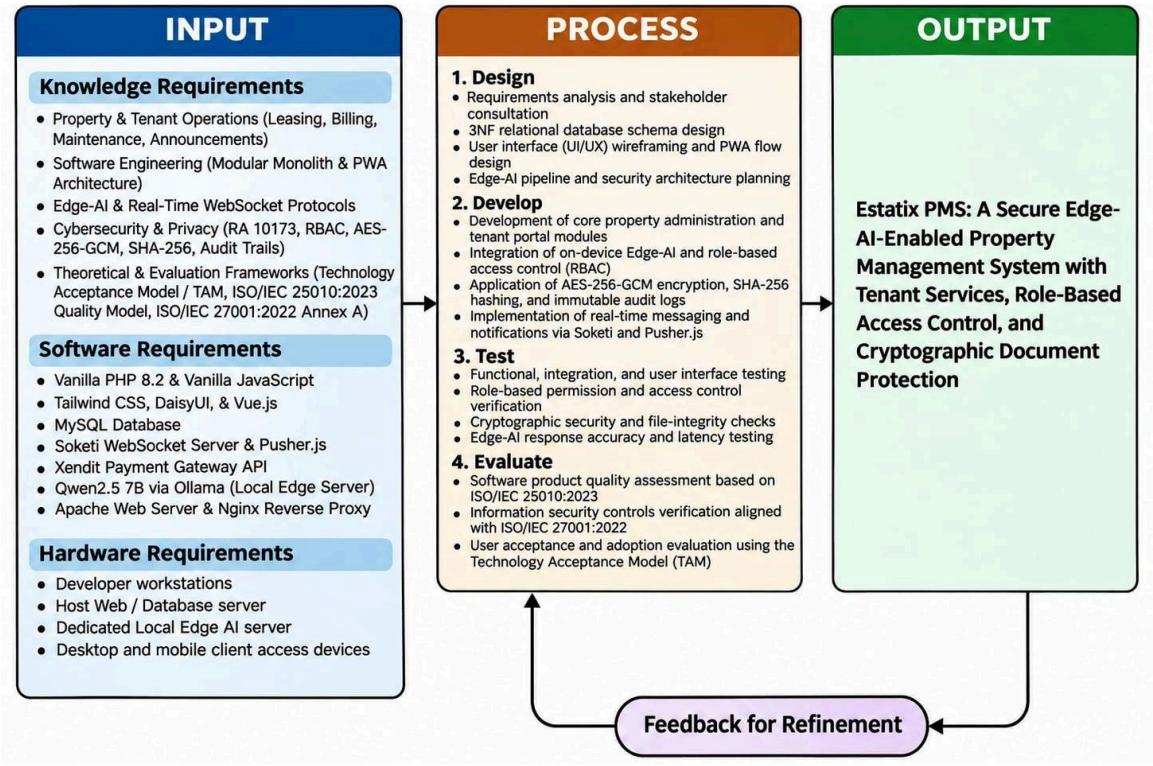
CONCEPTUAL MODEL OF THE STUDY

The study operationalizes the software engineering and evaluation lifecycle through the Input-Process-Output (IPO) conceptual model with an iterative feedback mechanism. The IPO model illustrates how domain knowledge, evaluation frameworks, software technologies, and hardware resources serve as inputs that undergo structured design, development, testing, and multi-tier evaluation phases to produce the proposed system output, with an empirical feedback loop ensuring continuous system refinement based on user and expert evaluation.

To guide the technical development and evaluation of the system, the conceptual model of the study is presented in Figure 2.1.1, illustrating the Input-Process-Output (IPO) framework with an iterative feedback mechanism.

Figure 2.1.1

Conceptual Model of Estatix PMS Using the Input–Process–Output Framework with Feedback Mechanisms



As illustrated in Figure 2.1.1, the conceptual framework outlines the knowledge, software, and hardware resources to be utilized in the design, development, testing, and evaluation of the proposed system. Furthermore, it emphasizes the developmental and assessment activities to be carried out across each phase and illustrates how empirical feedback from the output loops back to the process to refine and optimize the final system.

INPUT

The Input component represents three mandatory categories of resources required to build and validate the system. Under Knowledge Requirements, the study will integrate workflows for property and tenant administration, leasing protocols, billing and utility computation, principles of Progressive Web Applications (PWA), modular monolithic architecture, Edge-AI classification algorithms, and real-time communication protocols. This area also incorporates cybersecurity and regulatory frameworks, including Republic Act No. 10173 (Data Privacy Act of 2012), Role-Based Access Control (RBAC), AES-256-GCM authenticated encryption, SHA-256 file-integrity hashing, and append-only audit logging. Crucially, foundational evaluation frameworks are established as inputs, including the constructs of the TAM covering perceived usefulness, perceived ease of use, user trust, and behavioral adoption intention, alongside the software quality characteristics of ISO/IEC 25010:2023 and the information security controls of ISO/IEC 27001:2022 Annex A.

Complementing these domain principles, Software Requirements encompass the operational technology stack, which includes Vanilla PHP 8.2, Vanilla JavaScript, Tailwind CSS, DaisyUI, Vue.js, the Soketi WebSocket server, MySQL relational database, the Xendit Payment Gateway API, a local Qwen2.5 7B model executed via Ollama on a dedicated edge server, an Apache HTTP server, an Nginx reverse proxy, and modern web browsers. Finally, Hardware Requirements comprise developer workstations, web and database hosting server infrastructure, a dedicated on-premise local Edge-AI server, and various desktop,

tablet, and mobile client access devices necessary to develop, deploy, and conduct multi-device responsiveness testing.

PROCESS

The Process component details the engineering and assessment activities to be executed across four distinct sequential stages directly aligned with the specific objectives of the study. In the Design stage, the proponents will undertake field data collection and stakeholder consultation, database schema normalization up to Third Normal Form (3NF), responsive user interface (UI/UX) wireframing and PWA flow design, and comprehensive architectural planning for both the Edge-AI pipeline and cryptographic subsystems. During the Develop stage, the team will construct the core property administration and tenant portal modules, integrate on-device Edge-AI and role-based access control (RBAC), apply AES-256-GCM document encryption, SHA-256 hash verification, and server-side append-only audit logs, and implement real-time messaging and notifications via Soketi and Pusher.js. In the Test stage, the proponents will conduct functional, integration, and user interface testing, role-based permission and access control verification, cryptographic security and file-integrity checks, and Edge-AI classification accuracy and latency tests.

Finally, in the Evaluate stage, the system will be evaluated through a structured, multi-tier assessment. Technical software product quality will be evaluated by IT experts based on selected characteristics of ISO/IEC 25010:2023 (functional suitability, performance efficiency, interaction capability, reliability, security, and

maintainability) alongside information security controls verification aligned with ISO/IEC 27001:2022 Annex A. Concurrently, user acceptance and practical adoption will be evaluated among property administrators, on-site staff, and tenants using the TAM, measuring perceived usefulness, perceived ease of use, user trust, and behavioral intention to adopt the system. Diagnostic results and qualitative feedback derived from both testing and evaluation serve as an iterative feedback loop to resolve defects, refine database queries, tune model parameters, and optimize user experience prior to final deployment.

OUTPUT

The expected Output is the finalized, tested, and validated software artifact: Estatix PMS: A Secure Edge-AI-Enabled Property Management System with Tenant Services, Role-Based Access Control, and Cryptographic Document Protection. The Input component supplies the foundational domain expertise, quality standards, and technological resources. The Process component systematically transforms these inputs through structured software engineering, rigorous verification, and standards-based evaluation. Working cohesively, these components are designed to deliver a secure, operationally viable, and user-accepted property management platform tailored specifically to the operational realities of Torre De Manuel Residences 1.

Chapter III

RESEARCH METHODOLOGY

This chapter presents the research methodology, system design, technical architecture, developmental procedures, multi-tier testing protocols, and evaluation frameworks for Estatix PMS. It details the step-by-step procedures that will be conducted to evaluate software product quality, information security posture, and end-user technology acceptance.

Research Design

This study employs a quantitative, descriptive research design for the development, testing, and evaluation of Estatix PMS. Quantitative research focuses on measurable variables, using structured instruments and computational techniques to collect and analyze numerical data (Creswell & Creswell, 2018). A descriptive approach then presents the results as they are, such as system performance, user ratings, and compliance scores, without manipulating any variables. Together, they allow the study to objectively document how well Estatix PMS meets the operational requirements of Torre De Manuel Residences 1 (TDMR1), using numerical data from system benchmarks, functional tests, and evaluation questionnaires instead of subjective judgment.

The system will be evaluated along three primary dimensions: Software Product Quality, Information Security Posture, and User Acceptance and Adoption.

First, Software Product Quality will be evaluated using criteria adapted from the ISO/IEC 25010:2023 product quality model, specifically functional suitability, performance efficiency, interaction capability, reliability, security, and maintainability.

Second, Information Security Posture will be evaluated using selected technical controls adapted from ISO/IEC 27001:2022 Annex A, covering access control, identity management, secure authentication, append-only audit logging, application security, and cryptographic protection.

Third, User Acceptance and Adoption will be evaluated among building administrators, operational staff, and tenants using the TAM, measuring perceived usefulness, perceived ease of use, user trust, and behavioral intention to use the system.

Objective numerical data gathered from system benchmark logs, functional testing runs, and structured 4-point Likert evaluation instruments will be analyzed using descriptive statistics to determine overall system performance, security compliance, and user acceptance.

Sampling Technique

This study will employ a combination of purposive sampling and simple random sampling to select participants across alpha testing, technical evaluation, and beta operational evaluation.

Purposive sampling will be utilized for the technical evaluators and administrative personnel. A dedicated panel of external IT specialists will be purposively selected based on verified professional expertise in software engineering, cloud architecture, and cybersecurity. Their technical background qualifies them to rigorously evaluate system functionality, performance efficiency, interaction capability, reliability, maintainability, and security controls (including RBAC, append-only audit logging, AES-256-GCM encryption, and SHA-256 integrity verification) under the **ISO/IEC 25010:2023** and **ISO/IEC 27001:2022 Annex A** frameworks. In addition, the property owner, building administrator, and on-site maintenance staff of Torre De Manuel Residences 1 (TDMR1) will be purposively included as administrative evaluators due to their direct oversight of daily leasing, utility billing, and maintenance operations.

Simple random sampling will be employed for the beta testers comprising the residential tenants of TDMR1. Because the active tenant population represents a relatively homogeneous group with equal exposure to residential services, simple random sampling guarantees every tenant an equal probability of selection through a randomized selection method applied to the active tenant directory. This sampling technique mitigates selection bias and maintains population representativeness.

Together, these sampling techniques ensure comprehensive evaluation coverage: technical specialists will evaluate architectural quality and security posture, while randomly sampled tenants and administrative staff will evaluate user acceptance, perceived usefulness, perceived ease of use, and adoption intentions through the TAM.

Respondents of the Study

According to administrative records from Torre De Manuel Residences 1 (TDMR1), the property currently has an active population of 102 individuals, comprising 96 residential tenants, 5 shift staff, and 1 building administrator. Using Slovin's formula at a 90% confidence level with a 10% margin of error:

$$n = \frac{N}{1 + N(e)^2} = \frac{102}{1 + 102(0.10)^2} = \frac{102}{2.02} \approx 50.50$$

The computed sample size is rounded up to 51 internal respondents representing the property. This group serves as the beta testing cohort for end-user operational testing and Technology Acceptance Model (TAM) evaluation, comprising 46 randomly selected tenants and 5 purposively selected on-site administrative and operational personnel (1 building administrator and 4 staff).

In addition to the internal user sample, an independent panel of 5 external IT specialists with verified industry experience in software engineering and cybersecurity will be purposively selected to serve as alpha testers, evaluating the system under the ISO/IEC 25010:2023 and ISO/IEC 27001:2022 Annex A frameworks. In total, the study will engage 56 evaluators (5 technical alpha testers and 51 operational beta testers). Participation will be strictly voluntary, and written informed consent will be secured from all participants prior to evaluation.

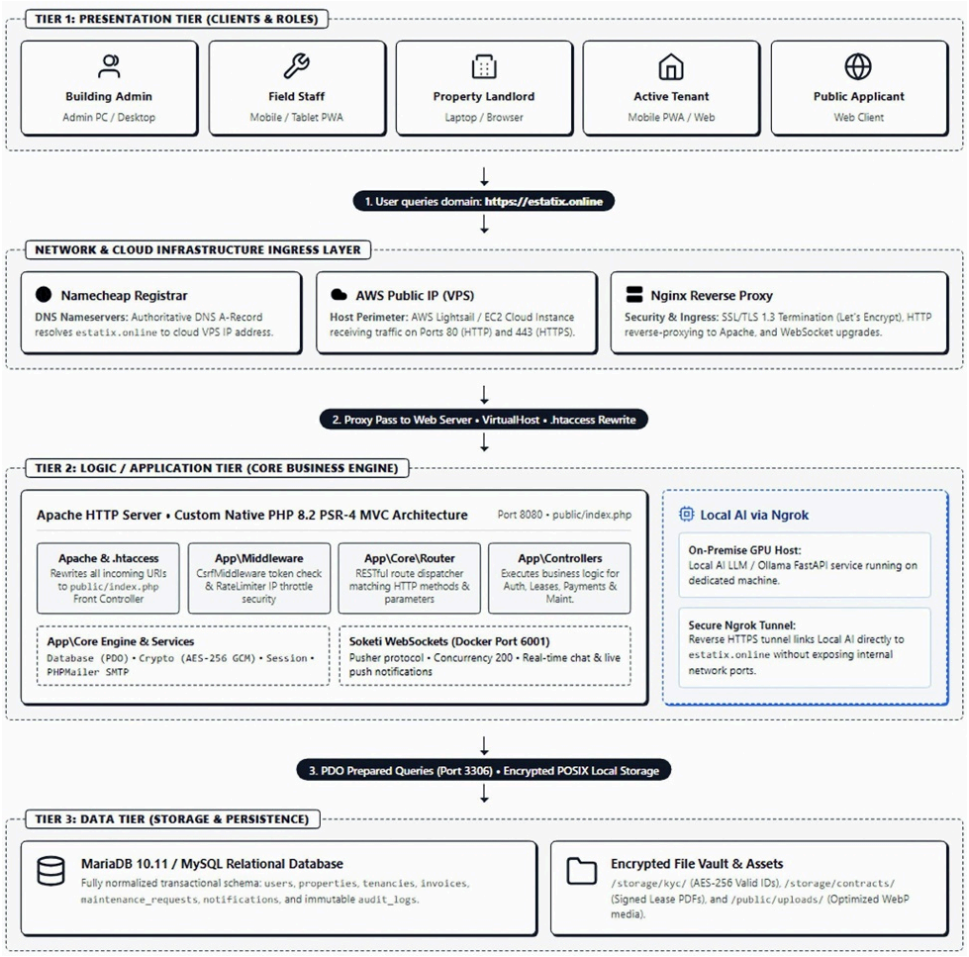
Project Design

Estatix PMS' architectural design features a comprehensive three-level system architecture and an ingress network layer, ensuring a secure, scalable, and efficient property management system. Crafted using a native PHP 8.2 PSR-4 MVC system, they are designed to adhere to strict security measures such as CSRF middleware checks, IP rate limits, and AES-256 cryptographic document protection. Operating efficiency is realised in real-time with a Soketi WebSockets engine for instant communications, and an on-premise Local AI service is linked via a secure HTTPS tunnel. All of the operational architecture is charted out in seventeen detailed system workflows that guide each process, starting from user onboarding to lease termination. This integration of hardware and software components is designed to support reliable operation, real-time communication, and robust security controls, providing a modern, structured, and responsive system.

To illustrate the technical organization, structural tiers, and network deployment of the platform, the overall system architecture is presented in Figure 3.1.1, depicting the multi-layered interaction between client roles, ingress security, application logic, and data persistence.

Figure 3.1.1

System Architecture of Estatix PMS Using a Three-Tier Client-Server Framework with Cloud Ingress



As shown in Figure 3.1.1, the Estatix PMS System Architecture is structured as a multi-layered framework consisting of three main operational tiers integrated with a cloud infrastructure ingress layer. Tier 1 (Presentation Tier) serves client interactions through web browsers and Progressive Web Applications (PWAs) across desktop and mobile devices, accommodating five distinct user roles: Building Admin, Building Staff, Property Landlord, Tenant, and Applicant. When a client queries the domain, requests pass through the Network and Cloud Infrastructure Ingress Layer, where authoritative DNS resolves the domain to a public IP address, and an Nginx reverse proxy terminates SSL/TLS 1.3 encryption before forwarding traffic to the web server. In Tier 2 (Logic and Application Tier), the core business engine is executed by an Apache HTTP Server hosting a custom PHP 8.2 PSR-4 MVC framework.

This tier incorporates middleware for Cross-Site Request Forgery (CSRF) protection and request rate limiting, a Soketi WebSocket service for real-time notifications, and a secure tunnel interfacing with an on-premise local Edge-AI model. Communication between the application logic and persistence layer is conducted over Port 3306 using PDO prepared statements, supplemented by POSIX-compliant encrypted file system operations. Finally, Tier 3 (Data Tier) provides structured persistence through a normalized, relational MySQL database storing transactional records of users, tenancies, and append-only audit logs, while sensitive documents and media assets are safeguarded within an AES-256-GCM encrypted file storage vault.

System Flowchart Diagram

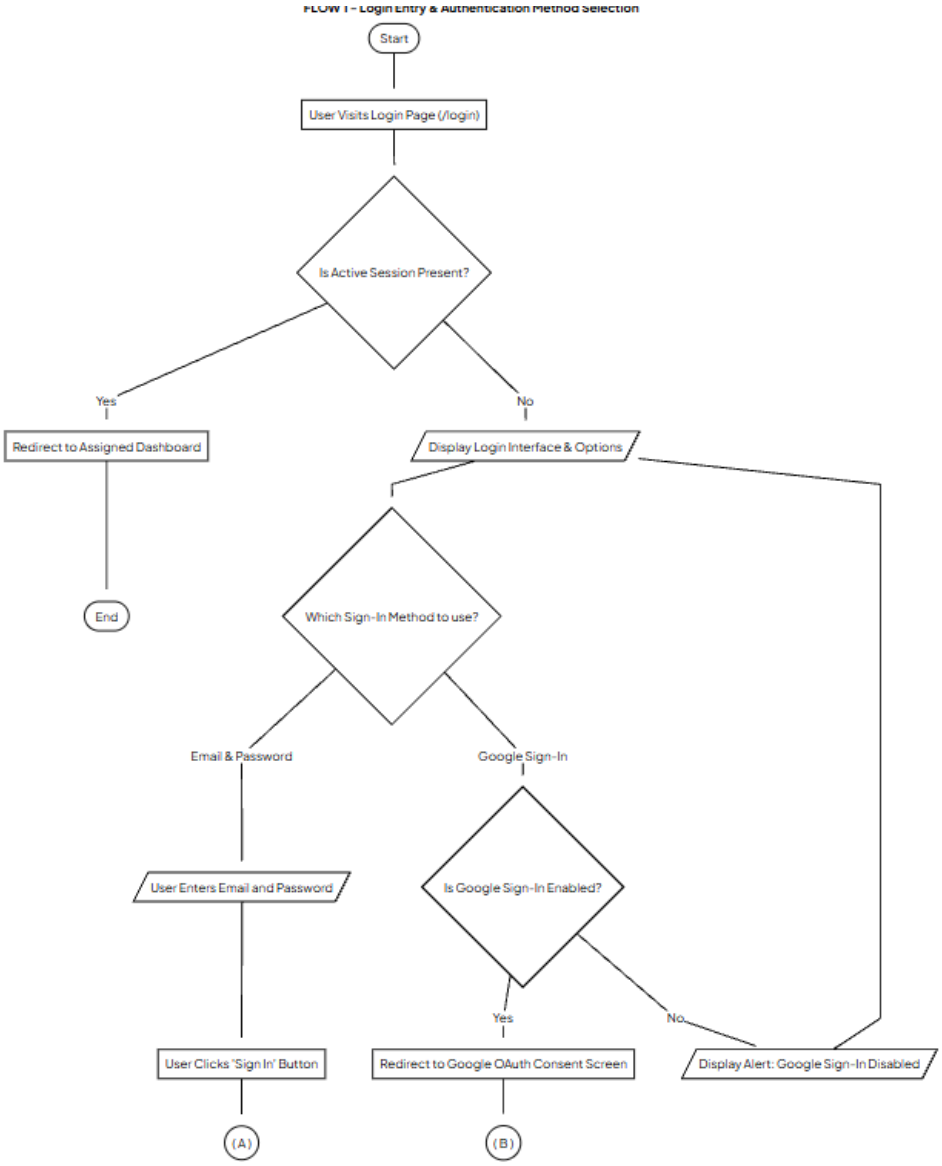
The system flowcharts illustrate the processes, activities, and decision-making logic planned for the operation of the Estatix PMS. Each flowchart provides a visual representation of the sequence of steps, user interactions, system validations, and expected outputs for specific system functionalities. These include the Login Flowchart, which presents user authentication; the Register Flowchart, which illustrates account creation; the Change Password Flowchart and Change Email Flowchart, which demonstrate credential management; the Forgot Password Flowchart, which illustrates account recovery; the Apply Viewing Schedule Flowchart, which shows the property viewing application process; the Lease Application Flowchart, which presents the submission and processing of tenancy agreements; the Maintenance Request Flowchart, which illustrates how tenants submit maintenance concerns; the Staff Tasks Flowchart, which demonstrates the resolution of assigned staff responsibilities; the Payments Flowchart, which presents billing and transaction processing; the Lease Termination Flowchart, which illustrates the procedure for ending a tenancy; the Security Deposit Refund Flowchart, which presents the processing of deposit refunds; the Session Governance Flowchart, which demonstrates session lifecycle and token validation; the Property Management Flowchart, which illustrates unit and property record maintenance; the Announcement Flowchart, which presents building-wide broadcasting; the Chat Support Flowchart, which demonstrates direct communication between tenants and management; and the Community

Forum Flowchart, which illustrates resident information sharing. Collectively, these flowcharts provide an architectural overview of the operational structure and explain how modules interact to support property management activities. The primary entry-point workflows are presented sequentially beginning with Figure 3.2.1, while the complete, detailed versions of all 17 system flowcharts are compiled in Appendix D.

The authentication process is designed as the primary security gateway of Estatix PMS to ensure that only authorized users with verified accounts can access building records and administrative services. As illustrated in Figure 3.2.1, the entry-point sequence will first evaluate whether an active authenticated session already exists upon accessing the sign-in interface. If an active session is detected, the system will route the user directly to their assigned role dashboard. If no active session exists, the interface will display the available sign-in options, allowing the user to authenticate using either registered email and password credentials or Google OAuth sign-in.

Figure 3.2.1

Login Entry and Authentication Method Selection Flow of Estatix PMS

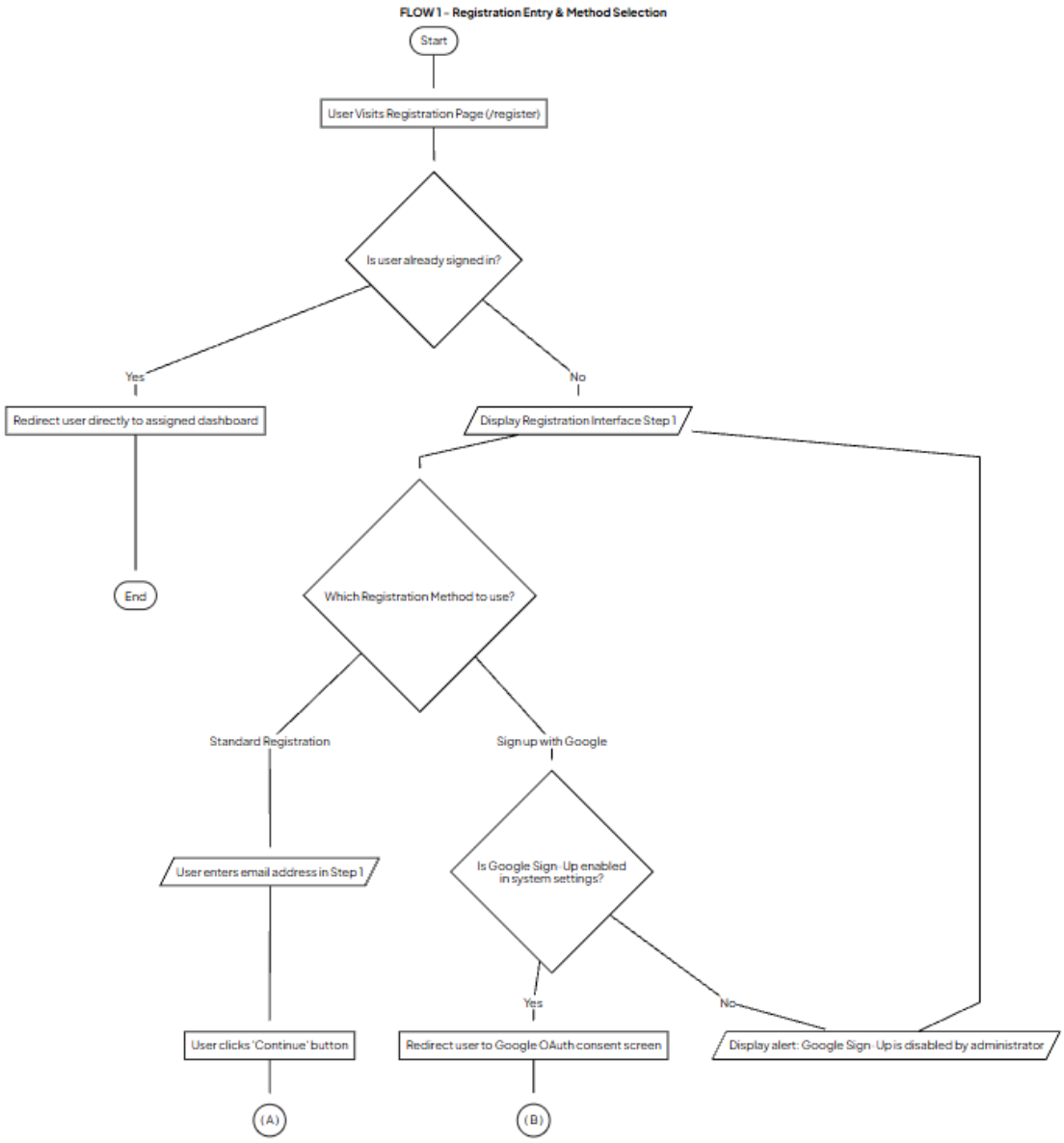


Note. As designed, this flowchart illustrates the entry point, active session verification, and authentication method selection for user access in Estatix PMS. Connector ***node (A)*** *leads to credential validation, password hash verification, and account lockout safeguards, while connector* ***node (B)*** *directs to external Google OAuth token verification and account binding. The complete login flow is detailed in* ***Appendix D under Flow 1: Login Flowchart.***

The registration process is designed as the onboarding gateway of Estatix PMS to allow prospective tenants and users to establish a verified account. As illustrated in Figure 3.2.2, the entry point sequence is planned to first evaluate whether an active authenticated session already exists upon accessing the registration screen. If an existing session is detected, the system is designed to route the user directly to their assigned role dashboard. If no active session exists, the interface will display the initial registration step, allowing the user to initiate account setup through either standard email registration or third-party Google OAuth sign-up.

Figure 3.2.2

User Registration Entry and Method Selection Flow of Estatix PMS

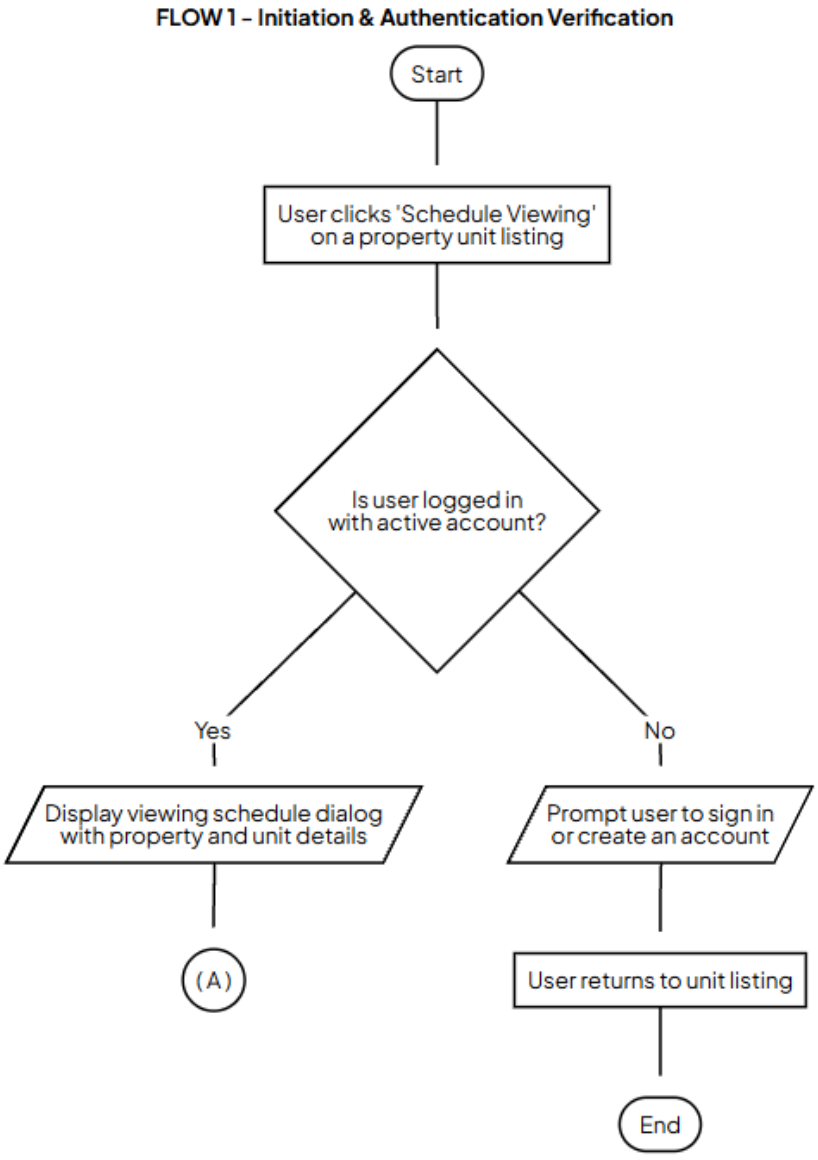


Note. As designed, this flowchart illustrates the entry point, session evaluation, and method selection for account registration in Estatix PMS. Connector ***node (A)*** *leads to the email verification and multi-step profile completion stages, while connector* ***node (B)*** *directs to external Google OAuth account validation. The complete registration flow is detailed in* ***Appendix D under Flow 2: Register Flowchart.***

The property viewing schedule process is designed as a core leasing facilitation feature of Estatix PMS, allowing prospective tenants to request and coordinate on-site or virtual property tours for available units. As illustrated in Figure 3.2.3, the entry point sequence is planned to initiate when an interested client selects the scheduling option on a specific unit listing. The system will first evaluate whether the user is authenticated with an active account. If the user is unauthenticated, the system is designed to prompt them to sign in or register before returning to the listing. If the user is actively logged in, the interface will display the viewing schedule dialog populated with the unit details, enabling the client to select their preferred date and time slot.

Figure 3.2.3

Property Viewing Schedule Initiation and Authentication Flow of Estatix PMS

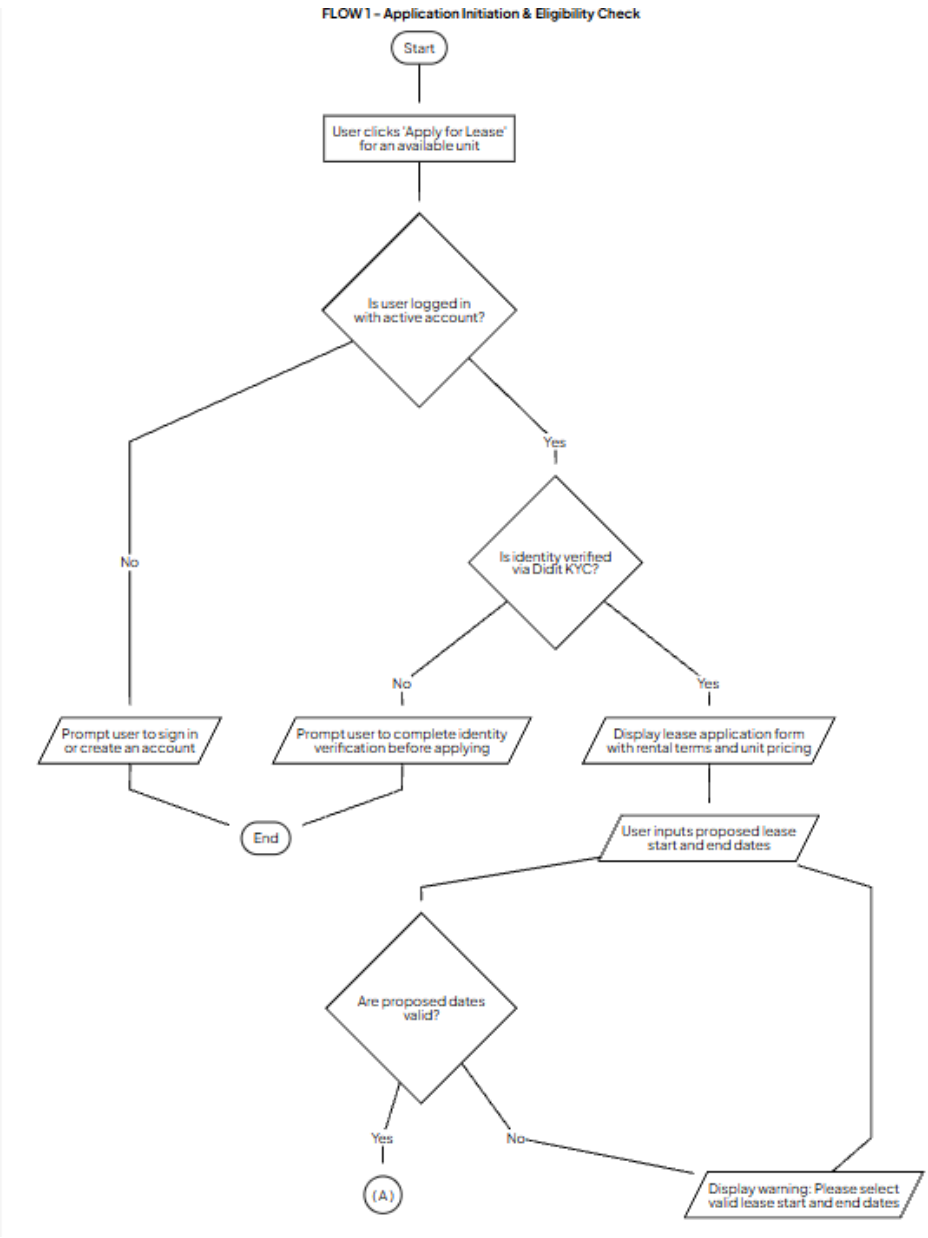


Note. As designed, this flowchart illustrates the entry point, user initiation, and account authentication verification for scheduling a property viewing in Estatix PMS. Connector ***node (A)*** *leads to the calendar date selection, dynamic slot availability calculation, and appointment reservation stages. The complete viewing schedule flow is detailed in* ***Appendix D under Flow 6: Apply Viewing Schedule Flowchart.***

The lease application process is designed as the primary contractual onboarding workflow of Estatix PMS, transitioning prospective clients into verified tenants. As illustrated in Figure 3.2.4, the entry point sequence is planned to initiate when an applicant selects the lease application option for a vacant unit. The system is designed to evaluate both user authentication and identity verification status through Didit KYC. If the client is unauthenticated or has not completed identity verification, the interface will prompt them to fulfill these prerequisites before proceeding. Once verified, the system displays the application form with unit pricing and rental terms, validating the user proposed lease start and end dates prior to submission.

Figure 3.2.4

Lease Application Initiation and Eligibility Verification Flow of Estatix PMS

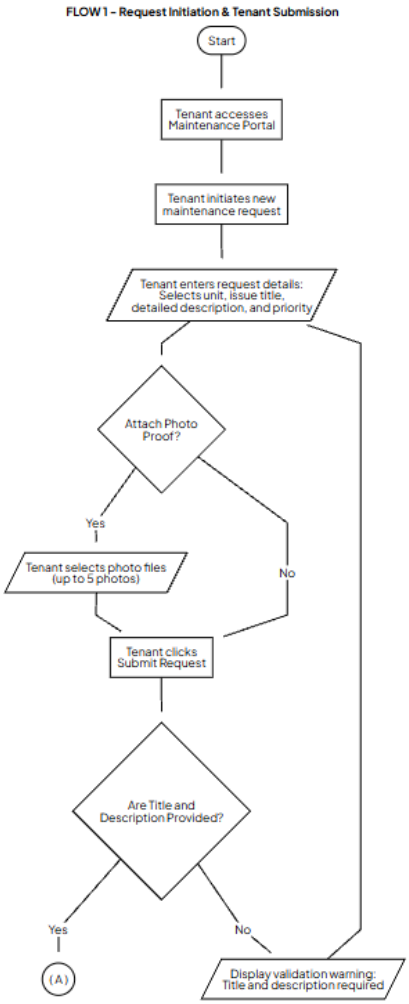


Note. As designed, this flowchart illustrates the entry point, account authentication check, identity verification guard, and proposed lease dates validation for initiating a tenancy application in Estatix PMS. Connector ***node (A)*** *leads to unit availability validation, duplicate submission safeguards, document upload, and administrative screening. The complete lease application flow is detailed in* ***Appendix D under Flow 7: Lease Application Flowchart.***

The maintenance request process is designed as the property upkeep and issue resolution module of Estatix PMS, providing tenants with a systematic channel to report unit defects and facility concerns. As illustrated in Figure 3.2.5, the entry point sequence is planned to begin when a tenant accesses the maintenance portal and initiates a service ticket. The tenant is prompted to specify their unit, select an issue category, define the priority level, and optionally upload up to five photographic proofs of the concern. The system is designed to validate that both a title and description are provided before permitting submission, displaying a prompt if required fields are missing.

Figure 3.2.5

Maintenance Request Initiation and Tenant Submission Flow of Estatix PMS

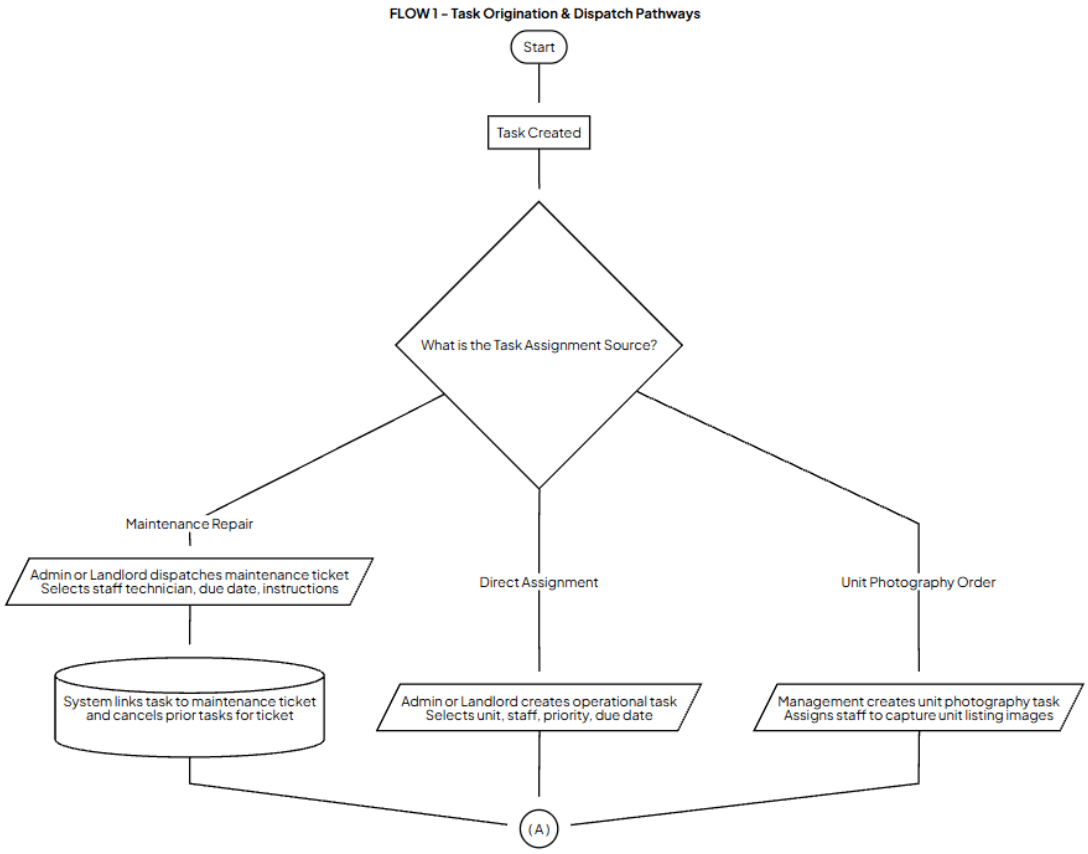


Note. As designed, this flowchart illustrates the entry point, issue specification, optional photographic attachment, and input validation for submitting a repair ticket in Estatix PMS. Connector ***node (A)*** *leads to ticket authorization checks, unique tracking code generation, and management dispatch to field staff. The complete maintenance request flow is detailed in* ***Appendix D under Flow 8: Maintenance Request Flowchart.***

The staff task management process is designed as the internal operational coordination module of Estatix PMS, enabling administrators and property managers to delegate and monitor building maintenance, unit preparations, and general staff responsibilities. As illustrated in Figure 3.2.6, the entry point sequence is planned to originate from one of three distinct administrative triggers: dispatching an approved maintenance repair ticket, assigning a direct operational duty, or issuing a unit photography order for property listings. The system will categorize the incoming assignment source, attach unit and technician specifications, and link existing maintenance records if applicable. Regardless of the originating pathway, all assignments converge to initiate task record generation and notification dispatch.

Figure 3.2.6

Staff Task Origination and Dispatch Pathways Flow of Estatix PMS

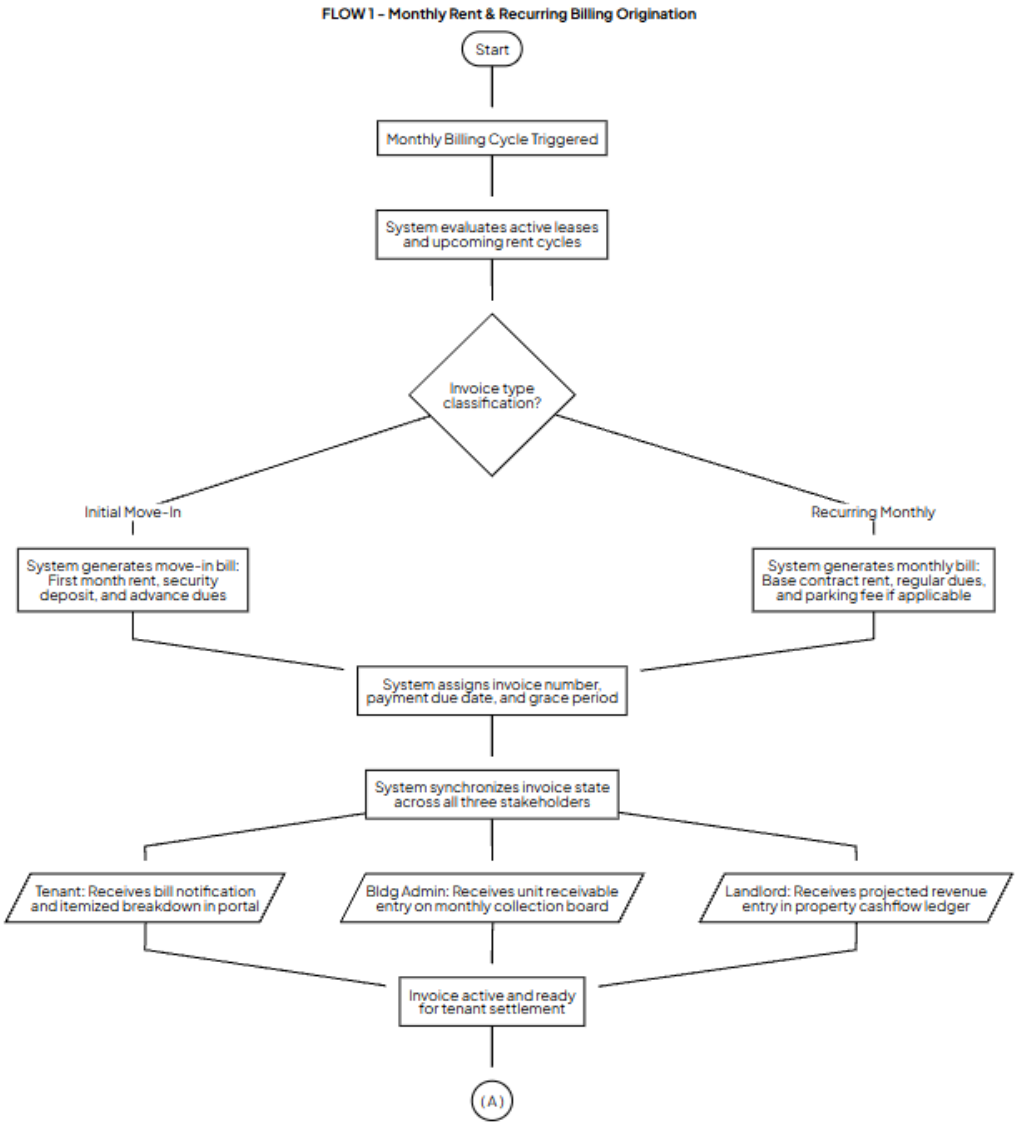


Note. As designed, this flowchart illustrates the entry point, assignment source evaluation, and dispatch pathways for building personnel duties in Estatix PMS. Connector ***node (A)*** *leads to task recording, unique task code generation, and automated notifications dispatched to assigned staff. The complete staff tasks flow is detailed in* ***Appendix D under Flow 9: Staff Tasks Flowchart.***

The payments and billing management process is designed as the primary financial engine of Estatix PMS, automating rent assessment, invoice generation, and ledger tracking across building administration, property owners, and tenants. As illustrated in Figure 3.2.7, the entry point sequence is planned to initiate through scheduled system triggers during each billing cycle, evaluating active lease agreements and upcoming payment deadlines. The system is designed to classify the transaction type, automatically generating either an initial move-in statement with security deposits or a recurring monthly invoice with base rent and regular dues. Upon establishing payment due dates and grace periods, the billing record is synchronized across all three stakeholders, notifying the tenant, updating administrative collection boards, and logging the landlord's financial ledger.

Figure 3.2.7

Monthly Rent and Recurring Billing Origination Flow of Estatix PMS

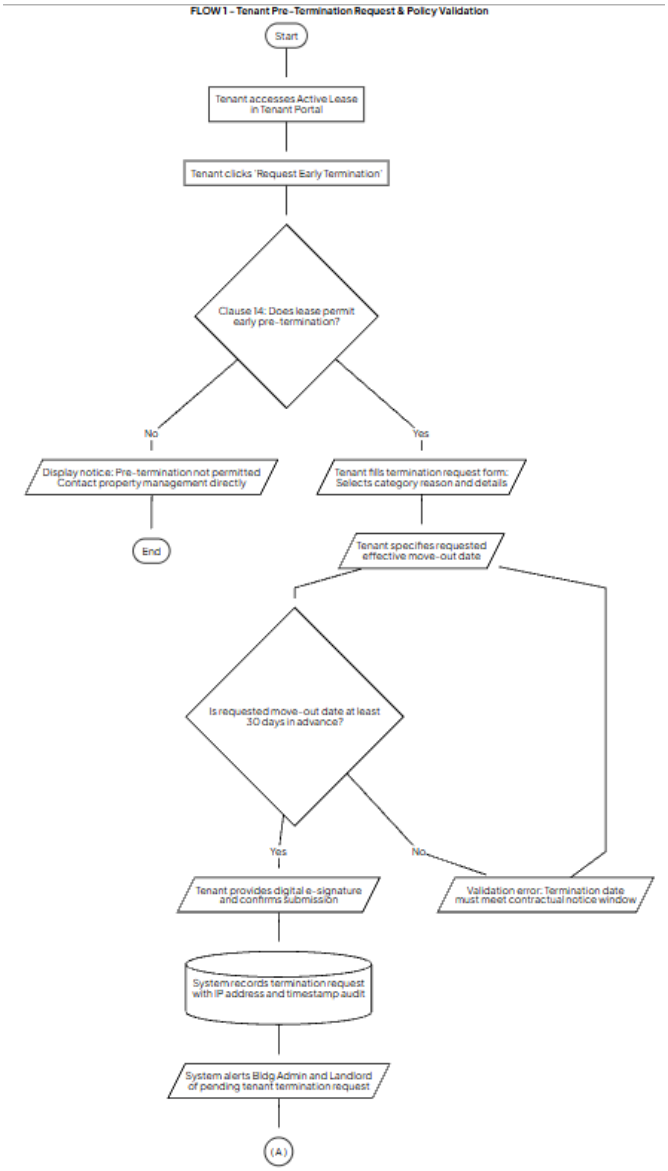


Note. As designed, this flowchart illustrates the automated billing cycle initiation,invoice classification, and tri-party financial synchronization in Estatix PMS. Connector ***node (A)*** *leads to invoice delivery, payment method selection through online payment gateways or post-dated check vaulting, overdue enforcement, and financial remittance reconciliation. The complete payments and financial operations flow is detailed in Appendix D under* ***Flow 10: Payments Flowchart.***

The lease termination process is designed as the tenant offboarding and contract conclusion framework of Estatix PMS, ensuring that early departures and scheduled expirations comply with legal terms and property policies. As illustrated in Figure 3.2.8, the entry point sequence is planned to initiate when an active tenant requests early pre-termination through the tenant portal. The system will evaluate the contract terms to verify whether early termination is permitted under the signed lease agreement, halting the request if a fixed-term lock applies. When permitted, the tenant is prompted to provide a termination reason and select an effective move-out date, with the system validating that a minimum 30-day advance notice requirement is satisfied before capturing a digital electronic signature and alerting property management.

Figure 3.2.8

Tenant Pre-Termination Request and Policy Validation Flow of Estatix PMS

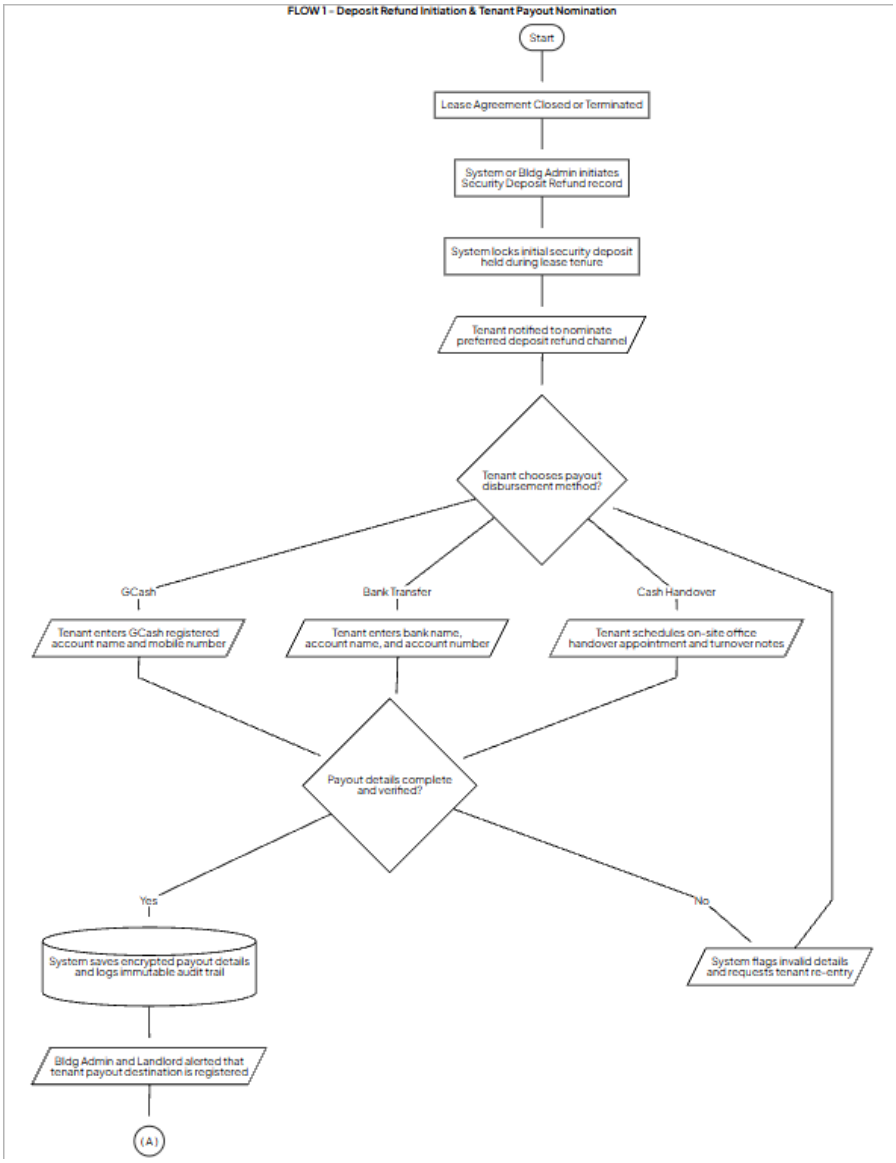


Note. As designed, this flowchart illustrates the entry point, contractual policy validation, minimum advance notice verification, and digital signature submission for early lease termination in Estatix PMS. Connector ***node (A)*** *leads to management review, outstanding balance audits, digital countersignature compilation, move-out inspections, and unit turnover. The complete lease termination flow is detailed in* ***Appendix D under Flow 11: Lease Termination Flowchart.***

The security deposit refund process is designed as the post-tenancy financial settlement module of Estatix PMS, ensuring the transparent reconciliation, deduction audit, and lawful return of tenant deposit funds. As illustrated in Figure 3.2.9, the entry point sequence is planned to trigger automatically upon the formal closure or termination of a lease agreement. The system will lock the baseline deposit amount held during the tenancy and prompt the departing tenant to nominate their preferred disbursement channel, offering choices between cash handover, bank transfer, or GCash. Once the payout information is validated and securely recorded into an append-only audit trail, the system alerts building administration and the landlord that the payout destination is established.

Figure 3.2.9

Security Deposit Refund Initiation and Tenant Payout Nomination Flow of Estatix PMS

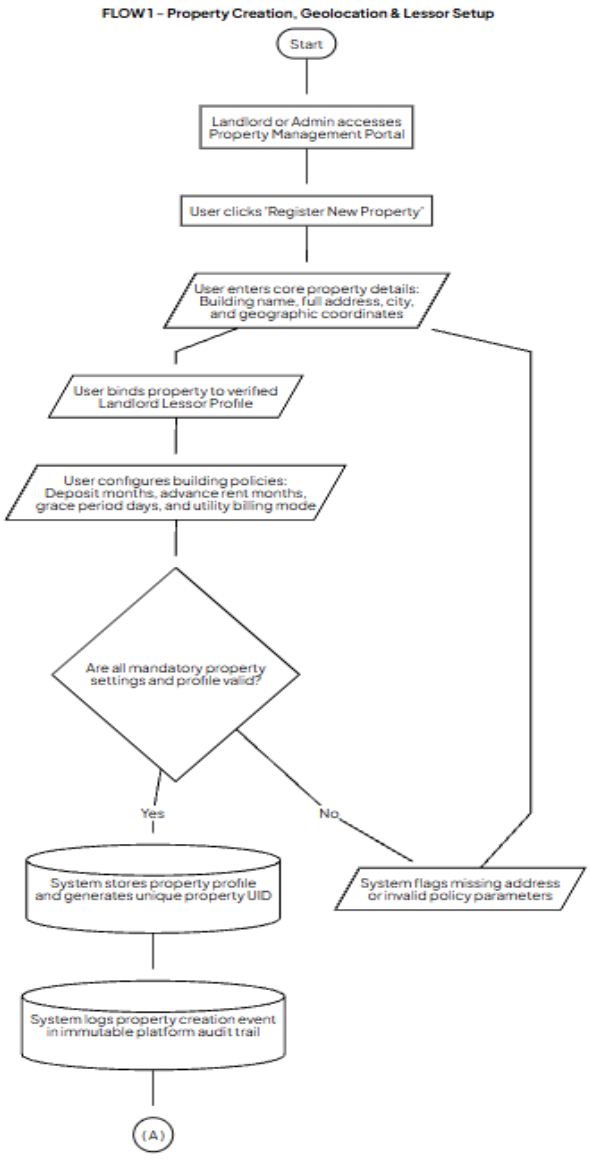


Note. As designed, this flowchart illustrates the automated trigger upon lease closure, baseline deposit locking, payout channel nomination, and encrypted credential validation in Estatix PMS. Connector ***node (A)*** *leads to move-out walkthrough inspections, itemized deduction audits for unpaid utilities or damages, and final fund disbursement or contractual forfeiture handling. The complete security deposit refund flow is detailed in* ***Appendix D under Flow 12: Security Deposit Refund Flowchart.***

The property management process is designed as the foundational asset administration module of Estatix PMS, structuring real estate inventories, physical building records, and leasing policies. As illustrated in Figure 3.2.10, the entry point sequence is planned to initiate when an administrator or landlord accesses the property portal to register a new building. The user inputs core property details, including the building name, address, and geographic coordinates, while binding the physical asset to a verified lessor profile. The system is designed to validate all mandatory policy parameters, including security deposit months, advance rent terms, and grace periods, before generating a unique property identification code and recording the transaction in the platform audit trail.

Figure 3.2.10

Property Creation, Geolocation, and Lessor Configuration Flow of Estatix PMS

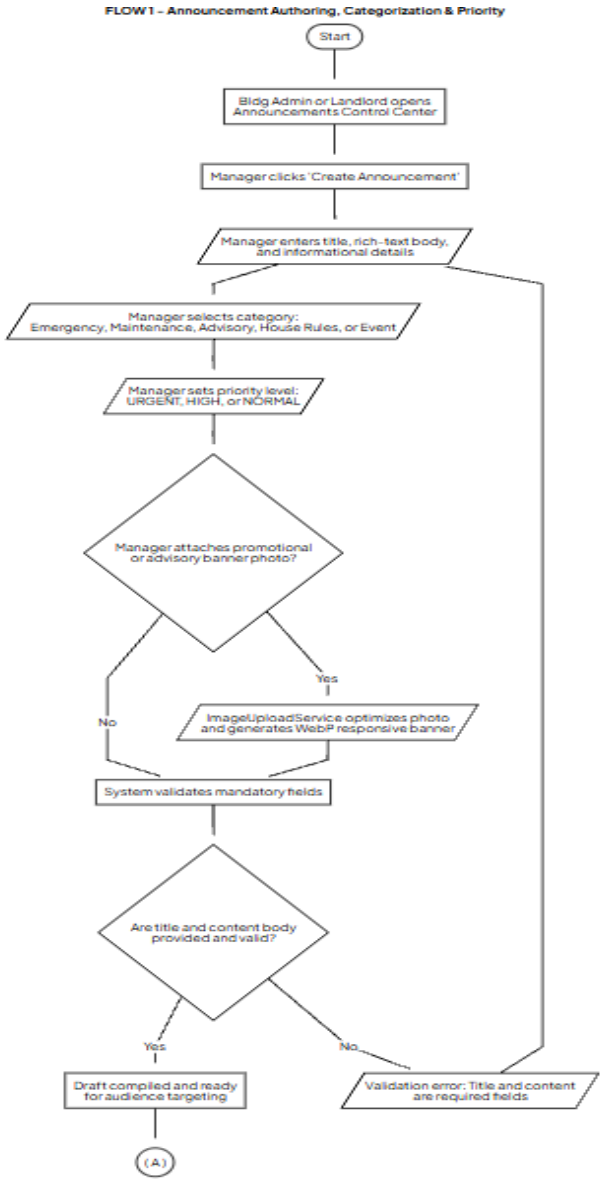


Note. As designed, this flowchart illustrates the entry point, building registration, geolocation setup, lessor profile binding, and policy parameter validation in Estatix PMS. Connector ***node (A)*** *leads to floor structure creation, unit inventory provisioning, unit interior photo uploads, and staff role assignment. The complete property management flow is detailed in* ***Appendix D under Flow 14: Property Management Flowchart.***

The announcements management process is designed as the property communication broadcast system of Estatix PMS, enabling building administrators and landlords to disseminate essential notices, emergency advisories, and facility updates to residents. As illustrated in Figure 3.2.11, the entry point sequence is planned to begin when management accesses the announcements control center to draft a new communication. The user authors the title and descriptive body text while selecting a relevant category and designating the priority tier as urgent, high, or normal. The system is designed to allow optional advisory banner photo attachments, verifying that all mandatory fields are completed before compiling the draft for audience targeting.

Figure 3.2.11

Announcement Authoring, Categorization, and Priority Assignment Flow of Estatix PMS

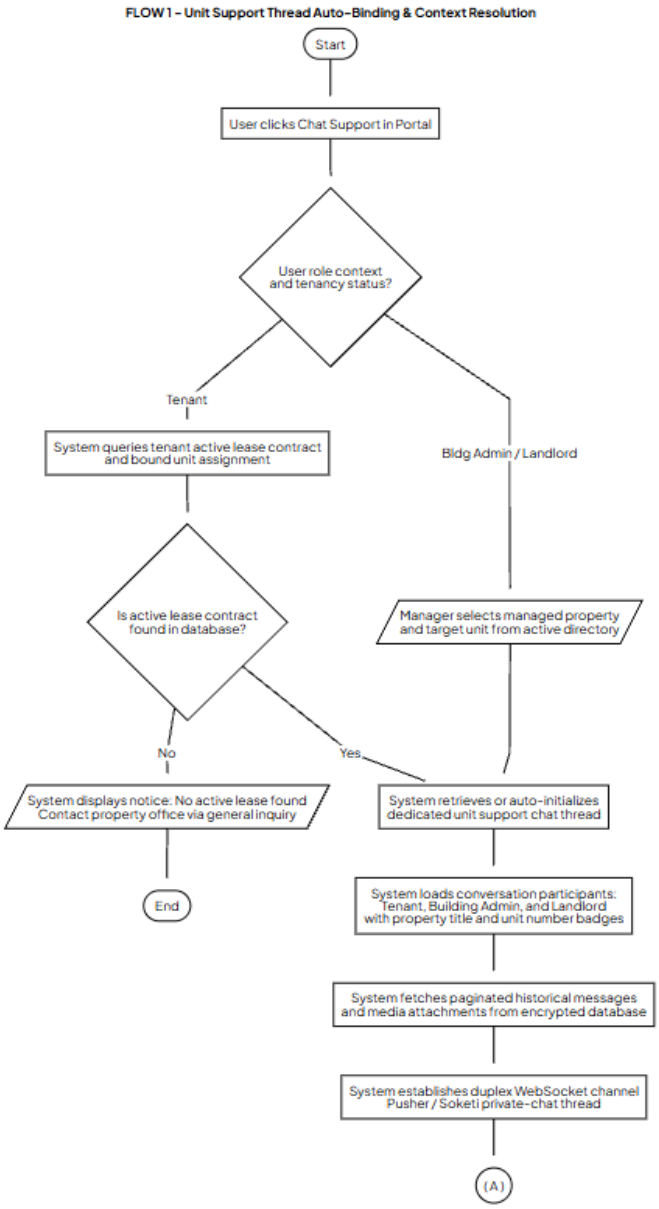


Note. As designed, this flowchart illustrates the entry point, content authoring, category classification, priority assignment, and input validation for property broadcasts in Estatix PMS. Connector ***node (A)*** *leads to audience demographic targeting, building-wide or floor-specific filtering, real-time broadcast dissemination, and mandatory acknowledgment tracking for urgent notices. The complete announcement flow is detailed in* ***Appendix D under Flow 15: Announcement Flowchart.***

The chat support process is designed as the real-time communication channel of Estatix PMS, facilitating responsive dialogue between tenants, building administrators, and property owners. As illustrated in Figure 3.2.12, the entry point sequence is planned to begin when a user accesses the chat interface from the portal. The system is designed to evaluate the user role and tenancy context, confirming active lease records for tenants or allowing administrators and landlords to select a target unit from their managed property inventory. If an active lease is verified, the system automatically binds the conversation thread to the unit, loads prior interaction history, and connects all parties to a private real-time communication channel.

Figure 3.2.12

Unit Support Thread Auto-Binding and Context Resolution Flow of Estatix PMS

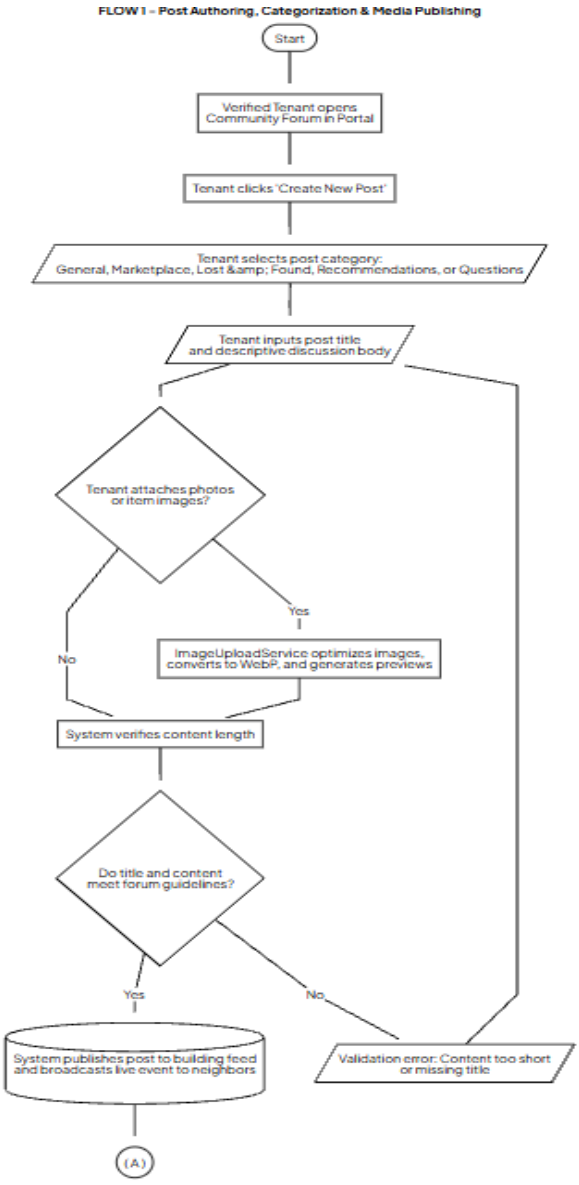


Note. As designed, this flowchart illustrates the entry point, user role evaluation, active lease verification, and unit thread auto-binding for in-app support communication in Estatix PMS. Connector ***node (A)*** *leads to real-time message transmission, photo attachment sharing, read receipt synchronization, and offline notification delivery. The complete chat support flow is detailed in* ***Appendix D under Flow 16: Chat Support Flowchart.***

The community forum process is designed as the resident social interaction and peer communication module of Estatix PMS, enabling verified tenants to participate in building discussions, marketplace exchanges, and community inquiries. As illustrated in Figure 3.2.13, the entry point sequence is planned to initiate when a tenant accesses the forum portal to create a new discussion thread. The user selects a relevant category, such as General, Marketplace, Lost and Found, Recommendations, or Questions, and inputs the title and descriptive body text. The system is designed to allow optional photo attachments, validating that the content complies with length requirements and community guidelines before publishing the post to the building feed and broadcasting the event to residents.

Figure 3.2.13

Community Forum Post Authoring, Categorization, and Publishing Flow of Estatix PMS



Note. As designed, this flowchart illustrates the entry point, discussion categorization, content authoring, optional photo attachment, and input validation for resident forum posts in Estatix PMS. Connector ***node (A)*** *leads to resident feed interactions, voting score tallies, threaded commenting, and administrative moderation controls such as pinning, locking, or content takedowns. The complete community forum flow is detailed in* ***Appendix D under Flow 17: Community Forum Flowchart.***

Requirements Gathering

1. Introduction

The requirements gathering phase will serve as the initial stage in developing Estatix PMS. It will focus on identifying, analyzing, and documenting the system requirements needed to address the property management challenges of TDMR1. The objective is to develop a secure and user-friendly Property Management System that would centralize property operations, improve tenant services, and protect sensitive records. The researchers will examine the existing procedures for managing tenant information, lease contracts, rental payments, utility bills, maintenance requests, and tenant concerns. These requirements will serve as the foundation for the system design, development, and evaluation.

2. Stakeholder Consultation

The requirements gathering process will focus on understanding the operational needs of TDMR1, including property management personnel and tenants. The researchers will identify challenges associated with maintaining paper-based records, spreadsheets, and communication messages for managing property-related transactions and tenant concerns. These processes may result in difficulties tracking payments, monitoring lease agreements, following up on maintenance requests, retrieving records, and controlling access to sensitive information.

The identified needs will guide the proposed features of Estatix PMS, including:

2.1. Centralized Property and Tenant Management: A centralized platform for maintaining tenant profiles, property information, lease records, and other relevant management data.

2.2. Lease and Payment Monitoring: Features for recording lease information, monitoring rental payments, tracking utility bills, and reviewing payment-related records.

2.3. Maintenance Request Management: A system for submitting, categorizing, tracking, and monitoring maintenance requests to support timely handling of tenant concerns.

2.4. Tenant-Administrator Communication: Communication features that allow tenants and administrators to exchange concerns and provide updates regarding property-related matters.

2.5. Announcements and Community: Features for broadcasting building advisories, emergency utility updates, and community notices, fostering transparent communication, resident engagement, and timely information dissemination between property administration and tenants.

2.6. Edge-AI-Assisted Tenant Services: Locally processed AI-assisted services for responding to routine tenant inquiries and helping classify and route maintenance requests. These services are intended to assist users rather than replace administrative judgment.

2.7. Role-Based Access Control (RBAC): Access restrictions based on assigned user roles to help prevent unauthorized users from accessing protected features and information.

2.8. Cryptographic Document Protection: AES-256-GCM encryption for applicable sensitive lease and billing documents, together with SHA-256 integrity verification to help detect unauthorized changes.

2.9. Audit Logging: Recording relevant login attempts, document access, record modifications, and administrative activities to support accountability and security monitoring.

3. Data Collection Methods

To establish the system specifications, the requirements gathering phase will utilize the following methods:

3.1. Review of Existing Property Management Procedures: Examining the existing processes for recording tenant information, managing lease contracts, tracking rental payments and utility bills, processing maintenance requests, and handling tenant concerns. This review will help identify areas where digitalization and centralized record management could improve operations.

3.2. Stakeholder Consultation: Gathering information about the operational needs of property management personnel and tenants, including their requirements for accessing records, monitoring

transactions, submitting maintenance concerns, and receiving announcements or updates. The researchers conducted key informant interviews with the building administrator and administered structured operational needs questionnaires to residing tenants.

3.3. Requirements Analysis: Organizing the identified needs into functional and non-functional requirements. Functional requirements specify the tasks that Estatix PMS must perform, while non-functional requirements describe the expected quality, security, reliability, and maintainability of the system.

3.4. Security Requirements Analysis: Identifying security measures needed to protect tenant information and sensitive documents. These include user authentication, RBAC, audit logging, AES-256-GCM document encryption, and SHA-256 document-integrity verification.

3.5. Technical Feasibility Assessment: Identifying the technologies and development environment suitable for implementing the proposed system, including Vanilla PHP, Vanilla JavaScript, DaisyUI, Tailwind CSS, MySQL, Soketi, Progressive Web Application (PWA) technologies, and a locally deployed Edge-AI model.

4. Functional Requirements

To address the operational requirements of Torre De Manuel Residences 1, the functional requirements of Estatix PMS are established and detailed in Table 3.3.1.

Table 3.3.1

Functional Requirements of Estatix PMS

Code	Functional Requirement	Description
FR-01	User Authentication and Account Access	Allows authorized users to authenticate before accessing protected system features and information.
FR-02	Role-Based Access Control	Restricts system functions and information according to assigned user roles and permissions.
FR-03	Tenant and Property Management	Allows authorized personnel to create, view, update, and manage tenant profiles and property records
FR-04	Lease Management	Supports recording and monitoring lease agreements, lease details, and relevant contract information.
FR-05	Payment and Utility Tracking	Records and monitors rental payments, water and electricity charges, and related payment information.
FR-06	Maintenance Request Management	Allows tenants to submit maintenance concerns and enables authorized personnel to review, categorize, track,

		and manage requests.
FR-07	Tenant-Administrator Communication	Provides communication features for exchanging tenant concerns and property-related updates between tenants and administrators.
FR-08	Announcements and Notifications	Supports publishing announcements and providing relevant updates to users through implemented communication features.
FR-09	Report Generation	Generates reports from available property, tenant, lease, payment, utility, and maintenance records to support management activities.
FR-10	Edge-AI-Assisted Tenant Services	Uses a locally deployed Edge-AI model to assist with routine tenant inquiries and maintenance request classification and routing.
FR-11	Cryptographic Document Protection	Encrypts applicable sensitive lease and billing documents using AES-256-GCM and verifies document integrity using SHA-256 hashing.
FR-12	Audit Logging and Activity Monitoring	Records relevant login attempts, document access, record modifications, and administrative activities for review and accountability.

5. Non Functional Requirements

The non-functional requirements define the expected quality attributes and operational constraints of Estatix PMS. These requirements guide the development of a system that is usable, reliable, secure, and maintainable.

The specific non-functional requirements and target quality benchmarks for Estatix PMS are summarized in Table 3.3.2.

Table 3.3.2

Non-Functional Requirements of Estatix PMS

Code	**Non-Functional Requirements**	**Description**
NFR-01	Security	Implements authentication, RBAC, audit logging, AES-256-GCM encryption for applicable sensitive documents, and SHA-256 integrity verification to help protect system information.
NFR-02	Reliability	Maintains consistent system operations and supports the accurate saving, retrieval, and updating of records during normal use.
NFR-03	Performance Efficiency	Aims to provide acceptable response times when accessing records, processing transactions, retrieving documents, and using major system functions.
NFR-04	Interaction Capability	Provides a clear and understandable interface that

		supports navigation and interaction for authorized administrators and tenants.
NFR-05	Maintainability	Uses an organized, modular code structure to support troubleshooting, corrections, updates, and future improvements.
NFR-06	Edge-AI Service Quality	Provides locally processed AI-assisted responses and maintenance classifications that are relevant to the submitted tenant inquiries and requests.
NFR-07	Platform Accessibility	Supports access through compatible desktop and mobile browsers using the Progressive Web Application approach. Internet connectivity is required for the system's intended online operations.

6. Tools, Technologies, and Environment

Table 3.3.3 presents the principal technologies identified for the development and implementation of Estatix PMS.

Table 3.3.3

Tools, Technologies, and Environment for Estatix PMS

Category	**Tool / Technology**	**Purpose**
Backend Development	Vanilla PHP	Implements server-side processing, business logic, database operations, and system functions.

Frontend Development	Vanilla JavaScript	Supports client-side interactions, interface behavior, and communication with system functions.
User Interface Frameworks	DaisyUI, Tailwind CSS and Vue.js	Support responsive layouts and the presentation of interface components.
Database Management	MySQL	Stores and manages tenant information, property records, leases, payments, utilities, maintenance requests, and other system data.
Payment Gateway	Xendit API	Generates digital invoices, processes secure online payments, and automatically updates tenant billing records in real time via webhooks.
Real-Time Communication	Soketi	Supports implemented real-time communication and update features.
Web Application Approach	Progressive Web Application (PWA)	Supports access through compatible browsers on desktop and mobile devices.
Edge-AI Technology	Quantized Qwen2.5: 7B model deployed on a dedicated edge server	Supports local inference for routine tenant inquiries and maintenance request classification and routing assistance.
Document Encryption	AES-256-GCM	Provides confidentiality and authentication protection for applicable sensitive documents.
Document Integrity Verification	SHA-256	Generates document hashes that can be compared to help detect changes to protected files
Access and Activity	Authentication, RBAC, and	Controls access to system functions and records

Security	audit logging	relevant user and administrative activities.
Development and Testing	Project development environment and compatible web browsers	Supports implementation, debugging, functional testing, and evaluation of the system.
Evaluation Frameworks	ISO/IEC 25010:2023, ISO/IEC 27001:2022 Annex A, and Technology Acceptance Model (TAM)	Guide the assessment of selected software quality characteristics, applicable information security controls, and end-user acceptance and adoption.

Project Development

In developing Estatix PMS, the researchers will employ the Agile Software Development Framework, specifically utilizing the Scrum methodology, to provide a flexible, iterative, and feedback-driven approach to system engineering. Rather than constructing the entire system in a single linear sequence, the Agile Scrum will allow the researchers to divide the complex development lifecycle into smaller, manageable, and time-boxed iterations known as Sprints. Within each Sprint, specific system requirements will be identified from the product backlog, designed, coded, integrated, tested, and reviewed continuously alongside the stakeholders of Torre De Manuel Residences 1 (TDMR1). To ensure clear accountability, streamlined task execution, and strict adherence to project timelines, the research team will observe defined Agile roles and responsibilities throughout the development lifecycle:

- **Product Owner Representative (Aileen Dela Torre):** Will serve as the primary stakeholder representative for TDMR1. This role will be

responsible for providing domain insights on property management workflows, defining operational priorities, validating user stories, and reviewing completed software increments during Sprint Reviews to ensure alignment with business needs.

- **Project Manager, System Architect, Lead Full-Stack Developer, UI/UX Designer, and QA Lead (Kivz M. Aragon):** Will spearhead the end-to-end technical execution, architectural modeling, and project management of Estatix PMS. This role will encompass Sprint planning, backlog management, designing all system workflows and architectural logic diagrams, UI/UX prototyping, frontend and backend development, database administration, cryptographic security integration, and comprehensive quality assurance and security testing across all iterations.
- **Research Writers and Documentation Specialists (Edrian Beronia, Jimboy A. Cajefe, and Rhenz P. Culaban):** Will focus primarily on authoring and structuring the academic research manuscript. This role will support the project lifecycle by writing the formal chapters, synthesizing related literature, administering survey instruments during user acceptance data collection, and compiling the written evaluation reports aligned with ISO/IEC 25010:2023 and ISO/IEC 27001:2022 Annex A security controls, and the TAM as the theoretical basis for assessing users' perceived usefulness and perceived ease of use of Estatix PMS.

The development of Estatix PMS will be executed across five (5)

structured Sprints, where each iteration will progress logically to deliver a functional, verifiable increment of the application:

- **Sprint 1: Planning, Requirements Analysis, and System Architecture Design:** Will focus on gathering operational requirements from the Product Owner Representative, establishing the product backlog, designing the database schema, and creating the foundational system architecture and UI/UX wireframes.
- **Sprint 2: Core Module Development and User Access Management:** Will center on building the foundational backend infrastructure, user authentication protocols, role-based access controls, and the primary management interfaces for system users and property records.
- **Sprint 3: Security Implementation and Document Management Integration:** Will focus on developing and integrating the core cryptographic security mechanisms, data protection controls, and secure document handling workflows required to safeguard sensitive organizational records.
- **Sprint 4: Feature Expansion, Workflow Automation, and Interface Refinement:** Will focus on constructing the remaining operational modules, automating routine business transactions, integrating user support features, and refining the responsiveness and usability of the user interface.

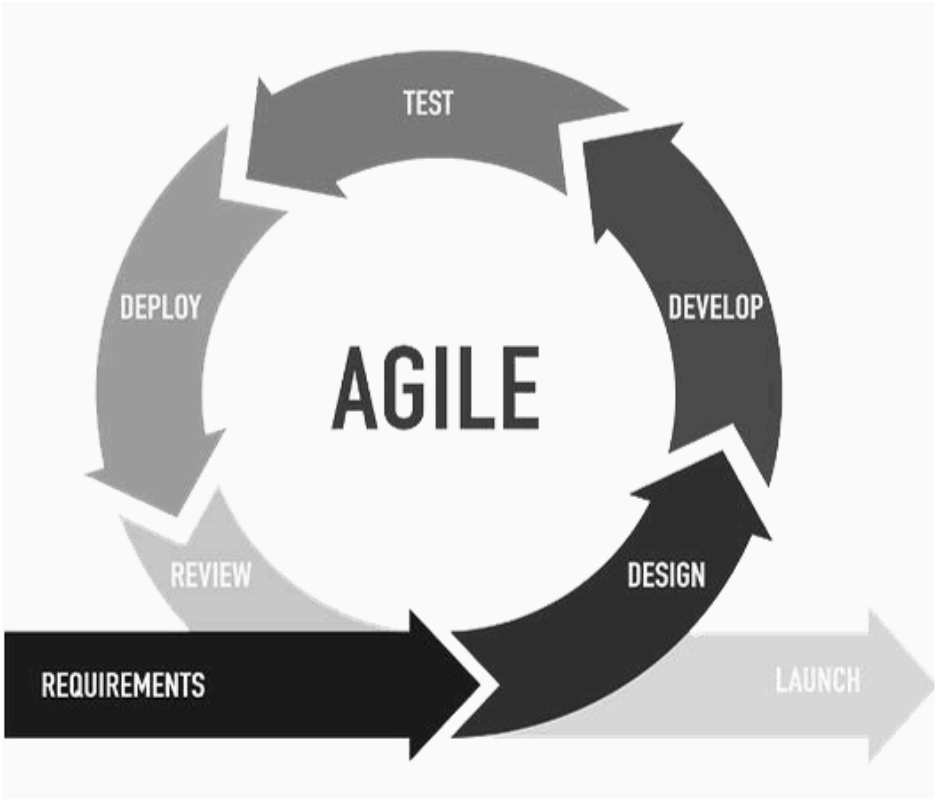
- **Sprint 5: System Integration Testing, Quality Assurance, and Deployment Readiness:** The final iteration will encompass end-to-end functional verification, performance benchmarking, security compliance auditing, bug fixing, and final stakeholder demonstration to confirm deployment readiness.

Throughout each Sprint, four core Agile ceremonies will be conducted: Sprint Planning to establish iteration objectives and prioritize backlog items; regular progress tracking to monitor development milestones and resolve technical bottlenecks; Sprint Reviews with the Product Owner Representative to inspect working software increments and gather immediate feedback; and Sprint Retrospectives to evaluate the development process and optimize subsequent iterations.

To visually represent the iterative lifecycle, stage progression, and continuous feedback mechanisms governing the system engineering process, the operational model is depicted in Figure 3.4.1.

Figure 3.4.1

Agile Software Development Framework



As illustrated in Figure 3.4.1, the development of Estatix PMS will follow the Agile Software Development Framework across iterative cycles. The development process will begin with the Requirements Gathering stage, in which the researchers will identify the operational challenges encountered in the existing property management procedures of Torre De Manuel Residences 1 (TDMR1). The researchers will examine the procedures for managing tenant records, lease contracts, rental payments, utility bills, maintenance requests, and tenant inquiries. The gathered information will be used to establish the system

requirements and determine the functional specifications needed to improve the administration of property-related operations and services.

The Design stage will focus on planning the system structure, database architecture, user interface, and security mechanisms. The relational database will be normalized to Third Normal Form (3NF), while the researchers will prepare interface wireframes and plan the implementation of Role-Based Access Control (RBAC), audit logging, AES-256-GCM encryption, and SHA-256 integrity verification.

The Development stage will involve implementing Estatix PMS using a modular monolithic architecture and Progressive Web Application (PWA) standards. Vanilla PHP, Vanilla JavaScript, DaisyUI, Tailwind CSS, MySQL, and Soketi will be used to develop the system modules for tenant and property management, leases, payments, utilities, maintenance, announcements, communication, and reporting. Edge-AI-assisted services will also be implemented for routine inquiries and maintenance request classification and routing.

The Testing stage will verify the functionality and integration of the developed modules. The researchers will test user permissions, encryption and decryption pipelines, file-integrity verification, audit logging, and Edge-AI-assisted services. Identified errors and anomalies will be documented and corrected for subsequent iterations.

The Deployment stage will involve preparing the tested system, database, server environment, and required network components for use at TDMR1. The PWA capabilities, security mechanisms, real-time communication services, and local Edge-AI infrastructure will be configured for operational readiness.

The Review stage will evaluate the system based on empirical testing results, user feedback, ISO/IEC 25010:2023 software product quality characteristics, applicable ISO/IEC 27001:2022 Annex A controls, and the TAM. The TAM will be used to assess users' perceived usefulness and perceived ease of use of Estatix PMS, while the ISO standards will provide the empirical basis for evaluating software quality and information security posture. The findings will be used to identify necessary improvements, assess stakeholder acceptance, and refine the system in subsequent sprint cycles.

To ensure a structured and transparent software engineering lifecycle, the technical developmental procedures, architectural tasks, and implementation milestones planned across the five Agile Scrum Sprints of Estatix PMS are outlined in Table 3.4.1.

Table 3.4.1

Developmental Procedures Across Agile Sprints of Estatix PMS

Sprint Cycle	Phase / Engineering Task	Technical Developmental Procedures Executed

Sprint 1 *(Architecture & Planning)*	**Environment Setup & Database Modeling**	• Set up PHP 8.2 development environment, Apache server, and Git repository. • Install MySQL 8.0 and configure database connection parameters. • Design and normalize relational database schema to Third Normal Form (3NF). • Draft responsive UI/UX wireframes and layout grids using Tailwind CSS and DaisyUI.
Sprint 2 *(Core Modules & Access)*	**Backend Architecture & RBAC Implementation**	• Construct PSR-4 compliant modular monolithic PHP folder structure and MVC controllers. • Implement Role-Based Access Control (RBAC) middleware for the 5 stakeholder roles. • Implement session governance, CSRF tokens, and secure password hashing via bcrypt. • Build core property, unit, and tenant profile management modules.
Sprint 3 *(Security & Cryptography)*	**Cryptographic Pipelines & Audit Logging**	• Develop AES-256-GCM encryption/decryption service for stored lease documents. • Implement SHA-256 cryptographic hashing service for document integrity verification. • Implement automated, append-only database audit logging capturing user telemetry, IP, and state changes. • Configure input sanitization and PDO parameterized queries to prevent SQL injection.

Sprint 4 *(Real-Time & AI Expansion)*	**API Integration & Local Edge-AI Deployment**	• Deploy Ollama with quantized Qwen2.5 7B model on dedicated local edge server. • Build HTTPS tunnel and API adapter to connect PHP backend to local Ollama inference endpoint. • Integrate Soketi WebSocket server for real-time chat and instantaneous push notifications. • Integrate Xendit payment gateway API and configure secure webhook listener for payment status updates.
Sprint 5 *(QA & Deployment Readiness)*	**Integration Testing & PWA Configuration**	• Configure PWA manifest, service workers, and client-side asset caching for offline resilience. • Execute end-to-end integration testing across frontend, backend, database, and local AI. • Resolve identified software defects and optimize database query indexing.

Operational and Testing Procedure

To establish a seamless and standardized workflow for daily property administration, the end-to-end operational procedures to be executed by building administrators, operational staff, and residing tenants within Estatix PMS are outlined in Table 3.5.1.

Table 3.5.1

Operational Procedure

Components/Phase	**Procedures to Be Conducted**
1. User Authentication & Session Governance	Users authenticate via email/password or Google OAuth; system verifies active session, enforces RBAC, and directs user to their specific dashboard.
2. Tenant Onboarding & Leasing Operations	Prospective tenants browse vacant units, complete Didit KYC verification, submit lease applications, and digitally sign lease agreements.
3. Utility Billing & Rent Collection	Management generates monthly water/electricity dues and rent invoices; tenants pay online via Xendit or log post-dated checks; system issues digital receipts.
4. Maintenance Ticket Routing & Resolution	Tenants submit repair tickets with photos; Edge-AI categorizes and routes the request; staff execute physical repairs and upload photo proof of completion.
5. Administrative Governance & Audit Logging	Management broadcasts emergency announcements; system automatically logs all critical transactions into append-only audit logs for legal compliance.

To verify that all system modules, security controls, and integrations function as intended, the multi-tier testing procedures to be conducted are outlined in Table 3.5.2.

Table 3.5.2

Testing Procedure

Components/Phase	Procedures to Be Conducted
Functional Testing	● Test the main system modules, including tenant and property records, lease management, payment monitoring, maintenance requests, announcements, messaging, and reports. ● Verify that each function produces the expected result.
Integration Testing	● Test the interaction between the frontend, backend, database, security components, local AI service, and the Xendit payment gateway. ● Verify that data, payment webhooks, and requests are processed correctly between all connected components.
Performance Testing	● Use the browser's developer tools to inspect page loading behavior, network requests, and interface responsiveness. ● Send multiple requests to the server and record the response time of each request. ● Observe the response time of MySQL database operations, Soketi real-time updates, and Edge-AI-assisted tenant services.
Usability Testing	● Navigate Estatix PMS to complete common tasks, such as viewing property records,

	checking payment information, and submitting maintenance requests. ● Observe whether users can understand the interface and complete tasks without unnecessary difficulty. ● Gather feedback regarding the system's navigation, readability, and ease of use.
Security Testing	● Log in using different test accounts and attempt to access functions restricted to other roles to verify whether Role-Based Access Control (RBAC) is working correctly. ● Test invalid login attempts and repeated login attempts to assess the implemented authentication and account protection mechanisms. ● Inspect designated sensitive documents to verify whether AES-256-GCM encryption is applied as intended. ● Modify a test document and compare its SHA-256 hash with the original to verify file-integrity checking. ● Perform selected login and document activities and check whether the events are recorded in the audit logs.
Compatibility Testing	● Access Estatix PMS using supported web browsers and desktop and mobile devices. ● Check whether the interface adjusts correctly to different screen sizes. ● Test the main functions, forms, navigation links, and displayed records across the tested browsers and devices.

Deployment Testing	● Run Estatix PMS in its intended deployment environment and access it using authorized test accounts. ● Verify whether the application loads correctly and its main functions operate as intended. ● Check the database connection, Soketi integration, security features, and communication with the locally deployed Edge-AI service. ● Verify that authorized users can access permitted functions and restricted functions remain inaccessible to unauthorized users.

The components of Estatix PMS will be evaluated throughout the development process to determine whether the system meets its functional, usability, performance, compatibility, and security requirements. Functionality and integration testing will be conducted to verify that the system features and integrated components operate as intended. Furthermore, performance and usability testing will assess the responsiveness and ease of use of the application, while security testing will examine the effectiveness of the implemented access controls and data protection mechanisms.

Evaluation Procedure

1. Purpose of Evaluation

The purpose of the evaluation is to assess the software quality, functional performance, usability, reliability, maintainability, and information security of

Estatix PMS, and to determine whether the proposed system meets the operational requirements of Torre De Manuel Residences 1 (TDMR1). Through the participation of intended users and IT specialists, the researchers will evaluate the system using selected software quality characteristics from ISO/IEC 25010:2023 and applicable information security controls from ISO/IEC 27001:2022 Annex A. Furthermore, the evaluation will incorporate the TAM to examine user acceptance of the system in terms of perceived usefulness, perceived ease of use, perceived security and trust, and behavioral intention to use. The evaluation will also assess the effectiveness of the system's tenant services, Role-Based Access Control (RBAC), audit logging, cryptographic document protection, and Edge-AI-assisted services.

2. Evaluation Framework or Standards Used

The evaluation of Estatix PMS will be guided by established international software engineering standards and user adoption models, as summarized in Table 3.6.1.

Table 3.6.1

Evaluation Frameworks and Standards Used

Framework / Standard	**Purpose in Evaluating Estatix PMS**
ISO/IEC 25010:2023 (Software Product Quality Model)	Evaluates the software quality of Estatix PMS using selected characteristics, including functional suitability, performance efficiency, interaction capability, reliability, security, and maintainability.

ISO/IEC 27001:2022 (Information Security Management Systems)	Guides the assessment of applicable security controls, including access control, identity and authentication management, access restriction, logging, cryptography, and secure system development and testing.
Technology Acceptance Model (TAM)	Evaluates users acceptance through perceived usefulness, perceived ease of use, perceived security and trust, and behavioral intention to adopt Estatix PMS.

3. Evaluation Criteria and Indicators

The specific evaluation criteria and their corresponding focus areas within Estatix PMS are presented in Table 3.6.2.

Table 3.6.2

Evaluation Criteria and Focus Areas

Criterion	**Description / Focus in Estatix PMS**
1.Functional Suitability	Completeness, correctness, and appropriateness of tenant and property record management, lease monitoring, rental payment and utility tracking, maintenance requests, announcements, communication, and report generation.
2. Performance Efficiency	System response time and performance when accessing records, processing transactions, retrieving documents, and using the system's major functions.
3. Interaction Capability	Clarity of the interface, ease of navigation, consistency of system controls, and ease of interaction for administrators and tenants.
4. Reliability	Consistency of system operations, accuracy of saved and retrieved records, and ability of

	implemented features to operate as intended during testing.
5. Security	Effectiveness of authentication, Role-Based Access Control (RBAC), audit logging, AES-256-GCM encryption for applicable sensitive documents, and SHA-256 document-integrity verification.
6. Maintainability	Organization and modularity of the system's code and the ease of identifying errors, modifying existing functions, and improving the system.
7. Edge-AI Functionality	Evaluated by technical specialists through functional demonstration and test prompt runs to assess response relevance and classification accuracy.

Technology Acceptance Model Evaluation

The Technology Acceptance Model will be used to evaluate the perceptions and acceptance of Estatix PMS among its intended users. The evaluation questionnaire incorporates the four core dimensions of the TAM, as described in Table 3.6.3.

Table 3.6.3

TAM Evaluation Dimensions

TAM Dimension	**Description**
Perceived Usefulness	The extent to which users believe Estatix PMS helps to improve property management activities, access to information, request tracking, and communication
Perceived Ease of Use	The extent to which users believe the system is understandable, easy to navigate, and

	convenient to use when performing their assigned tasks.
Perceived Security and Trust	The extent to which users feel that their accounts, personal information, records, and sensitive documents are protected by the system's security mechanisms.
Behavioral Intention to Adopt	The extent to which users intend to use or continue using Estatix PMS for relevant property management activities.

4. Evaluation Instrument and Rating Scale

The researchers will prepare structured evaluation questionnaires based on the selected software quality characteristics of ISO/IEC 25010:2023, applicable security requirements guided by ISO/IEC 27001:2022 Annex A, and the Technology Acceptance Model (TAM). The questionnaire will contain statements about the functionality, performance, interaction capability, reliability, security, and maintainability of Estatix PMS. Additional items will address the perceived usefulness of the Edge-AI-assisted tenant services.

A four-point Likert scale will be used to obtain the respondents' assessment of the system without a neutral midpoint. The proposed rating scale is presented in Table 3.6.4

Table 3.6.4

Four-Point Rating Scale Used in the Evaluation Instrument

Numerical Scale	Verbal Interpretation
4	Strongly Agree
3	Agree
2	Disagree
1	Strongly Disagree

5. Data Gathering and Evaluation Process

The evaluation procedure will be conducted through the following sequential steps:

1. Preparation: The researchers will prepare the evaluation questionnaires based on the selected ISO/IEC 25010:2023 software product quality characteristics, the applicable ISO/IEC 27001:2022 Annex A information security controls, and the Technology Acceptance Model (TAM) constructs. The instruments will be submitted to the capstone adviser for review and validation before they are administered.

2. Respondent Selection and Distribution: The 5 alpha testers will be IT experts selected through purposive sampling. The 51 beta testers will comprise 46 randomly selected TDMR1 tenants and 5 purposively selected on-site administrative and operational staff (1 building administrator and 4 staff).

3. System Demonstration and Hands-On Interaction: The IT experts will be given a demonstration of Estatix PMS and an opportunity to use its features. They will evaluate the system and answer the expert questionnaire. Errors and concerns identified during this phase will be addressed before the system is deployed for beta testing.

4. Technical Testing and Security Verification: The researchers will conduct functional and technical tests to verify the behavior of the implemented features, including Edge-AI inference, AES-256-GCM encryption, and audit log tracking. Test results will be documented to support the evaluation of functions that cannot be adequately assessed through survey responses alone.

5. Beta Testing. The beta testers (tenants and on-site staff) will be given a walkthrough of the system and the opportunity to interact with the operational modules relevant to their roles. They will then complete the end-user questionnaire covering software usability and TAM constructs, distributed in printed or digital format along with instructions and rating scale definitions.

6. Response Collection and Validation: The completed questionnaires will be collected and verified for completeness and consistency. Responses will be kept confidential in accordance with the Data Privacy Act of 2012 and utilized strictly for academic research.

7. Data Encoding and Statistical Analysis: The ratings will be encoded, tabulated, and analyzed using the weighted mean for each questionnaire item and evaluation criterion. Each weighted mean will be interpreted using the

4-point Likert scale. Software product quality, information security controls, and user technology acceptance (TAM) results will be presented separately, and technical test results will be integrated alongside the survey findings where applicable. After the study, all collected data will be securely disposed of.

Treatment of Data

The researchers will administer an evaluation questionnaire to the intended evaluators of Estatix PMS to determine whether the system meets the identified functional, operational, and security requirements of TDMR1.

The researchers will collect, organize, and statistically analyze the responses using the weighted mean. The results will be interpreted using a four-point Likert scale, ranging from 1 (Strongly Disagree) to 4 (Strongly Agree), to determine the evaluators' level of agreement with the statements presented in the questionnaire.

The computed weighted mean scores for both technical quality and user acceptance will be evaluated against the interpretation ranges established in Table 3.7.1.

Table 3.7.1

Weighted Mean Interpretation and Acceptability Scale

Weighted Mean			
Scale	**Range - Value**	**Acceptability Level Equivalent**	**Agreement Level Equivalent**
4	3.25 - 4	Highly Acceptable	Strongly Agree
3	2.50 - 3.24	Acceptable	Agree
2	1.75 - 2.49	Slightly Acceptable	Disagree
1	1 - 1.74	Unacceptable	Strongly Disagree

Weighted Mean

The weighted mean will be computed to determine the central tendency of evaluator ratings across each software quality characteristic and security domain. Rather than determining overall project reliability from a single composite average, the computed mean will be interpreted against the established four-point evaluation criteria to assess specific levels of stakeholder acceptance and functional performance.

In computing the weighted mean, the formula is as follows

$$\text{weighted mean} = \frac{\sum(X_i \cdot W_i)}{N}$$

Where:

Σ : represents the sum of the values

X_i : It represents the i-th data value in the data set

W_i : It represents the weight or frequency of the i-th data value

N: represents the total number of respondents

Ethical Consideration

This study will observe ethical standards in the conduct of research, particularly in protecting the rights and privacy of its respondents. Before data collection, the purpose and procedures of the study will be clearly explained to the building administrator, the IT experts, and the tenants of Torre De Manuel Residences 1 (TDMR1). Participation will be voluntary, and all respondents will give their informed consent before answering the evaluation questionnaire. They will be free to decline or withdraw at any time without any penalty or effect on their tenancy or employment. Permission to conduct the study within the property will likewise be secured from the building administrator.

All information gathered will be handled in accordance with the Data Privacy Act of 2012 (Republic Act No. 10173). Respondents' identities and responses will be kept confidential and will be used only for the purposes of this research. Results will be reported in aggregate form so that no individual can be identified, and the data will be accessed only by the proponents. Once the study is completed, all collected data will be permanently deleted. These measures will

ensure that respondents' information is never used for any other purpose, is never shared with third parties, and does not remain stored beyond the period needed to complete the study.

BIBLIOGRAPHY

Adam, M., Wessel, M., & Benlian, A. (2022). AI-based chatbots in customer service: Investigating user trust and automated task compliance. Electronic Commerce Research and Applications, 51, 101112. https://doi.org/10.1016/j.elerap.2021.101112

Akadal, E., & Satman, M. H. (2022). A novel automatic relational database normalization method. Acta Informatica Pragensia, 11(3), 293–308. https://doi.org/10.18267/j.aip.193

Alyoussef, I. Y. (2022). Acceptance of a flipped classroom to improve university students' learning: An empirical study on the TAM model and the unified theory of acceptance and use of technology (UTAUT). Heliyon, 8(12), e12529. https://doi.org/10.1016/j.heliyon.2022.e12529

Anisah, S., Ridha, A. A., & Ristian, U. (2023). Usability characteristics evaluation on food delivery service applications using ISO/IEC 25010 quality model. In 2023 14th International Conference on Information & Communication Technology and System (ICTS) (pp. 138–143). IEEE. https://doi.org/10.1109/ICTS58770.2023.10330853

Bhattacharya, S., & Roy, A. (2023). Cryptographic hashing using SHA-256 for tamper-proof document verification in web architectures. Journal of

Information Security and Applications, 75, 103550. https://doi.org/10.1016/j.jisa.2023.103550

Buradkar, K. (2022). Property rental management system. International Journal of Computer Science and Mobile Computing, 11(11), 84–91. https://doi.org/10.47760/ijcsmc.2022.v11i11.014

Caparos, E., Gabriel, C., Prudente, R., & Romero, R. (2025). Cloud based property management system: Enhancing real estate operation and tenant experience. International Journal of Innovative Science and Research Technology, 10(5), 1782–1791. https://doi.org/10.38124/ijisrt/25may851

Chauhan, D. (2024, January 23). The consequences of poor data and record-keeping: A detriment to residents. Housemark. https://www.housemark.co.uk/news/the-consequences-of-poor-data-and-record-keeping-a-detriment-to-residents/

Chen, A. C. H. (2024). Performance comparison of various modes of Advanced Encryption Standard. arXiv preprint arXiv:2407.09490. https://arxiv.org/abs/2407.09490

Cinco, G. M. D., & Devaraj, M. (2026). LexCAT: A lexicon-enhanced transformer for Tagalog-English code-switched sentiment analysis. In 2026 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA) (pp. 88–94). IEEE. https://doi.org/10.1109/ACDSA67686.2026.11467593

Coda, F., Busch, K., & Bauer, B. (2024). Modular monolith: Is this the trend in software architecture? In Companion Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering (ASE-Companion '24) (pp. 52–56). Association for Computing Machinery. https://doi.org/10.1145/3643657.3643911

Cosme, C. J., & De Leon, M. M. (2024). Sentiment analysis of code-switched Filipino-English product and service reviews using transformers-based large language models. In Proceedings of the 12th International Conference on Computer and Communications Management (ICCCM 2024) (pp. 95–104). Springer. https://doi.org/10.1007/978-981-99-8349-0_11

Creswell, J. W., & Creswell, J. D. (2018). Research design: Qualitative, quantitative, and mixed methods approaches (5th ed.). SAGE Publications.

Cruz, J. C. B., & Cheng, C. (2022). Improving large-scale language models and resources for Filipino. In Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC 2022) (pp. 4939–4945). European Language Resources Association. https://aclanthology.org/2022.lrec-1.528/

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly, 13(3), 319–340. https://doi.org/10.2307/249008

Deng, Y., & Chen, Z. (2024). TickIt: Automated customer service ticket routing and state tracking with localized large language models. ACM Transactions on Software Engineering, 31(4), 1042–1058. https://doi.org/10.48550/arXiv.2504.08475

Gurses, S., & Del Alamo, J. M. (2022). Privacy engineering in modern software architectures: From principles to implementation. IEEE Software, 39(5), 34–42. https://doi.org/10.1109/MS.2022.3181249

International Organization for Standardization. (2022). Information security, cybersecurity and privacy protection: Information security management systems: Requirements (ISO/IEC 27001:2022). ISO. https://www.iso.org/standard/27001

International Organization for Standardization. (2023). Systems and software engineering: Systems and software Quality Requirements and Evaluation (SQuaRE): Product quality model (ISO/IEC 25010:2023). ISO. https://www.iso.org/standard/78176.html

Łasocha, W., & Badurowicz, M. (2022). Comparison of WebSocket and HTTP protocol performance. Journal of Computer Sciences Institute, 24, 252–257. https://doi.org/10.35784/jcsi.2452

Magno, J. M., Pelayo, R. S., & Soberano, K. T. (2024). Apartment rental management system for real-time transaction and task organization. International Journal of Multidisciplinary Research and Analysis, 7(7), 3326–3334. https://doi.org/10.47191/ijmra/v7-i07-29

Muawwal, A. (2024). Analisis kinerja progressive web app (PWA) dalam meningkatkan aksesibilitas dan performa website. Buletin Pos dan Telekomunikasi, 22(1), 35–48. https://bpostel.komdigi.go.id/index.php/bpostel/article/view/395

National Privacy Commission. (2022). Guidelines on administrative fines (NPC Circular No. 2022-01). Republic of the Philippines. https://privacy.gov.ph/circulars/npc-circular-no-2022-01/

National Privacy Commission. (2023). Security of personal data in the government and private sectors (NPC Circular No. 2023-06). Republic of the Philippines. https://privacy.gov.ph/wp-content/uploads/2023/12/NPC-Circular-No.-2023-06.pdf

Ngo, T., & Nguyen, H. (2023). Preventing SQL injection and cross-site scripting in PHP MVC architectures using PDO parameter binding and sanitization middleware. Computers & Security, 132, 103310. https://doi.org/10.1016/j.cose.2023.103310

Nguyen, H., & Le, T. (2023). Evaluating multilingual large language models on low-resource Austronesian languages: Tagalog, Indonesian, and Vietnamese. Transactions of the Association for Computational Linguistics, 11, 789–806. https://doi.org/10.1162/tacl_a_00572

Obonguko, U. A. (2024). WebSockets vs. HTTP polling: Implications for high-frequency data streaming applications. International Journal of

Modern Technology and Engineering Research, 11(2), 112–125. https://doi.org/10.5281/zenodo.16893947

OWASP Foundation. (2023). OWASP Top 10 API security risks: 2023. Open Web Application Security Project. https://owasp.org/API-Security/editions/2023/en/0x11-t10/

Oyetunji, T. S., Erinjogunola, F. L., Ajirotutu, R. O., Adeyemi, A. B., Ohakawa, T. C., & Adio, S. A. (2025). Digital solutions for streamlining property management operations in affordable housing communities. Engineering and Technology Journal, 10(3), 3619–3628. https://doi.org/10.47191/etj/v10i03.40

Pandya, K., & Holia, M. (2023). Automating customer service using LangChain: Building open-source solutions for private and custom data. arXiv preprint arXiv:2310.05421. https://arxiv.org/abs/2310.05421

Qu, G., Dong, R., Xu, M., Shen, X., & Ni, J. (2024). Mobile edge intelligence for large language models: A contemporary survey. arXiv preprint arXiv:2407.18921. https://arxiv.org/abs/2407.18921

Ramos, M. J., Go, K., & Tan, L. (2025). BATAYAN: A Filipino NLP benchmark for evaluating large language models. In Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025) (pp. 1509–1524). Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.acl-long.1509

Republic of the Philippines. (2012). *Data Privacy Act of 2012 (Republic Act No. 10173)*. Official Gazette of the Republic of the Philippines. https://www.officialgazette.gov.ph/2012/08/15/republic-act-no-10173/

Ri, O. C., Kim, C. S., & Kim, C. N. (2022). Blockchain-based RBAC model with separation of duties constraint in cloud environment. arXiv preprint arXiv:2203.04505. https://arxiv.org/abs/2203.04505

Roumeliotis, K. I., & Tselikas, N. D. (2022). Evaluating progressive web app accessibility for people with disabilities. Network, 2(2), 271–289. https://doi.org/10.3390/network2020018

Saha, R., & Das, S. (2023). Authenticated encryption with AES-GCM in web architectures: Performance and vulnerability mitigation. Journal of Cyber Security and Mobility, 12(4), 511–538. https://doi.org/10.13052/jcsm2245-1439.1242

Salunkhe, M. S. (2026). Property rental & management system. International Journal for Research in Applied Science & Engineering Technology, 14(3), 3596–3602. https://doi.org/10.22214/ijraset.2026.78551

Schulz, M., & Weber, K. (2022). Forensic readiness and audit logging in web architectures: Meeting compliance and non-repudiation requirements. Computers & Security, 118, 102725. https://doi.org/10.1016/j.cose.2022.102725

Sharma, P., & Gupta, A. (2023). On-premise deployment of quantized open-source LLMs using Ollama for enterprise inquiry automation. In 2023 IEEE International Conference on Big Data (pp. 3120–3127). IEEE. https://doi.org/10.1109/BigData59044.2023.10386412

Silva, R., & Santos, M. (2023). Evaluating utility-first CSS frameworks for responsive web application performance and maintainability. Journal of Web Engineering, 22(5), 781–802. https://doi.org/10.13052/jwe1540-9589.2254

Thazhath, M. B., Michalak, J., & Hoang, T. (2022). Harpocrates: Privacy-preserving and immutable audit log for sensitive data operations. In 2022 IEEE 4th International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA) (pp. 226–235). IEEE. https://arxiv.org/abs/2211.04741

Truss, M., Müller, M., & Weber, P. (2024). AI-based classification of customer support tickets: State of the art and implementation with AutoML. arXiv preprint arXiv:2406.01789. https://arxiv.org/abs/2406.01789

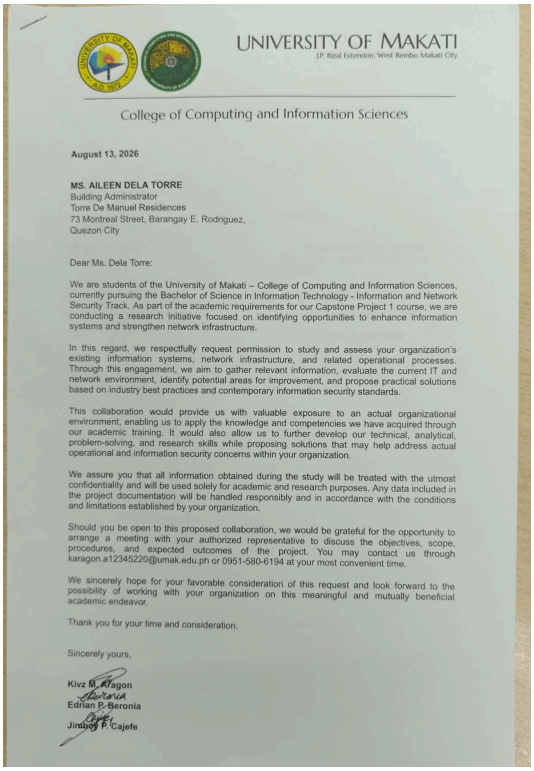
Villasin, J. Y., & Encarnacion, R. E. (2024). Perceptions of realtors on property information processing: Basis for a web-based real estate property management system. International Journal of Advanced Research in Science, Communication and Technology, 4(3), 94–103. https://doi.org/10.48175/ijarsct-18910

Wagh, S., Kulkarni, P., & Joshi, R. (2023). User experience comparison of progressive web apps and native mobile applications: A mixed-methods approach. Journal of Engineering Research and Reports, 25(11), 120–132. https://doi.org/10.9734/jerr/2023/v25i111029

APPENDICES

Appendix A

Intent Letter



UNIVERSITY OF MAKATI
J.P. Rizal Extension, West Rembo Makati City

College of Computing and Information Sciences

August 13, 2026

MS. AILEEN DELA TORRE
Building Administrator
Torre De Manuel Residences
73 Montreal Street, Barangay E. Rodriguez,
Quezon City

Dear Ms. Dela Torre:

We are students of the University of Makati – College of Computing and Information Sciences, currently pursuing the Bachelor of Science in Information Technology - Information and Network Security Track. As part of the academic requirements for our Capstone Project 1 course, we are conducting a research initiative focused on identifying opportunities to enhance information systems and strengthen network infrastructure.

In this regard, we respectfully request permission to study and assess your organization's existing information systems, network infrastructure, and related operational processes. Through this engagement, we aim to gather relevant information, evaluate the current IT and network environment, identify potential areas for improvement, and propose practical solutions based on industry best practices and contemporary information security standards.

This collaboration would provide us with valuable exposure to an actual organizational environment, enabling us to apply the knowledge and competencies we have acquired through our academic training. It would also allow us to further develop our technical, analytical, problem-solving, and research skills while proposing solutions that may help address actual operational and information security concerns within your organization.

We assure you that all information obtained during the study will be treated with the utmost confidentiality and will be used solely for academic and research purposes. Any data included in the project documentation will be handled responsibly and in accordance with the conditions and limitations established by your organization.

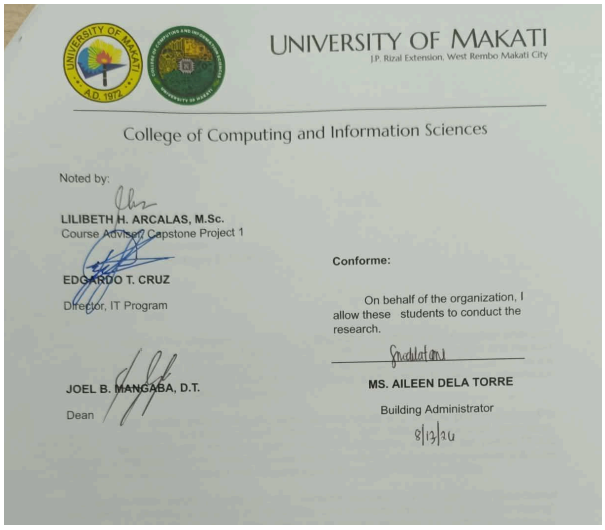
Should you be open to this proposed collaboration, we would be grateful for the opportunity to arrange a meeting with your authorized representative to discuss the objectives, scope, procedures, and expected outcomes of the project. You may contact us through karagon.a12345220@umak.edu.ph or 0951-580-6194 at your most convenient time.

We sincerely hope for your favorable consideration of this request and look forward to the possibility of working with your organization on this meaningful and mutually beneficial academic endeavor.

Thank you for your time and consideration.

Sincerely yours,

Kivz M. Aragon
Edrian P. Beronia
Jimboy P. Cajefe



UNIVERSITY OF MAKATI
J.P. Rizal Extension, West Rembo Makati City

College of Computing and Information Sciences

Noted by:

LILIBETH H. ARCALAS, M.Sc.
Course Adviser, Capstone Project 1

EDGARDO T. CRUZ
Director, IT Program

JOEL B. MANGABA, D.T.
Dean

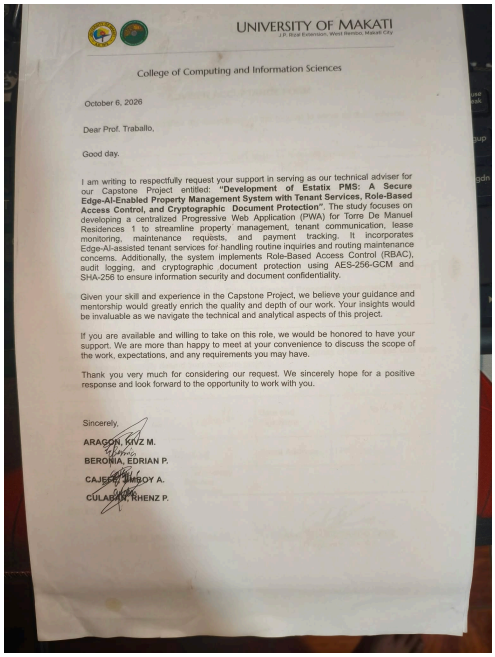
Conforme:

On behalf of the organization, I allow these students to conduct the research.

MS. AILEEN DELA TORRE
Building Administrator
8/13/26

Appendix B

Letter for Technical Adviser



UNIVERSITY OF MAKATI
J.P. Rizal Extension, West Rembo, Makati City

College of Computing and Information Sciences

October 6, 2026

Dear Prof. Traballo,

Good day.

I am writing to respectfully request your support in serving as our technical adviser for our Capstone Project entitled: **"Development of Estatix PMS: A Secure Edge-AI-Enabled Property Management System with Tenant Services, Role-Based Access Control, and Cryptographic Document Protection"**. The study focuses on developing a centralized Progressive Web Application (PWA) for Torre De Manuel Residences 1 to streamline property management, tenant communication, lease monitoring, maintenance requests, and payment tracking. It incorporates Edge-AI-assisted tenant services for handling routine inquiries and routing maintenance concerns. Additionally, the system implements Role-Based Access Control (RBAC), audit logging, and cryptographic document protection using AES-256-GCM and SHA-256 to ensure information security and document confidentiality.

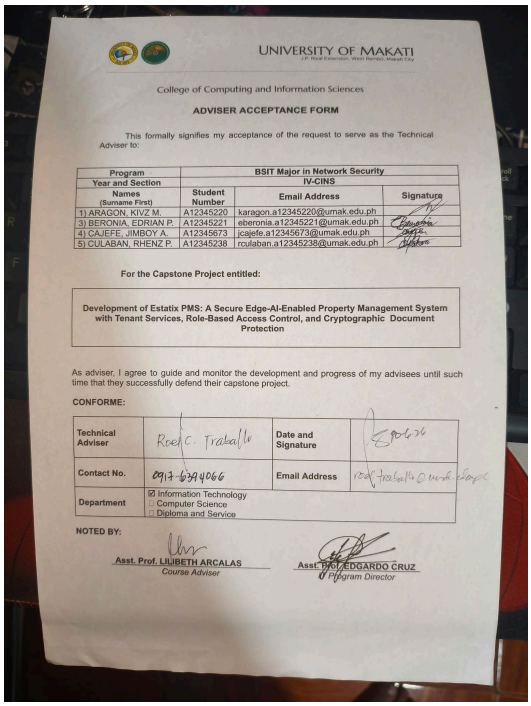
Given your skill and experience in the Capstone Project, we believe your guidance and mentorship would greatly enrich the quality and depth of our work. Your insights would be invaluable as we navigate the technical and analytical aspects of this project.

If you are available and willing to take on this role, we would be honored to have your support. We are more than happy to meet at your convenience to discuss the scope of the work, expectations, and any requirements you may have.

Thank you very much for considering our request. We sincerely hope for a positive response and look forward to the opportunity to work with you.

Sincerely,

ARAGON, KIVZ M.
BERONIA, EDRIAN P.
CAJEFE, JIMBOY A.
CULABAN, RHENZ P.



UNIVERSITY OF MAKATI
J.P. Rizal Extension, West Rembo, Makati City

College of Computing and Information Sciences

ADVISER ACCEPTANCE FORM

This formally signifies my acceptance of the request to serve as the Technical Adviser to:

Program	BSIT Major in Network Security		
Year and Section	IV-CINS		
Names (Surname First)	Student Number	Email Address	Signature
1) ARAGON, KIVZ M.	A12345220	karagon.a12345220@umak.edu.ph	
3) BERONIA, EDRIAN P.	A12345221	eberonia.a12345221@umak.edu.ph	
4) CAJEFE, JIMBOY A.	A1234567 3	jcajefe.a12345673@umak.edu.ph	
5) CULABAN, RHENZ P.	A12345238	rculaban.a12345238@umak.edu.ph	

For the Capstone Project entitled:

Development of Estatix PMS: A Secure Edge-AI-Enabled Property Management System with Tenant Services, Role-Based Access Control, and Cryptographic Document Protection

As adviser, I agree to guide and monitor the development and progress of my advisees until such time that they successfully defend their capstone project.

CONFORME:

Technical Adviser	Roel C. Traballo	Date and Signature	10/6/26
Contact No.	0917-637-4066	Email Address	rcd.traballo@umak.edu.ph
Department	☑ Information Technology ☐ Computer Science ☐ Diploma and Service		

NOTED BY:

Asst. Prof. LILIBETH ARCALAS
Course Adviser

Asst. Prof. EDGARDO CRUZ
Program Director

Appendix C

Evaluation Instrument

Dear Respondents,

We are students from **University of Makati (UMak) IV-CINS**, and we are currently developing a Capstone project titled ***Estatix PMS: A Secure Edge-AI-Enabled Property Management System with Tenant Services, Role-Based Access Control and Cryptographic Document Protection*** for the Academic Year 2026–2027. This Project is currently developing for **Torre De Manuel Residences 1**, It aims to improve property management operations by providing centralized management of tenant records, lease contracts, rental payments, utility bills, maintenance requests, announcements, communication, and reports.

As part of our research, we are conducting a survey to gather insights and feedback from key stakeholders to refine the system and ensure it meets community needs. We highly value your input and assure you that your responses will be treated with the outmost confidentiality in compliance with data privacy laws. All information collected will be used solely for academic purposes and will not be shared with unauthorized third parties.

PART I. RESPONDENT INFORMATION

Name:(Optional) _____

Respondent Category (please check one):

☐ Tenant

☐ Building Staff

☐ Landlord

☐ Building Administrator

☐ IT Expert

PART II. SYSTEM EVALUATION

Instructions: Put a check mark (✓) under the rating that best describes your experience with Estatix PMS.

Rating Scale

Rating	Description
4	Strongly Agree
3	Agree
2	Disagree
1	Strongly Disagree

A. FUNCTIONALITY

Evaluates whether the system has the features needed to perform its tasks.

No.	Statement	4	3	2	1
1	The system provides useful features for managing property-related activities.				
2	The system displays correct information about tenants, rooms, leases, and payments.				
3	The system allows users to perform tasks based on their assigned accounts.				
4	The system makes it easy to submit and check the status of maintenance requests.				
5	The system makes it easy to view announcements, messages, and room viewing schedules.				

B. PERFORMANCE

Evaluates how quickly and smoothly the system works.

No.	Statement	4	3	2	1
1	The system opens its pages without taking too much time.				
2	The system responds quickly when I click buttons or submit information.				
3	The system works smoothly when I use its features.				

4	The system displays information without unnecessary delays.				
5	The system can handle normal activities without becoming too slow.				

C. EASE OF USE

Evaluates how easy the system is to understand and navigate.

No.	Statement	4	3	2	1
1	The system is easy to understand, even for first-time users.				
2	The menus and buttons are easy to find.				
3	The instructions and information shown on the system are clear.				
4	The system provides clear messages when I complete an action or encounter a problem.				
5	The system is convenient to use on both computers and mobile devices.				

D. RELIABILITY

Evaluates whether the system works properly and consistently.

No.	Statement	4	3	2	1
1	The system works properly whenever I use its features.				
2	The system saves information correctly when I add or update records.				
3	The system does not frequently encounter errors while I am using it.				
4	The system keeps its features available during normal use.				
5	The system informs me when an action cannot be completed.				

E. SECURITY

Evaluates whether the system protects accounts and important information.

No.	Statement	4	3	2	1
1	The system requires users to log in before accessing their accounts.				
2	Users can only access the features allowed for their accounts.				
3	The system helps protect personal and property information from unauthorized access.				

4	The system keeps important documents and records protected.				
5	The system keeps records of important activities performed by users.				

PART III. SYSTEM USEFULNESS AND USER EXPERIENCE

A. Perceived Usefulness

Evaluates whether Estatix PMS helps users accomplish their tasks.

No.	Statement	4	3	2	1
1	Estatix PMS helps me complete property-related tasks more easily.				
2	Estatix PMS helps me find the information I need.				
3	Estatix PMS makes property-related tasks more organized and convenient.				

B. Perceived Ease of Use

Evaluates whether Estatix PMS is easy to learn and use.

No.	Statement	4	3	2	1
1	Learning how to use Estatix PMS is easy for me.				
2	I can use the system without needing too much assistance.				
3	The steps for using the system's features are easy to understand and follow. .				

C. Attitude Toward Using

Evaluates users' feelings and experiences when using Estatix PMS.

No.	Statement	4	3	2	1
1	I feel comfortable using Estatix PMS.				
2	I think using Estatix PMS is a good way to manage property-related tasks.				
3	I have a positive experience when using Estatix PMS.				

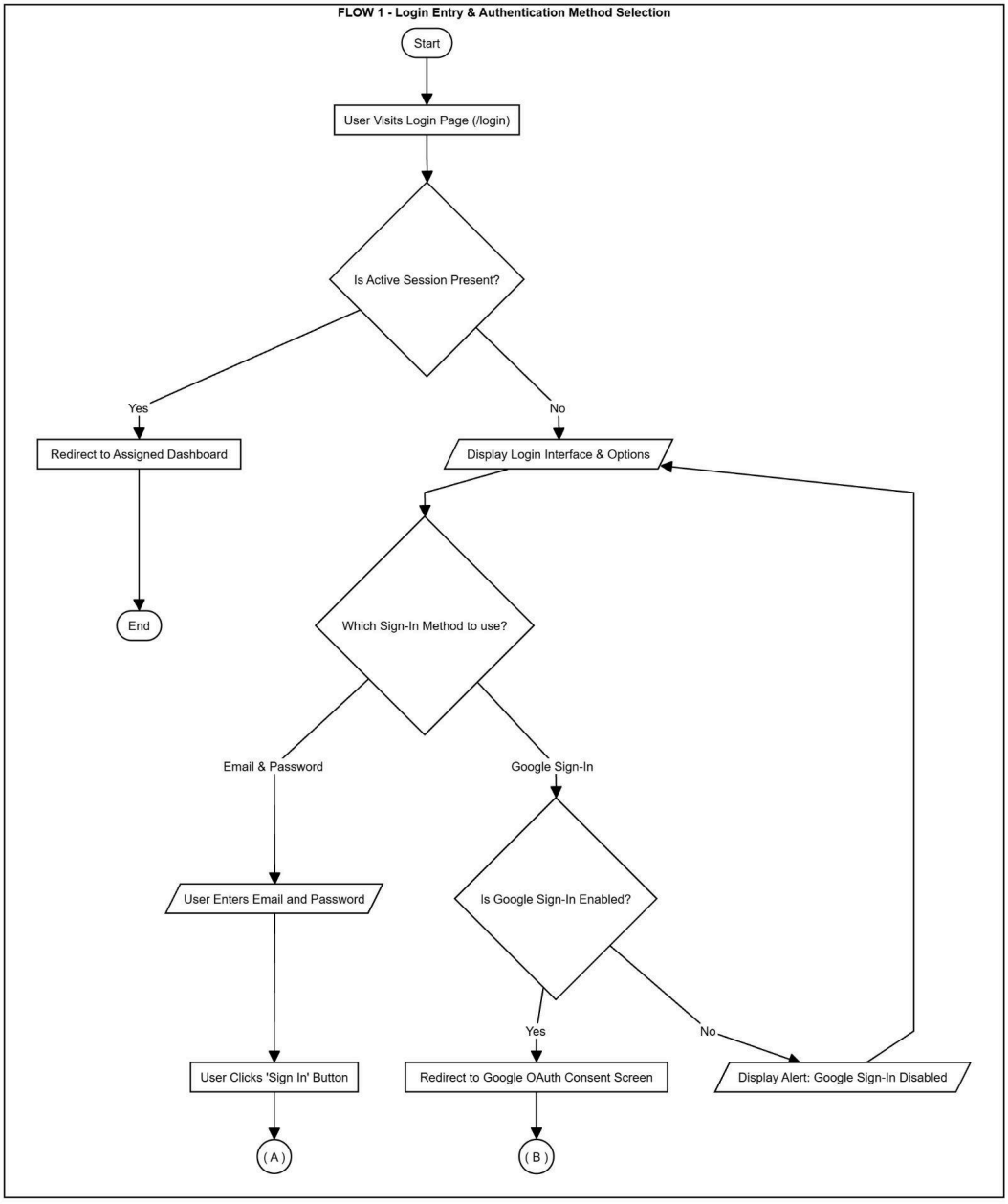
D. Behavioral Intention to Use

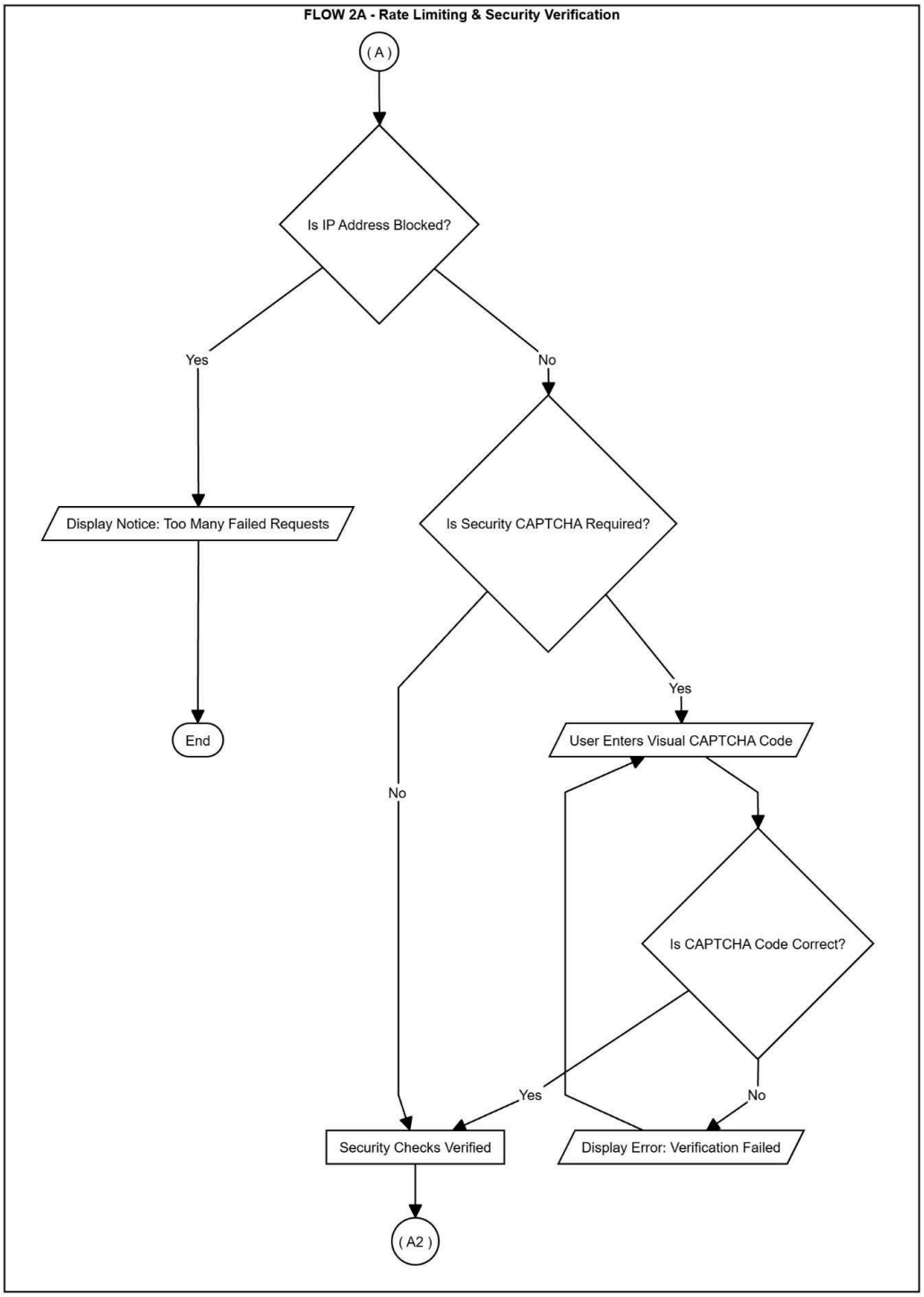
Evaluates users' willingness to continue using the system.

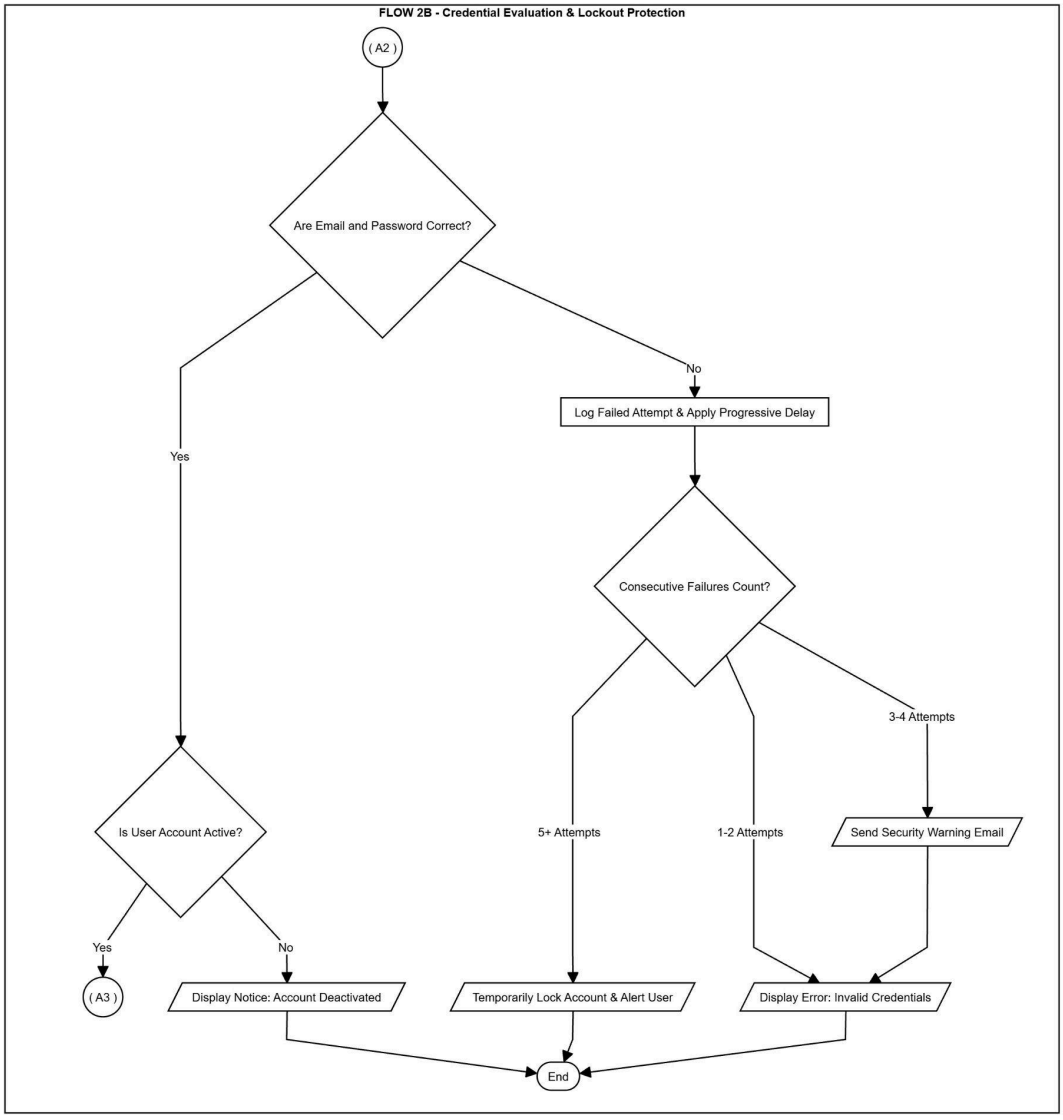
No.	Statement	4	3	2	1
1	I intend to use Estatix PMS whenever I need its features.				
2	I would choose Estatix PMS to perform property-related tasks when it is available.				
3	I would recommend Estatix PMS to other people who need a property management system.				

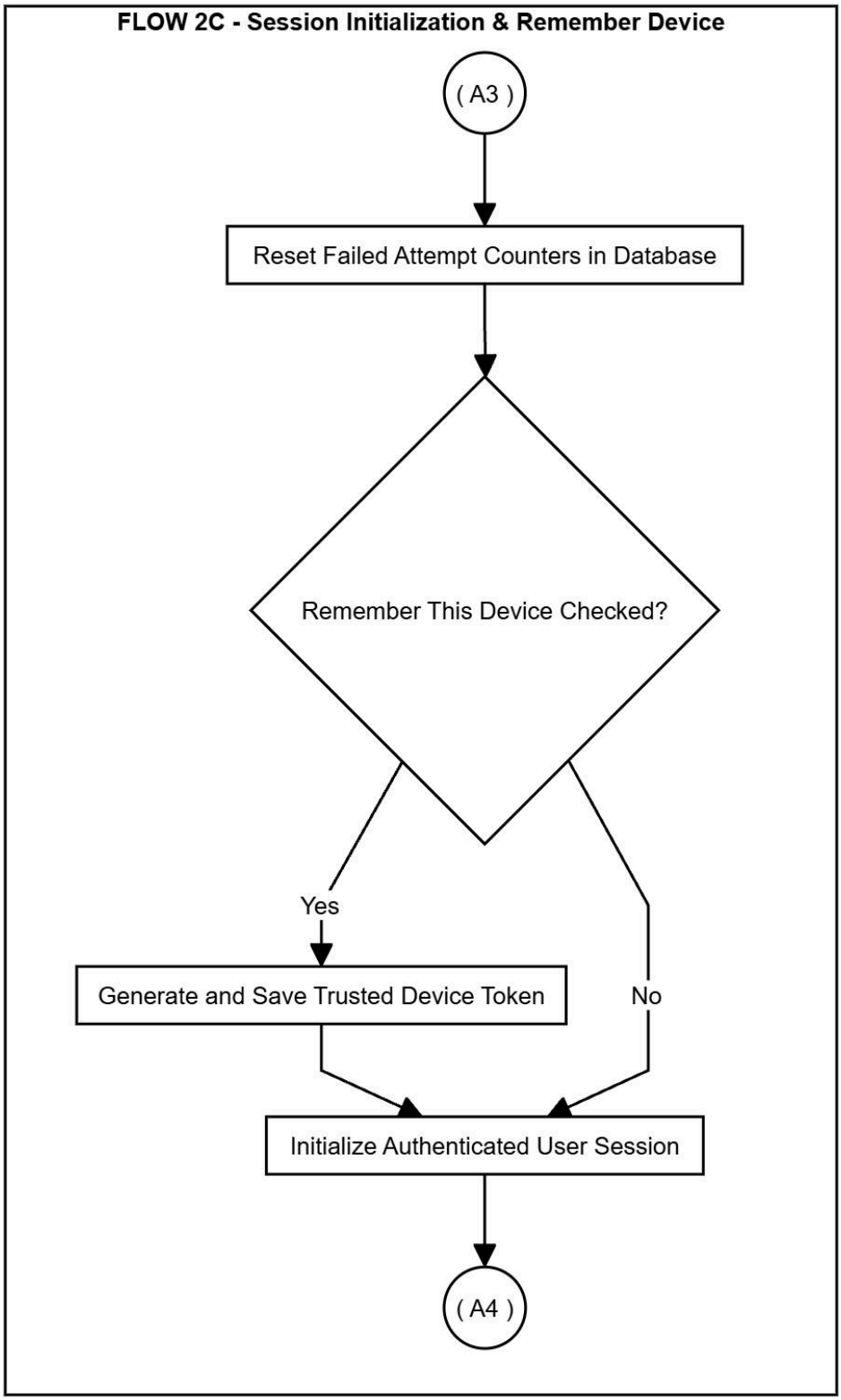
Appendix D

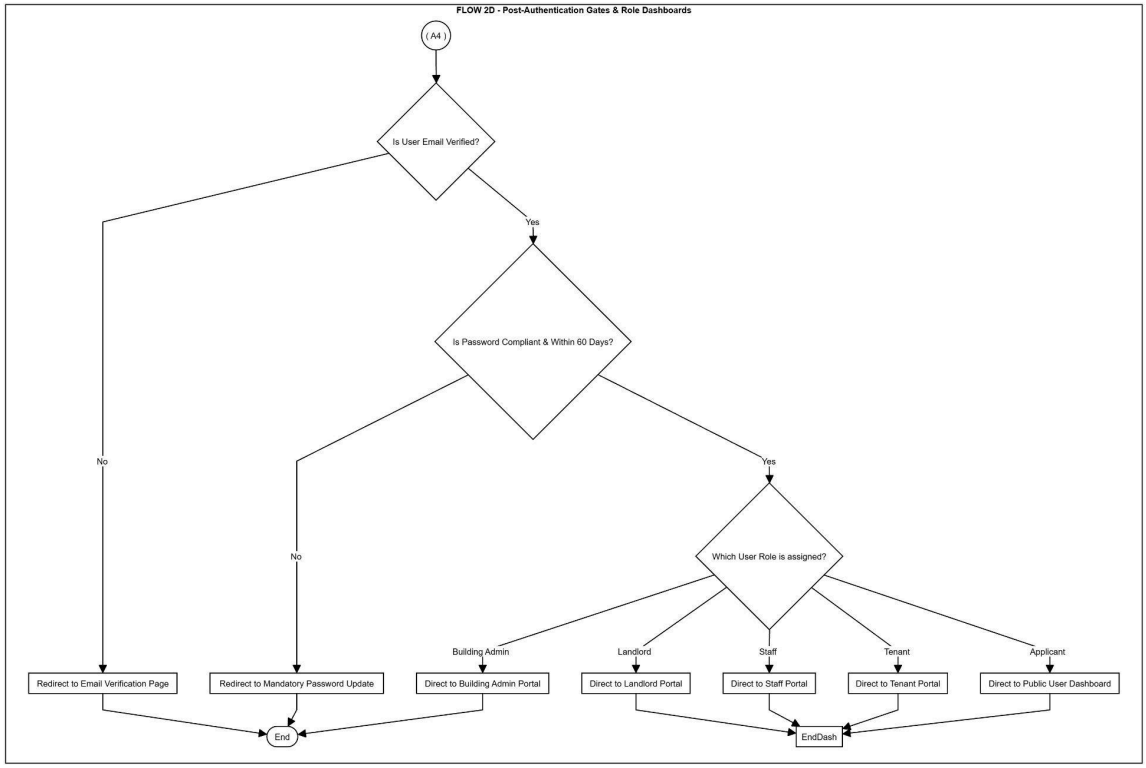
Flow 1: Login Flowchart

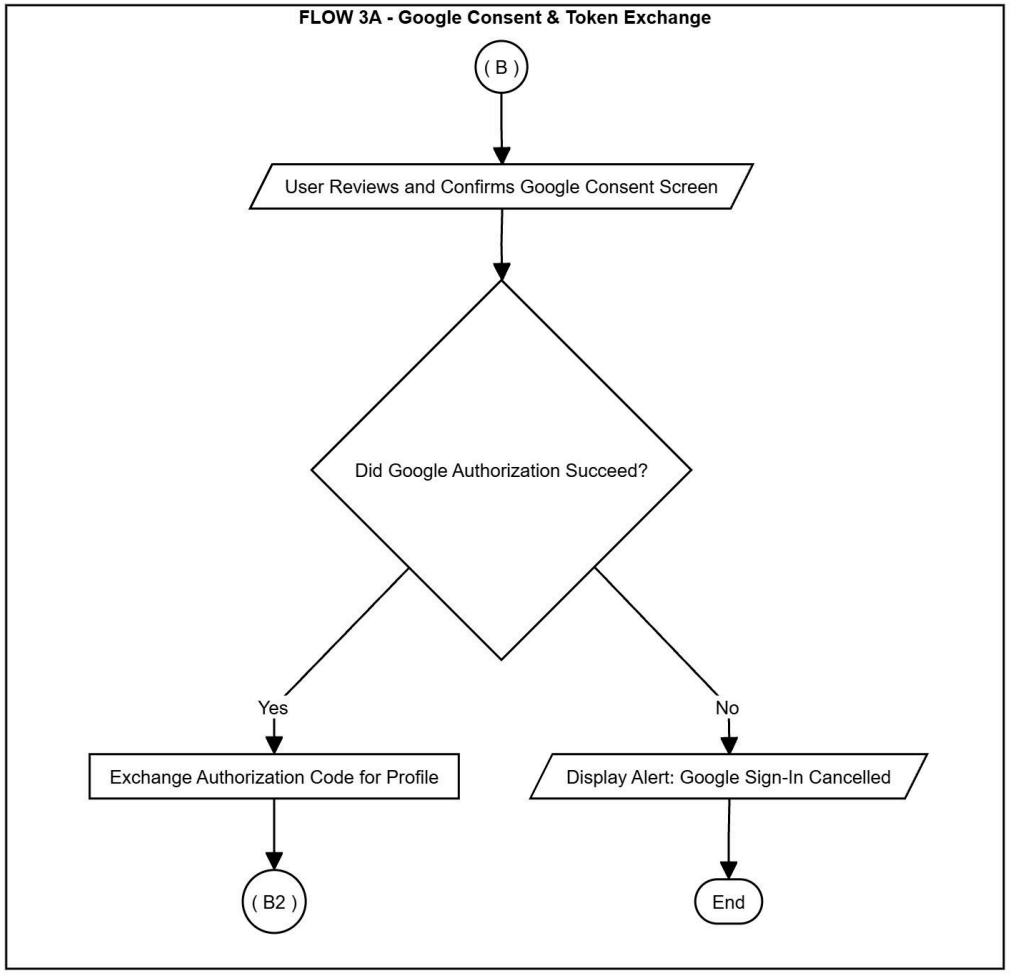


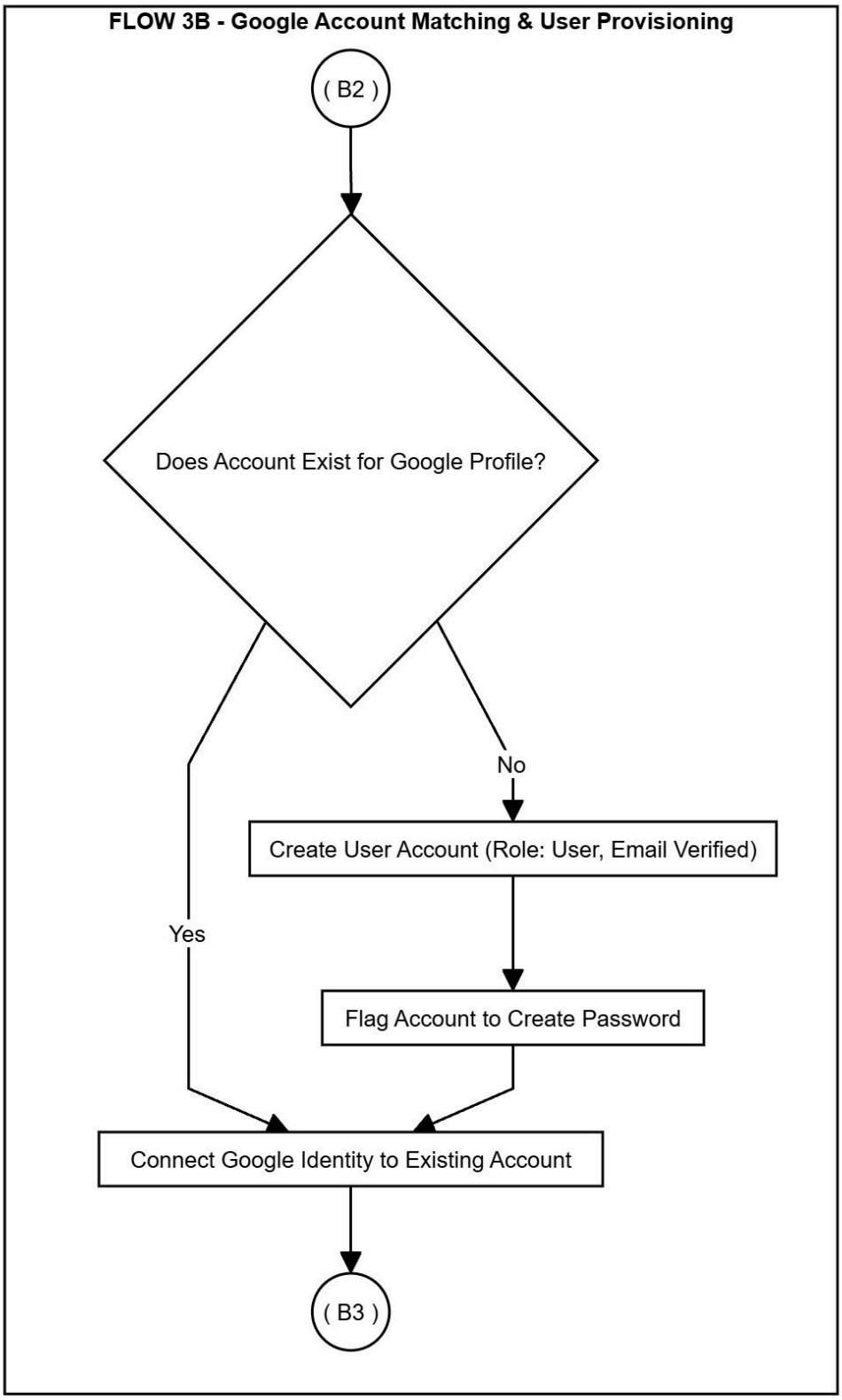

FLOW 2A - Rate Limiting & Security Verification
(A)
Is IP Address Blocked?
Yes
No
Display Notice: Too Many Failed Requests
Is Security CAPTCHA Required?
End
Yes
User Enters Visual CAPTCHA Code
No
Is CAPTCHA Code Correct?
Yes
No
Security Checks Verified
Display Error: Verification Failed
(A2)

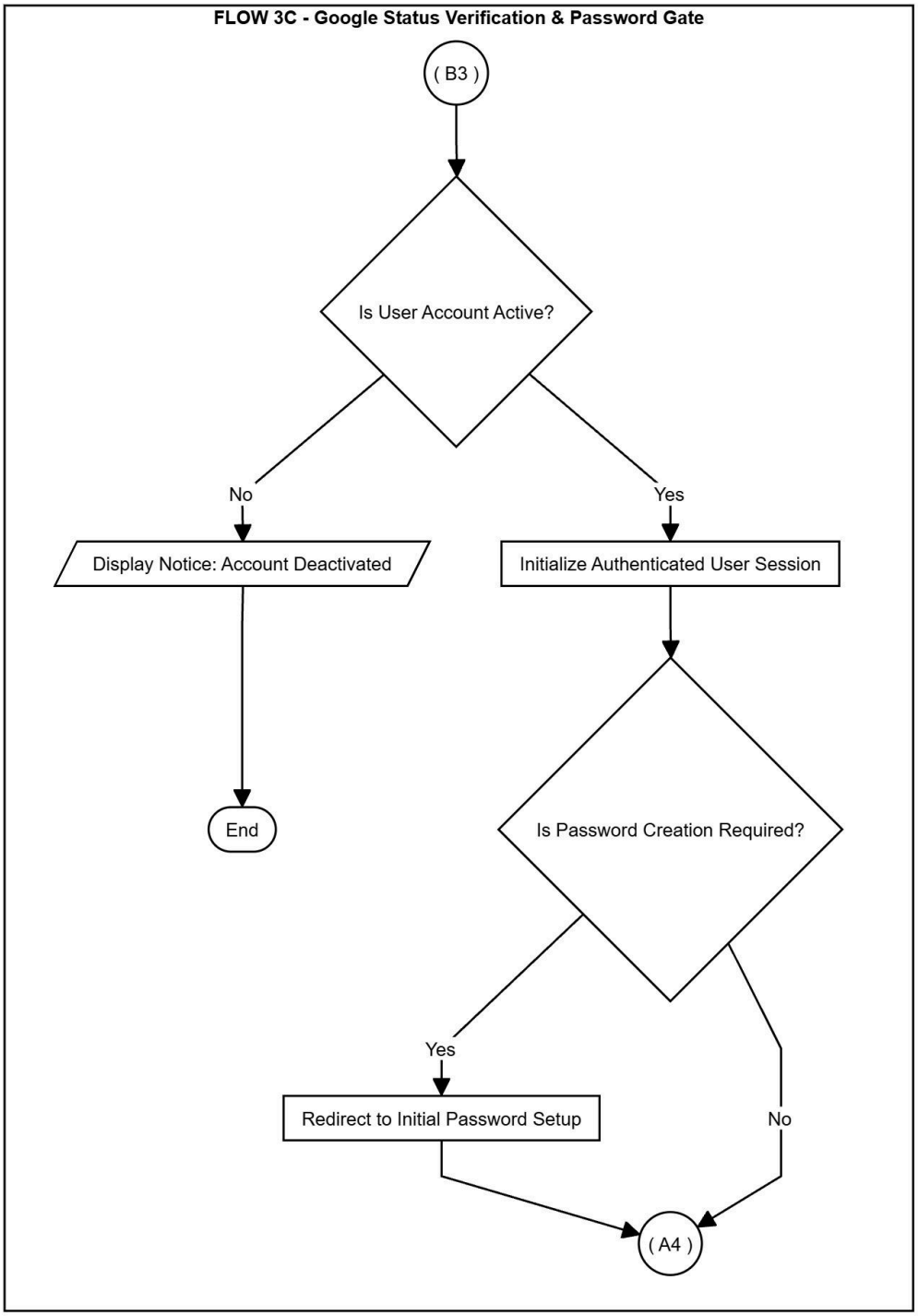

FLOW 2B - Credential Evaluation & Lockout Protection
(A2)
Are Email and Password Correct?
No
Log Failed Attempt & Apply Progressive Delay
Yes
Consecutive Failures Count?
3-4 Attempts
Is User Account Active?
5+ Attempts
1-2 Attempts
Send Security Warning Email
Yes
No
(A3)
Display Notice: Account Deactivated
Temporarily Lock Account & Alert User
Display Error: Invalid Credentials
End


FLOW 2C - Session Initialization & Remember Device
(A3)
Reset Failed Attempt Counters in Database
Remember This Device Checked?
Yes
No
Generate and Save Trusted Device Token
Initialize Authenticated User Session
(A4)

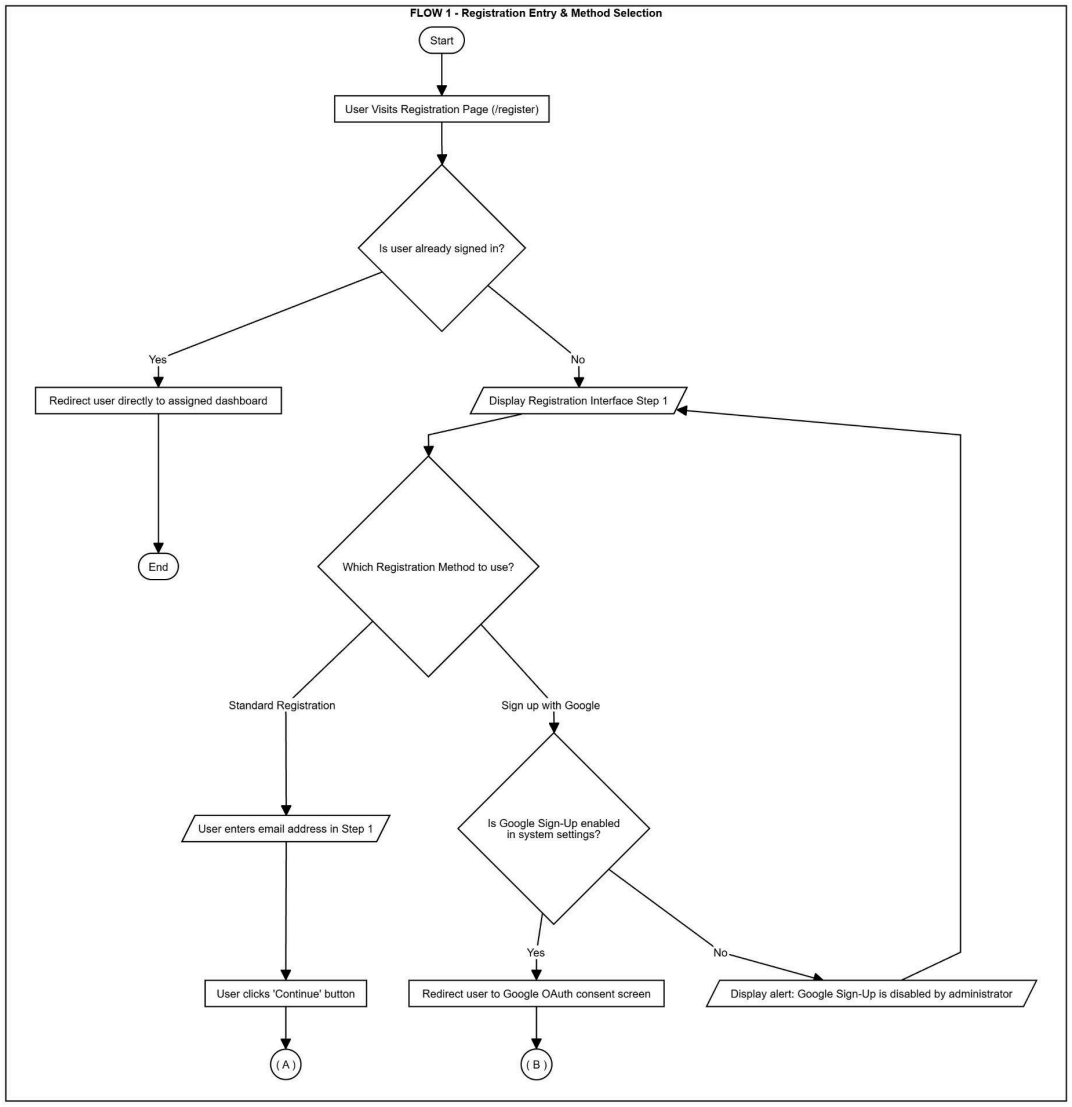

FLOW 2D - Post-Authentication Gates & Role Dashboards
(A4)
Is User Email Verified?
No
Yes
Is Password Compliant & Within 60 Days?
No
Yes
Which User Role is assigned?
Building Admin
Landlord
Staff
Tenant
Applicant
Redirect to Email Verification Page
Redirect to Mandatory Password Update
Direct to Building Admin Portal
Direct to Landlord Portal
Direct to Staff Portal
Direct to Tenant Portal
Direct to Public User Dashboard
End
EndDash

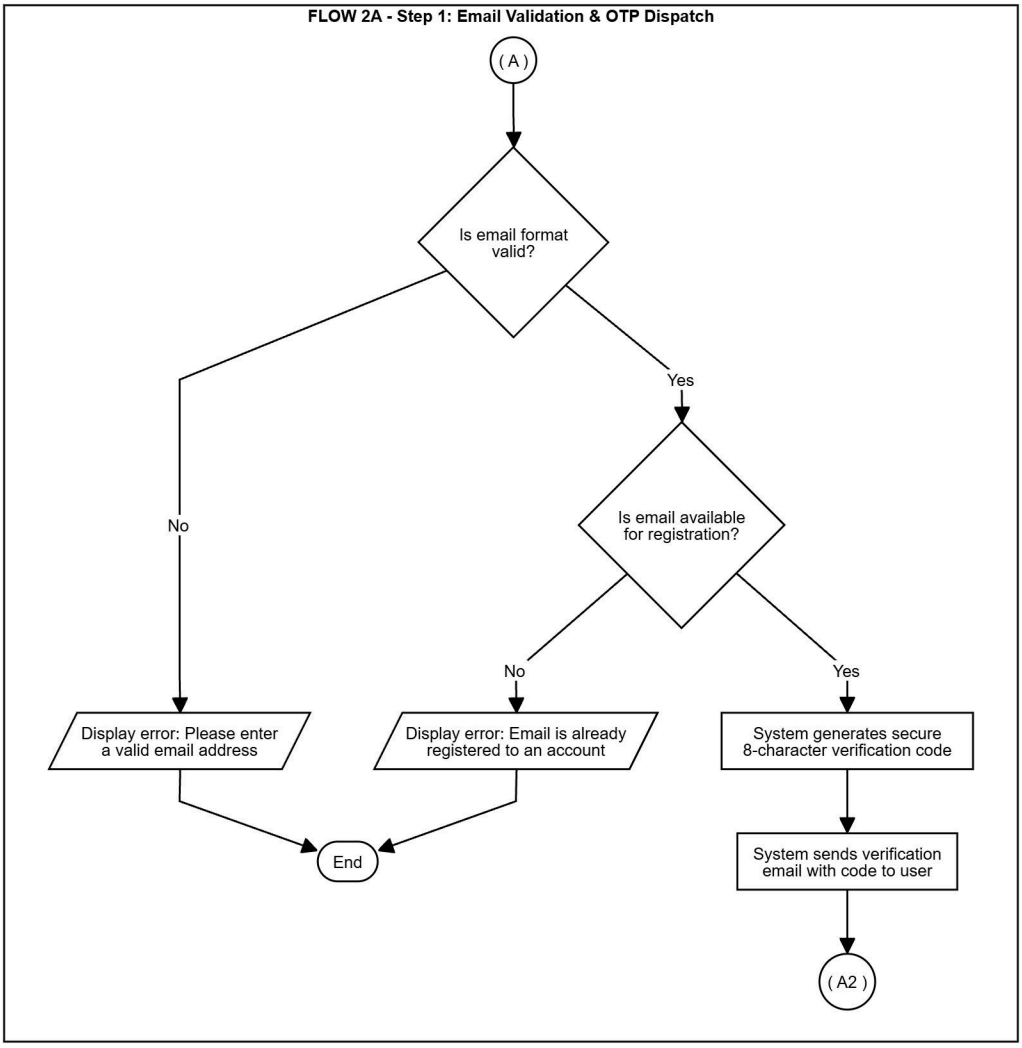

FLOW 3A - Google Consent & Token Exchange
(B)
User Reviews and Confirms Google Consent Screen
Did Google Authorization Succeed?
Yes
No
Exchange Authorization Code for Profile
Display Alert: Google Sign-In Cancelled
(B2)
End

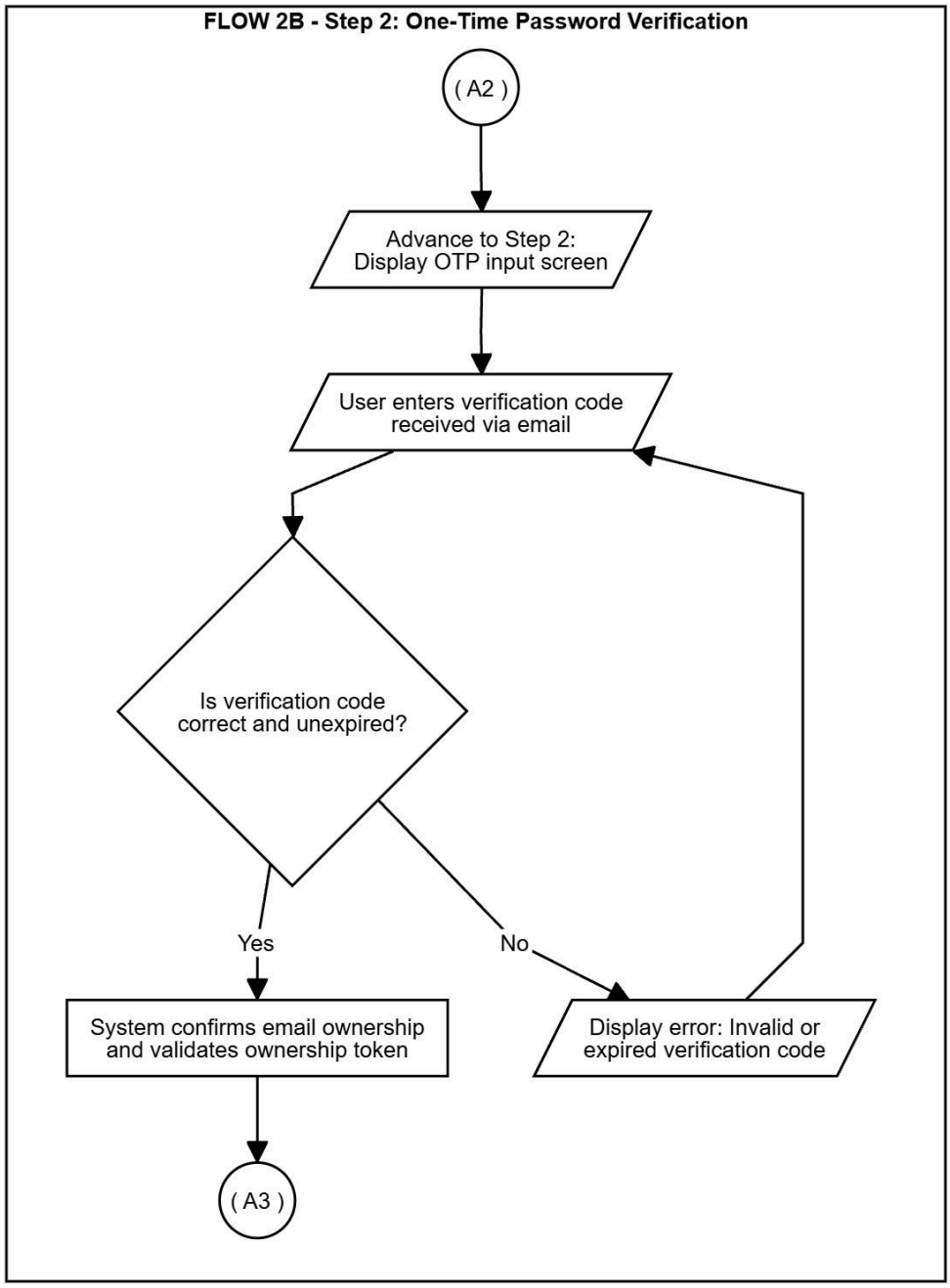

FLOW 3B - Google Account Matching & User Provisioning
(B2)
Does Account Exist for Google Profile?
No
Create User Account (Role: User, Email Verified)
Yes
Flag Account to Create Password
Connect Google Identity to Existing Account
(B3)

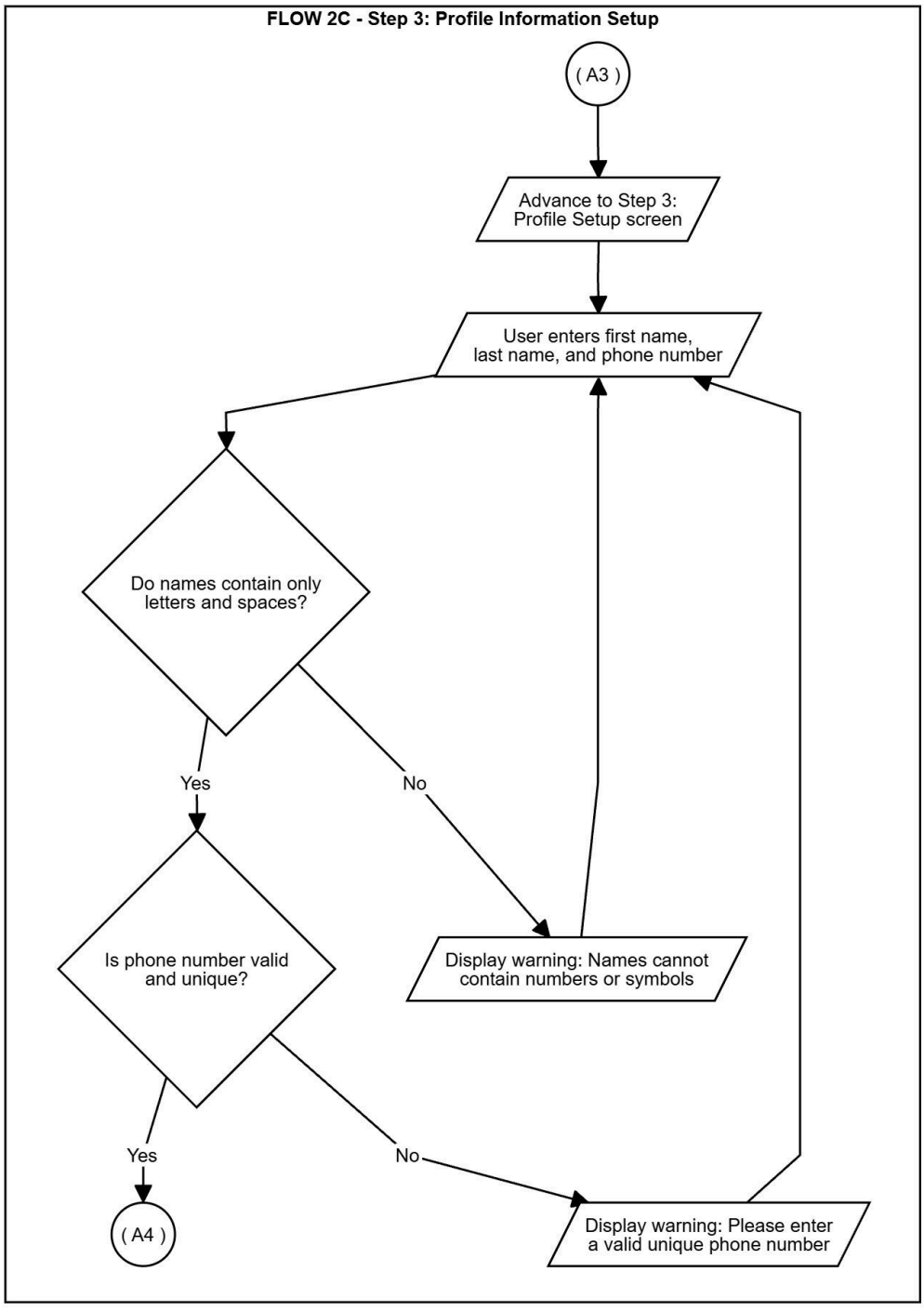

FLOW 3C - Google Status Verification & Password Gate
(B3)
Is User Account Active?
No
Yes
Display Notice: Account Deactivated
Initialize Authenticated User Session
End
Is Password Creation Required?
Yes
No
Redirect to Initial Password Setup
(A4)

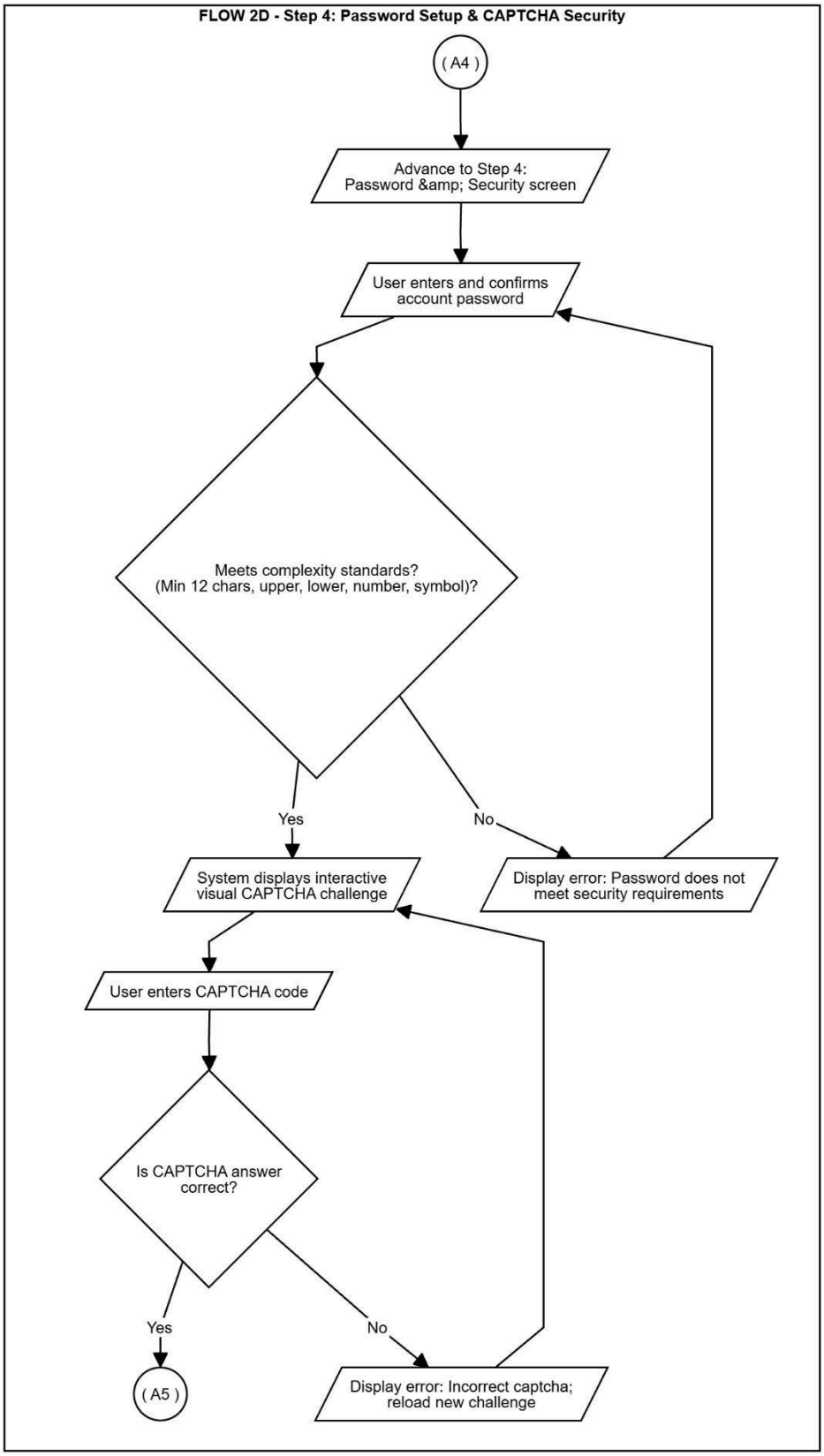
Flow 2: Register Flowchart

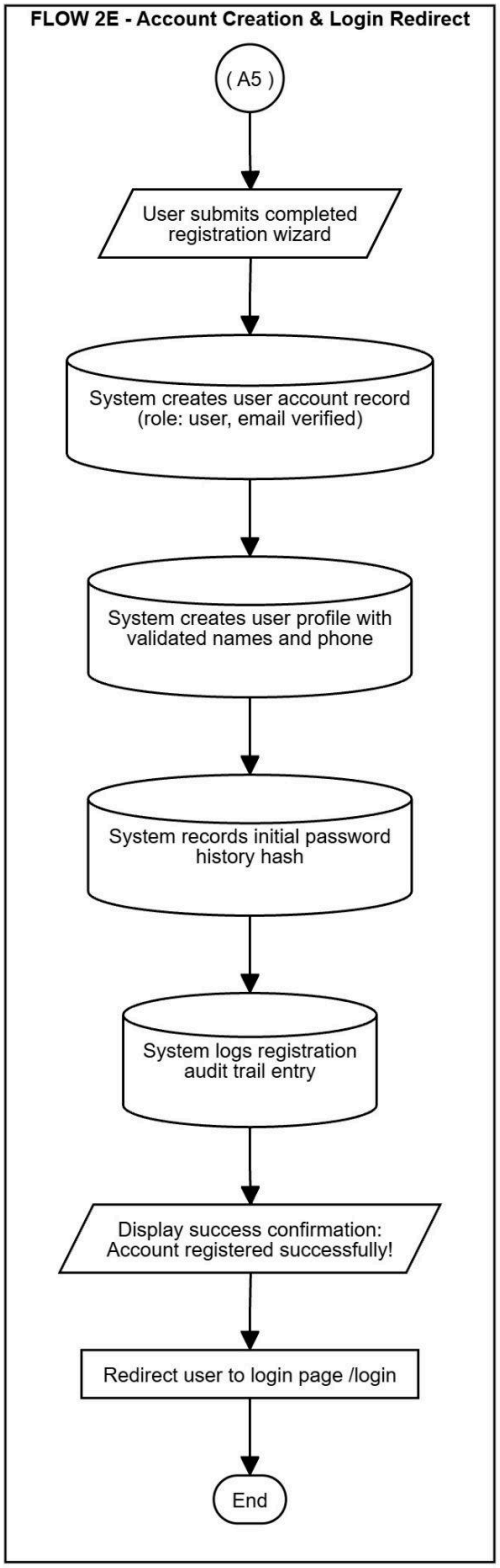


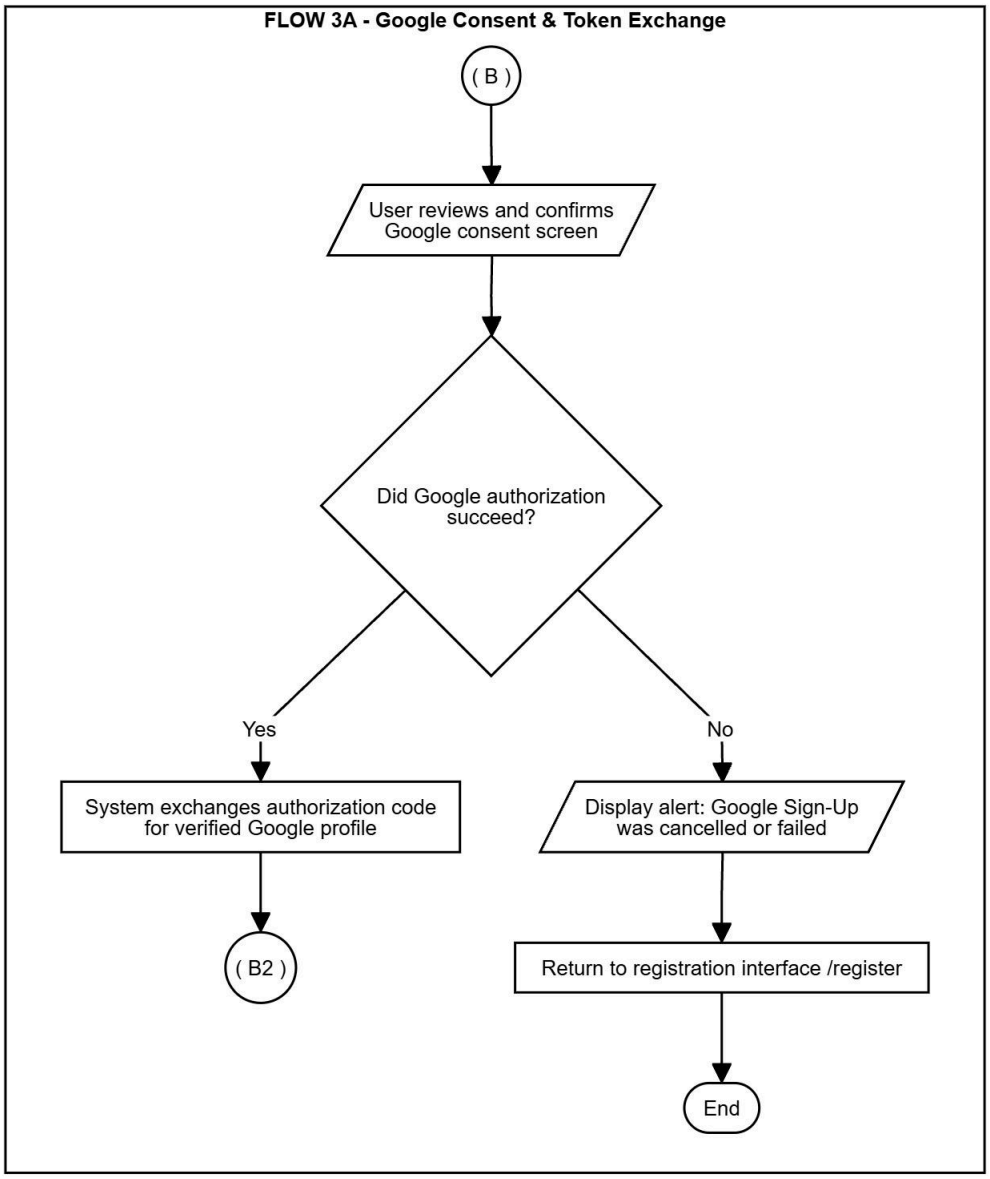

FLOW 2A - Step 1: Email Validation & OTP Dispatch
(A)
Is email format valid?
No
Yes
Is email available for registration?
No
Yes
Display error: Please enter a valid email address
Display error: Email is already registered to an account
System generates secure 8-character verification code
End
System sends verification email with code to user
(A2)

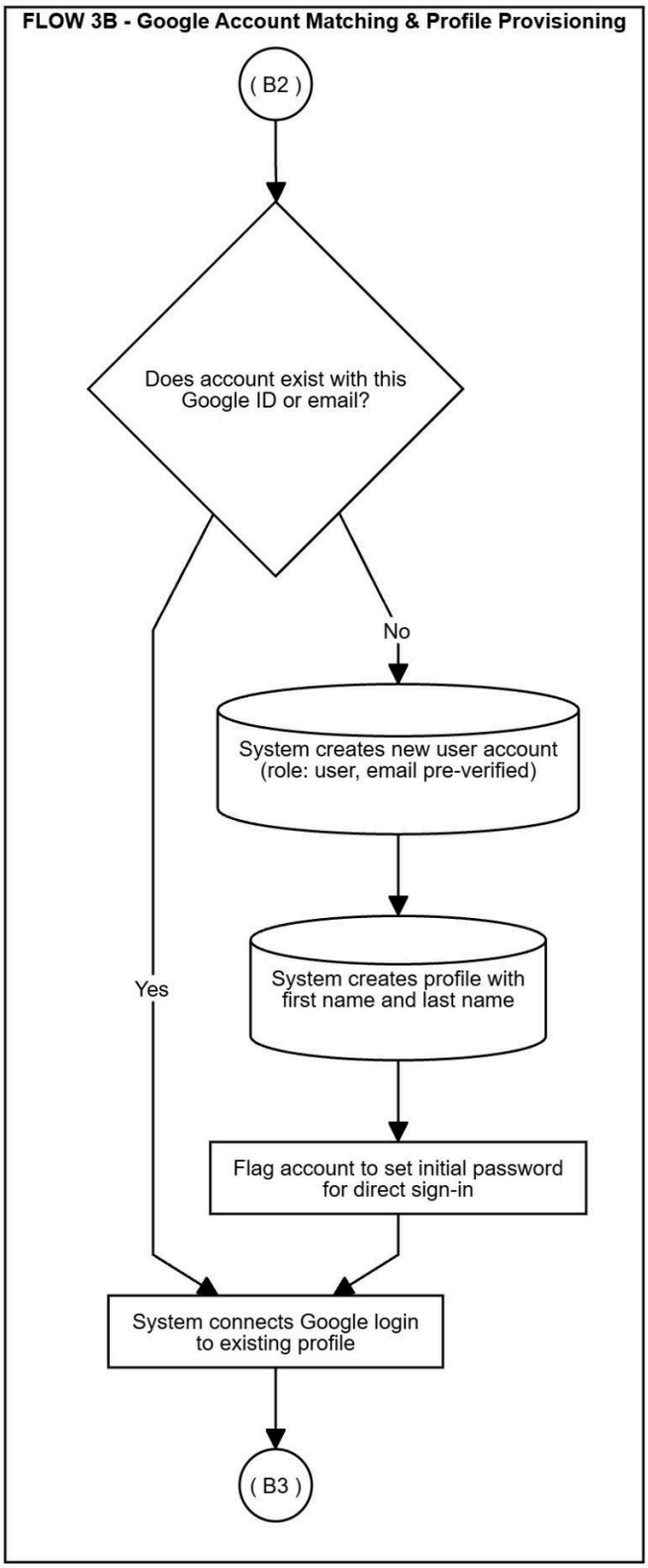

FLOW 2B - Step 2: One-Time Password Verification
(A2)
Advance to Step 2:
Display OTP input screen
User enters verification code
received via email
Is verification code
correct and unexpired?
Yes
No
System confirms email ownership
and validates ownership token
Display error: Invalid or
expired verification code
(A3)

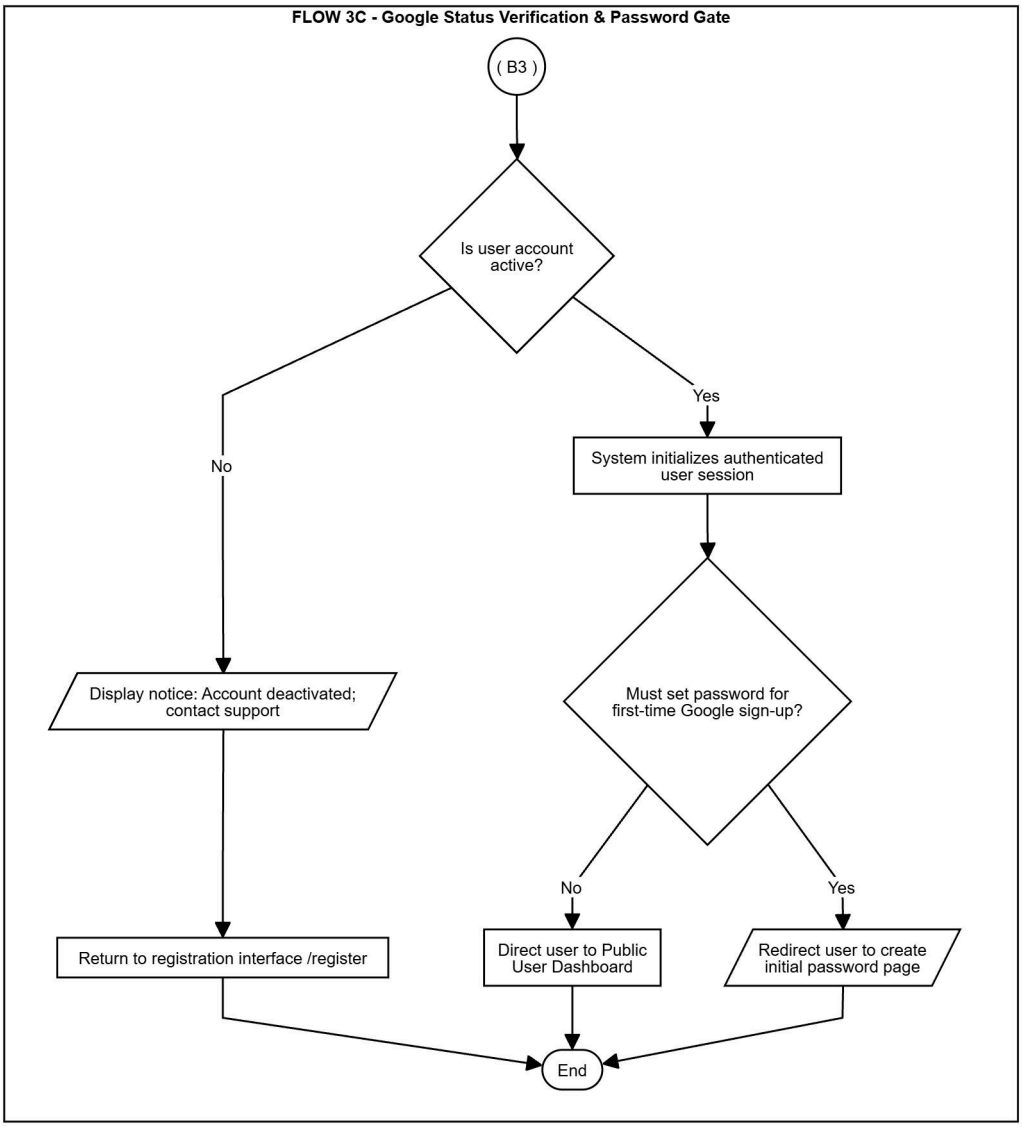

FLOW 2C - Step 3: Profile Information Setup
(A3)
Advance to Step 3: Profile Setup screen
User enters first name, last name, and phone number
Do names contain only letters and spaces?
Yes
No
Is phone number valid and unique?
Display warning: Names cannot contain numbers or symbols
Yes
No
(A4)
Display warning: Please enter a valid unique phone number


FLOW 2D - Step 4: Password Setup & CAPTCHA Security
(A4)
Advance to Step 4:
Password & Security screen
User enters and confirms
account password
Meets complexity standards?
(Min 12 chars, upper, lower, number, symbol)?
Yes
No
System displays interactive
visual CAPTCHA challenge
Display error: Password does not
meet security requirements
User enters CAPTCHA code
Is CAPTCHA answer
correct?
Yes
No
(A5)
Display error: Incorrect captcha;
reload new challenge

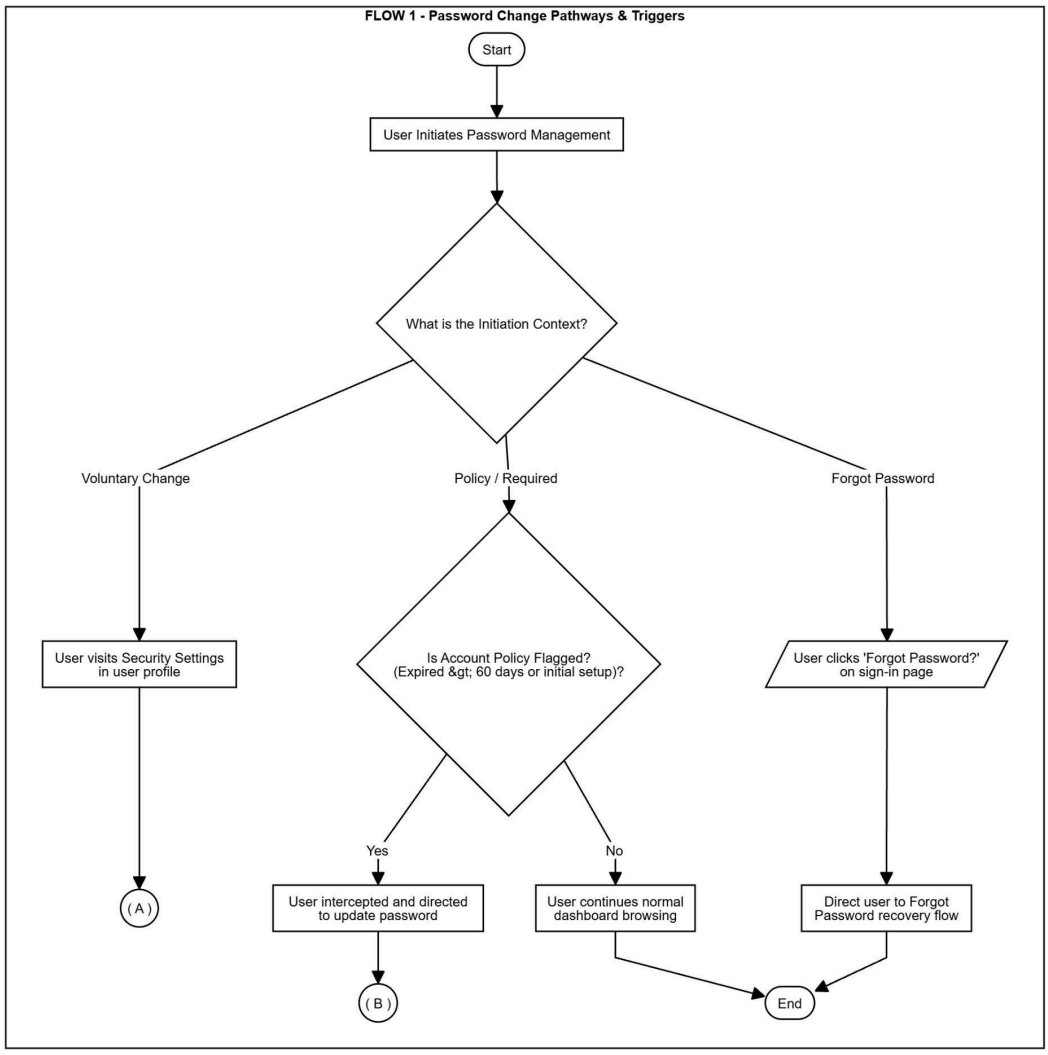

FLOW 2E - Account Creation & Login Redirect
(A5)
User submits completed registration wizard
System creates user account record (role: user, email verified)
System creates user profile with validated names and phone
System records initial password history hash
System logs registration audit trail entry
Display success confirmation: Account registered successfully!
Redirect user to login page /login
End

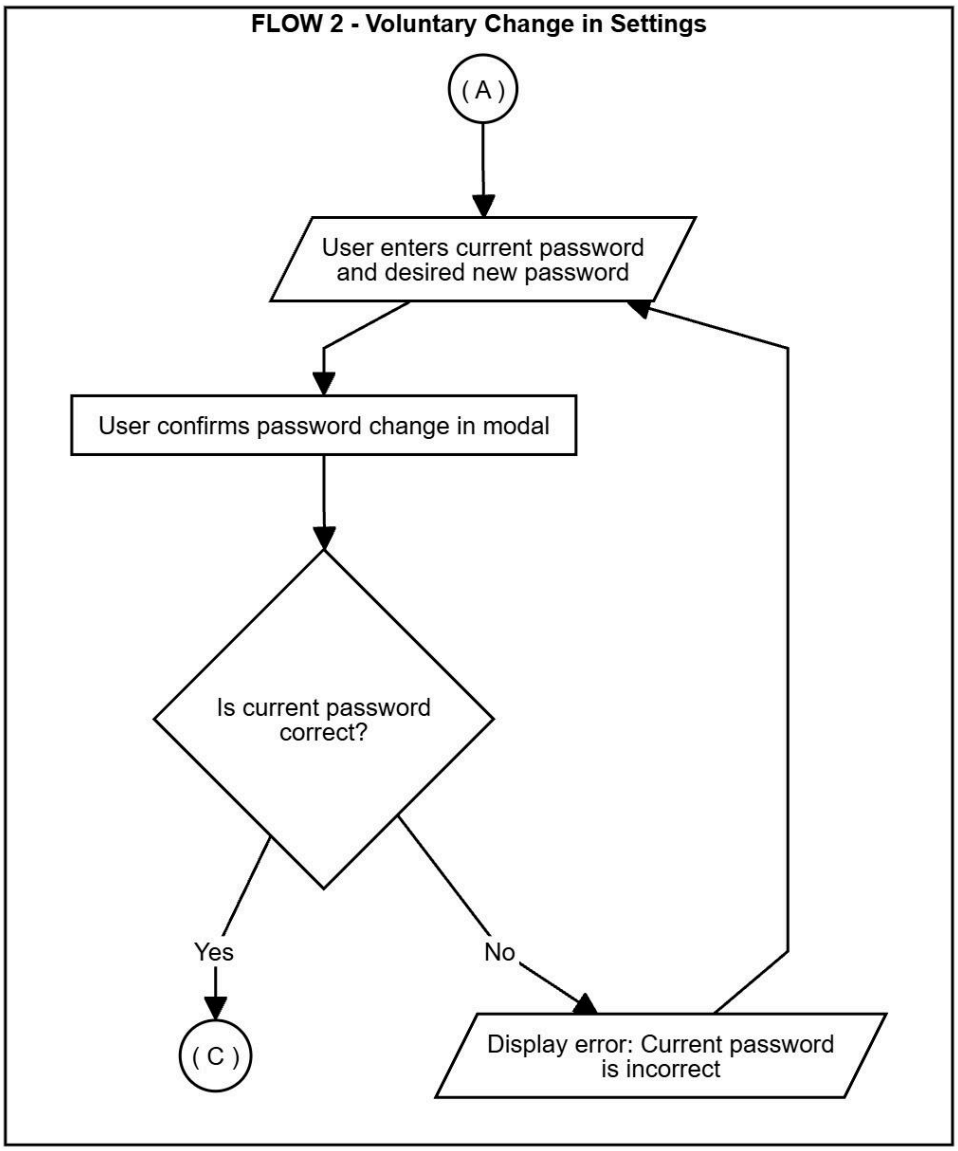

FLOW 3A - Google Consent & Token Exchange
(B)
User reviews and confirms Google consent screen
Did Google authorization succeed?
Yes
No
System exchanges authorization code for verified Google profile
Display alert: Google Sign-Up was cancelled or failed
(B2)
Return to registration interface /register
End

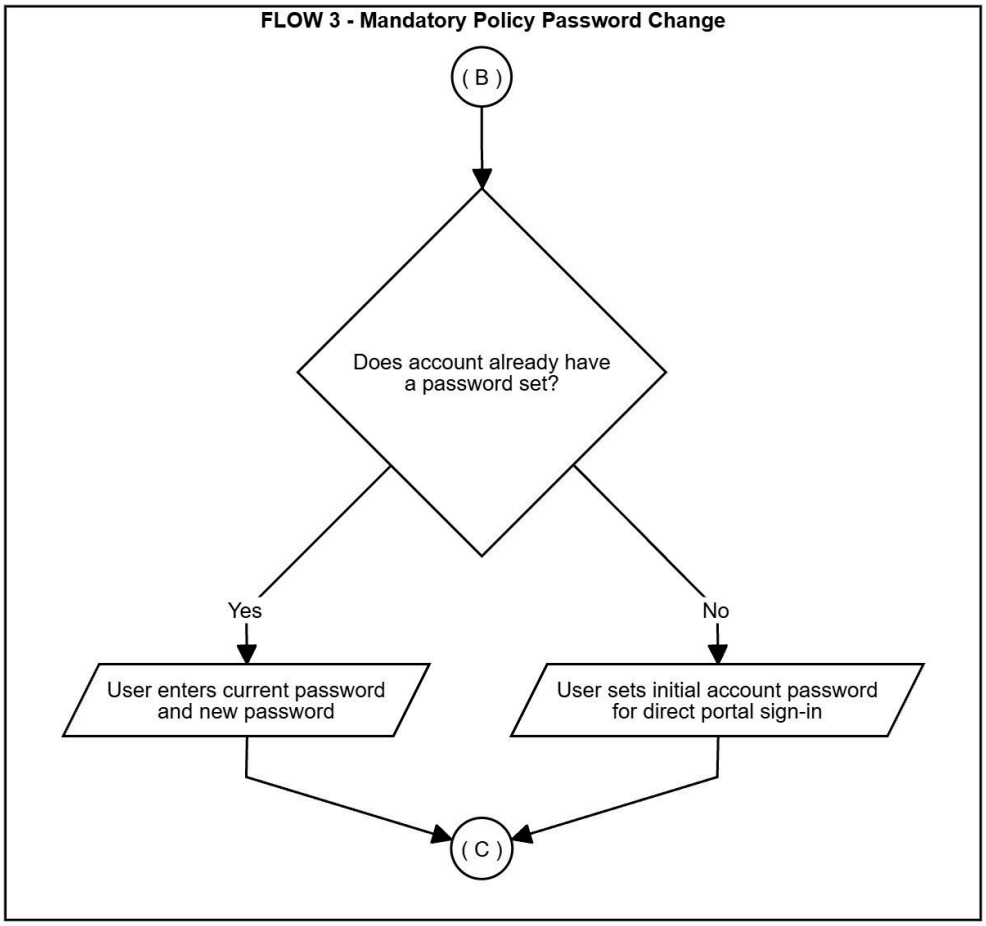

FLOW 3B - Google Account Matching & Profile Provisioning
(B2)
Does account exist with this Google ID or email?
No
Yes
System creates new user account (role: user, email pre-verified)
System creates profile with first name and last name
Flag account to set initial password for direct sign-in
System connects Google login to existing profile
(B3)

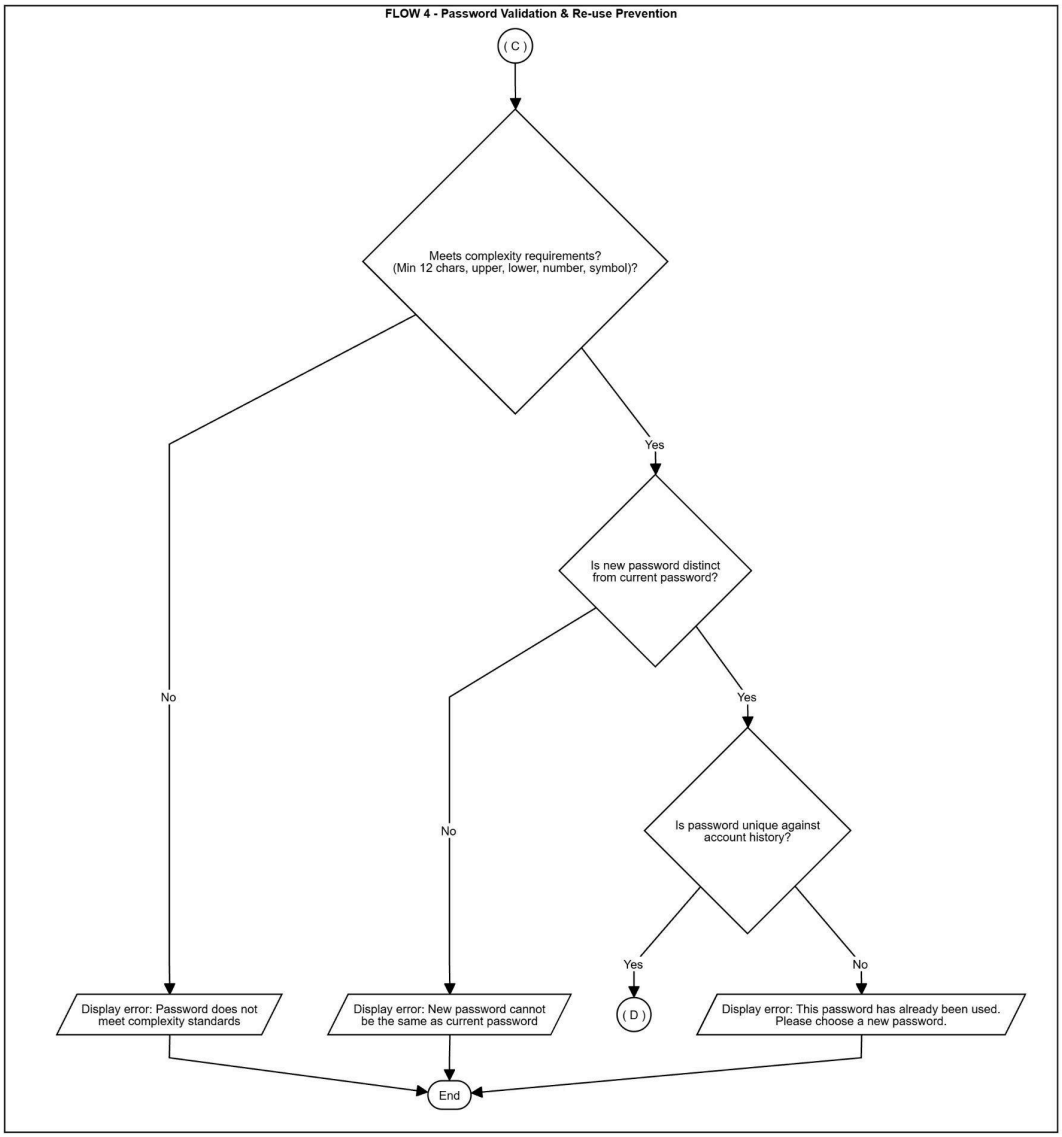

FLOW 3C - Google Status Verification & Password Gate
(B3)
Is user account active?
No
Yes
Display notice: Account deactivated; contact support
System initializes authenticated user session
Must set password for first-time Google sign-up?
No
Yes
Return to registration interface /register
Direct user to Public User Dashboard
Redirect user to create initial password page
End

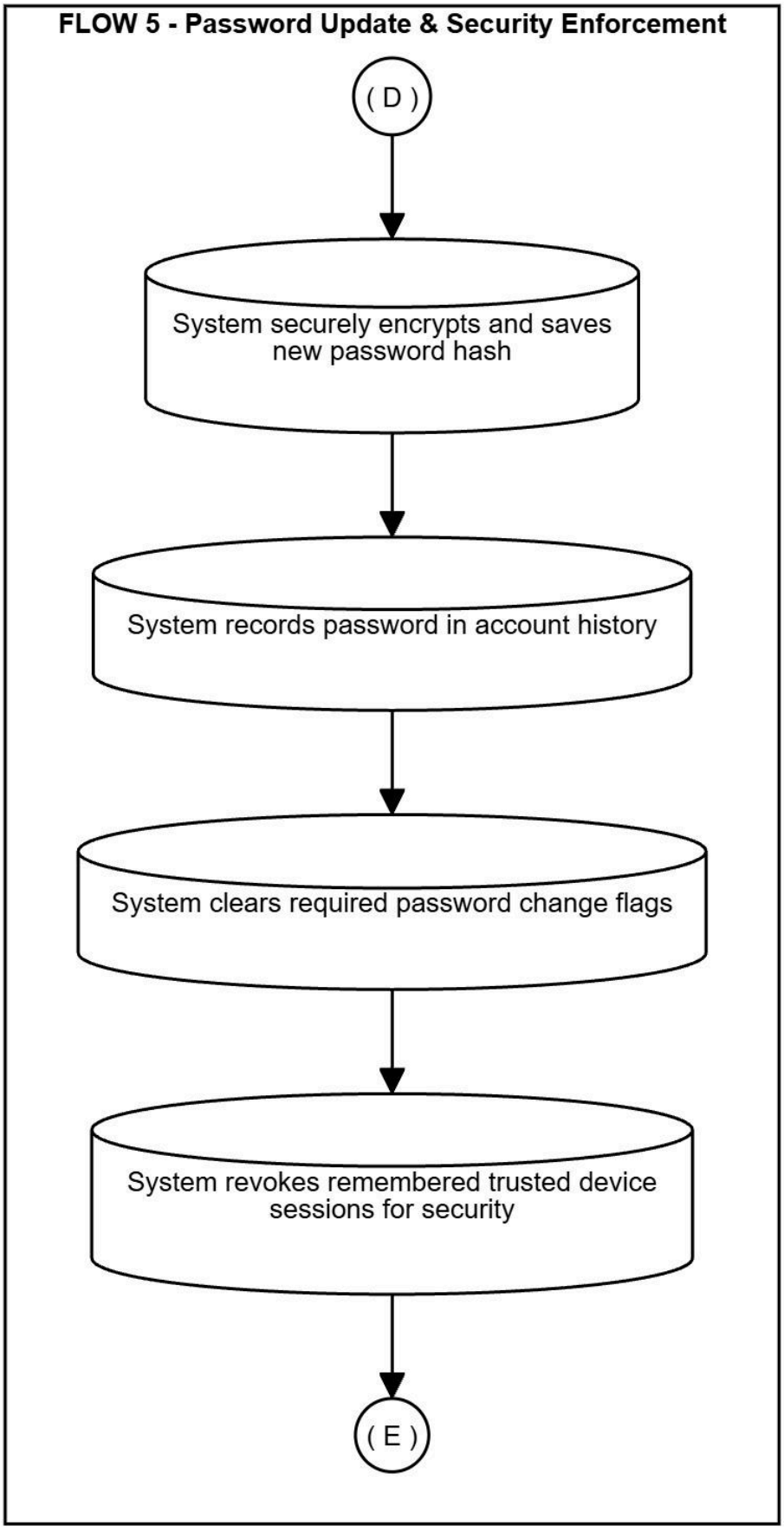
Flow 3: Change Password Flowchart

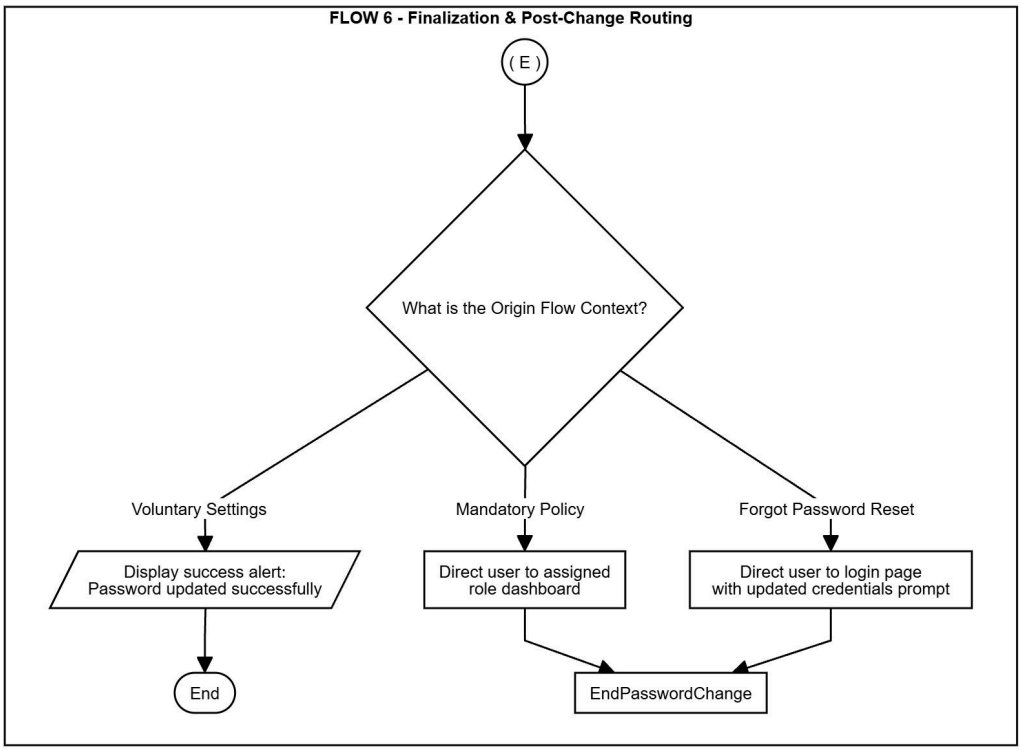



FLOW 2 - Voluntary Change in Settings
(A)
User enters current password and desired new password
User confirms password change in modal
Is current password correct?
Yes
No
(C)
Display error: Current password is incorrect

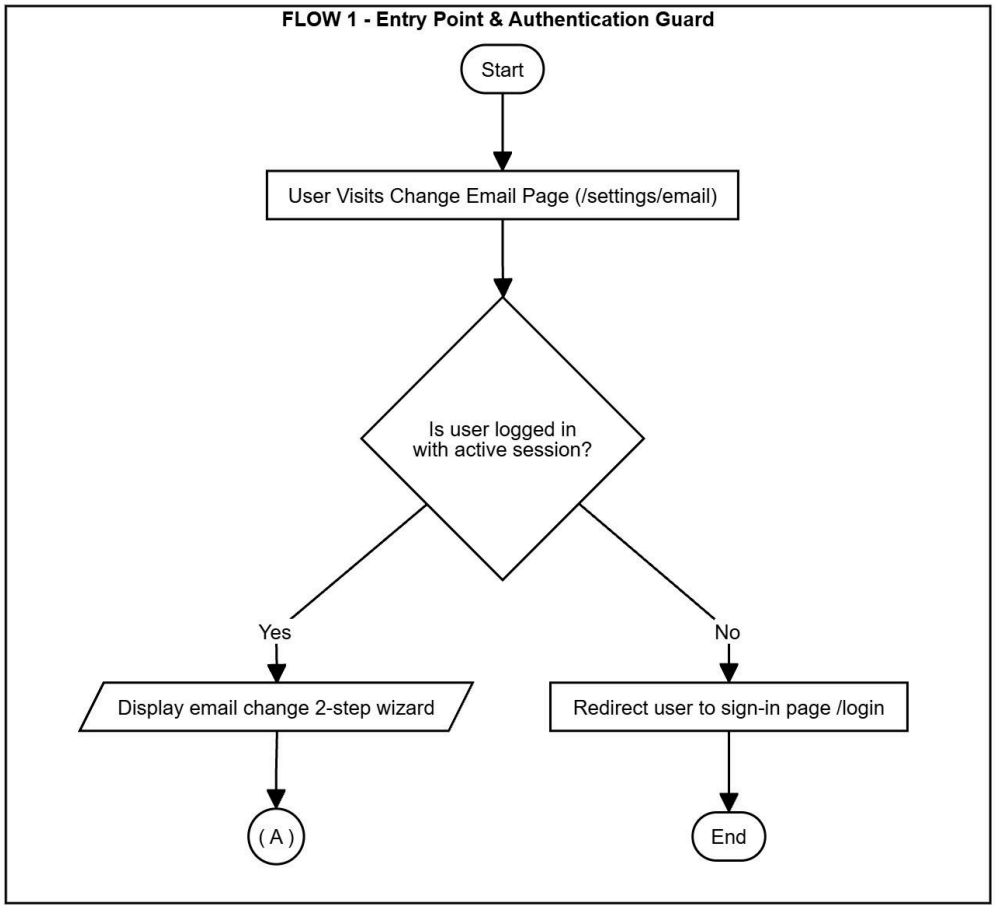

FLOW 3 - Mandatory Policy Password Change
(B)
Does account already have a password set?
Yes
No
User enters current password and new password
User sets initial account password for direct portal sign-in
(C)

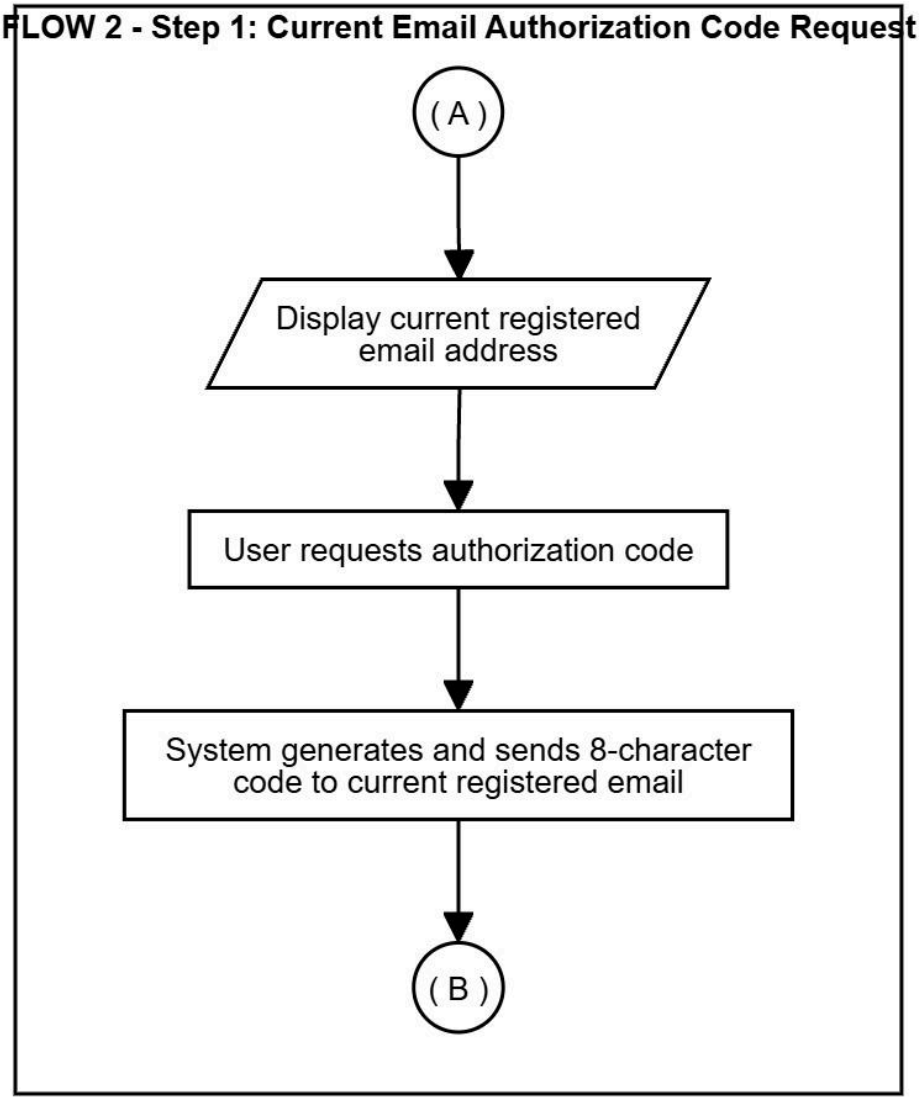

FLOW 4 - Password Validation & Re-use Prevention
(C)
Meets complexity requirements?
(Min 12 chars, upper, lower, number, symbol)?
No
Yes
Is new password distinct
from current password?
No
Yes
Is password unique against
account history?
Yes
No
Display error: Password does not
meet complexity standards
Display error: New password cannot
be the same as current password
(D)
Display error: This password has already been used.
Please choose a new password.
End

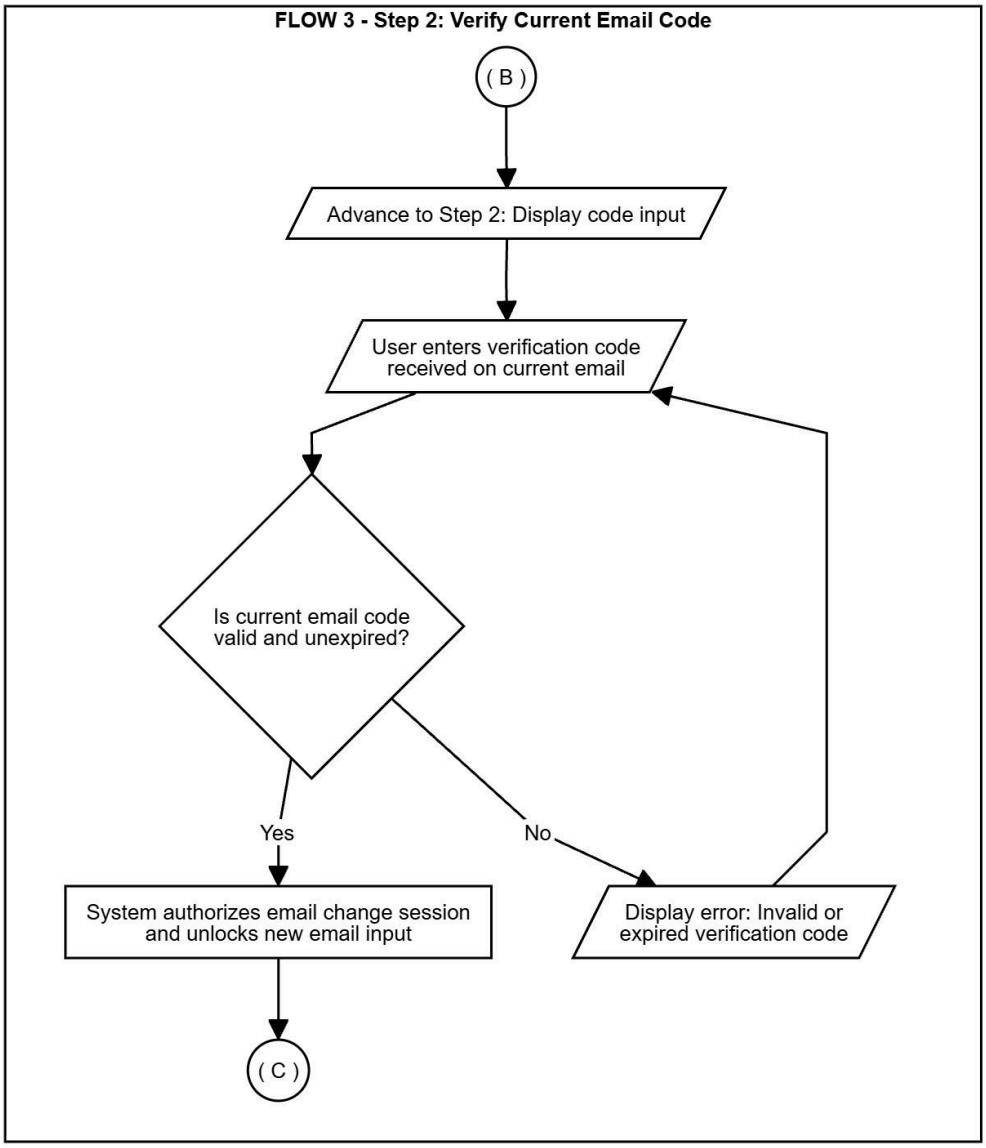

FLOW 5 - Password Update & Security Enforcement
(D)
System securely encrypts and saves new password hash
System records password in account history
System clears required password change flags
System revokes remembered trusted device sessions for security
(E)

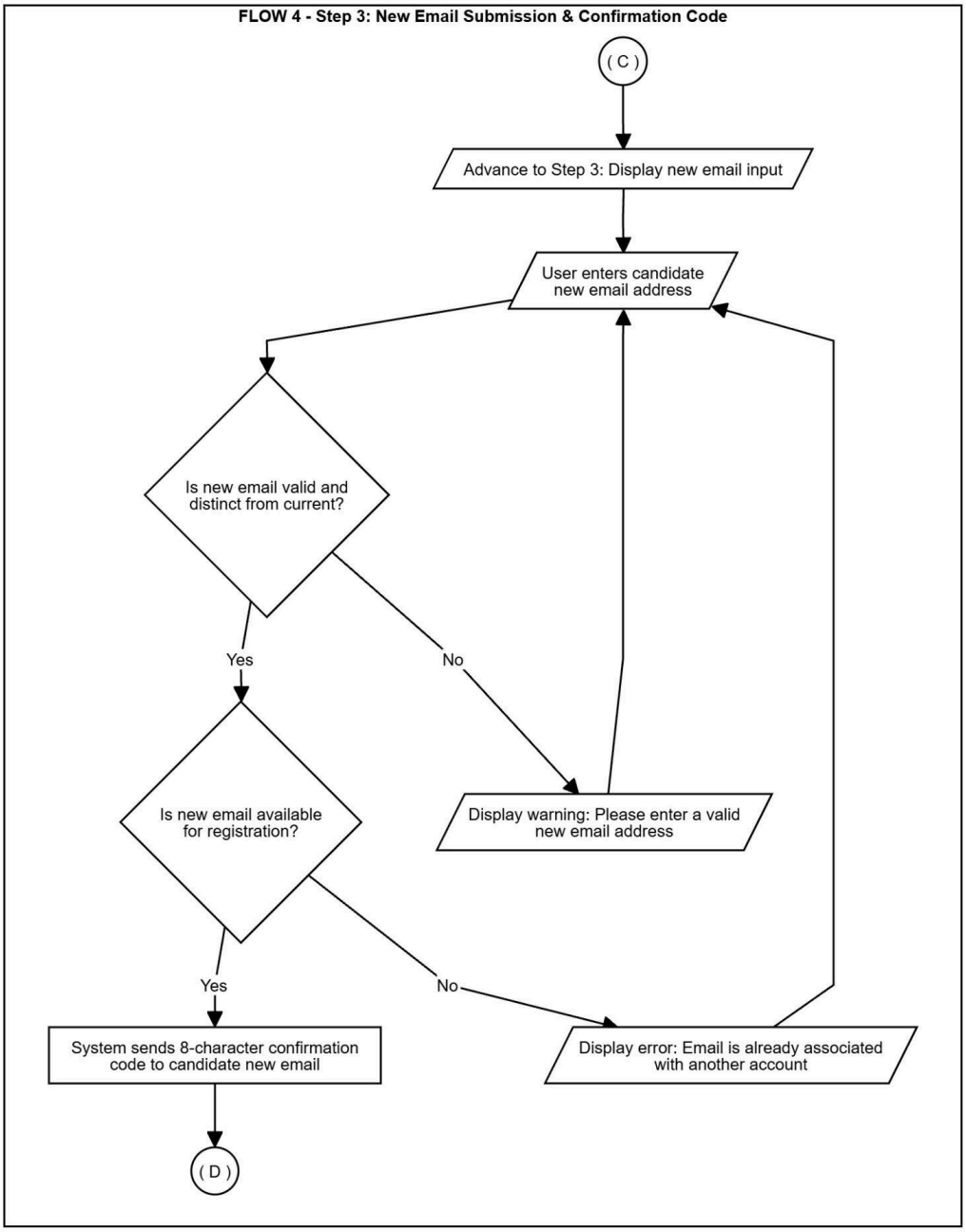

FLOW 6 - Finalization & Post-Change Routing
(E)
What is the Origin Flow Context?
Voluntary Settings
Mandatory Policy
Forgot Password Reset
Display success alert:
Password updated successfully
Direct user to assigned
role dashboard
Direct user to login page
with updated credentials prompt
End
EndPasswordChange

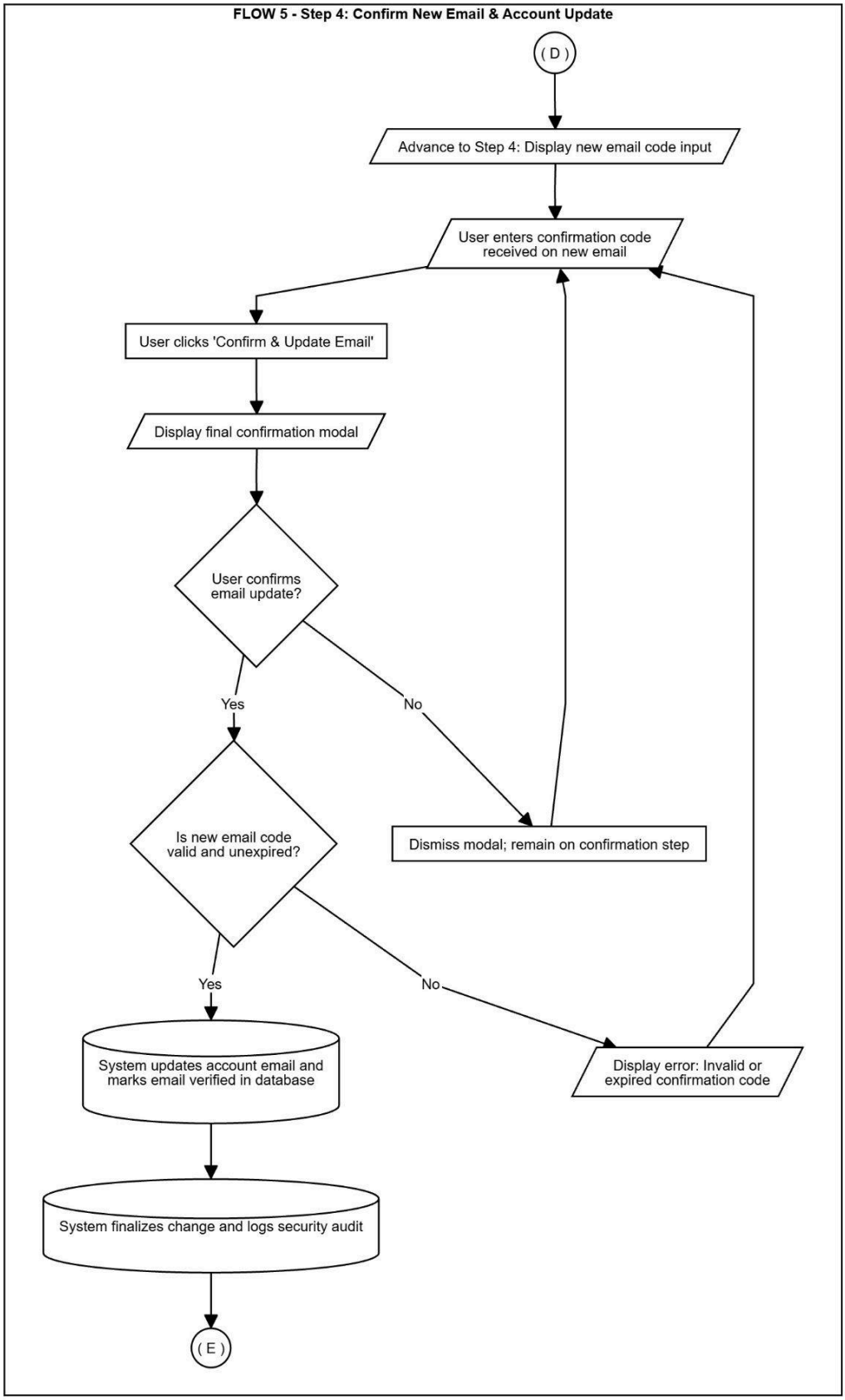
Flow 4: Change Email Flowchart

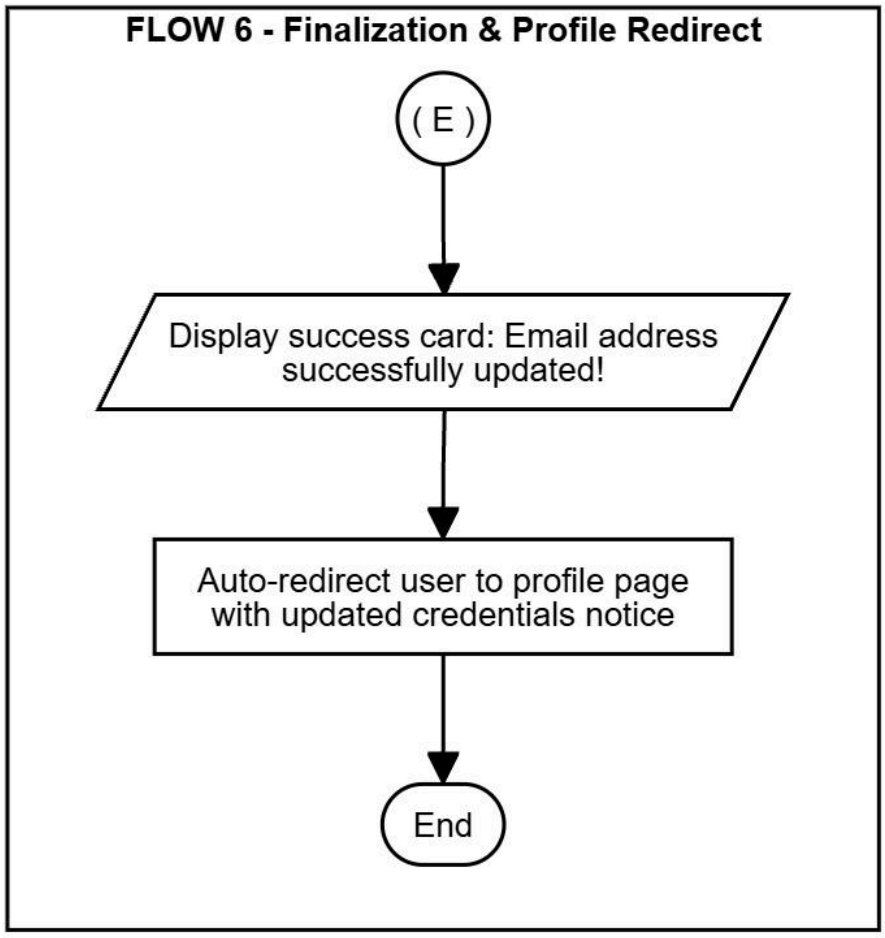



FLOW 2 - Step 1: Current Email Authorization Code Request
(A)
Display current registered email address
User requests authorization code
System generates and sends 8-character code to current registered email
(B)

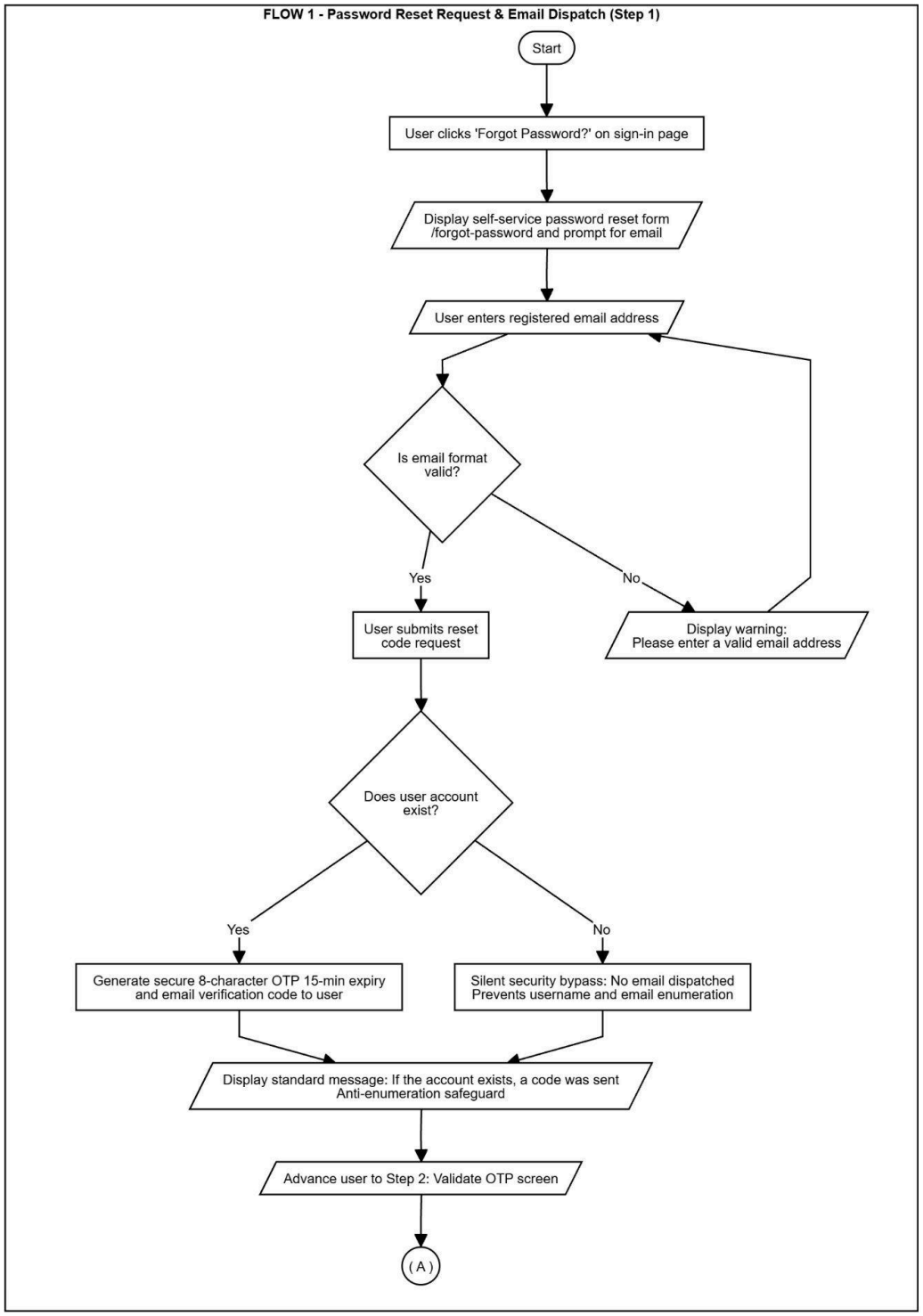

FLOW 3 - Step 2: Verify Current Email Code
(B)
Advance to Step 2: Display code input
User enters verification code received on current email
Is current email code valid and unexpired?
Yes
No
System authorizes email change session and unlocks new email input
Display error: Invalid or expired verification code
(C)

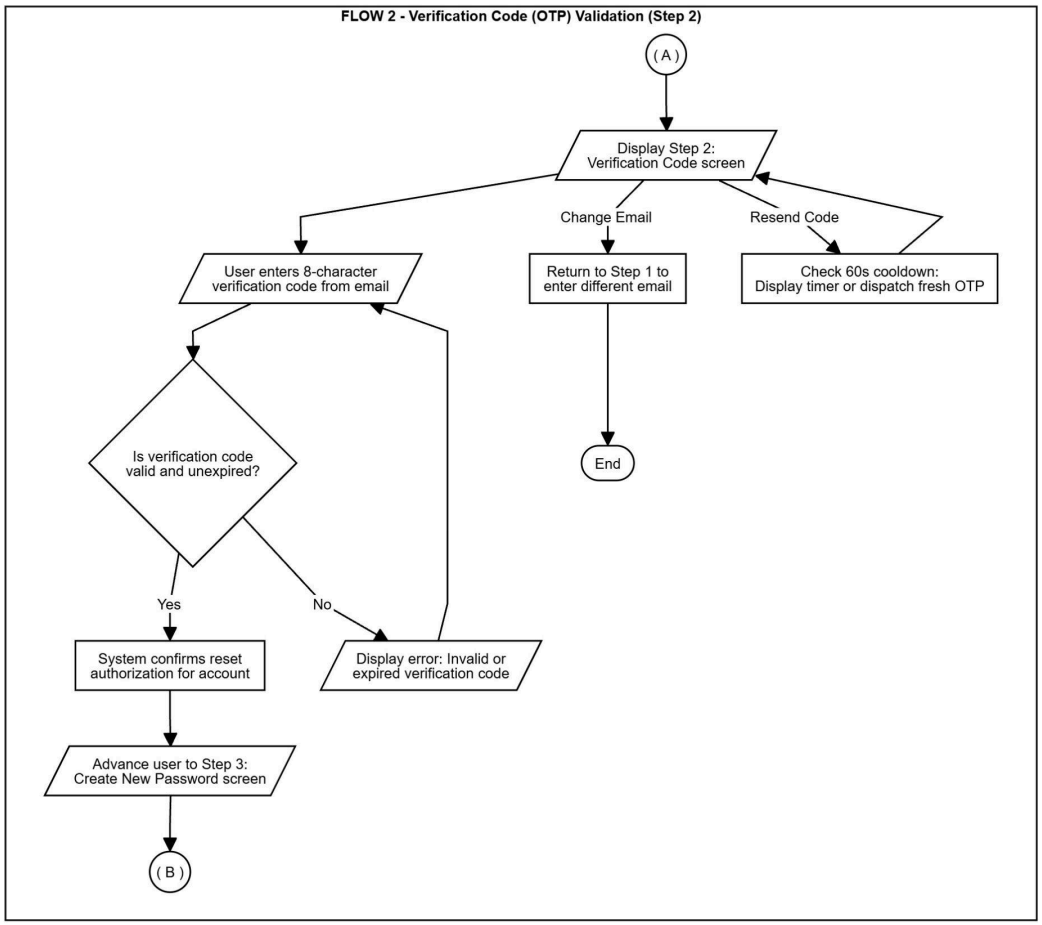

FLOW 4 - Step 3: New Email Submission & Confirmation Code
(C)
Advance to Step 3: Display new email input
User enters candidate new email address
Is new email valid and distinct from current?
Yes
No
Is new email available for registration?
Display warning: Please enter a valid new email address
Yes
No
System sends 8-character confirmation code to candidate new email
Display error: Email is already associated with another account
(D)

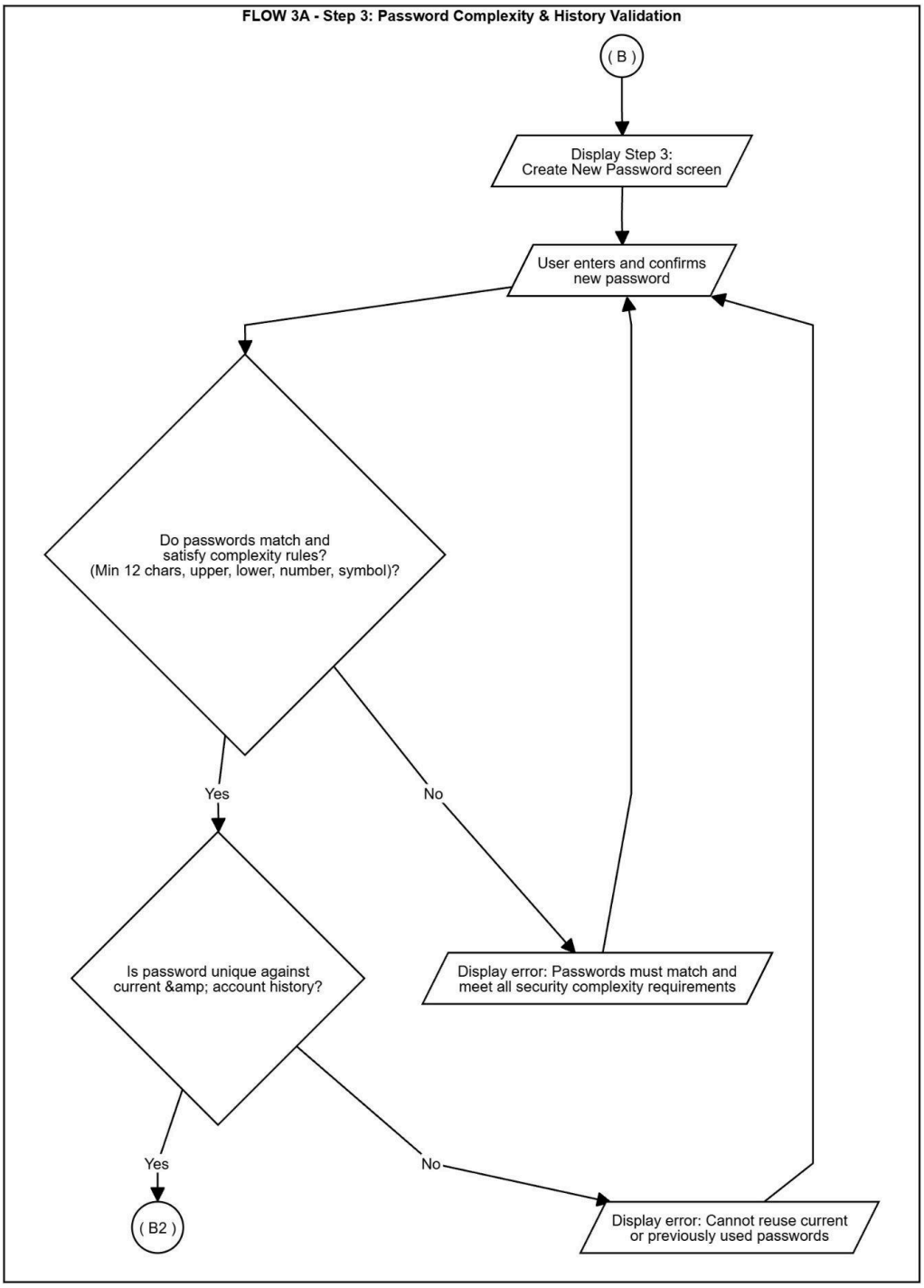

FLOW 5 - Step 4: Confirm New Email & Account Update
(D)
Advance to Step 4: Display new email code input
User enters confirmation code received on new email
User clicks 'Confirm & Update Email'
Display final confirmation modal
User confirms email update?
Yes
No
Is new email code valid and unexpired?
Dismiss modal; remain on confirmation step
Yes
No
System updates account email and marks email verified in database
Display error: Invalid or expired confirmation code
System finalizes change and logs security audit
(E)

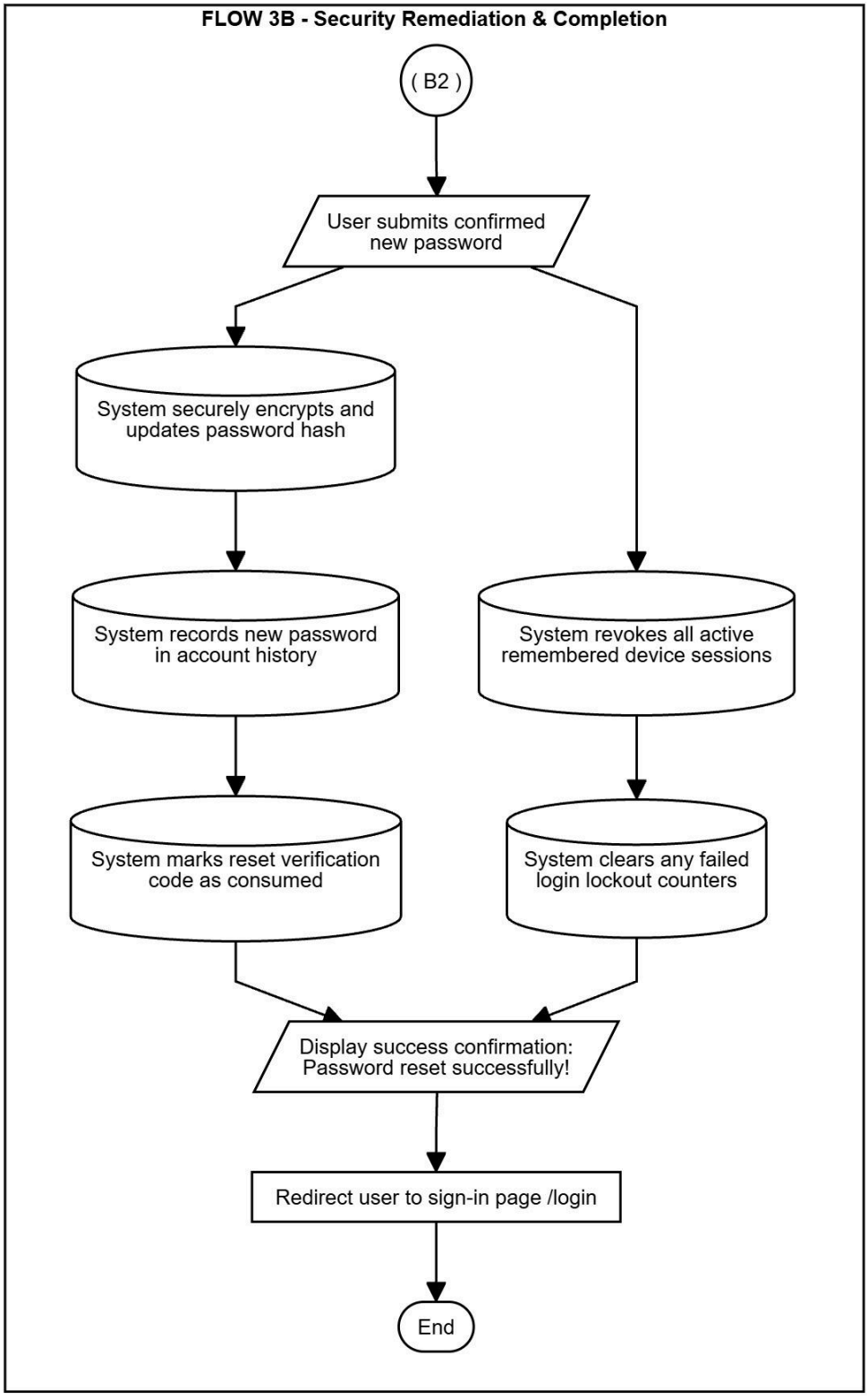

FLOW 6 - Finalization & Profile Redirect
(E)
Display success card: Email address successfully updated!
Auto-redirect user to profile page with updated credentials notice
End

Flow 5: Forgot Password Flowchart

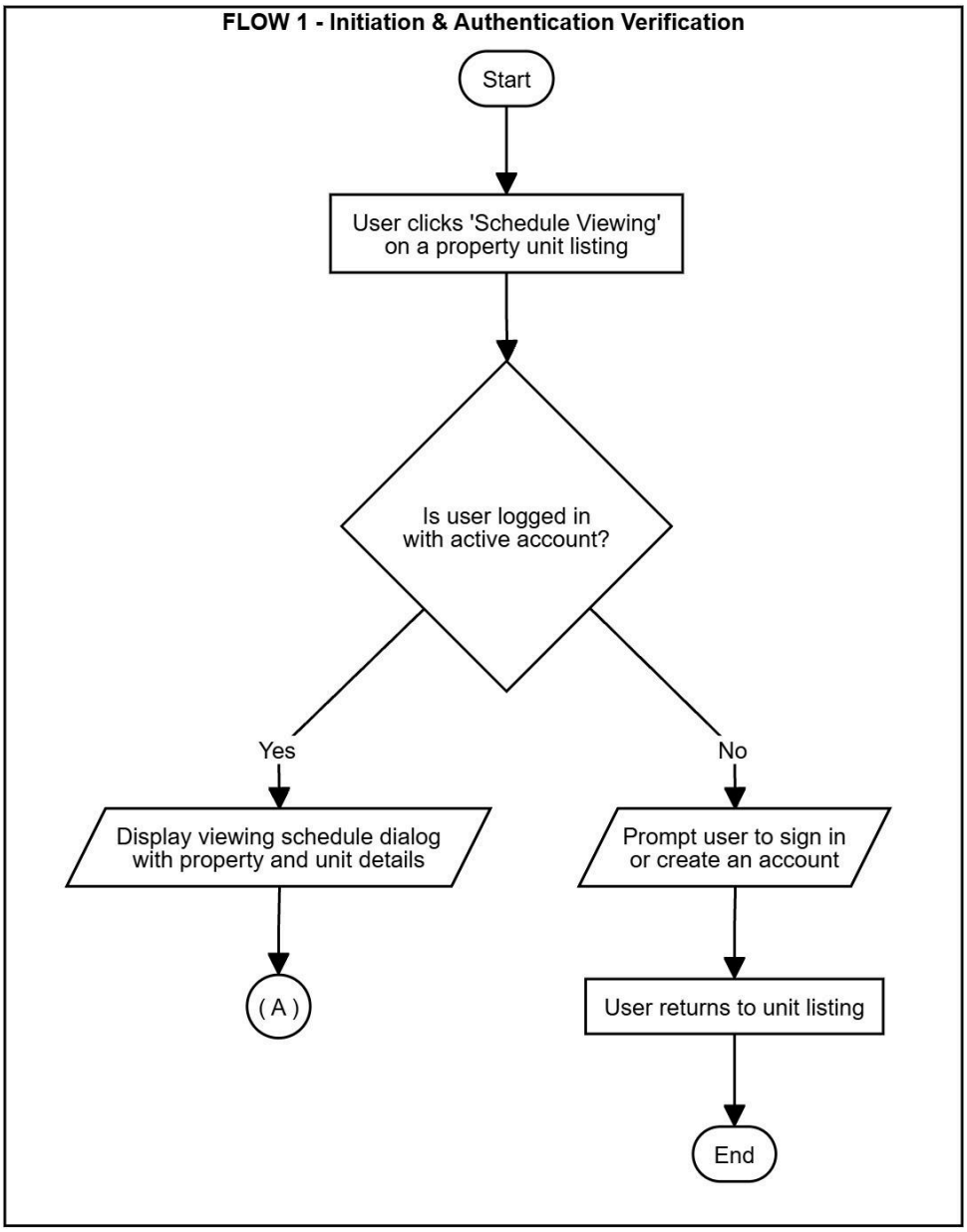


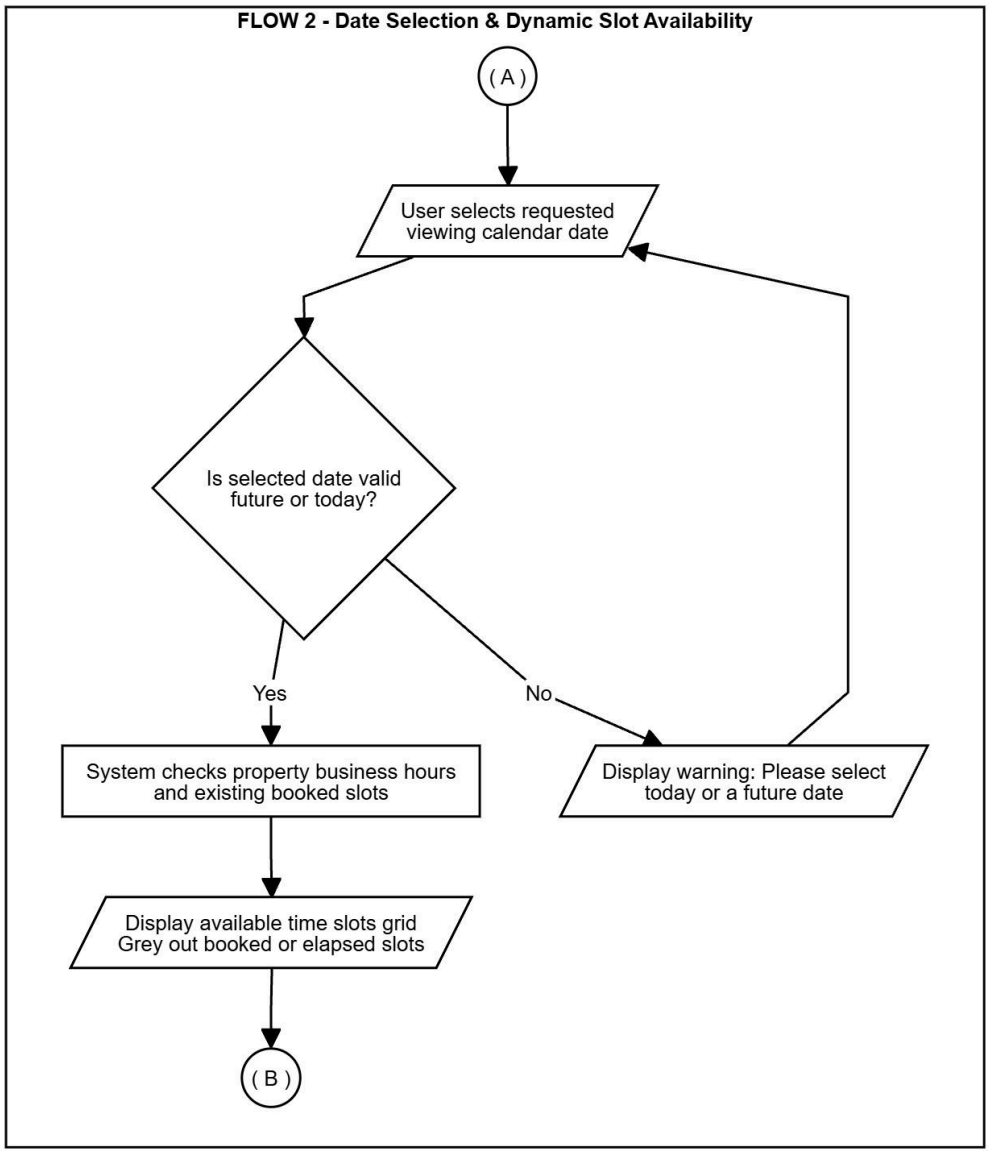

FLOW 2 - Verification Code (OTP) Validation (Step 2)
(A)
Display Step 2:
Verification Code screen
Change Email
Resend Code
User enters 8-character
verification code from email
Return to Step 1 to
enter different email
Check 60s cooldown:
Display timer or dispatch fresh OTP
End
Is verification code
valid and unexpired?
Yes
No
System confirms reset
authorization for account
Display error: Invalid or
expired verification code
Advance user to Step 3:
Create New Password screen
(B)

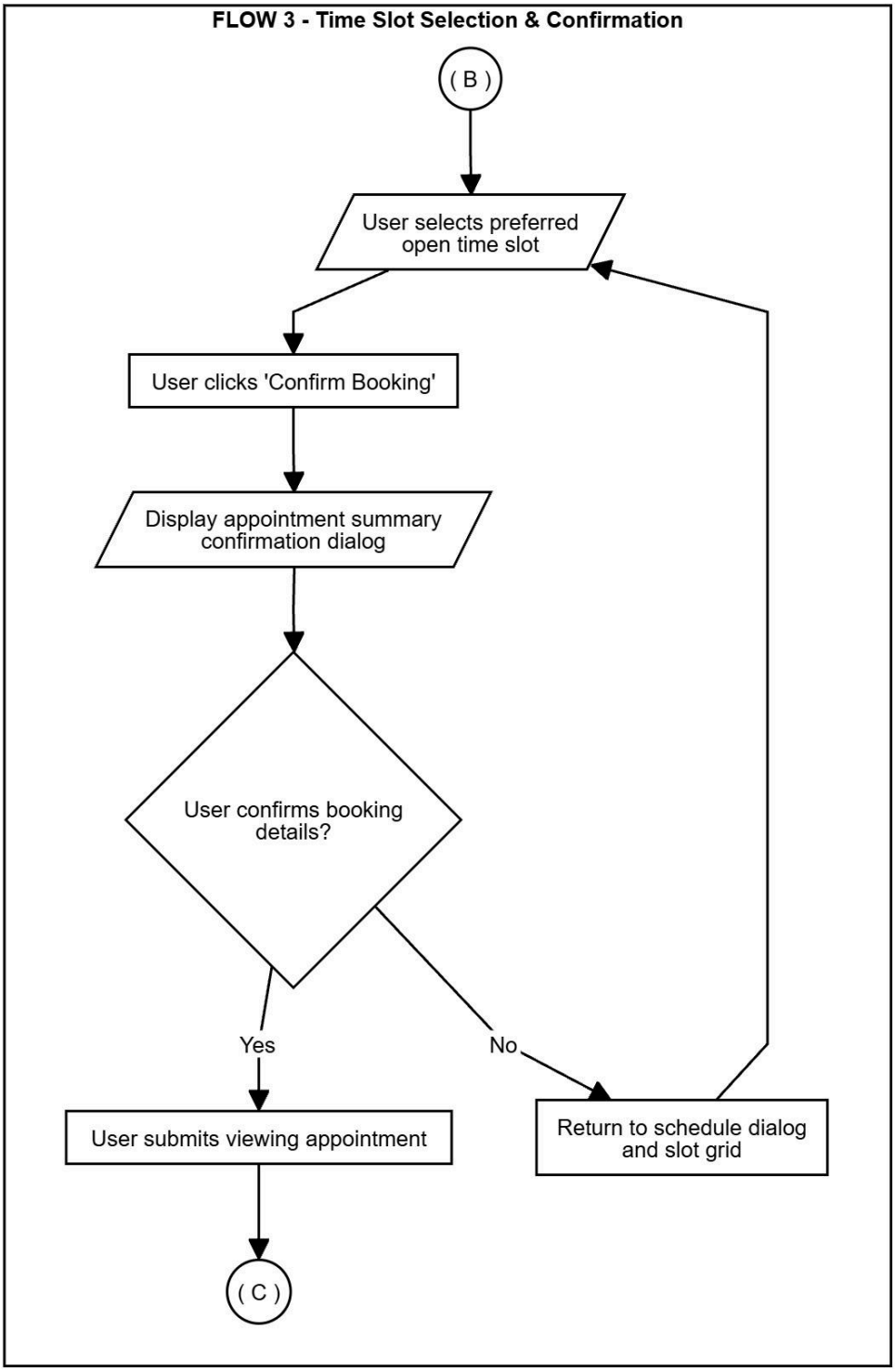

FLOW 3A - Step 3: Password Complexity & History Validation
(B)
Display Step 3:
Create New Password screen
User enters and confirms
new password
Do passwords match and
satisfy complexity rules?
(Min 12 chars, upper, lower, number, symbol)?
Yes
No
Is password unique against
current & account history?
Display error: Passwords must match and
meet all security complexity requirements
Yes
No
(B2)
Display error: Cannot reuse current
or previously used passwords

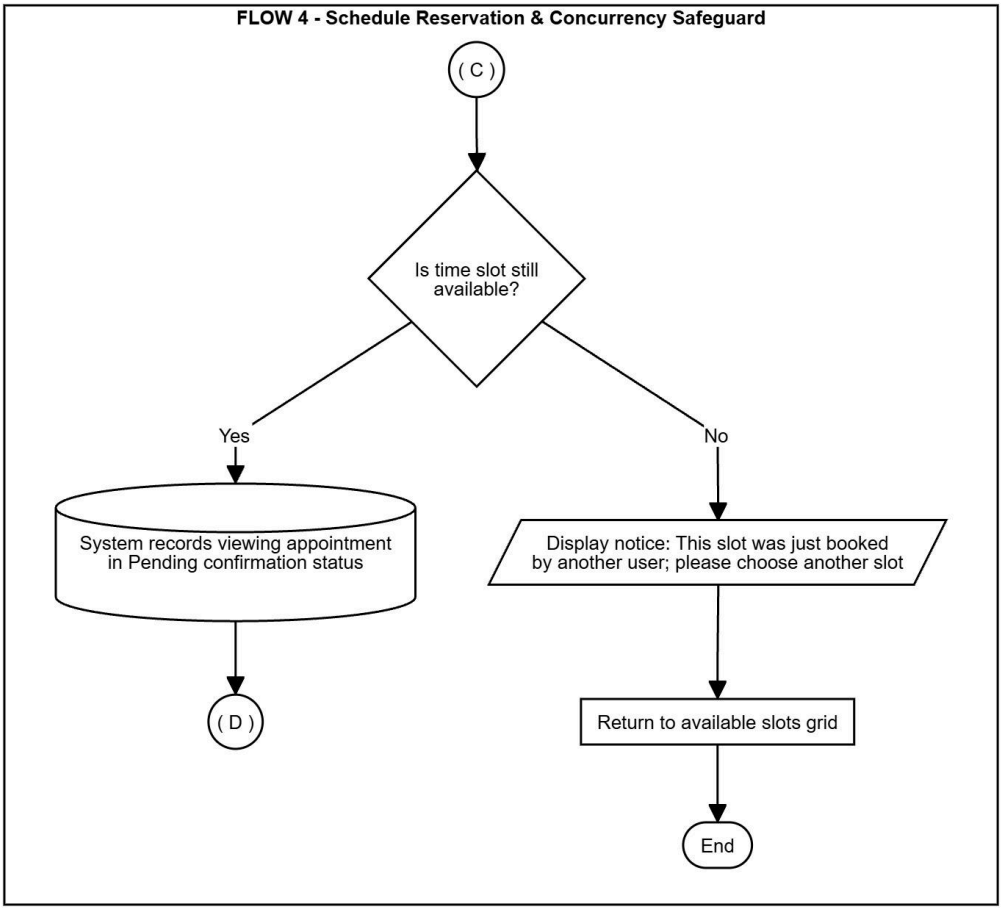

FLOW 3B - Security Remediation & Completion
(B2)
User submits confirmed new password
System securely encrypts and updates password hash
System records new password in account history
System revokes all active remembered device sessions
System marks reset verification code as consumed
System clears any failed login lockout counters
Display success confirmation: Password reset successfully!
Redirect user to sign-in page /login
End

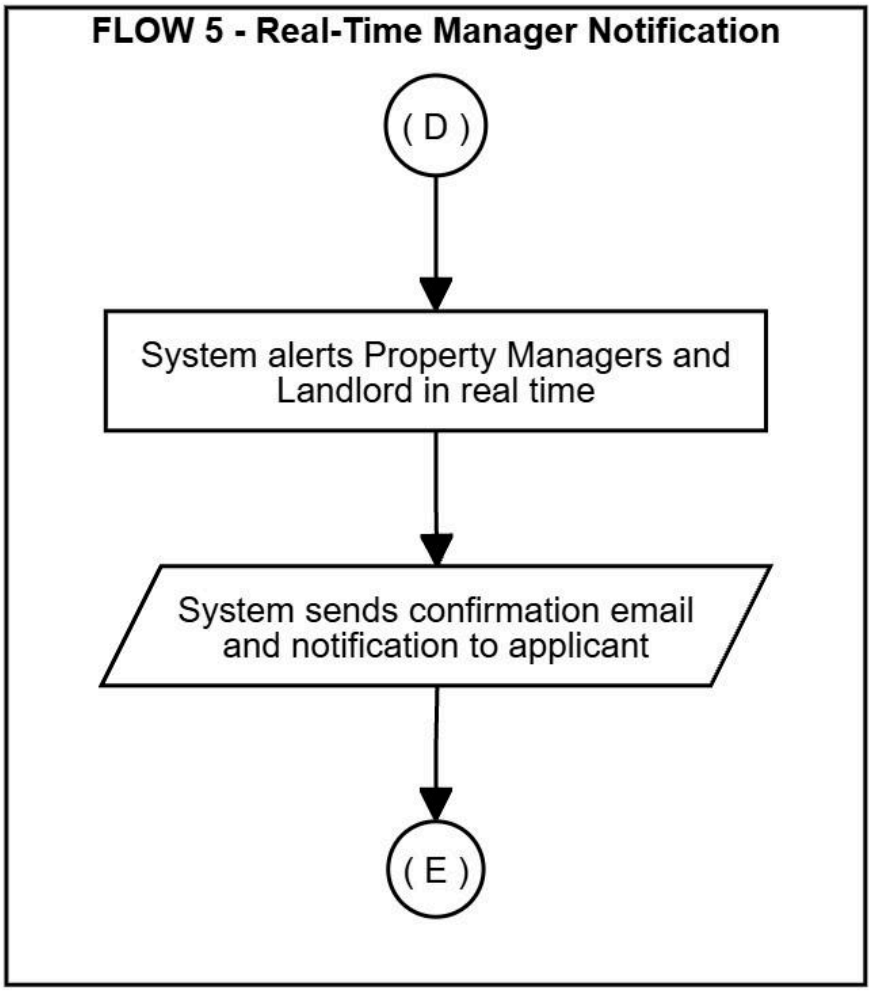
Flow 6: Apply Viewing Schedule Flowchart

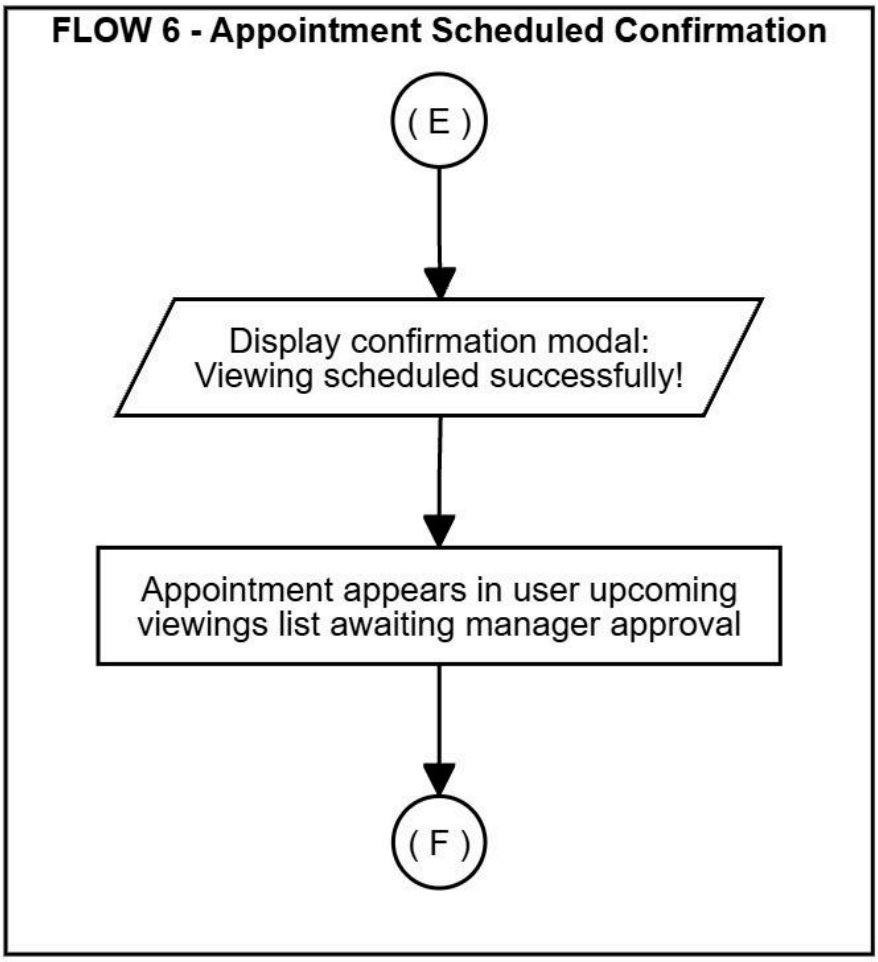


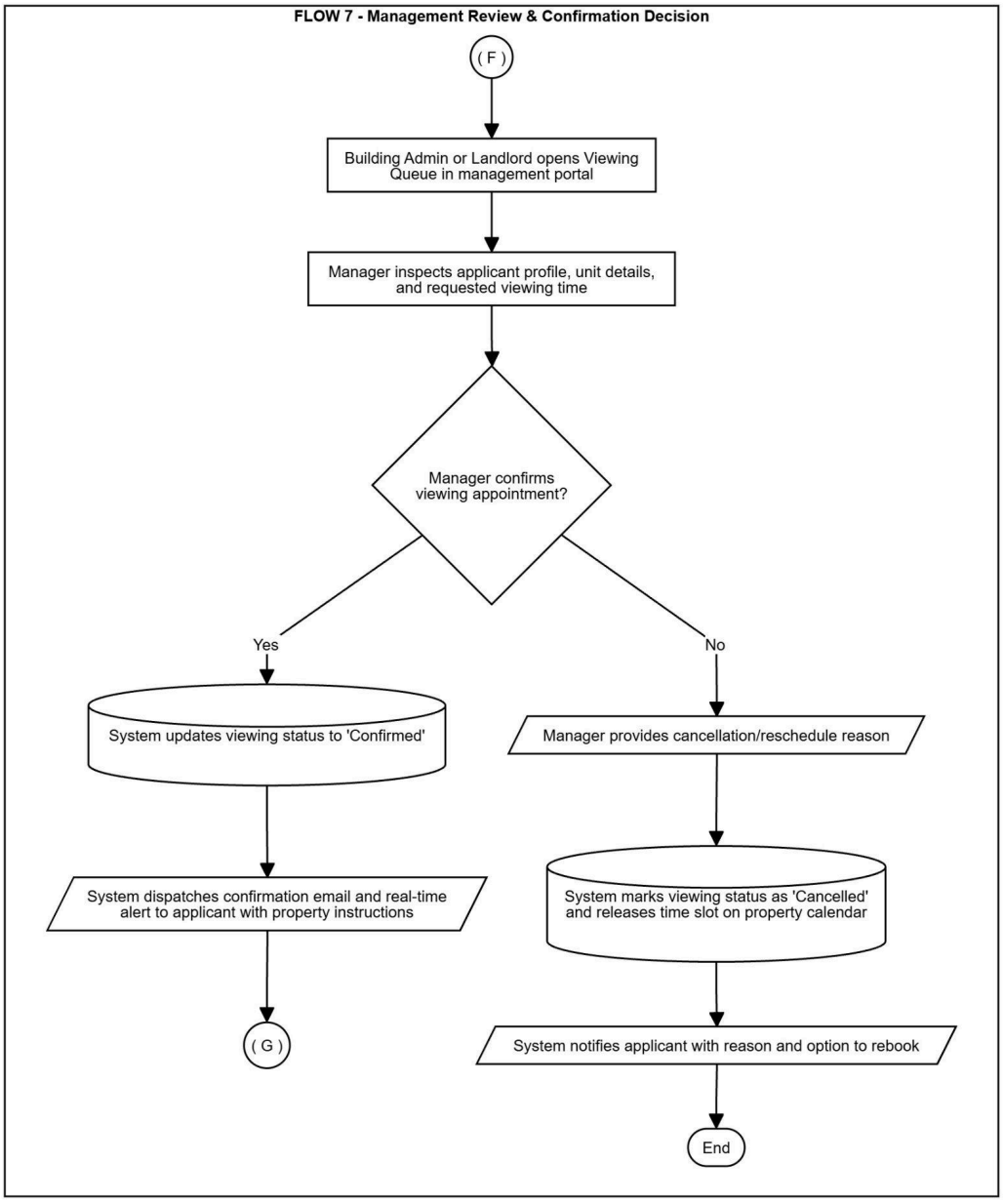

FLOW 2 - Date Selection & Dynamic Slot Availability
(A)
User selects requested viewing calendar date
Is selected date valid future or today?
Yes
No
System checks property business hours and existing booked slots
Display warning: Please select today or a future date
Display available time slots grid Grey out booked or elapsed slots
(B)

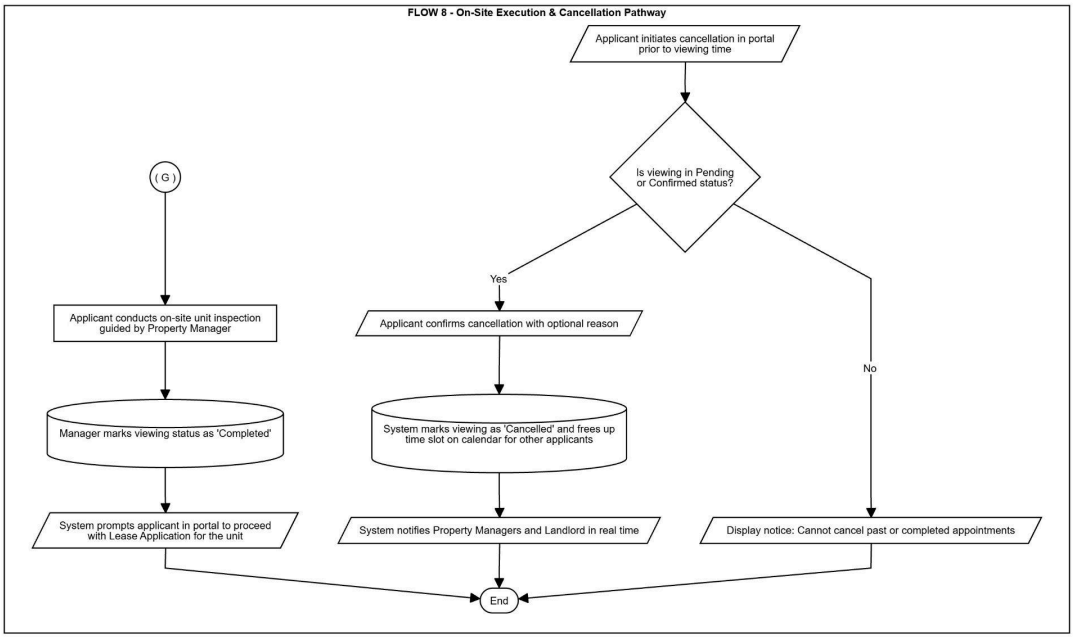

FLOW 3 - Time Slot Selection & Confirmation
(B)
User selects preferred open time slot
User clicks 'Confirm Booking'
Display appointment summary confirmation dialog
User confirms booking details?
Yes
No
User submits viewing appointment
Return to schedule dialog and slot grid
(C)


FLOW 4 - Schedule Reservation & Concurrency Safeguard
(C)
Is time slot still available?
Yes
No
System records viewing appointment in Pending confirmation status
Display notice: This slot was just booked by another user; please choose another slot
(D)
Return to available slots grid
End

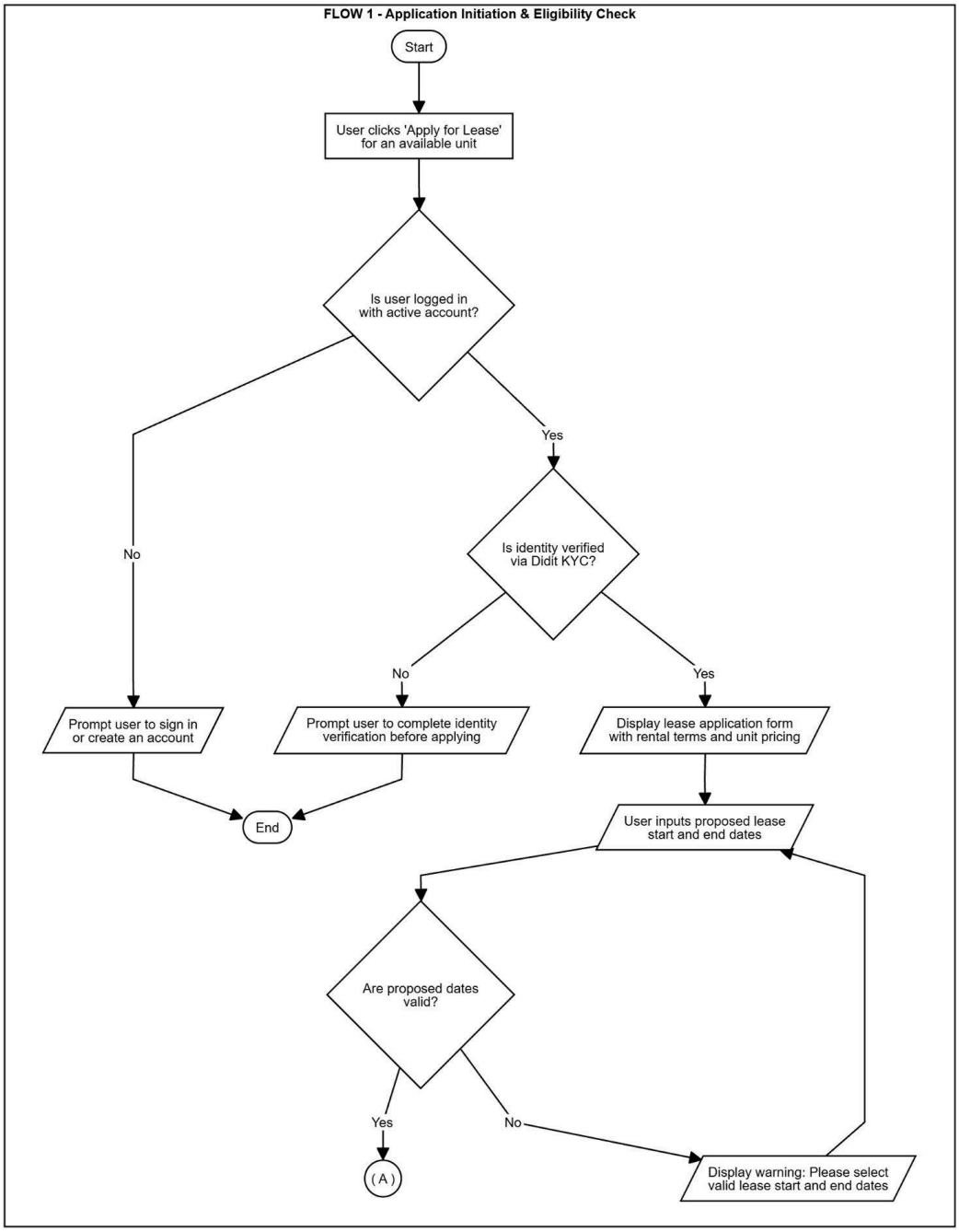

FLOW 5 - Real-Time Manager Notification
(D)
System alerts Property Managers and Landlord in real time
System sends confirmation email and notification to applicant
(E)

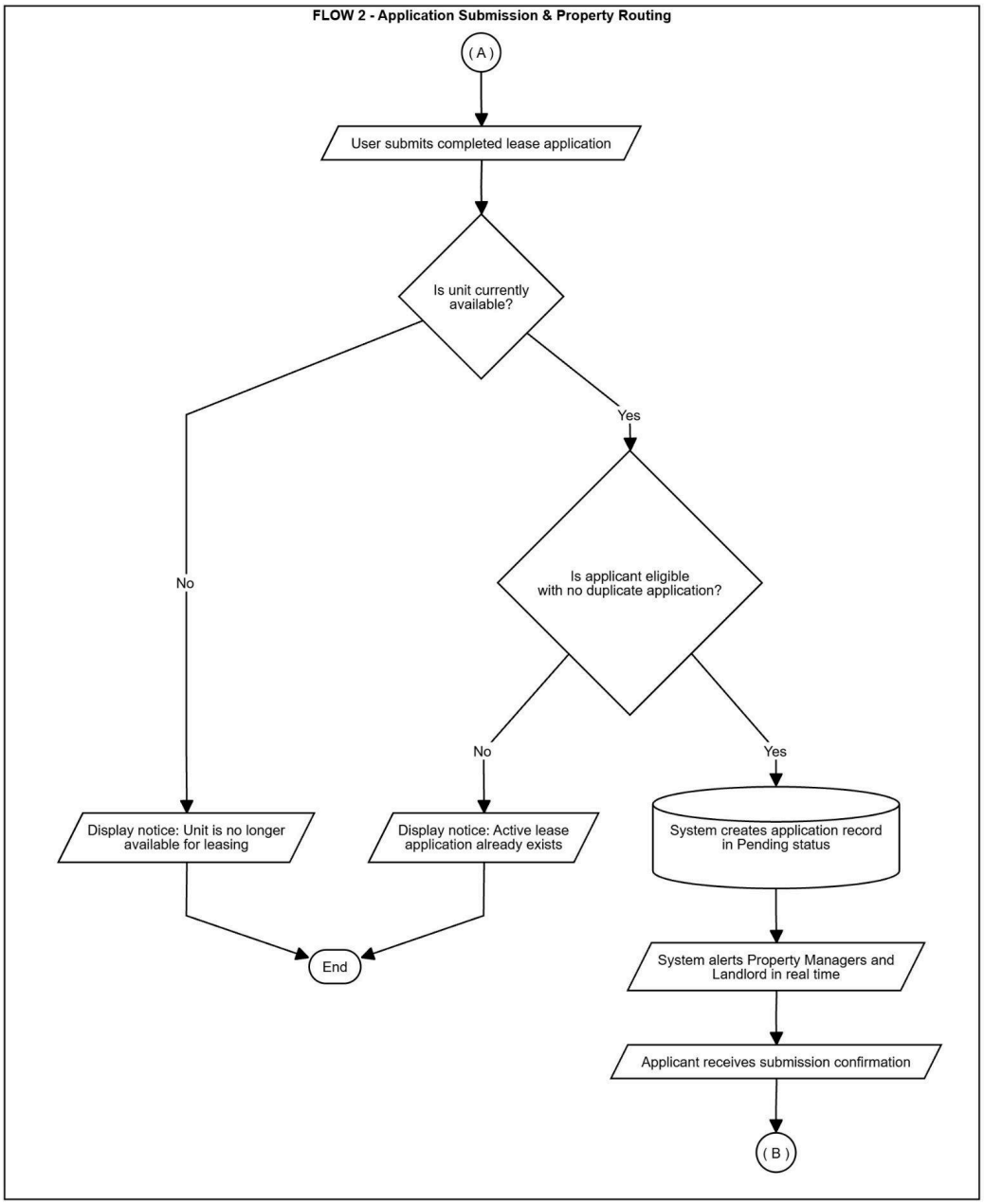

FLOW 6 - Appointment Scheduled Confirmation
(E)
Display confirmation modal:
Viewing scheduled successfully!
Appointment appears in user upcoming
viewings list awaiting manager approval
(F)

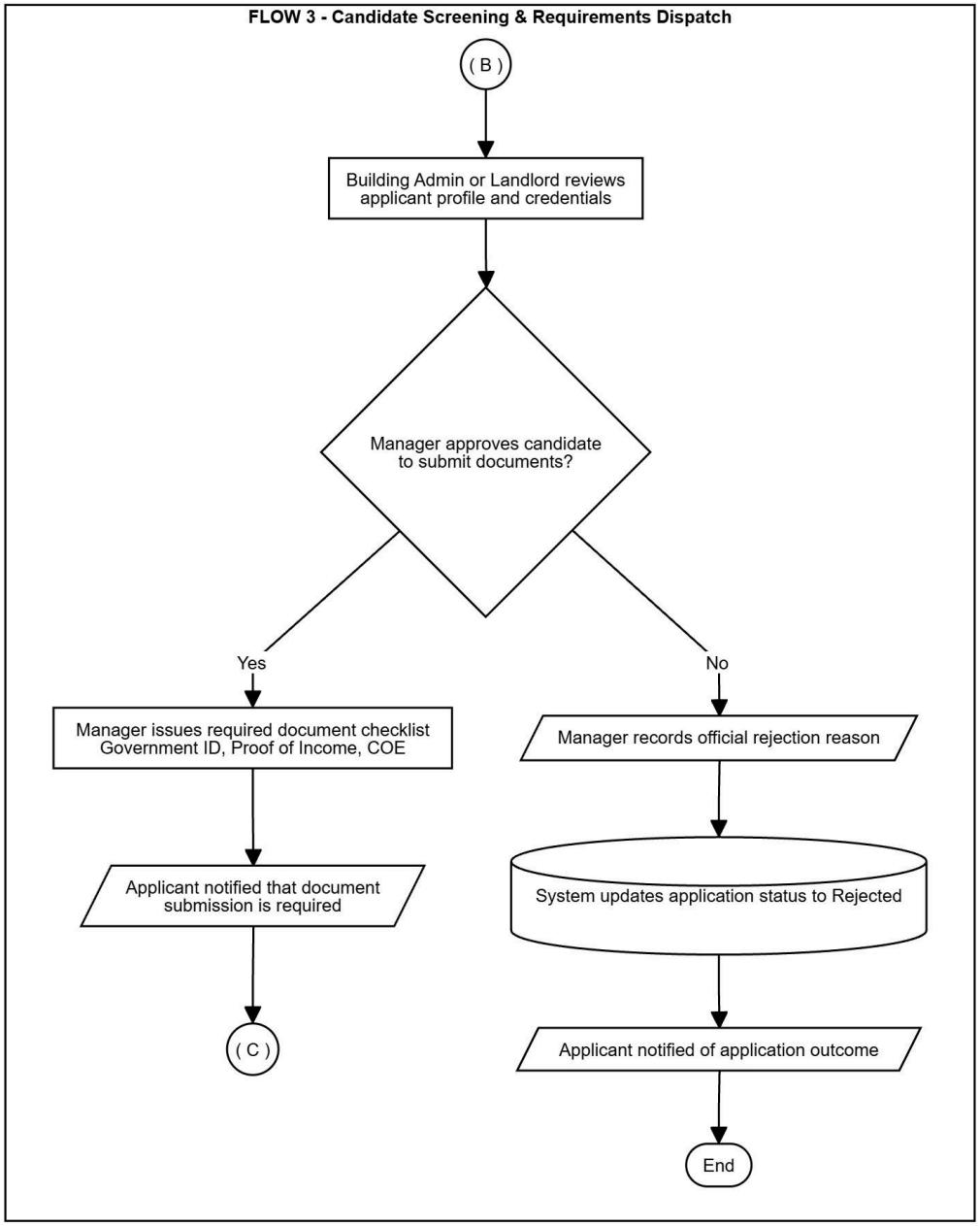

FLOW 7 - Management Review & Confirmation Decision
(F)
Building Admin or Landlord opens Viewing Queue in management portal
Manager inspects applicant profile, unit details, and requested viewing time
Manager confirms viewing appointment?
Yes
No
System updates viewing status to 'Confirmed'
Manager provides cancellation/reschedule reason
System dispatches confirmation email and real-time alert to applicant with property instructions
System marks viewing status as 'Cancelled' and releases time slot on property calendar
(G)
System notifies applicant with reason and option to rebook
End

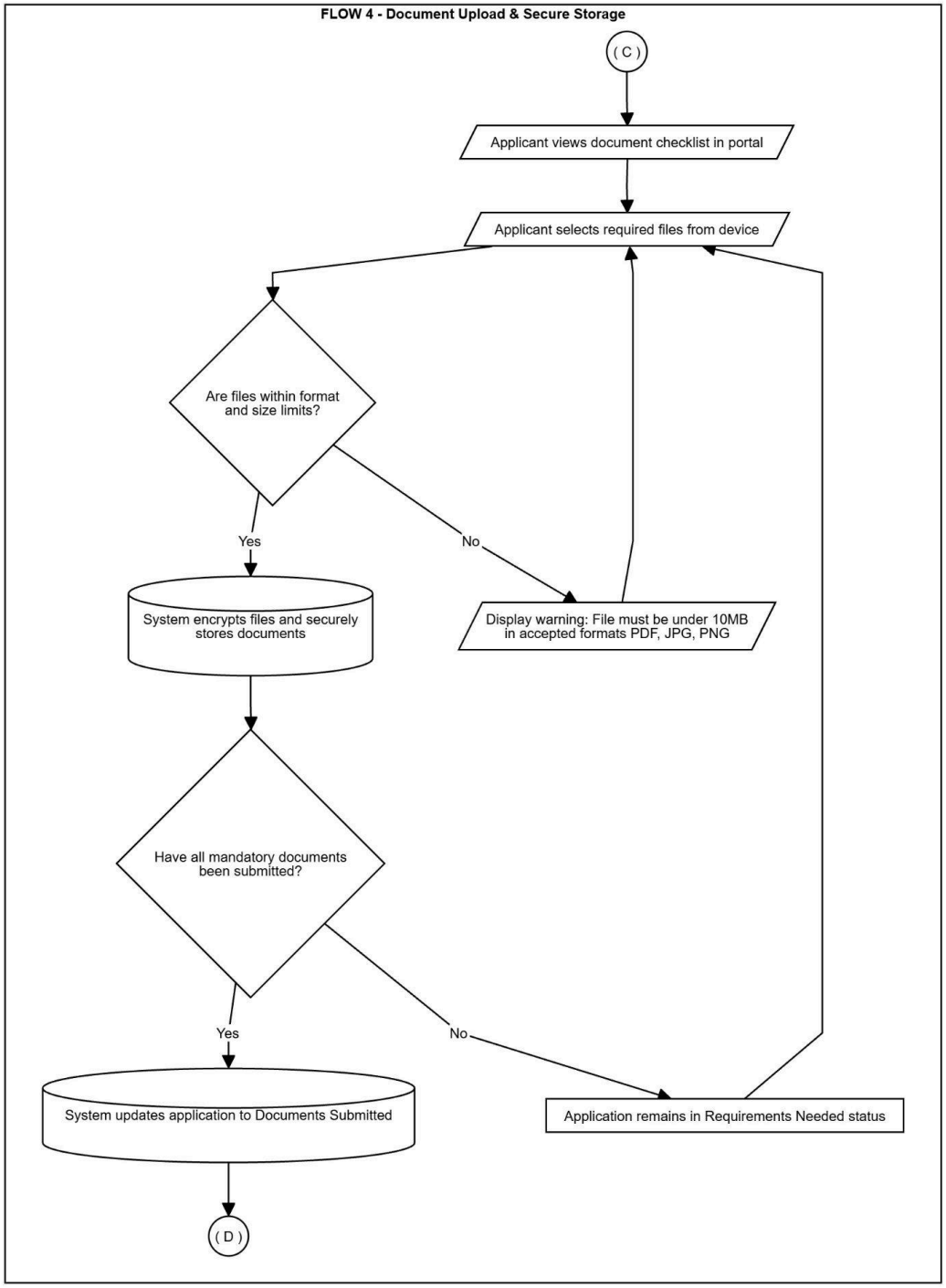

FLOW 8 - On-Site Execution & Cancellation Pathway
Applicant initiates cancellation in portal prior to viewing time
Is viewing in Pending or Confirmed status?
Yes
No
(G)
Applicant conducts on-site unit inspection guided by Property Manager
Applicant confirms cancellation with optional reason
Manager marks viewing status as 'Completed'
System marks viewing as 'Cancelled' and frees up time slot on calendar for other applicants
System prompts applicant in portal to proceed with Lease Application for the unit
System notifies Property Managers and Landlord in real time
Display notice: Cannot cancel past or completed appointments
End

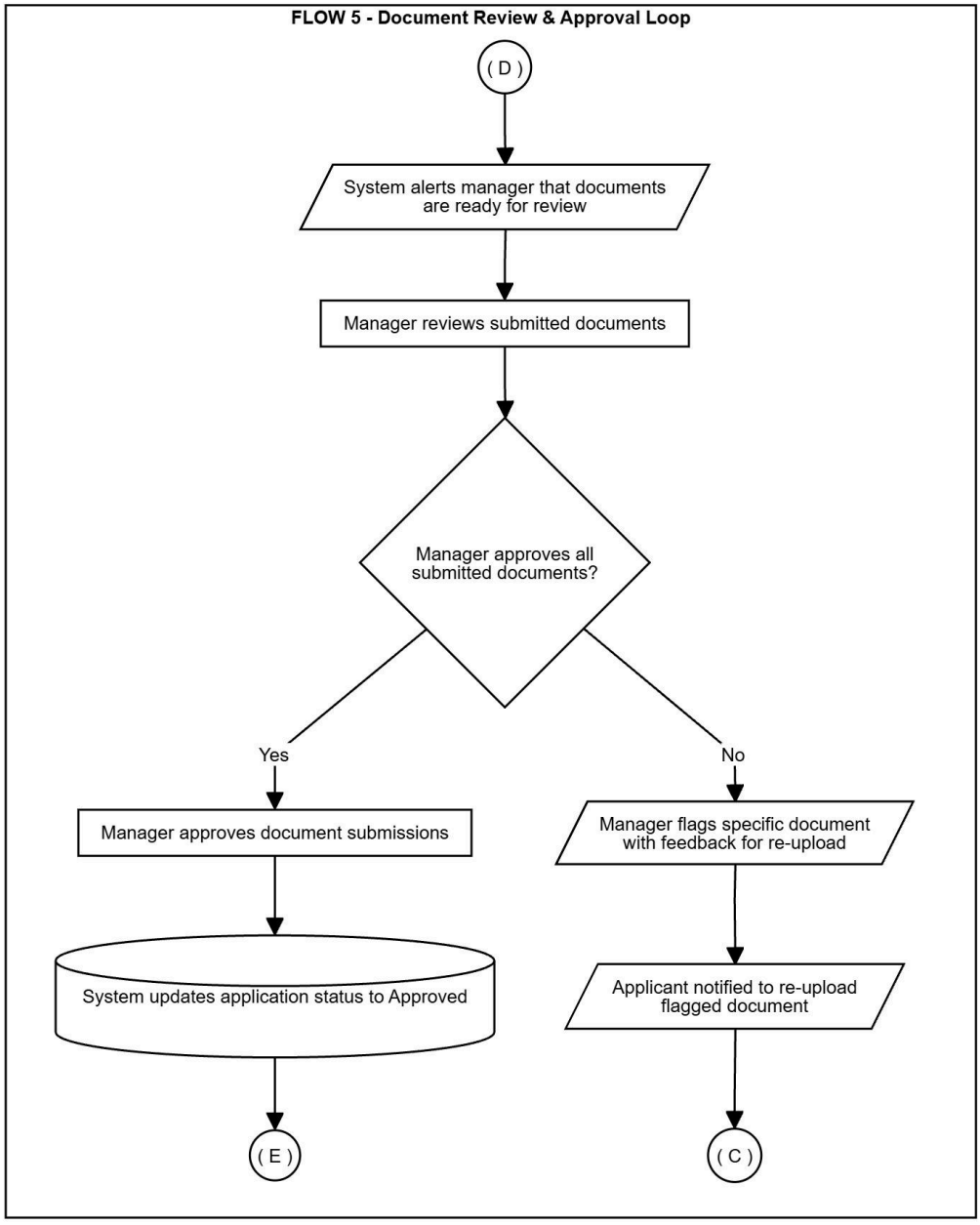
Flow 7: Lease Application Flowchart

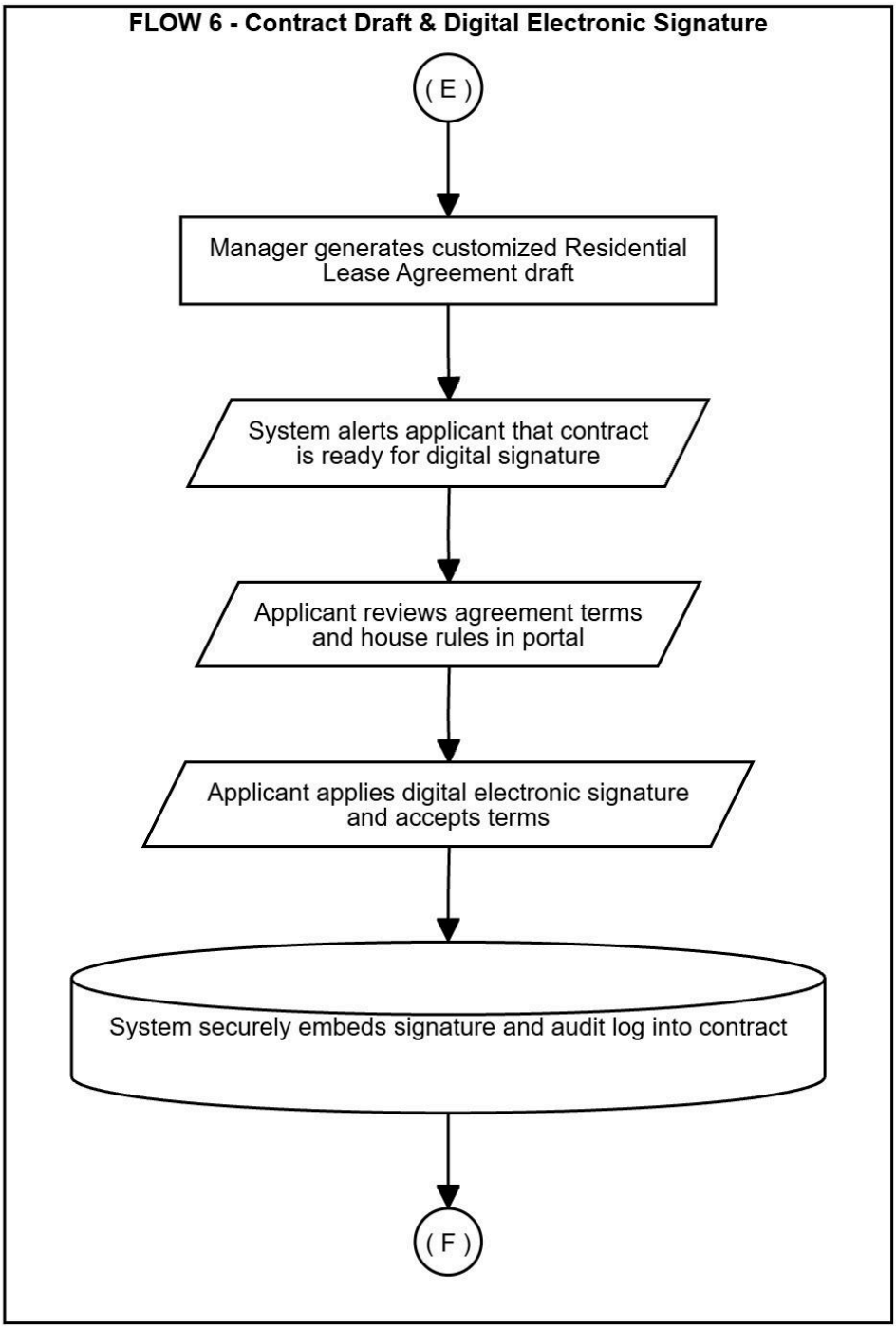


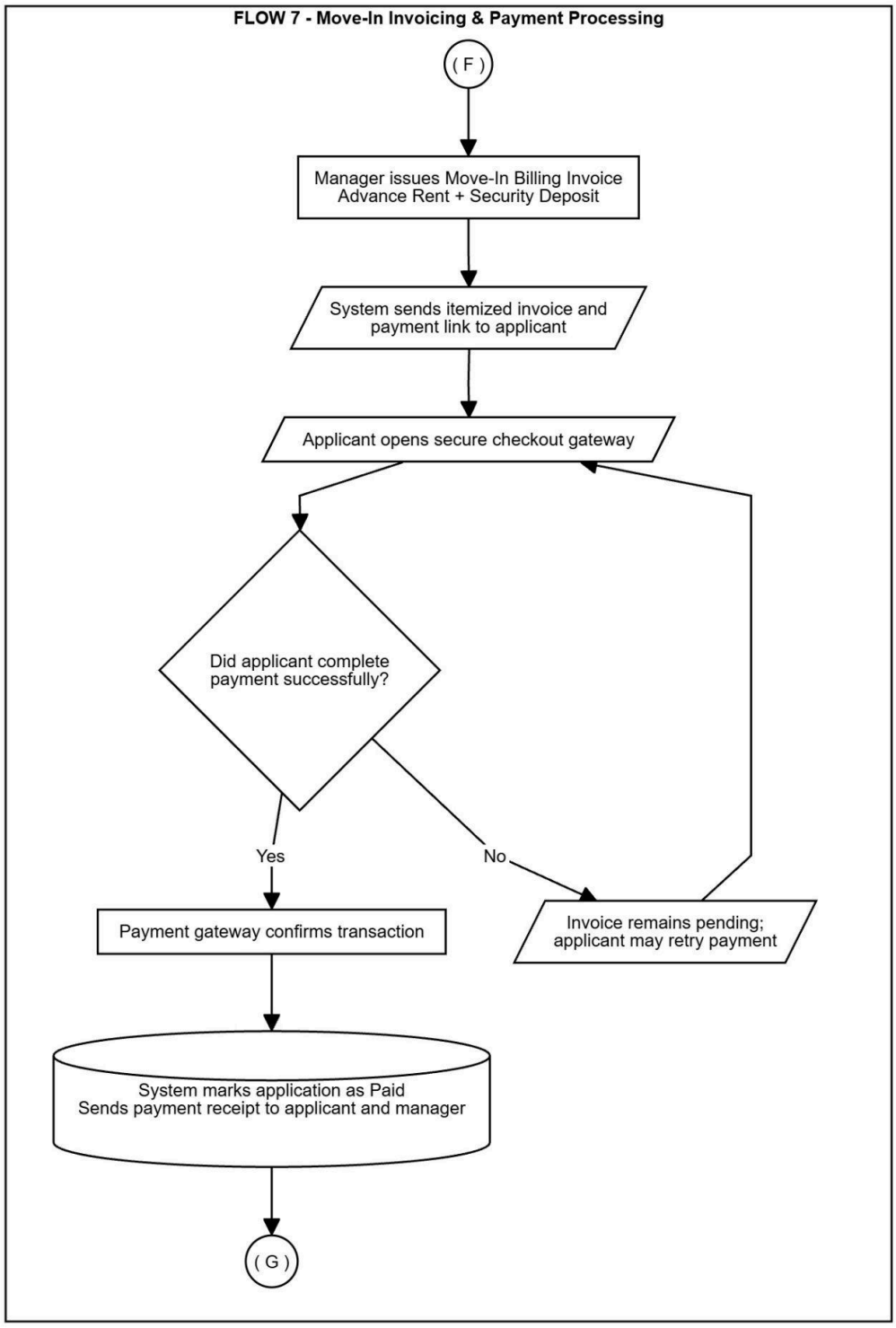

FLOW 2 - Application Submission & Property Routing
(A)
User submits completed lease application
Is unit currently available?
No
Yes
Is applicant eligible with no duplicate application?
No
Yes
Display notice: Unit is no longer available for leasing
Display notice: Active lease application already exists
System creates application record in Pending status
End
System alerts Property Managers and Landlord in real time
Applicant receives submission confirmation
(B)

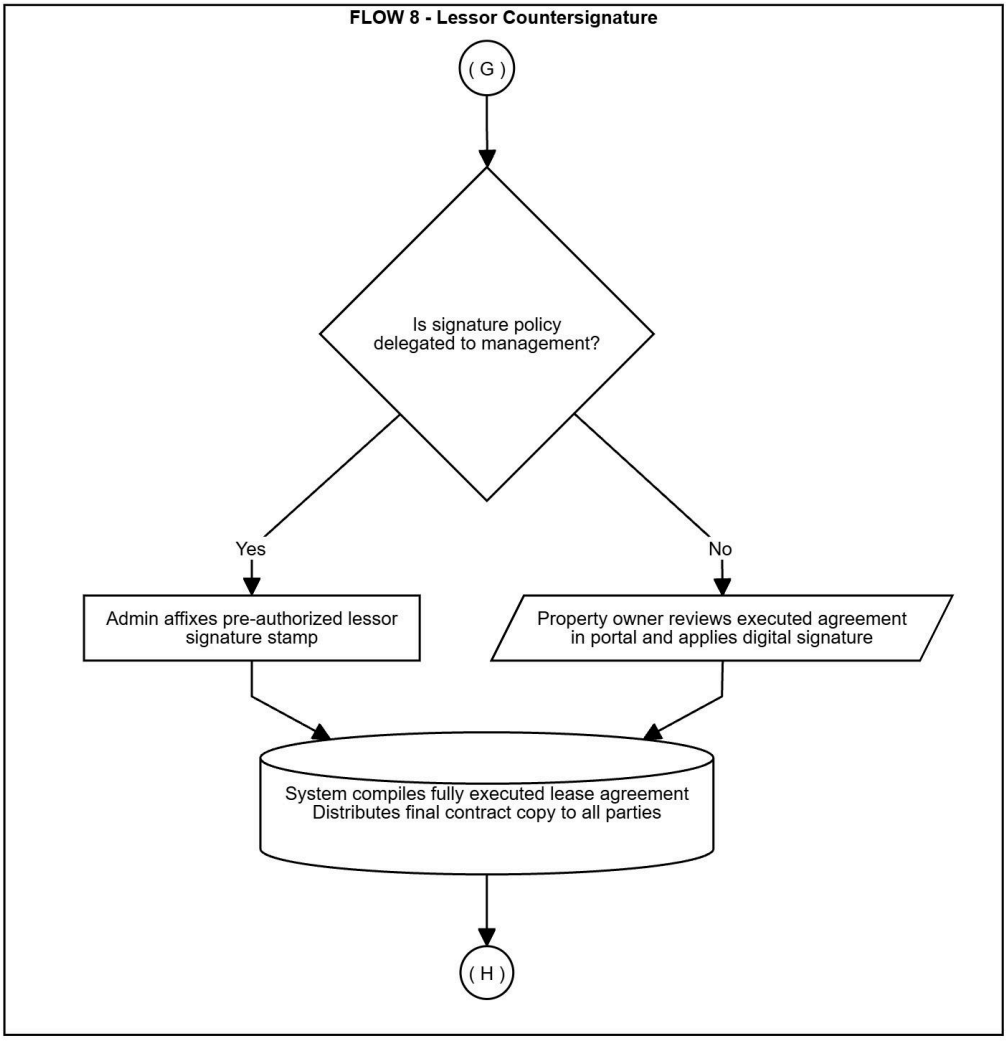

FLOW 3 - Candidate Screening & Requirements Dispatch
(B)
Building Admin or Landlord reviews applicant profile and credentials
Manager approves candidate to submit documents?
Yes
No
Manager issues required document checklist Government ID, Proof of Income, COE
Applicant notified that document submission is required
(C)
Manager records official rejection reason
System updates application status to Rejected
Applicant notified of application outcome
End

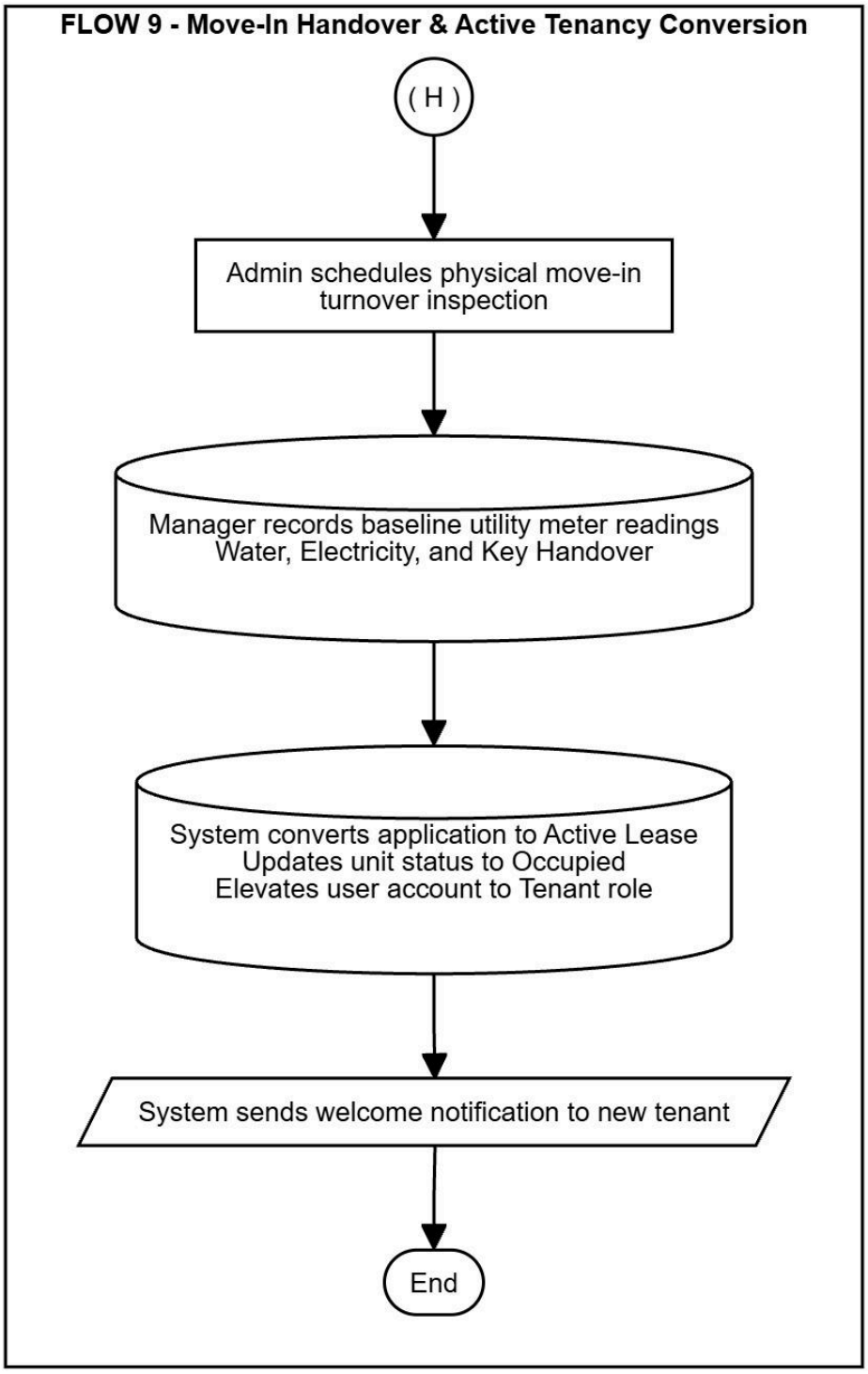

FLOW 4 - Document Upload & Secure Storage
(C)
Applicant views document checklist in portal
Applicant selects required files from device
Are files within format and size limits?
Yes
No
System encrypts files and securely stores documents
Display warning: File must be under 10MB in accepted formats PDF, JPG, PNG
Have all mandatory documents been submitted?
Yes
No
System updates application to Documents Submitted
Application remains in Requirements Needed status
(D)

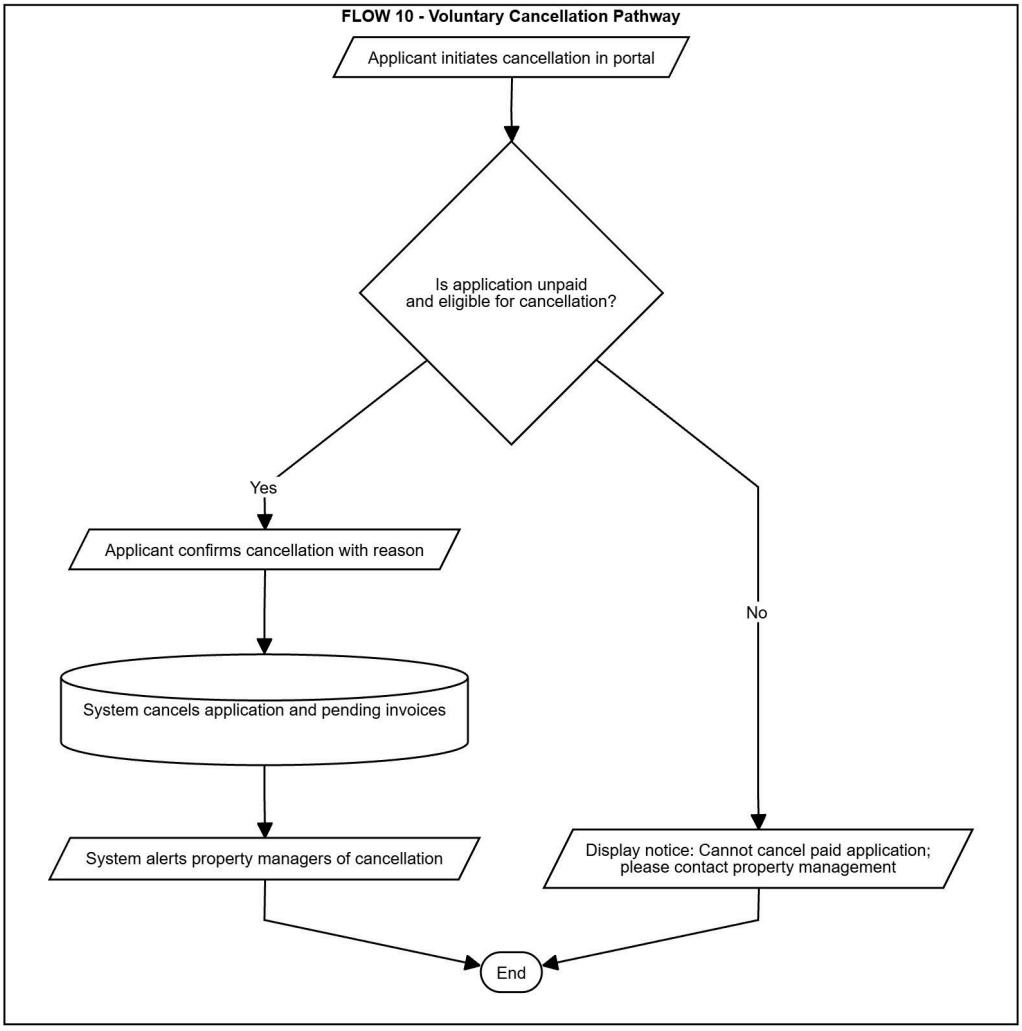

FLOW 5 - Document Review & Approval Loop
(D)
System alerts manager that documents are ready for review
Manager reviews submitted documents
Manager approves all submitted documents?
Yes
No
Manager approves document submissions
Manager flags specific document with feedback for re-upload
System updates application status to Approved
Applicant notified to re-upload flagged document
(E)
(C)


FLOW 6 - Contract Draft & Digital Electronic Signature
(E)
Manager generates customized Residential Lease Agreement draft
System alerts applicant that contract is ready for digital signature
Applicant reviews agreement terms and house rules in portal
Applicant applies digital electronic signature and accepts terms
System securely embeds signature and audit log into contract
(F)

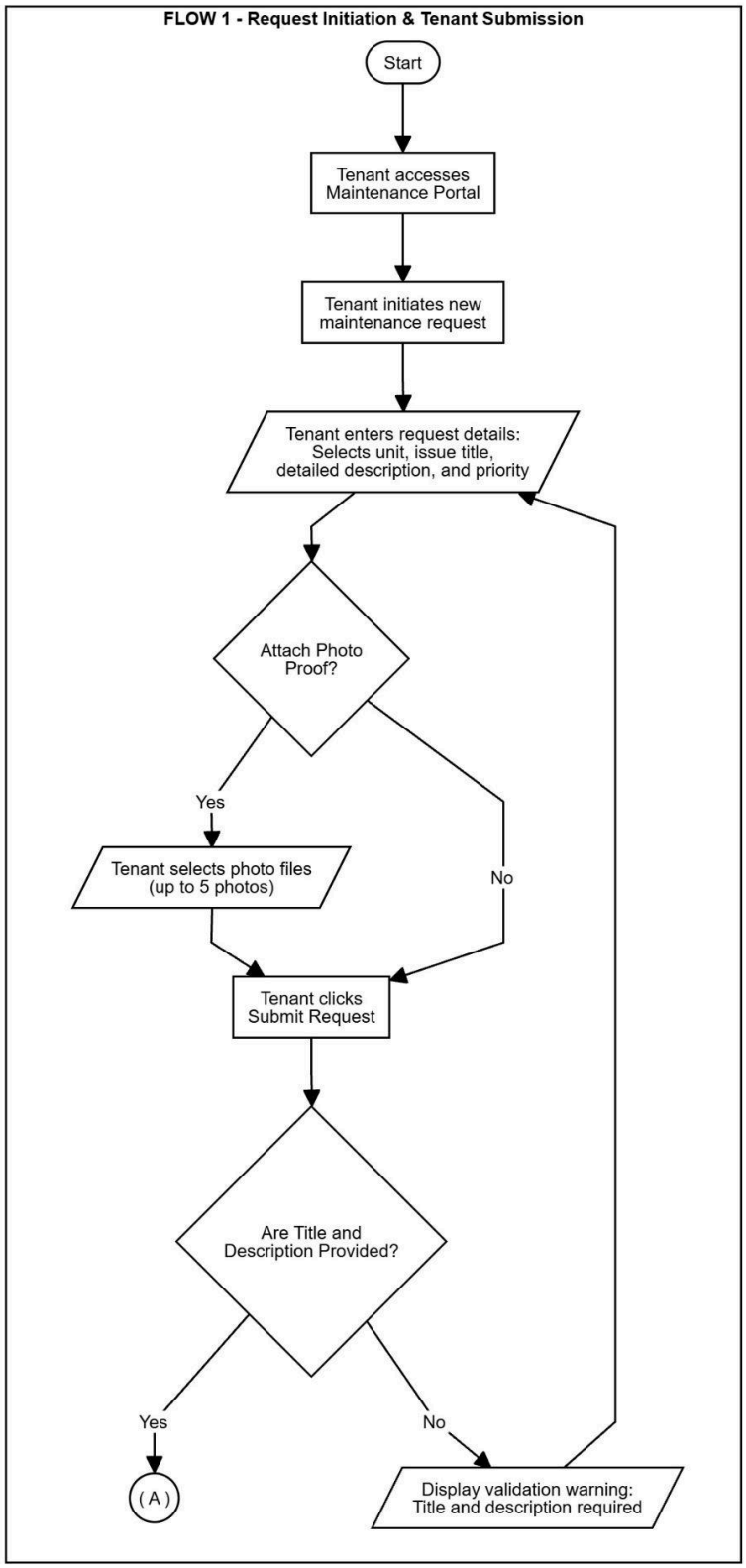

FLOW 7 - Move-In Invoicing & Payment Processing
(F)
Manager issues Move-In Billing Invoice
Advance Rent + Security Deposit
System sends itemized invoice and
payment link to applicant
Applicant opens secure checkout gateway
Did applicant complete
payment successfully?
Yes
No
Payment gateway confirms transaction
Invoice remains pending;
applicant may retry payment
System marks application as Paid
Sends payment receipt to applicant and manager
(G)

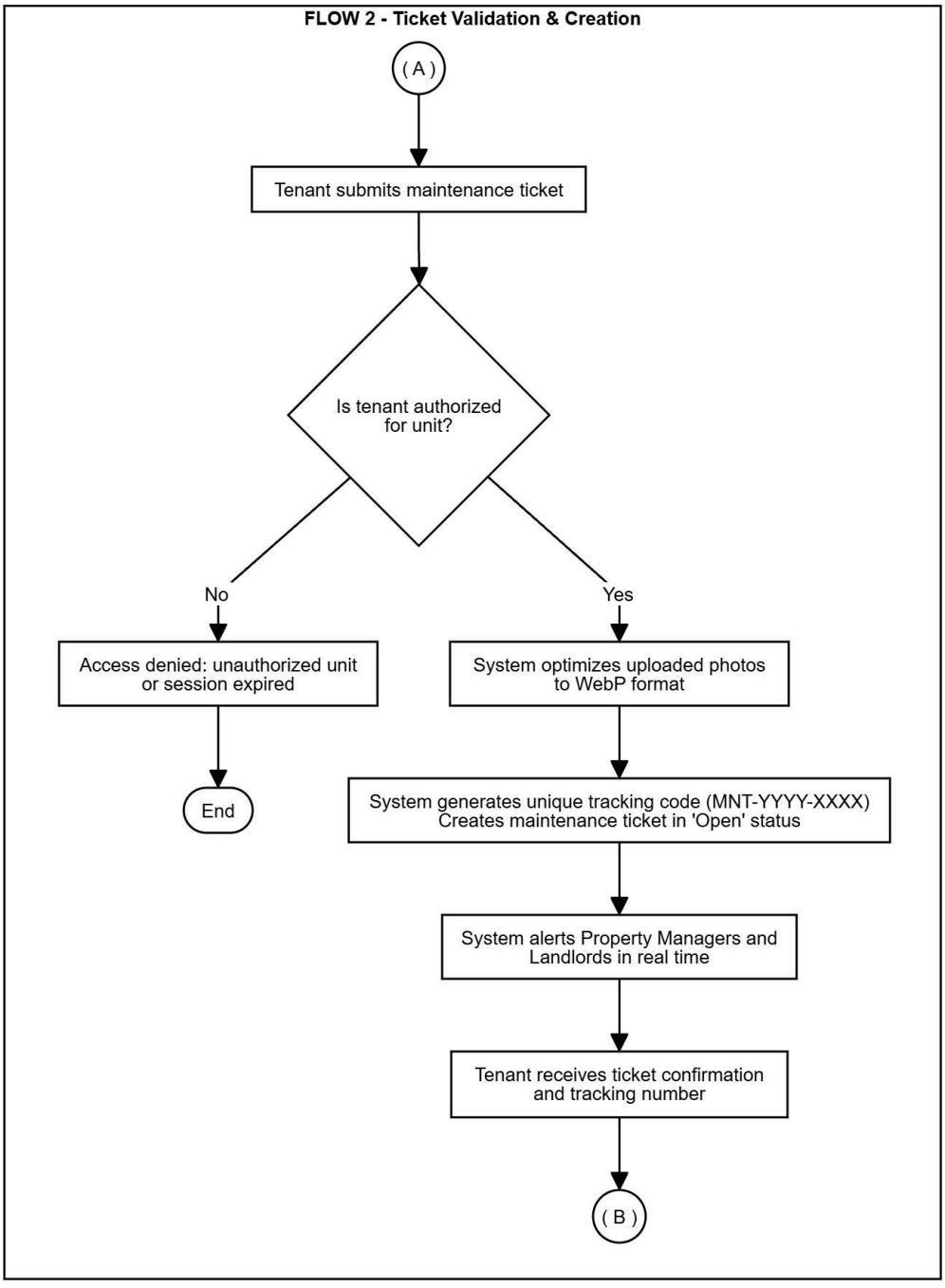

FLOW 8 - Lessor Countersignature
(G)
Is signature policy delegated to management?
Yes
No
Admin affixes pre-authorized lessor signature stamp
Property owner reviews executed agreement in portal and applies digital signature
System compiles fully executed lease agreement
Distributes final contract copy to all parties
(H)

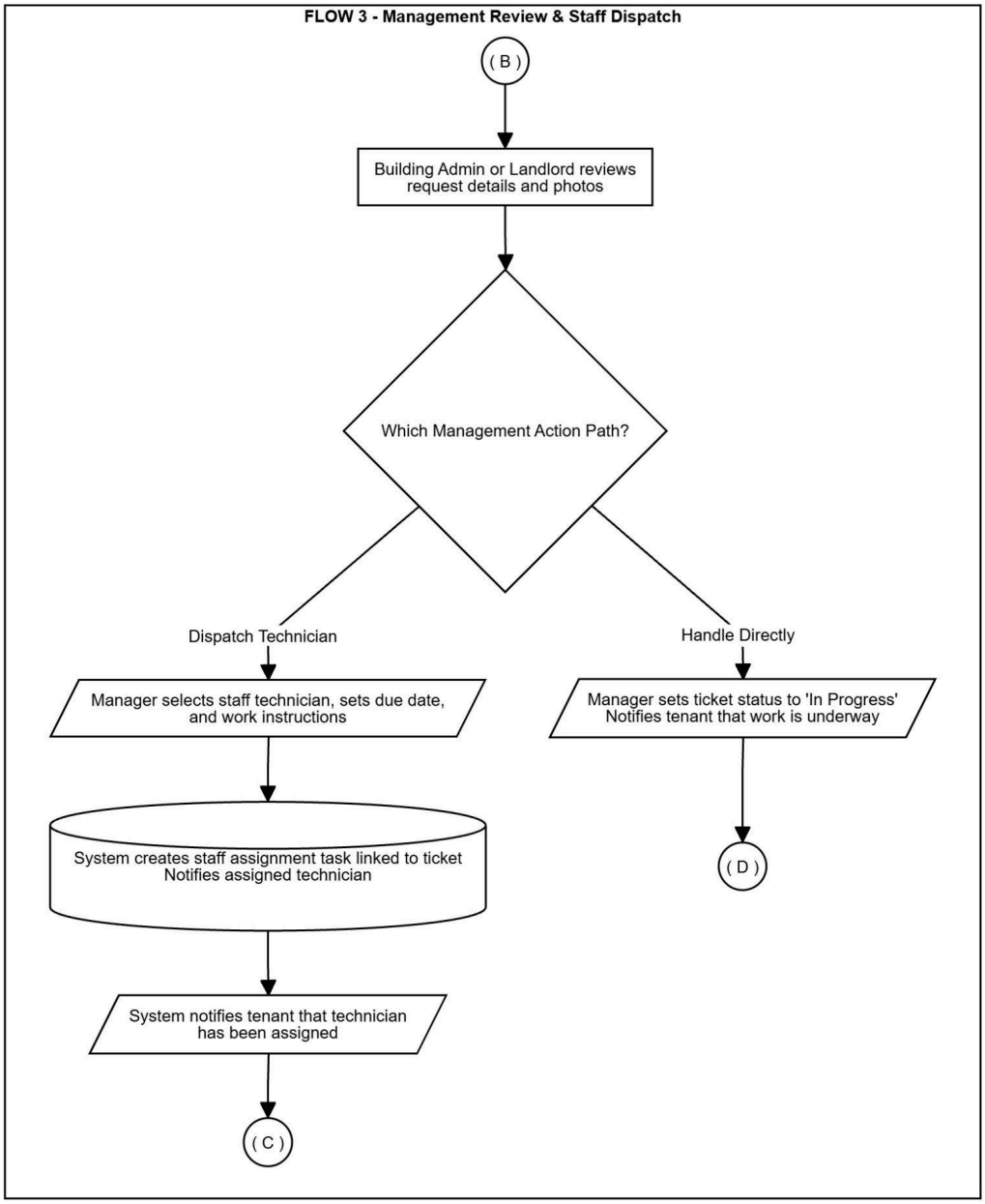

FLOW 9 - Move-In Handover & Active Tenancy Conversion
(H)
Admin schedules physical move-in turnover inspection
Manager records baseline utility meter readings Water, Electricity, and Key Handover
System converts application to Active Lease Updates unit status to Occupied Elevates user account to Tenant role
System sends welcome notification to new tenant
End

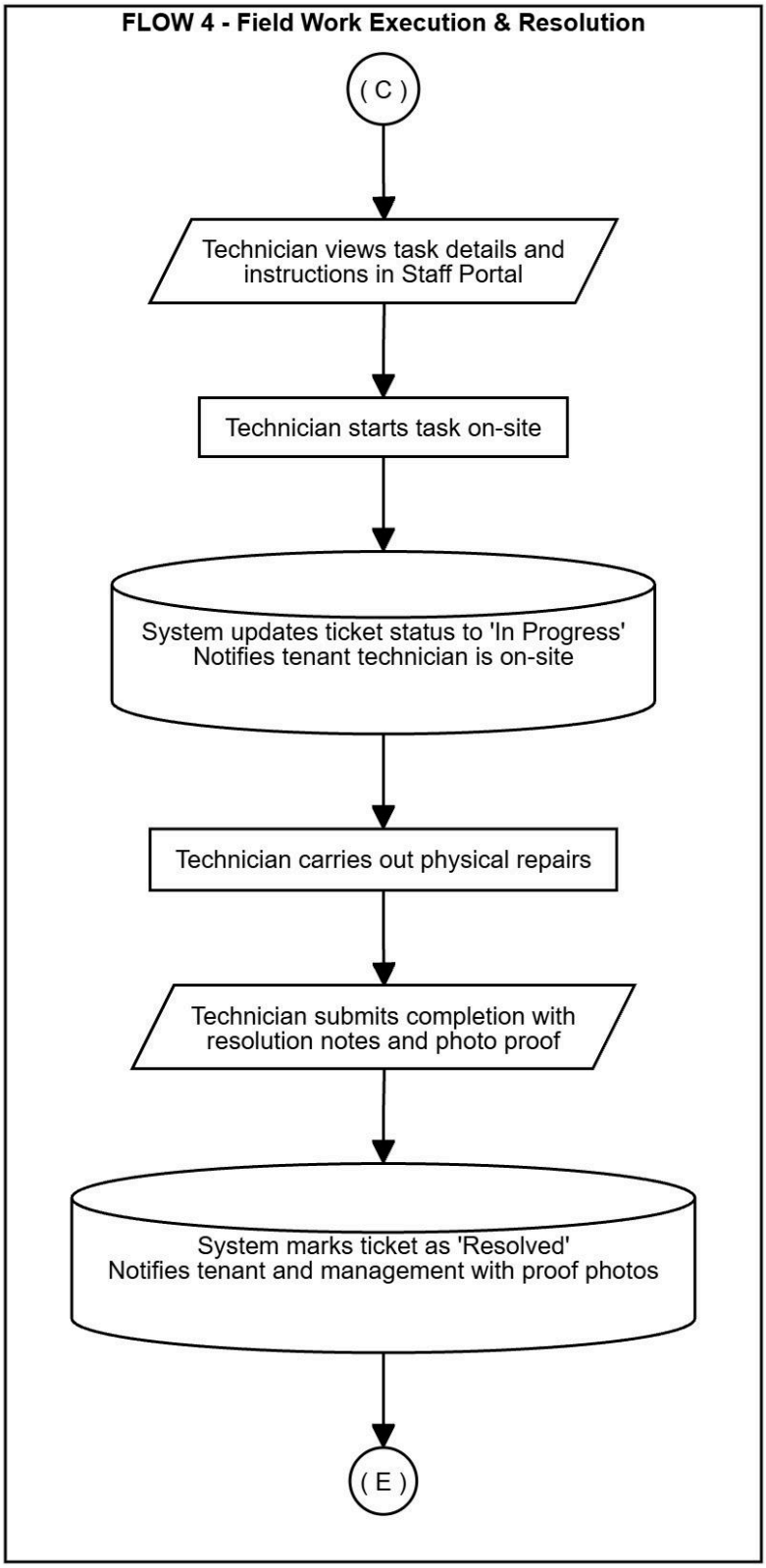

FLOW 10 - Voluntary Cancellation Pathway
Applicant initiates cancellation in portal
Is application unpaid and eligible for cancellation?
Yes
No
Applicant confirms cancellation with reason
System cancels application and pending invoices
System alerts property managers of cancellation
Display notice: Cannot cancel paid application; please contact property management
End

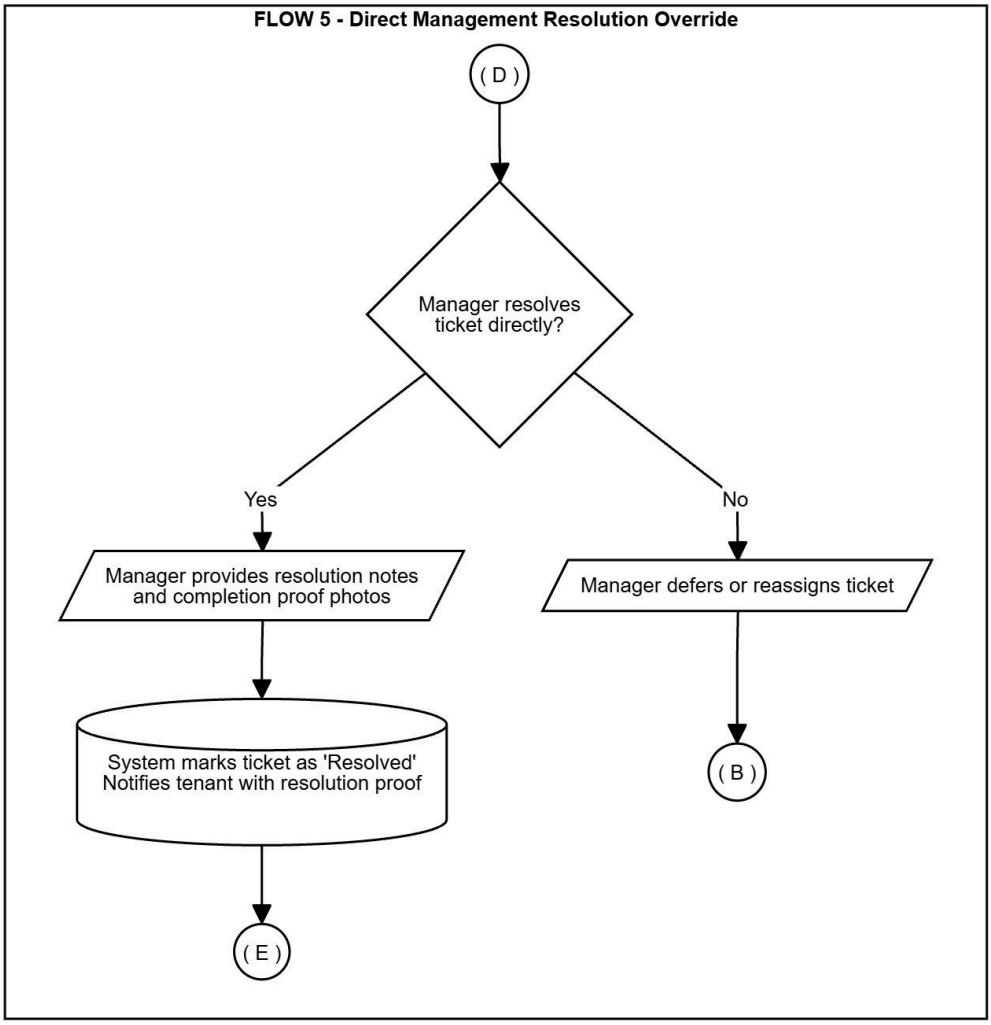
Flow 8: Maintenance Request Flowchart

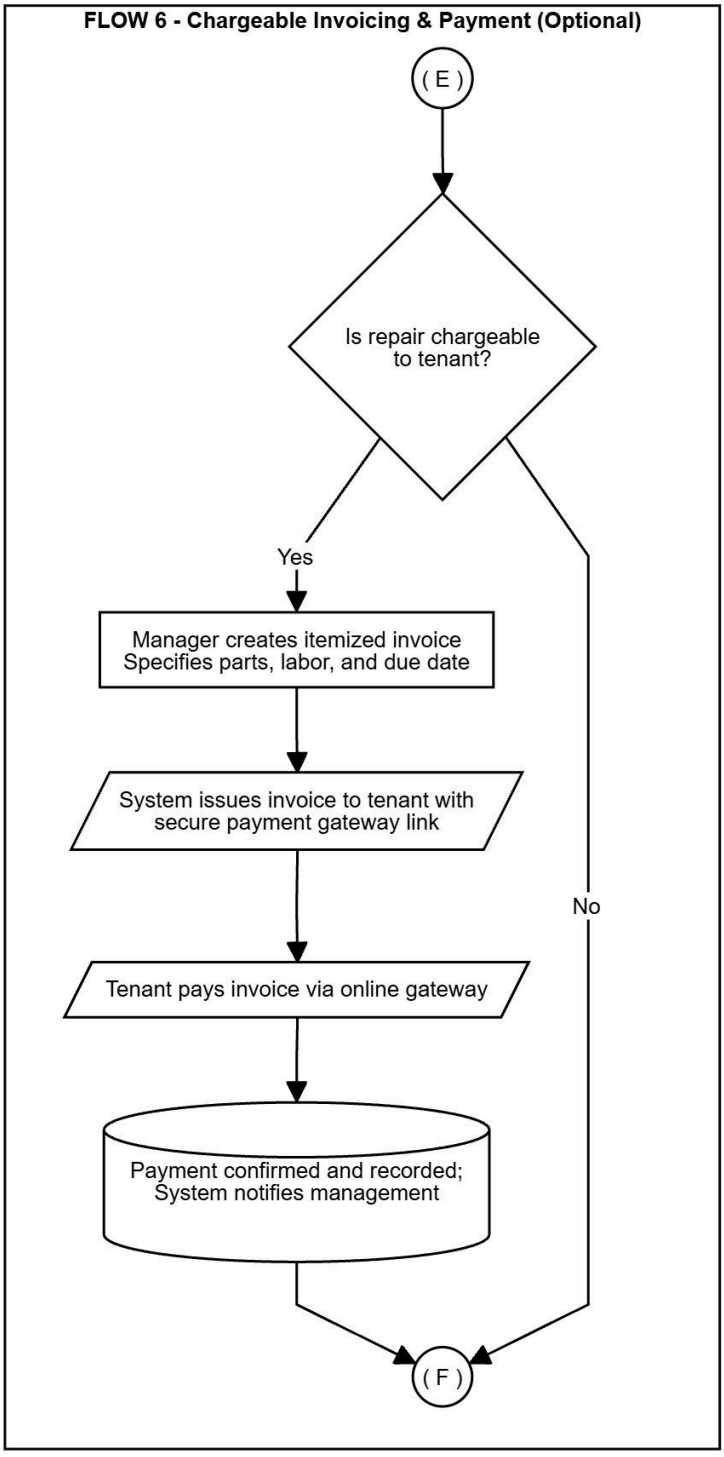


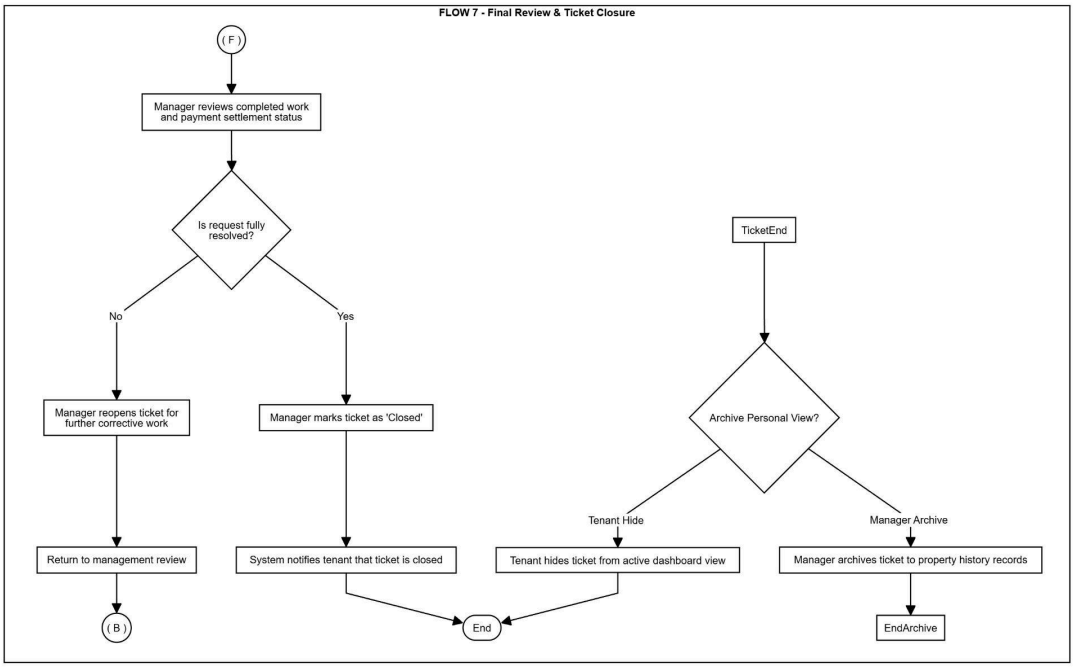

FLOW 2 - Ticket Validation & Creation
(A)
Tenant submits maintenance ticket
Is tenant authorized for unit?
No
Yes
Access denied: unauthorized unit or session expired
End
System optimizes uploaded photos to WebP format
System generates unique tracking code (MNT-YYYY-XXXX)
Creates maintenance ticket in 'Open' status
System alerts Property Managers and Landlords in real time
Tenant receives ticket confirmation and tracking number
(B)

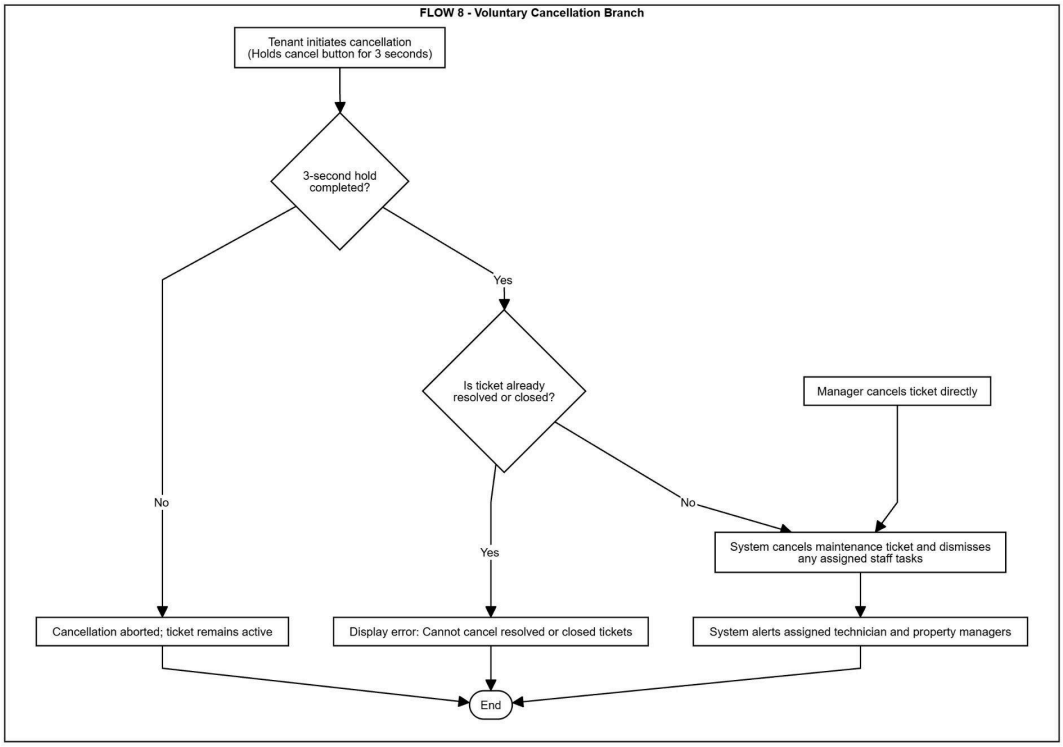

FLOW 3 - Management Review & Staff Dispatch
(B)
Building Admin or Landlord reviews request details and photos
Which Management Action Path?
Dispatch Technician
Handle Directly
Manager selects staff technician, sets due date, and work instructions
Manager sets ticket status to 'In Progress' Notifies tenant that work is underway
System creates staff assignment task linked to ticket Notifies assigned technician
(D)
System notifies tenant that technician has been assigned
(C)


FLOW 4 - Field Work Execution & Resolution
(C)
Technician views task details and instructions in Staff Portal
Technician starts task on-site
System updates ticket status to 'In Progress'
Notifies tenant technician is on-site
Technician carries out physical repairs
Technician submits completion with resolution notes and photo proof
System marks ticket as 'Resolved'
Notifies tenant and management with proof photos
(E)

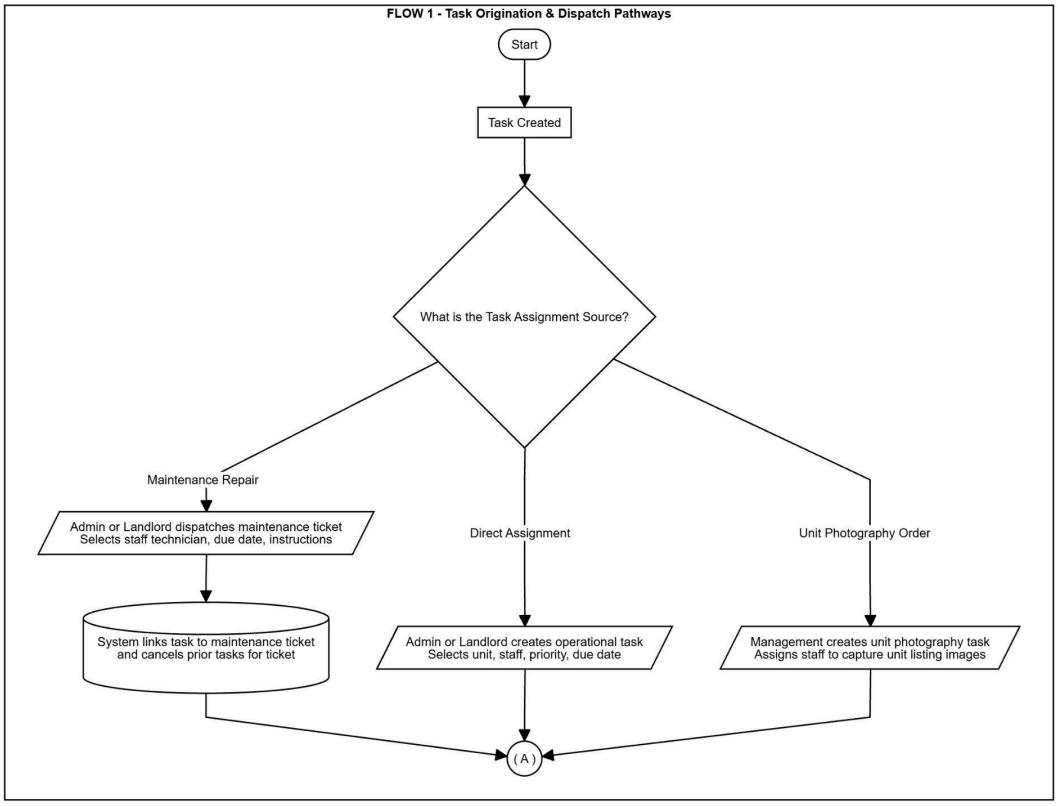

FLOW 5 - Direct Management Resolution Override
(D)
Manager resolves ticket directly?
Yes
No
Manager provides resolution notes and completion proof photos
Manager defers or reassigns ticket
System marks ticket as 'Resolved' Notifies tenant with resolution proof
(B)
(E)

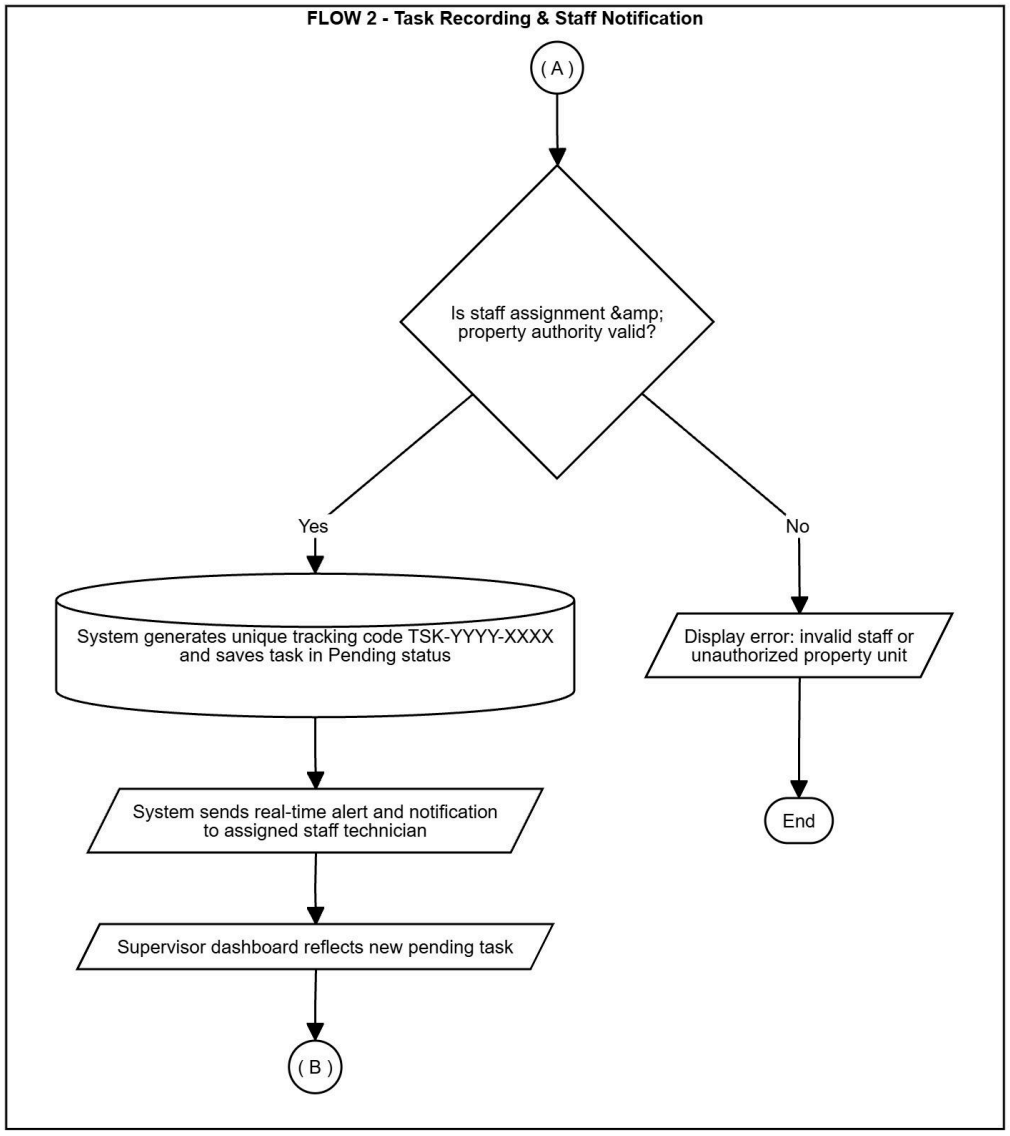

FLOW 6 - Chargeable Invoicing & Payment (Optional)
(E)
Is repair chargeable to tenant?
Yes
No
Manager creates itemized invoice Specifies parts, labor, and due date
System issues invoice to tenant with secure payment gateway link
Tenant pays invoice via online gateway
Payment confirmed and recorded; System notifies management
(F)

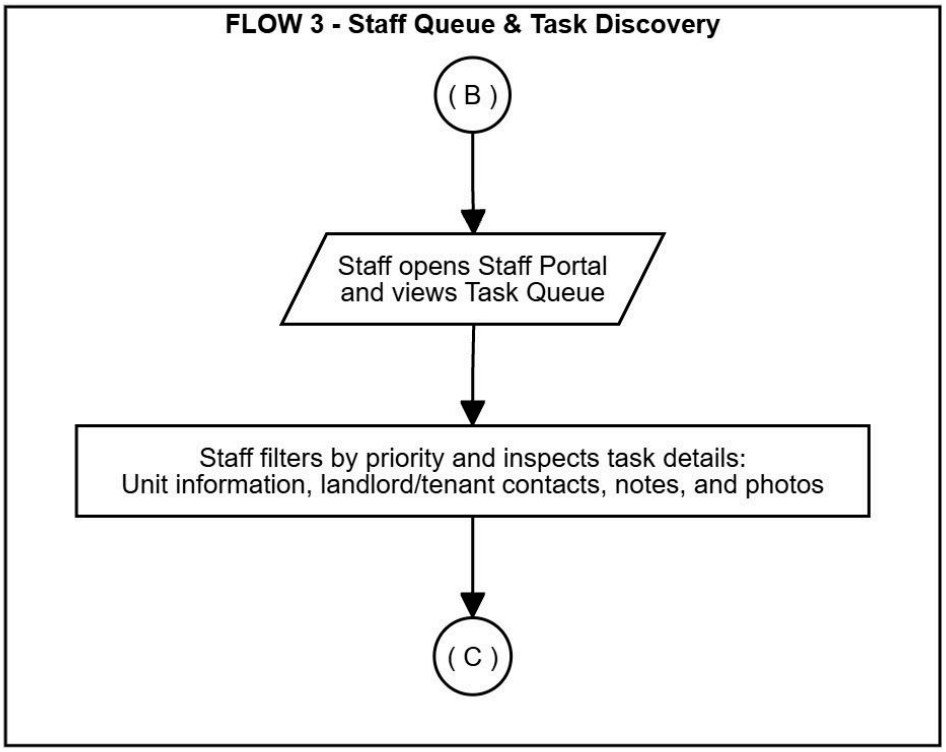

FLOW 7 - Final Review & Ticket Closure
(F)
Manager reviews completed work and payment settlement status
Is request fully resolved?
No
Yes
Manager reopens ticket for further corrective work
Manager marks ticket as 'Closed'
Return to management review
System notifies tenant that ticket is closed
(B)
End
TicketEnd
Archive Personal View?
Tenant Hide
Manager Archive
Tenant hides ticket from active dashboard view
Manager archives ticket to property history records
EndArchive

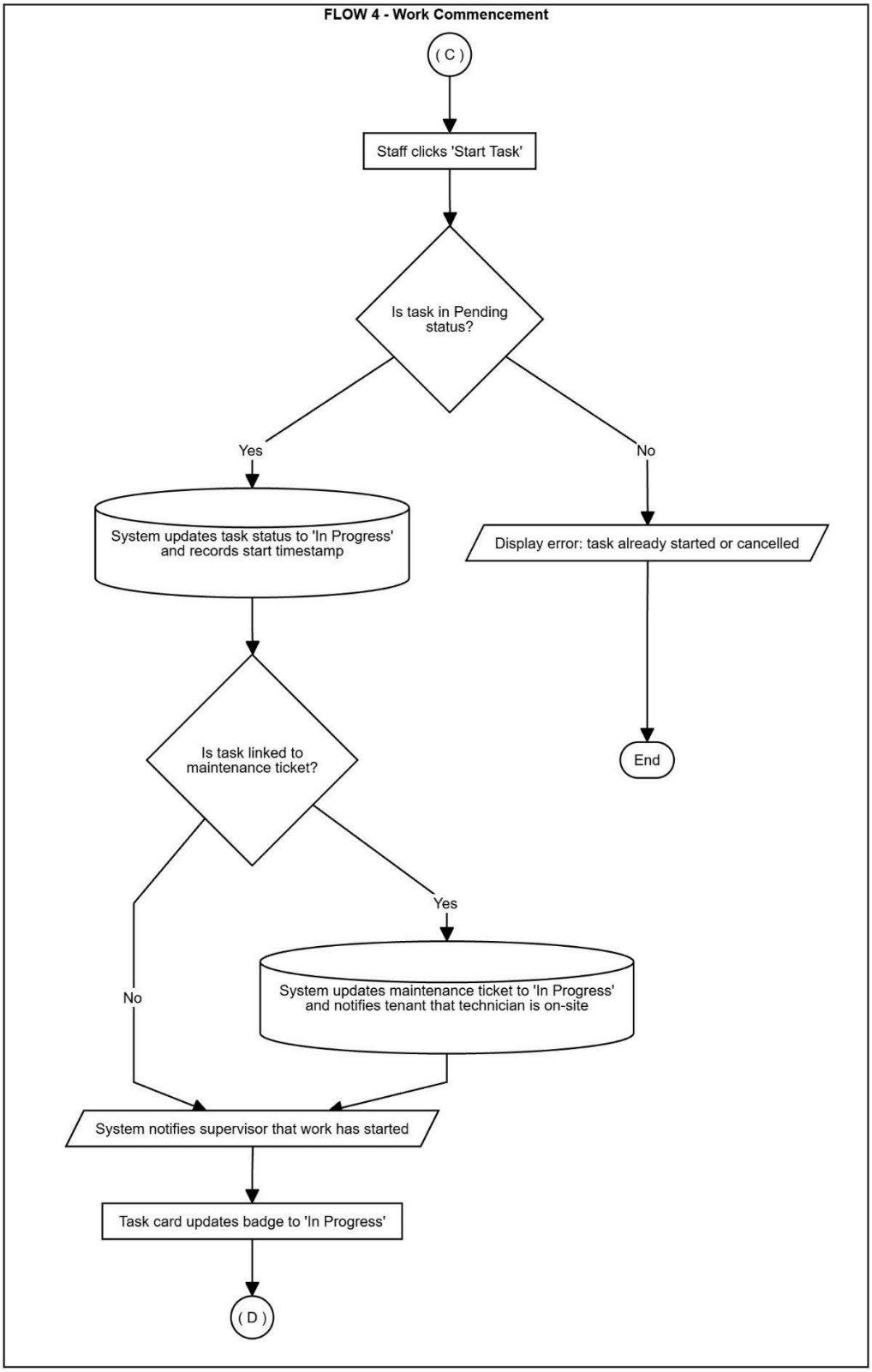

FLOW 8 - Voluntary Cancellation Branch
Tenant initiates cancellation (Holds cancel button for 3 seconds)
3-second hold completed?
No
Yes
Is ticket already resolved or closed?
Yes
No
Manager cancels ticket directly
System cancels maintenance ticket and dismisses any assigned staff tasks
Cancellation aborted; ticket remains active
Display error: Cannot cancel resolved or closed tickets
System alerts assigned technician and property managers
End

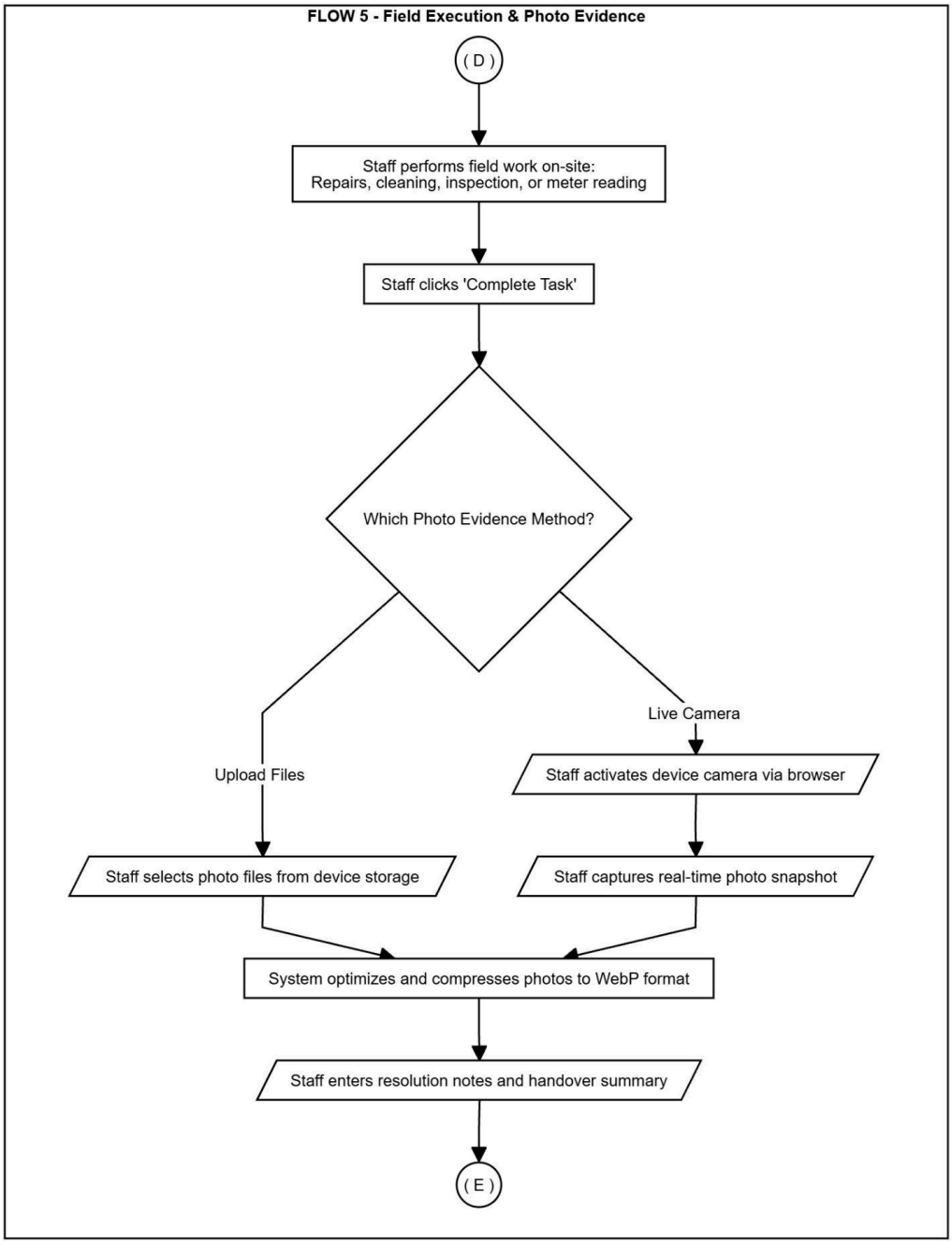
Flow 9: Staff Tasks Flowchart

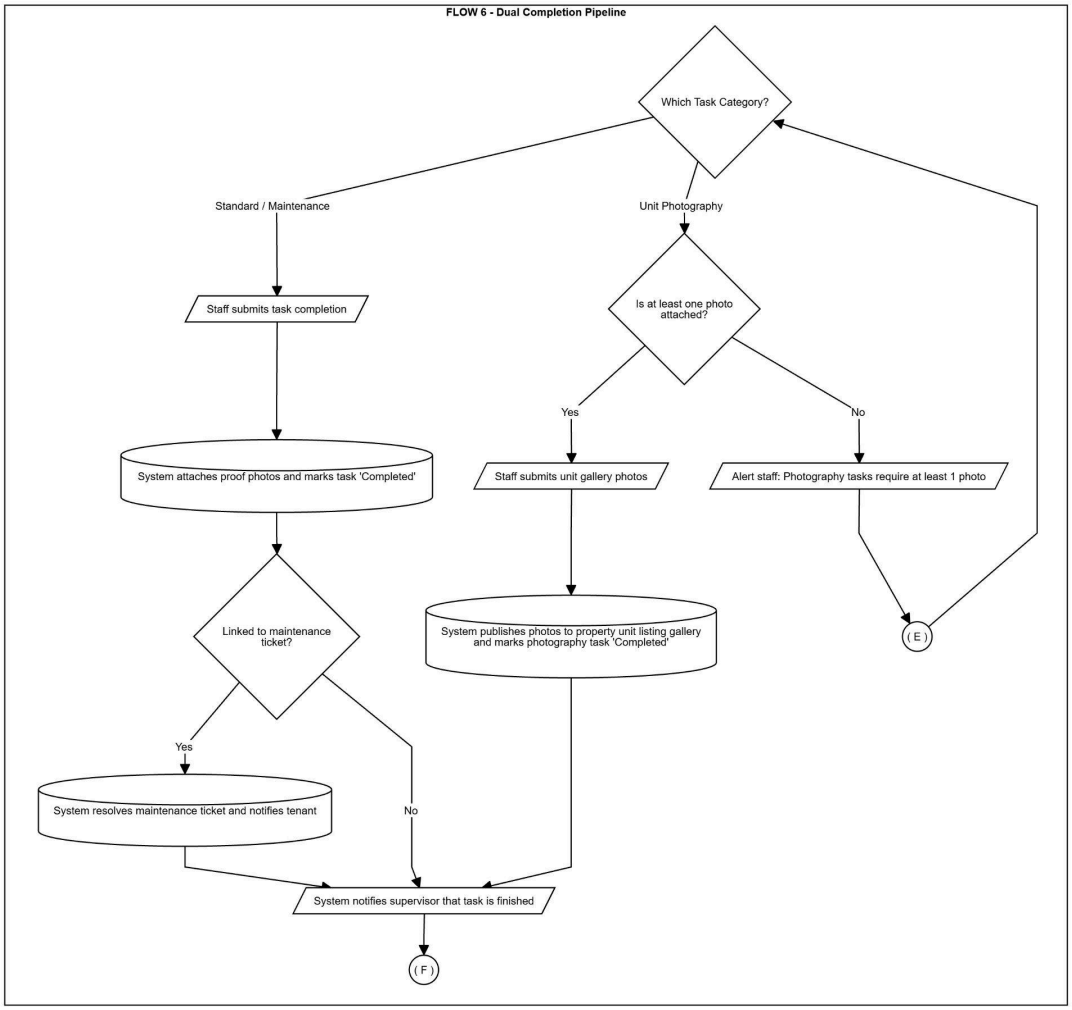


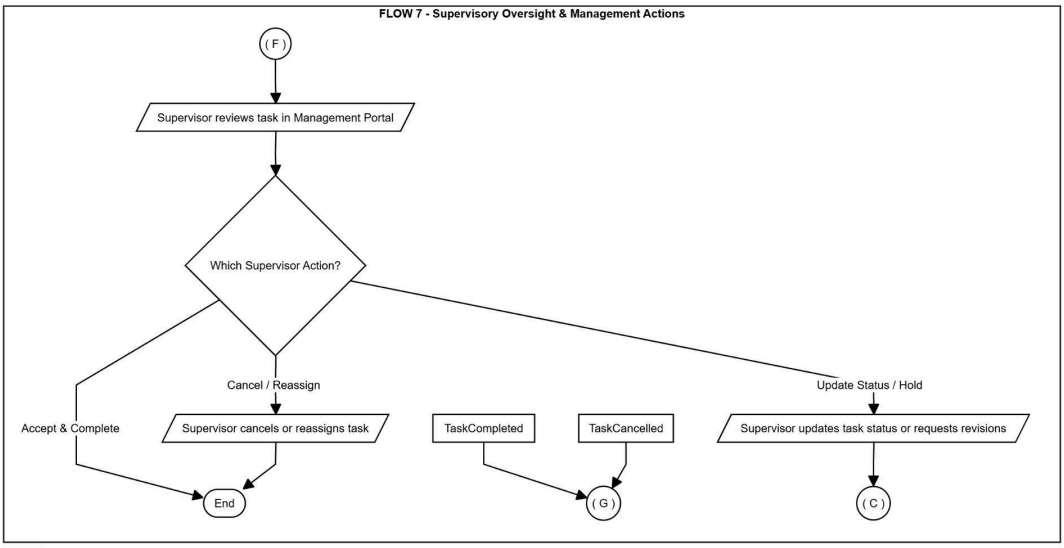

FLOW 2 - Task Recording & Staff Notification
(A)
Is staff assignment & property authority valid?
Yes
No
System generates unique tracking code TSK-YYYY-XXXX and saves task in Pending status
Display error: invalid staff or unauthorized property unit
End
System sends real-time alert and notification to assigned staff technician
Supervisor dashboard reflects new pending task
(B)

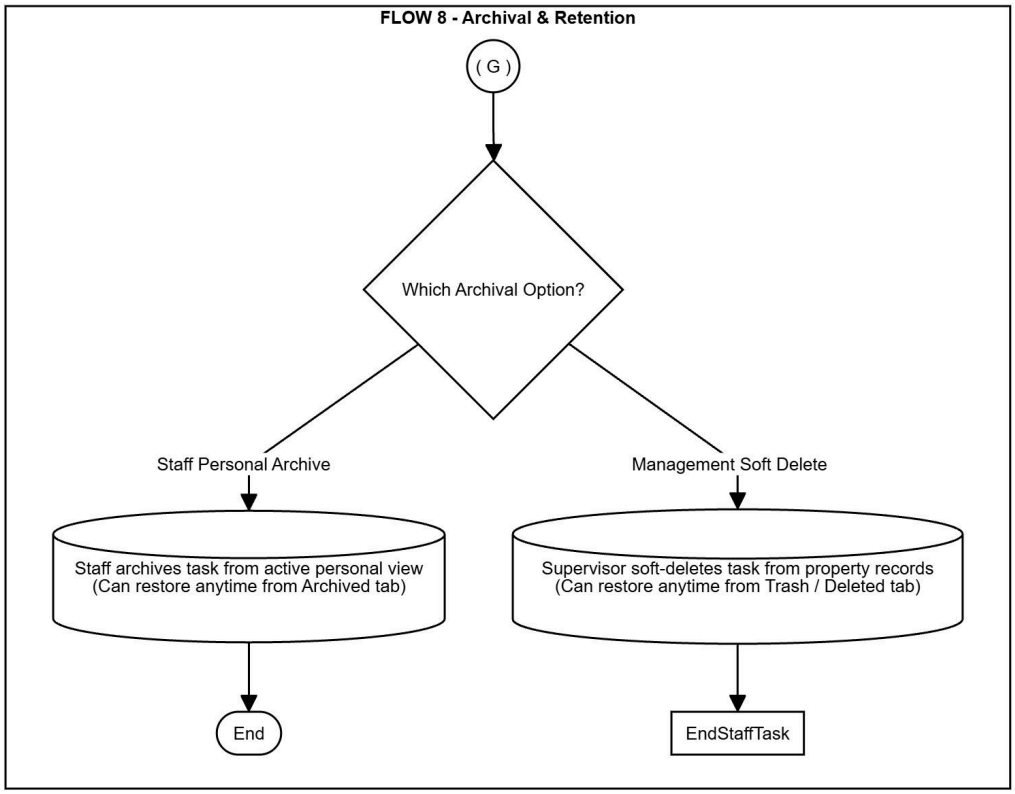

FLOW 3 - Staff Queue & Task Discovery
(B)
Staff opens Staff Portal and views Task Queue
Staff filters by priority and inspects task details: Unit information, landlord/tenant contacts, notes, and photos
(C)


FLOW 4 - Work Commencement
(C)
Staff clicks 'Start Task'
Is task in Pending status?
Yes
No
System updates task status to 'In Progress' and records start timestamp
Display error: task already started or cancelled
End
Is task linked to maintenance ticket?
Yes
No
System updates maintenance ticket to 'In Progress' and notifies tenant that technician is on-site
System notifies supervisor that work has started
Task card updates badge to 'In Progress'
(D)

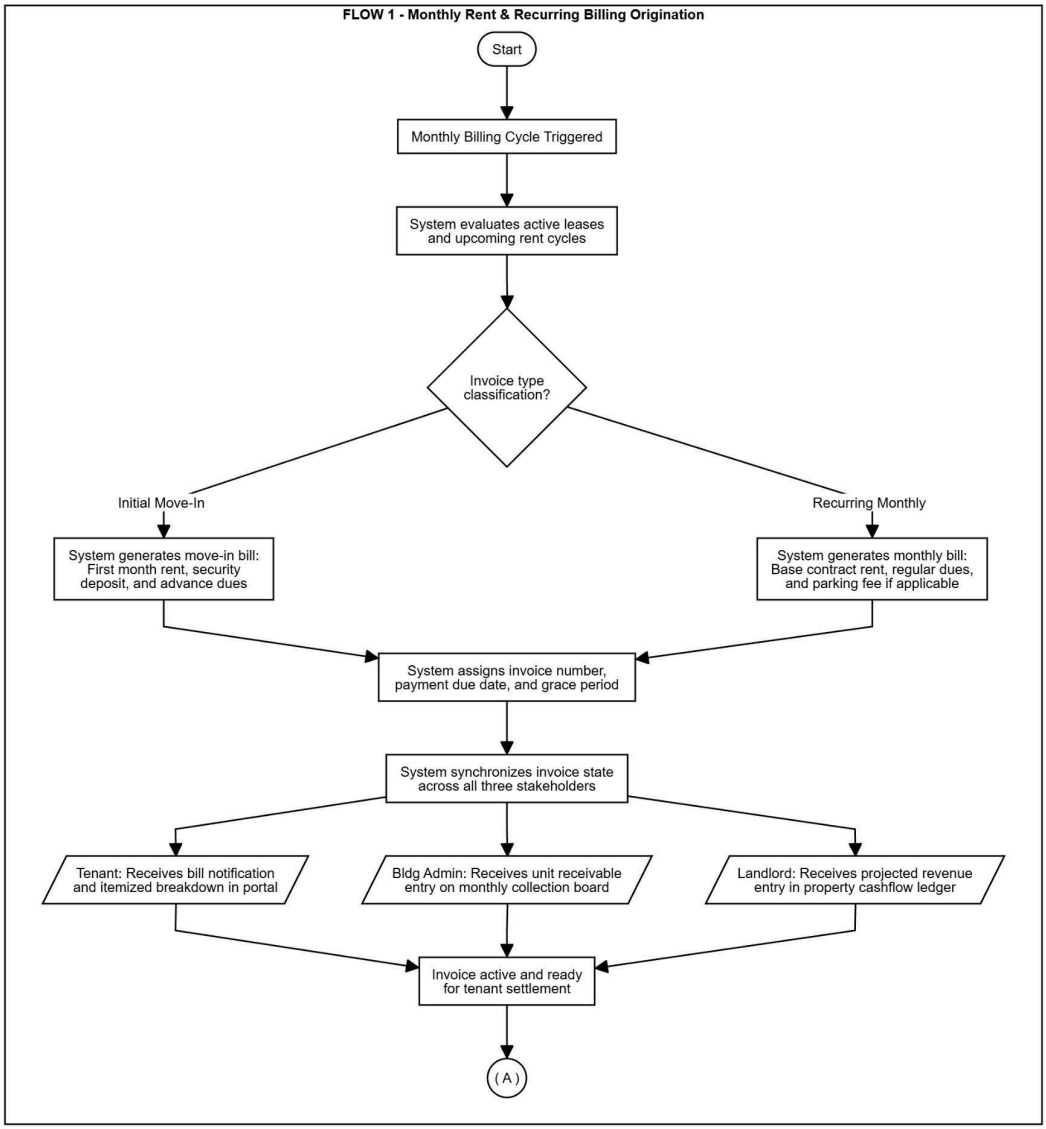

FLOW 5 - Field Execution & Photo Evidence
(D)
Staff performs field work on-site:
Repairs, cleaning, inspection, or meter reading
Staff clicks 'Complete Task'
Which Photo Evidence Method?
Upload Files
Live Camera
Staff activates device camera via browser
Staff selects photo files from device storage
Staff captures real-time photo snapshot
System optimizes and compresses photos to WebP format
Staff enters resolution notes and handover summary
(E)

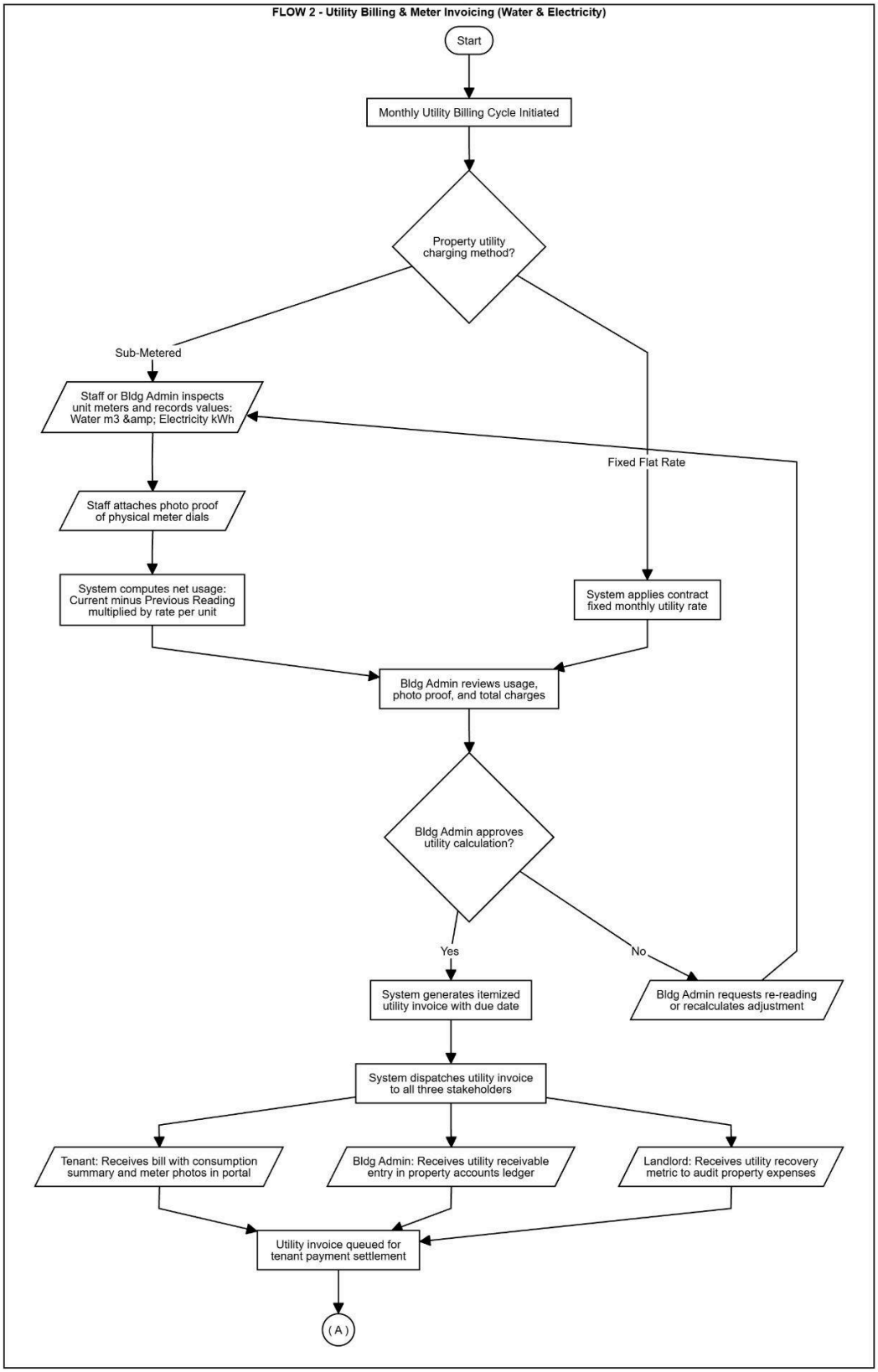

FLOW 6 - Dual Completion Pipeline
Which Task Category?
Standard / Maintenance
Unit Photography
Staff submits task completion
Is at least one photo attached?
Yes
No
System attaches proof photos and marks task 'Completed'
Staff submits unit gallery photos
Alert staff: Photography tasks require at least 1 photo
Linked to maintenance ticket?
System publishes photos to property unit listing gallery and marks photography task 'Completed'
(E)
Yes
System resolves maintenance ticket and notifies tenant
No
System notifies supervisor that task is finished
(F)

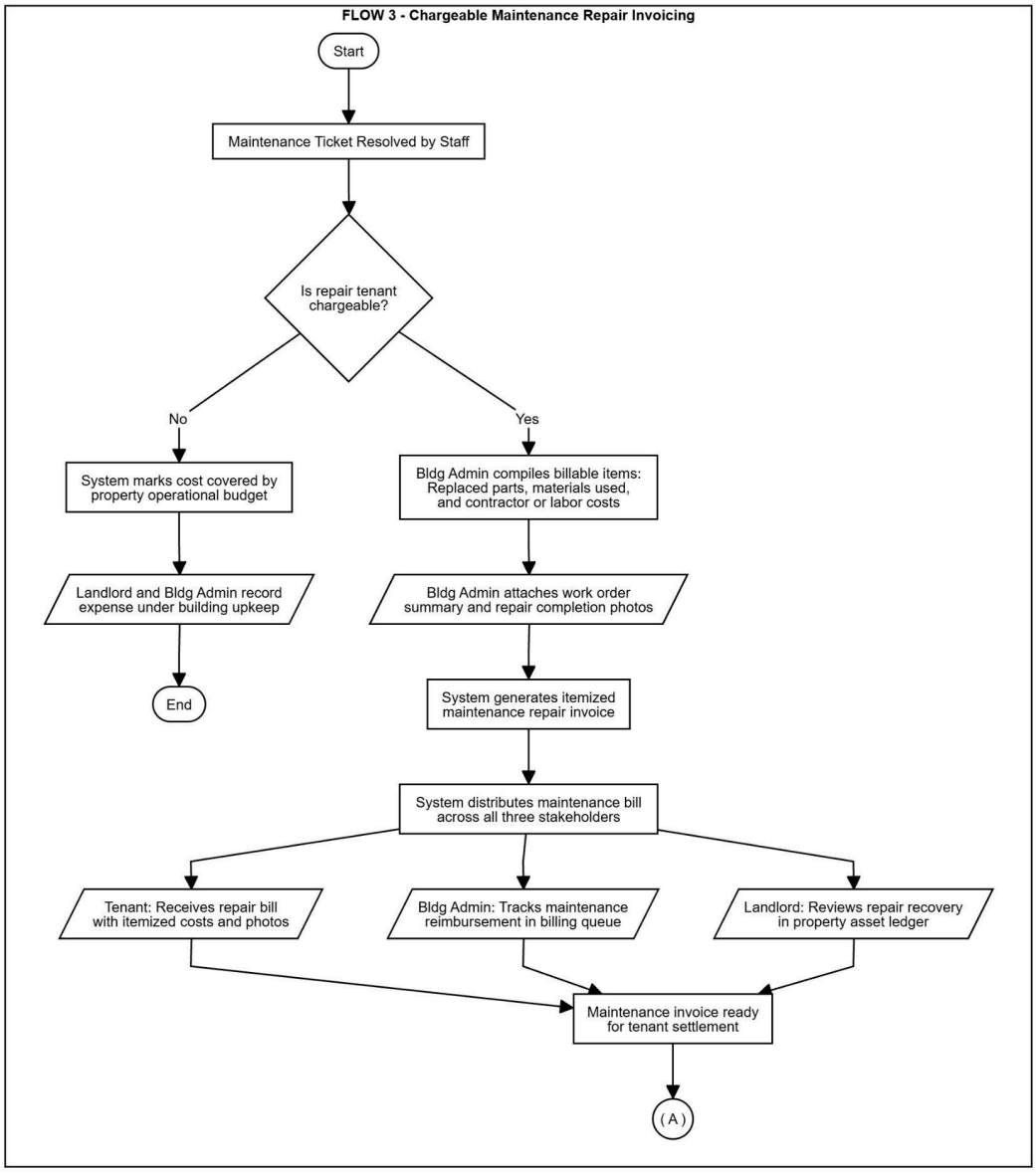

FLOW 7 - Supervisory Oversight & Management Actions
(F)
Supervisor reviews task in Management Portal
Which Supervisor Action?
Cancel / Reassign
Update Status / Hold
Accept & Complete
Supervisor cancels or reassigns task
TaskCompleted
TaskCancelled
Supervisor updates task status or requests revisions
End
(G)
(C)

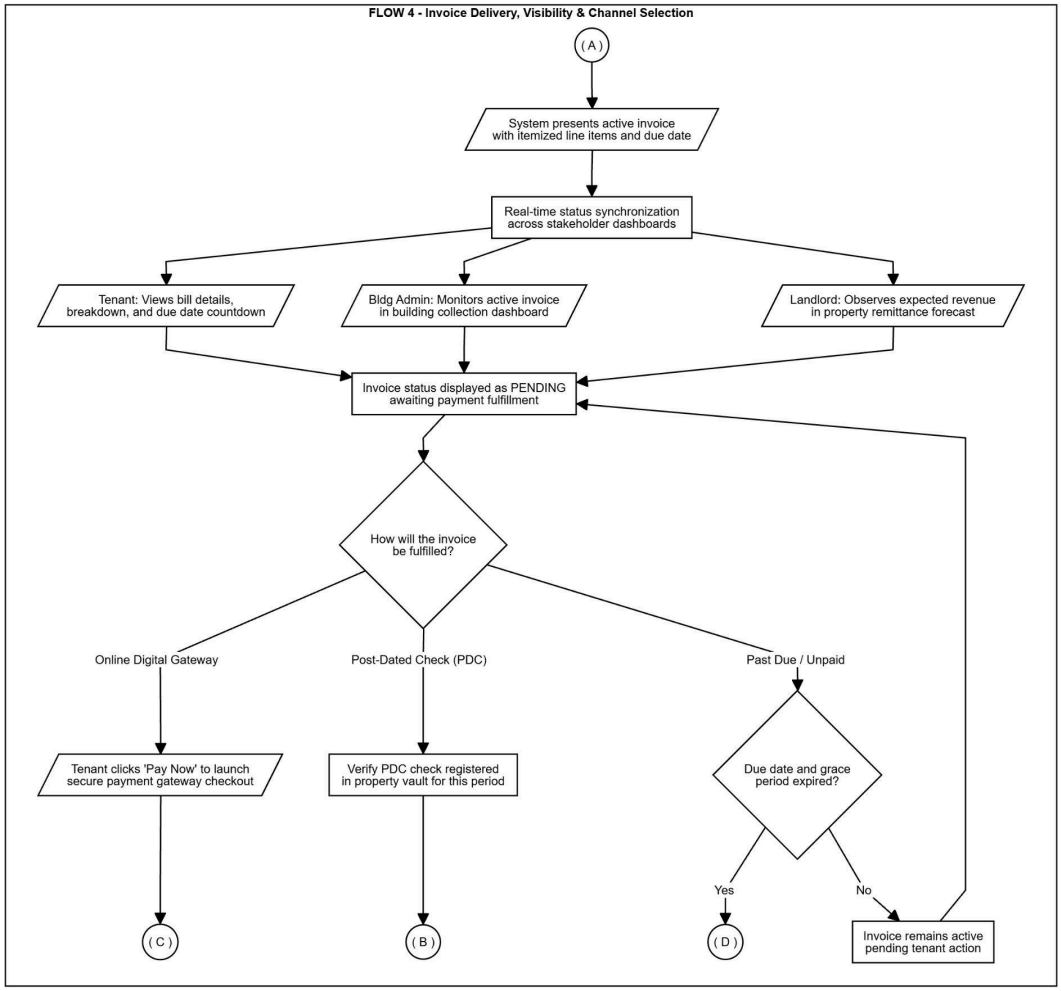

FLOW 8 - Archival & Retention
(G)
Which Archival Option?
Staff Personal Archive
Management Soft Delete
Staff archives task from active personal view
(Can restore anytime from Archived tab)
Supervisor soft-deletes task from property records
(Can restore anytime from Trash / Deleted tab)
End
EndStaffTask

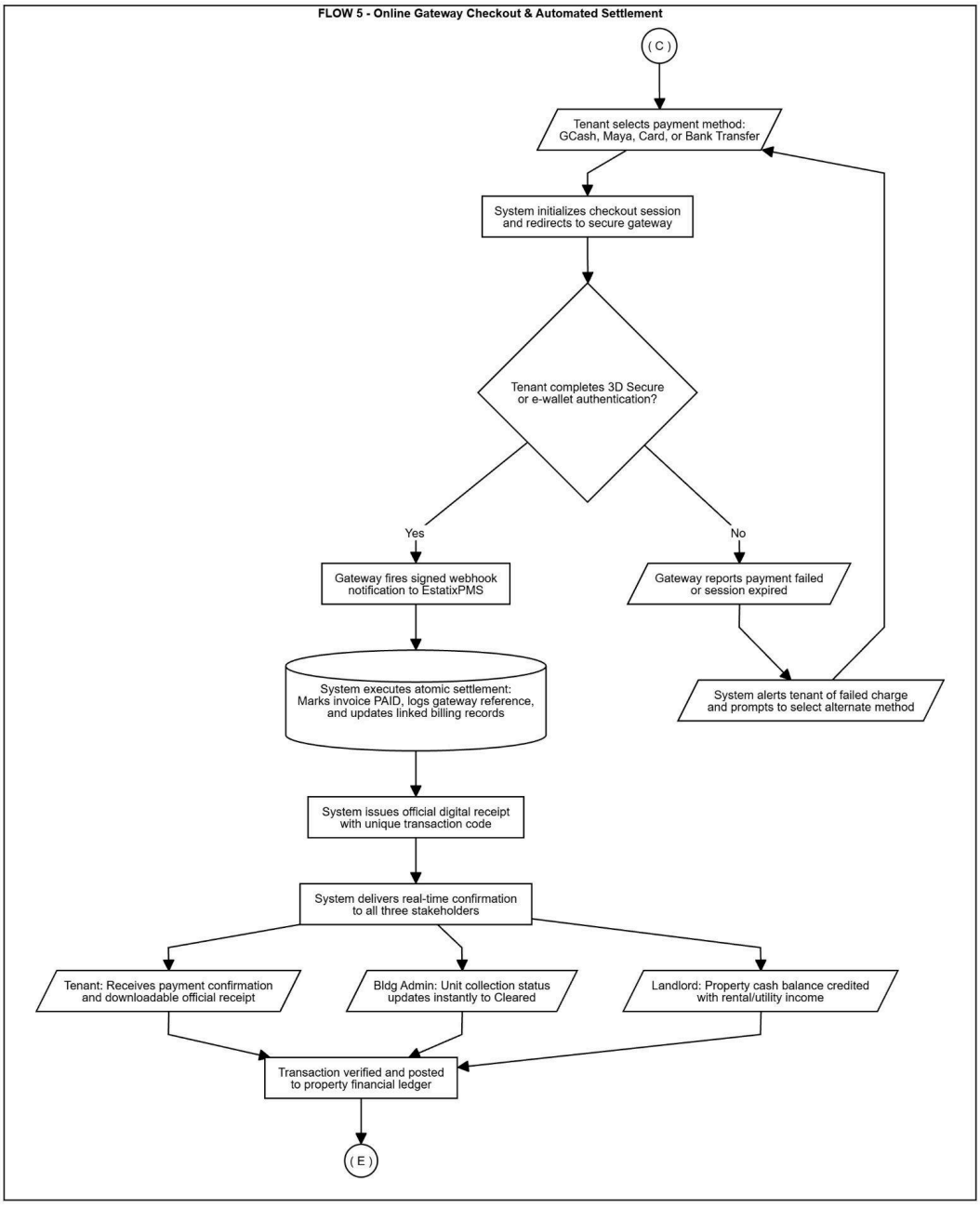
Flow 10: Payments Flowchart

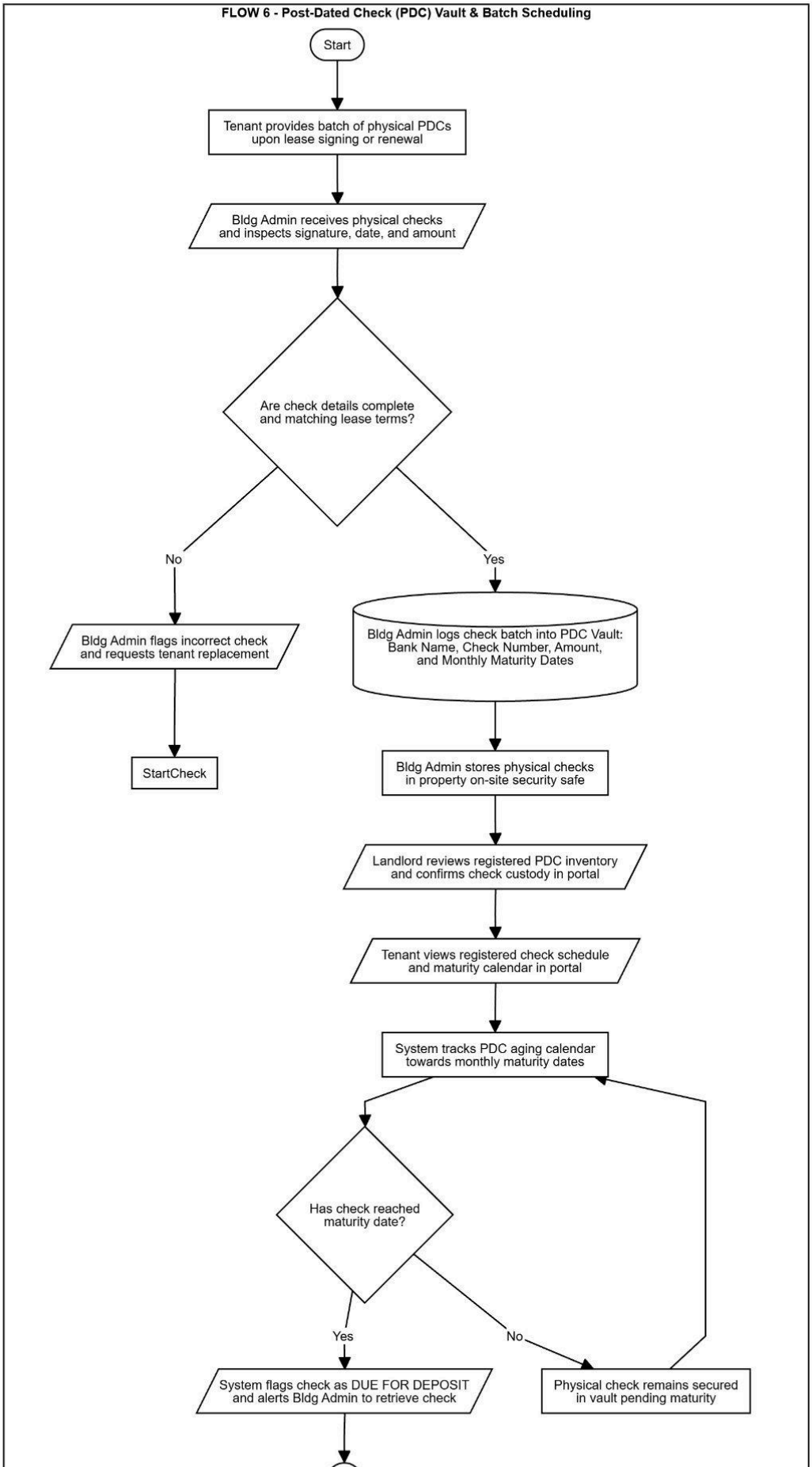


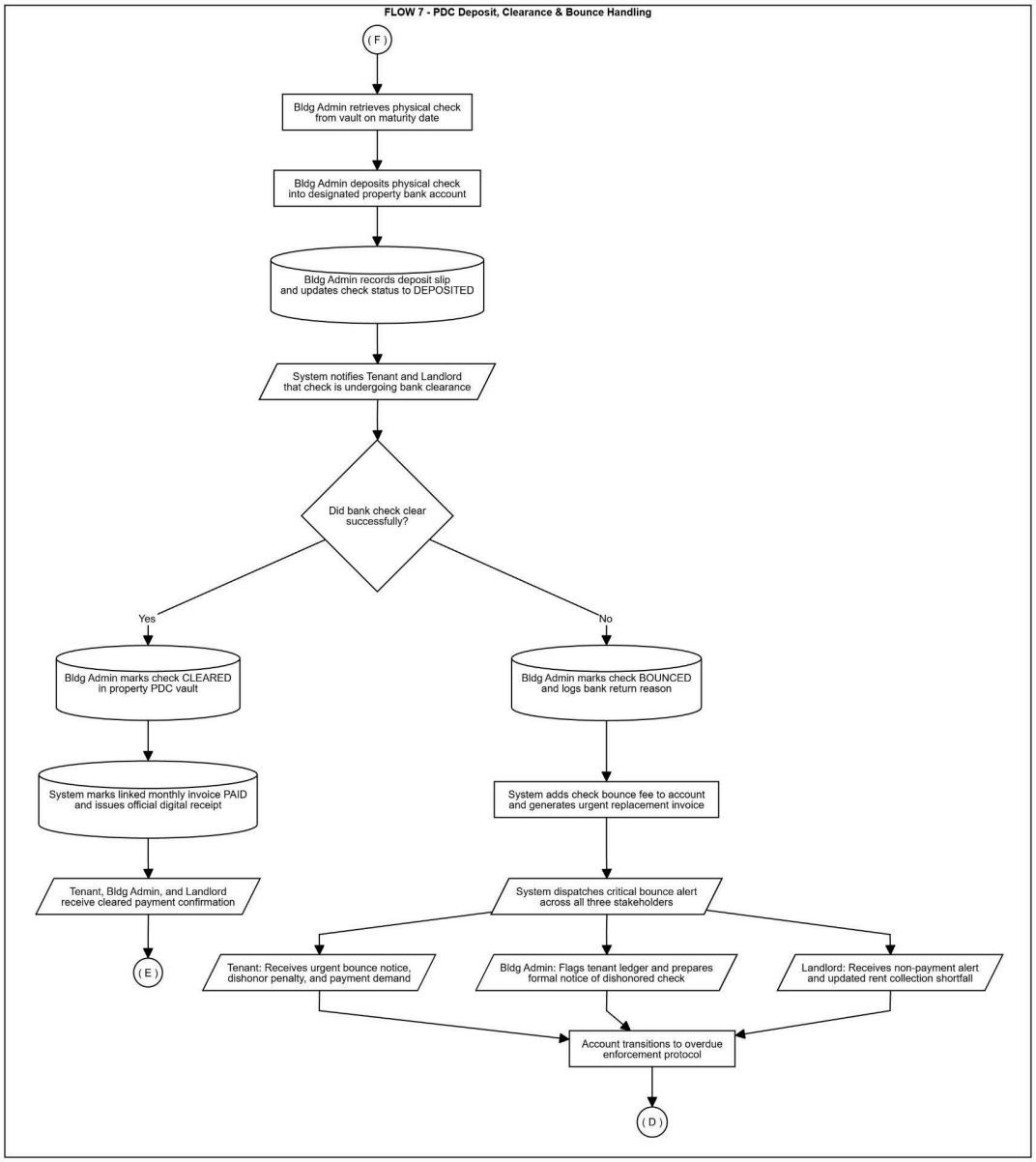

FLOW 2 - Utility Billing & Meter Invoicing (Water & Electricity)
Start
Monthly Utility Billing Cycle Initiated
Property utility charging method?
Sub-Metered
Fixed Flat Rate
Staff or Bldg Admin inspects unit meters and records values: Water m3 & Electricity kWh
Staff attaches photo proof of physical meter dials
System computes net usage: Current minus Previous Reading multiplied by rate per unit
System applies contract fixed monthly utility rate
Bldg Admin reviews usage, photo proof, and total charges
Bldg Admin approves utility calculation?
Yes
No
System generates itemized utility invoice with due date
Bldg Admin requests re-reading or recalculates adjustment
System dispatches utility invoice to all three stakeholders
Tenant: Receives bill with consumption summary and meter photos in portal
Bldg Admin: Receives utility receivable entry in property accounts ledger
Landlord: Receives utility recovery metric to audit property expenses
Utility invoice queued for tenant payment settlement
(A)

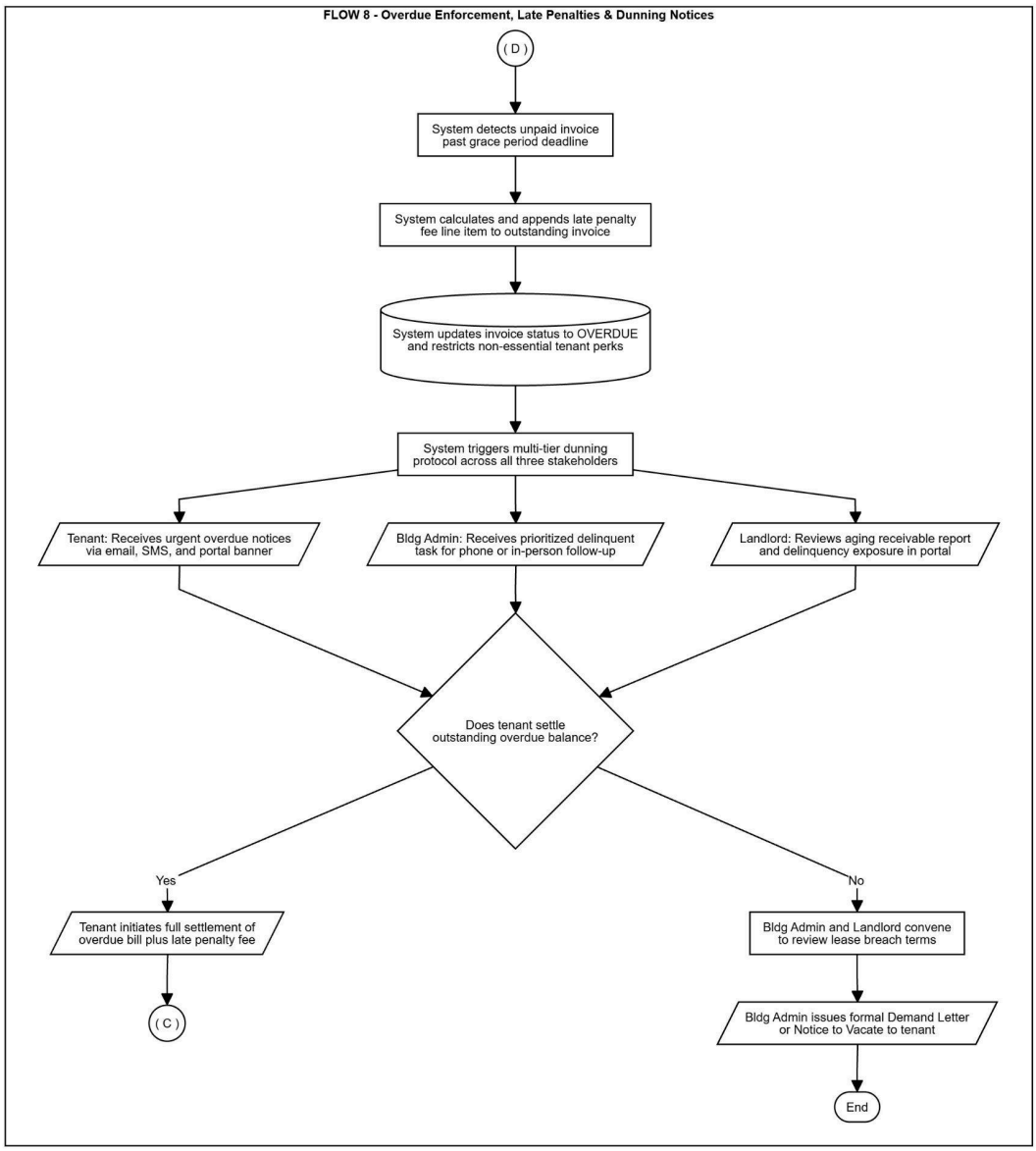

FLOW 3 - Chargeable Maintenance Repair Invoicing
Start
Maintenance Ticket Resolved by Staff
Is repair tenant chargeable?
No
Yes
System marks cost covered by property operational budget
Bldg Admin compiles billable items: Replaced parts, materials used, and contractor or labor costs
Landlord and Bldg Admin record expense under building upkeep
Bldg Admin attaches work order summary and repair completion photos
End
System generates itemized maintenance repair invoice
System distributes maintenance bill across all three stakeholders
Tenant: Receives repair bill with itemized costs and photos
Bldg Admin: Tracks maintenance reimbursement in billing queue
Landlord: Reviews repair recovery in property asset ledger
Maintenance invoice ready for tenant settlement
(A)

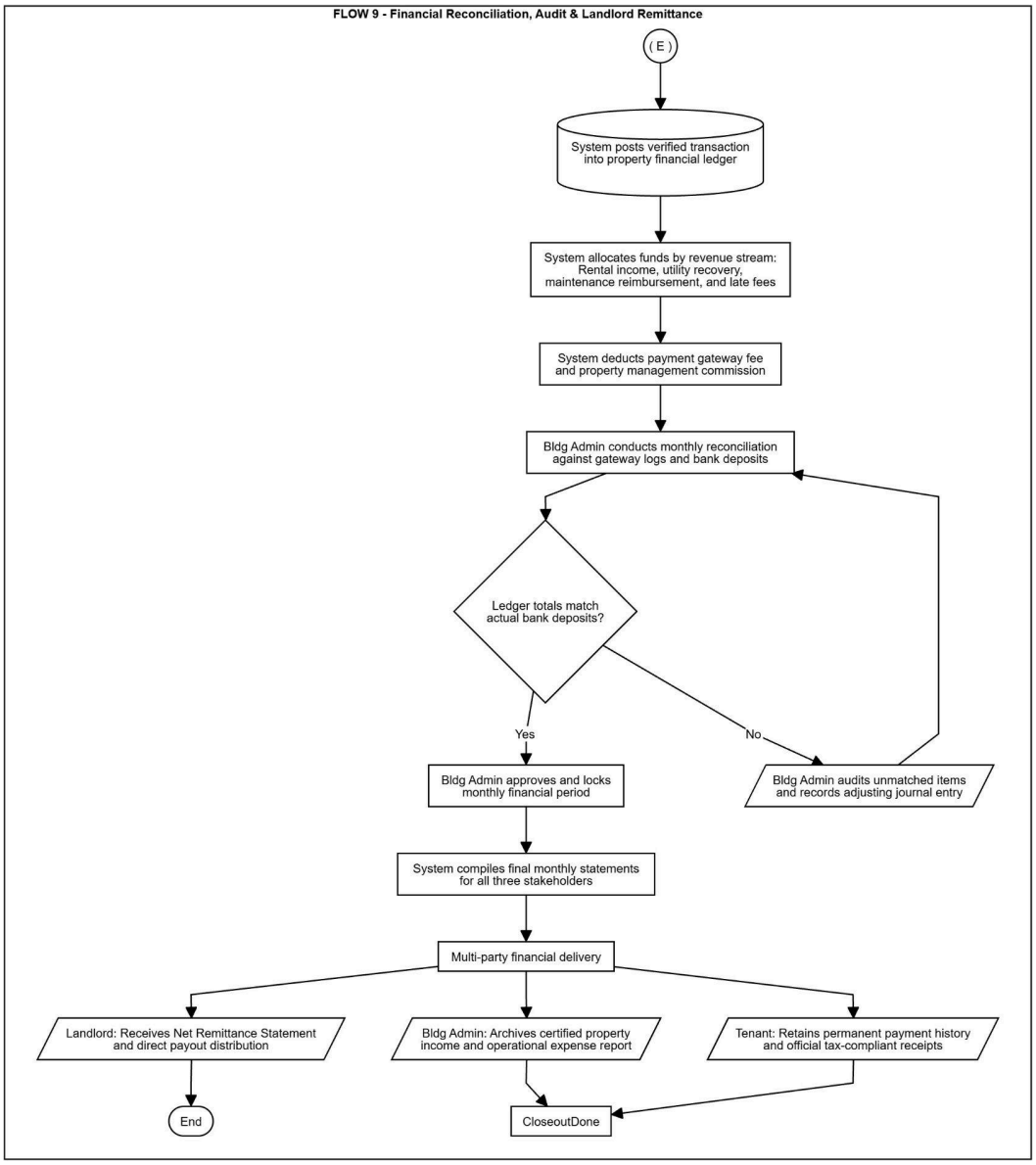

FLOW 4 - Invoice Delivery, Visibility & Channel Selection
(A)
System presents active invoice with itemized line items and due date
Real-time status synchronization across stakeholder dashboards
Tenant: Views bill details, breakdown, and due date countdown
Bldg Admin: Monitors active invoice in building collection dashboard
Landlord: Observes expected revenue in property remittance forecast
Invoice status displayed as PENDING awaiting payment fulfillment
How will the invoice be fulfilled?
Online Digital Gateway
Post-Dated Check (PDC)
Past Due / Unpaid
Tenant clicks 'Pay Now' to launch secure payment gateway checkout
Verify PDC check registered in property vault for this period
Due date and grace period expired?
Yes
No
(C)
(B)
(D)
Invoice remains active pending tenant action


FLOW 5 - Online Gateway Checkout & Automated Settlement
(C)
Tenant selects payment method: GCash, Maya, Card, or Bank Transfer
System initializes checkout session and redirects to secure gateway
Tenant completes 3D Secure or e-wallet authentication?
Yes
No
Gateway fires signed webhook notification to EstatixPMS
Gateway reports payment failed or session expired
System executes atomic settlement: Marks invoice PAID, logs gateway reference, and updates linked billing records
System alerts tenant of failed charge and prompts to select alternate method
System issues official digital receipt with unique transaction code
System delivers real-time confirmation to all three stakeholders
Tenant: Receives payment confirmation and downloadable official receipt
Bldg Admin: Unit collection status updates instantly to Cleared
Landlord: Property cash balance credited with rental/utility income
Transaction verified and posted to property financial ledger
(E)

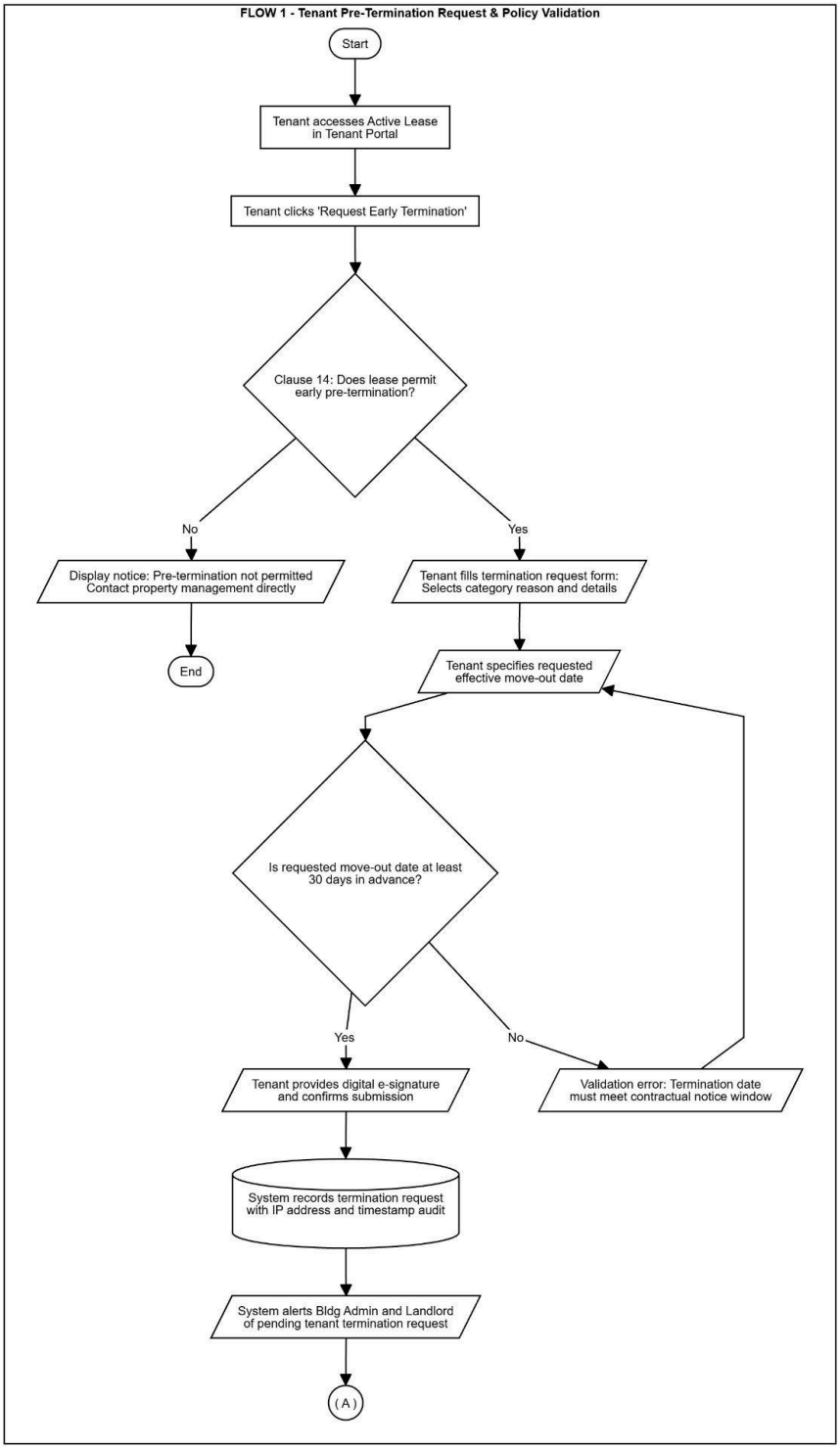

FLOW 6 - Post-Dated Check (PDC) Vault & Batch Scheduling
Start
Tenant provides batch of physical PDCs upon lease signing or renewal
Bldg Admin receives physical checks and inspects signature, date, and amount
Are check details complete and matching lease terms?
No
Yes
Bldg Admin flags incorrect check and requests tenant replacement
StartCheck
Bldg Admin logs check batch into PDC Vault: Bank Name, Check Number, Amount, and Monthly Maturity Dates
Bldg Admin stores physical checks in property on-site security safe
Landlord reviews registered PDC inventory and confirms check custody in portal
Tenant views registered check schedule and maturity calendar in portal
System tracks PDC aging calendar towards monthly maturity dates
Has check reached maturity date?
Yes
No
System flags check as DUE FOR DEPOSIT and alerts Bldg Admin to retrieve check
Physical check remains secured in vault pending maturity

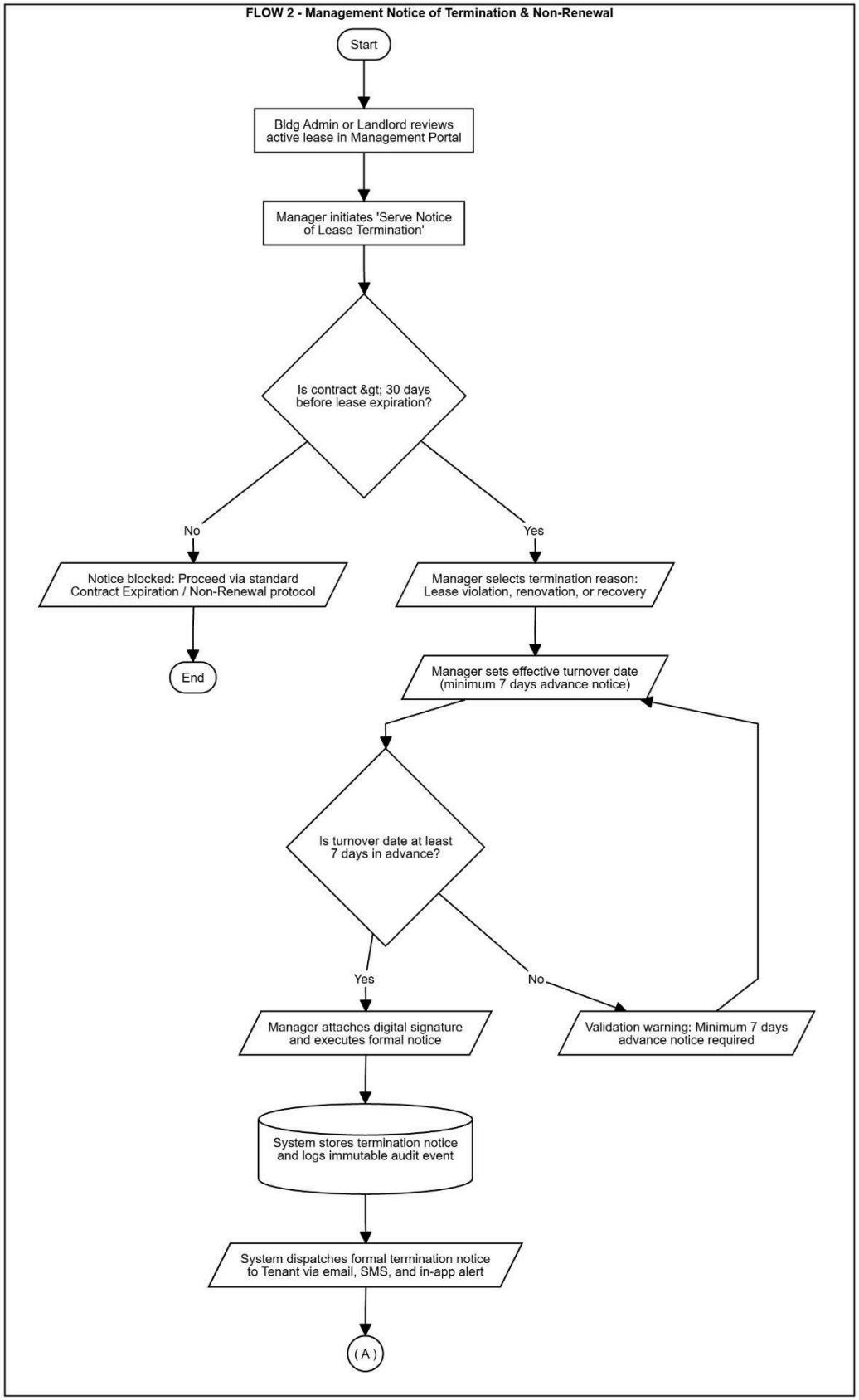

FLOW 7 - PDC Deposit, Clearance & Bounce Handling
(F)
Bldg Admin retrieves physical check from vault on maturity date
Bldg Admin deposits physical check into designated property bank account
Bldg Admin records deposit slip and updates check status to DEPOSITED
System notifies Tenant and Landlord that check is undergoing bank clearance
Did bank check clear successfully?
Yes
No
Bldg Admin marks check CLEARED in property PDC vault
Bldg Admin marks check BOUNCED and logs bank return reason
System marks linked monthly invoice PAID and issues official digital receipt
System adds check bounce fee to account and generates urgent replacement invoice
Tenant, Bldg Admin, and Landlord receive cleared payment confirmation
System dispatches critical bounce alert across all three stakeholders
(E)
Tenant: Receives urgent bounce notice, dishonor penalty, and payment demand
Bldg Admin: Flags tenant ledger and prepares formal notice of dishonored check
Landlord: Receives non-payment alert and updated rent collection shortfall
Account transitions to overdue enforcement protocol
(D)

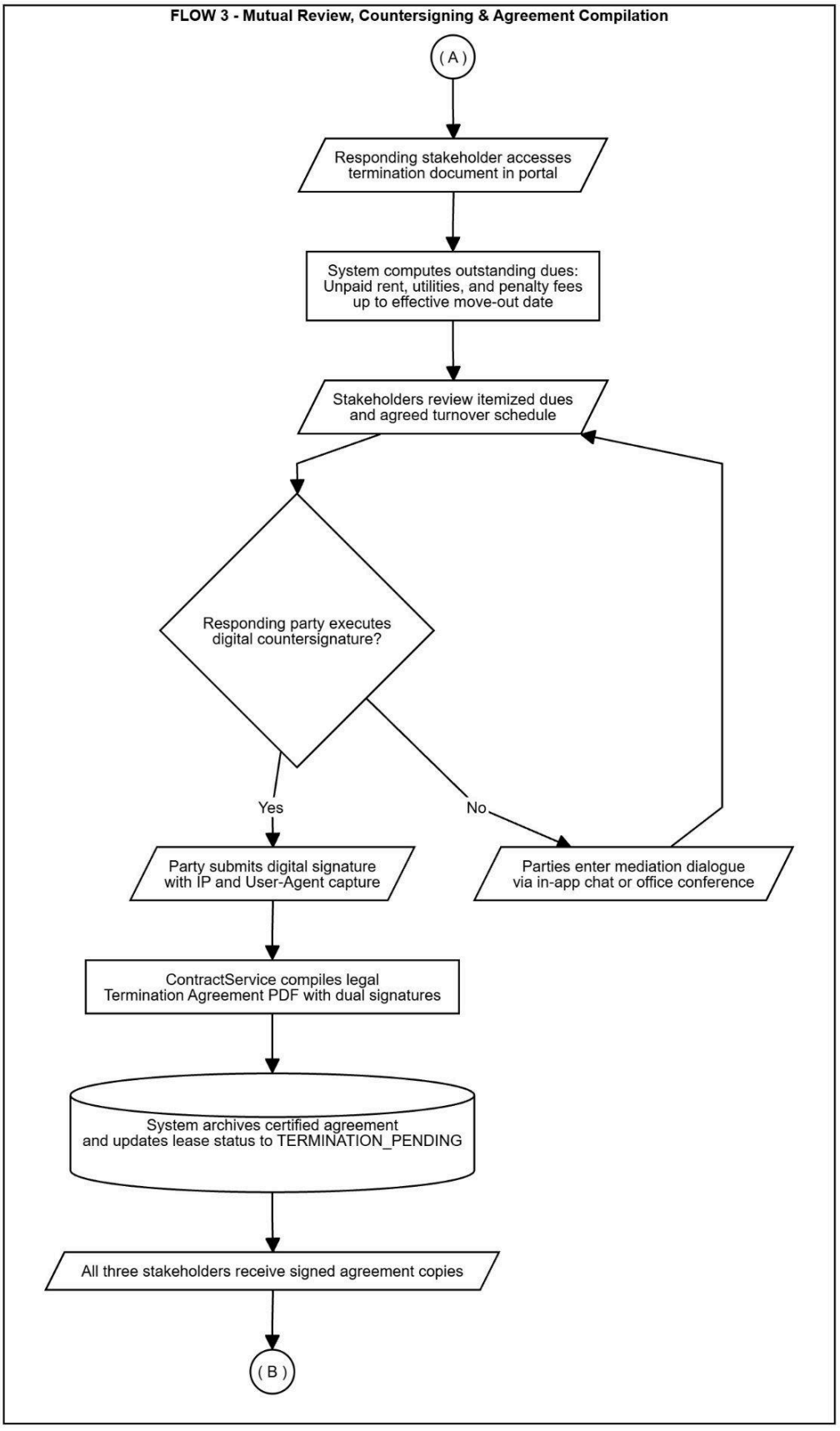

FLOW 8 - Overdue Enforcement, Late Penalties & Dunning Notices
(D)
System detects unpaid invoice past grace period deadline
System calculates and appends late penalty fee line item to outstanding invoice
System updates invoice status to OVERDUE and restricts non-essential tenant perks
System triggers multi-tier dunning protocol across all three stakeholders
Tenant: Receives urgent overdue notices via email, SMS, and portal banner
Bldg Admin: Receives prioritized delinquent task for phone or in-person follow-up
Landlord: Reviews aging receivable report and delinquency exposure in portal
Does tenant settle outstanding overdue balance?
Yes
No
Tenant initiates full settlement of overdue bill plus late penalty fee
(C)
Bldg Admin and Landlord convene to review lease breach terms
Bldg Admin issues formal Demand Letter or Notice to Vacate to tenant
End

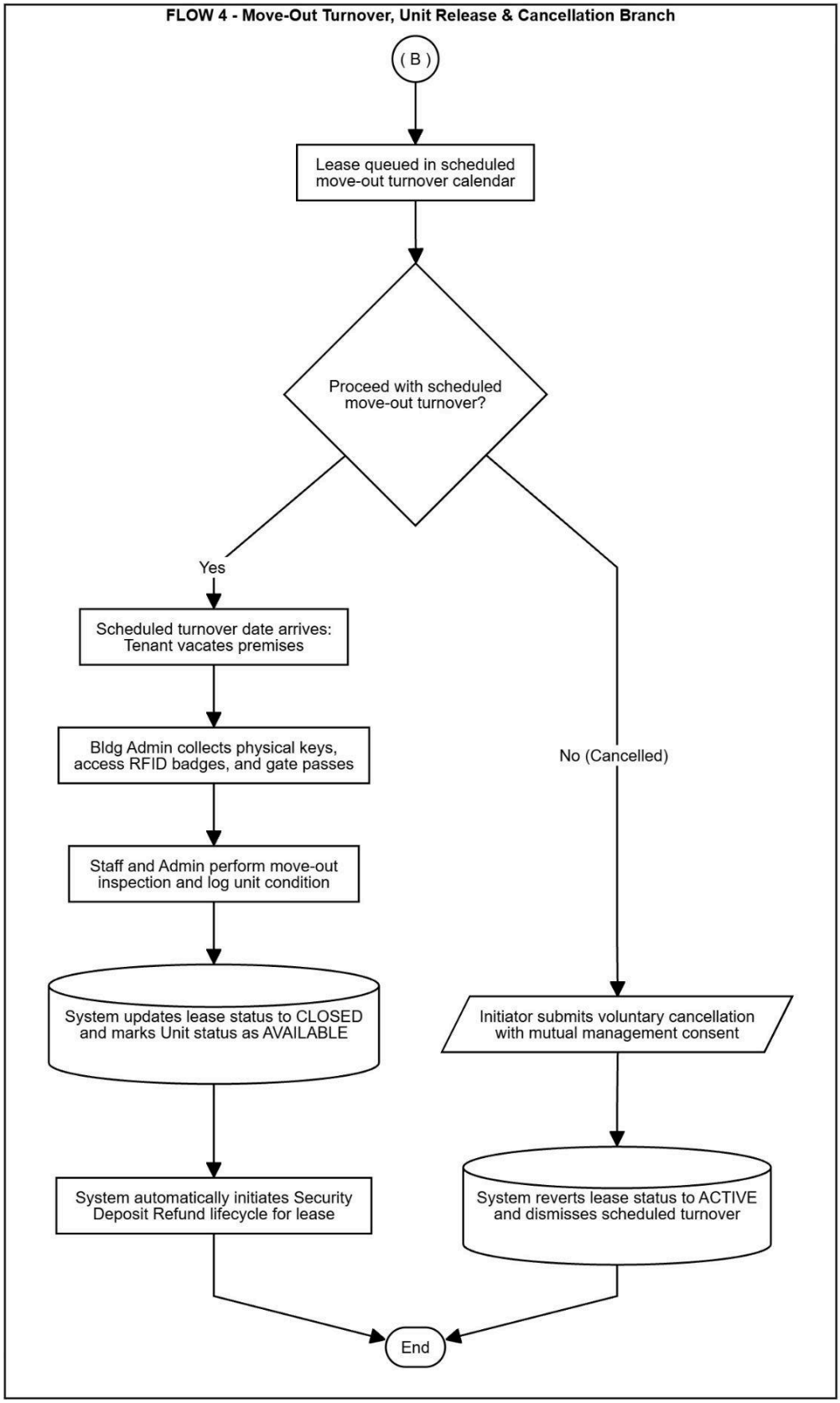

FLOW 9 - Financial Reconciliation, Audit & Landlord Remittance
(E)
System posts verified transaction into property financial ledger
System allocates funds by revenue stream: Rental income, utility recovery, maintenance reimbursement, and late fees
System deducts payment gateway fee and property management commission
Bldg Admin conducts monthly reconciliation against gateway logs and bank deposits
Ledger totals match actual bank deposits?
Yes
No
Bldg Admin approves and locks monthly financial period
Bldg Admin audits unmatched items and records adjusting journal entry
System compiles final monthly statements for all three stakeholders
Multi-party financial delivery
Landlord: Receives Net Remittance Statement and direct payout distribution
Bldg Admin: Archives certified property income and operational expense report
Tenant: Retains permanent payment history and official tax-compliant receipts
End
CloseoutDone

Flow 11: Lease Termination Flowchart

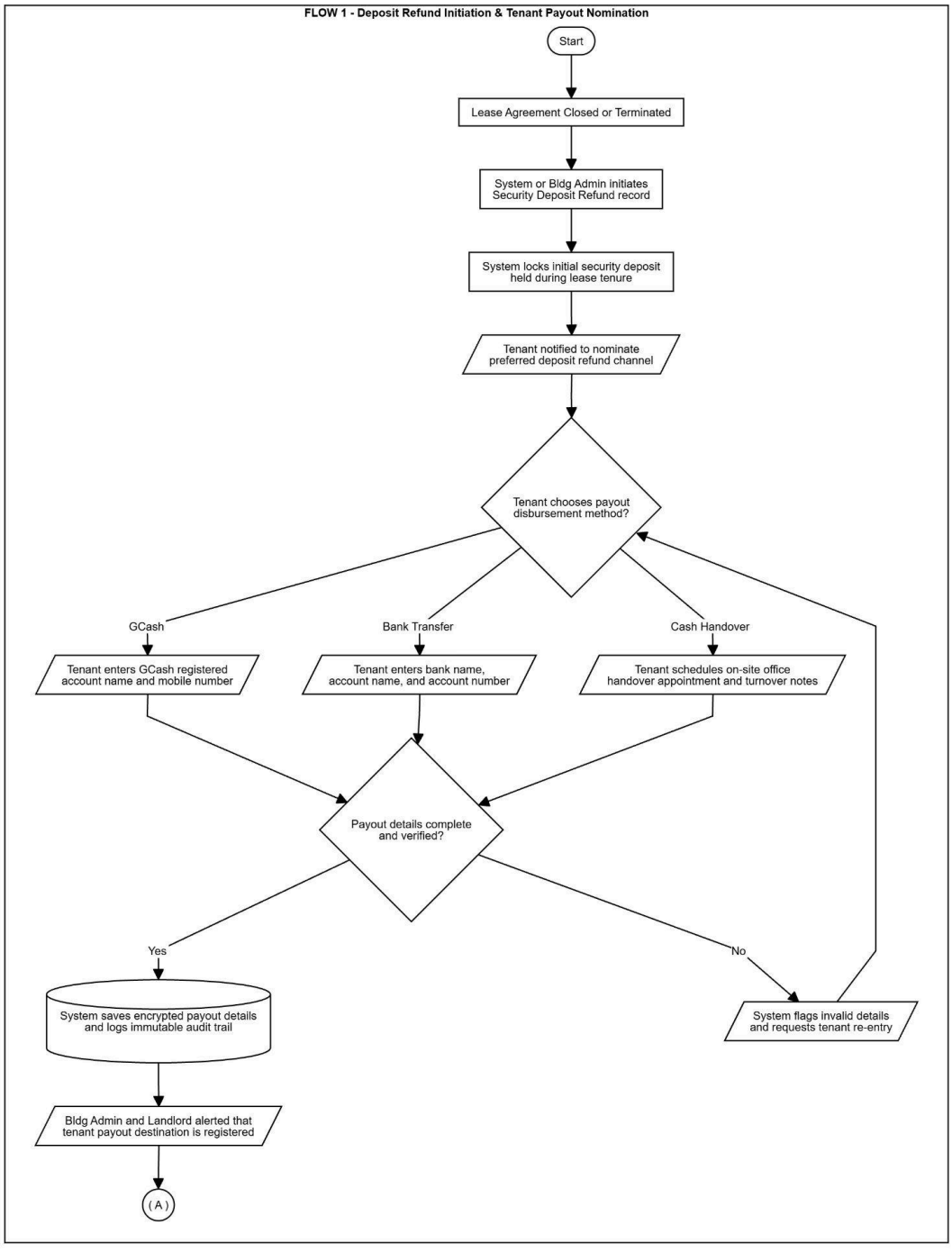


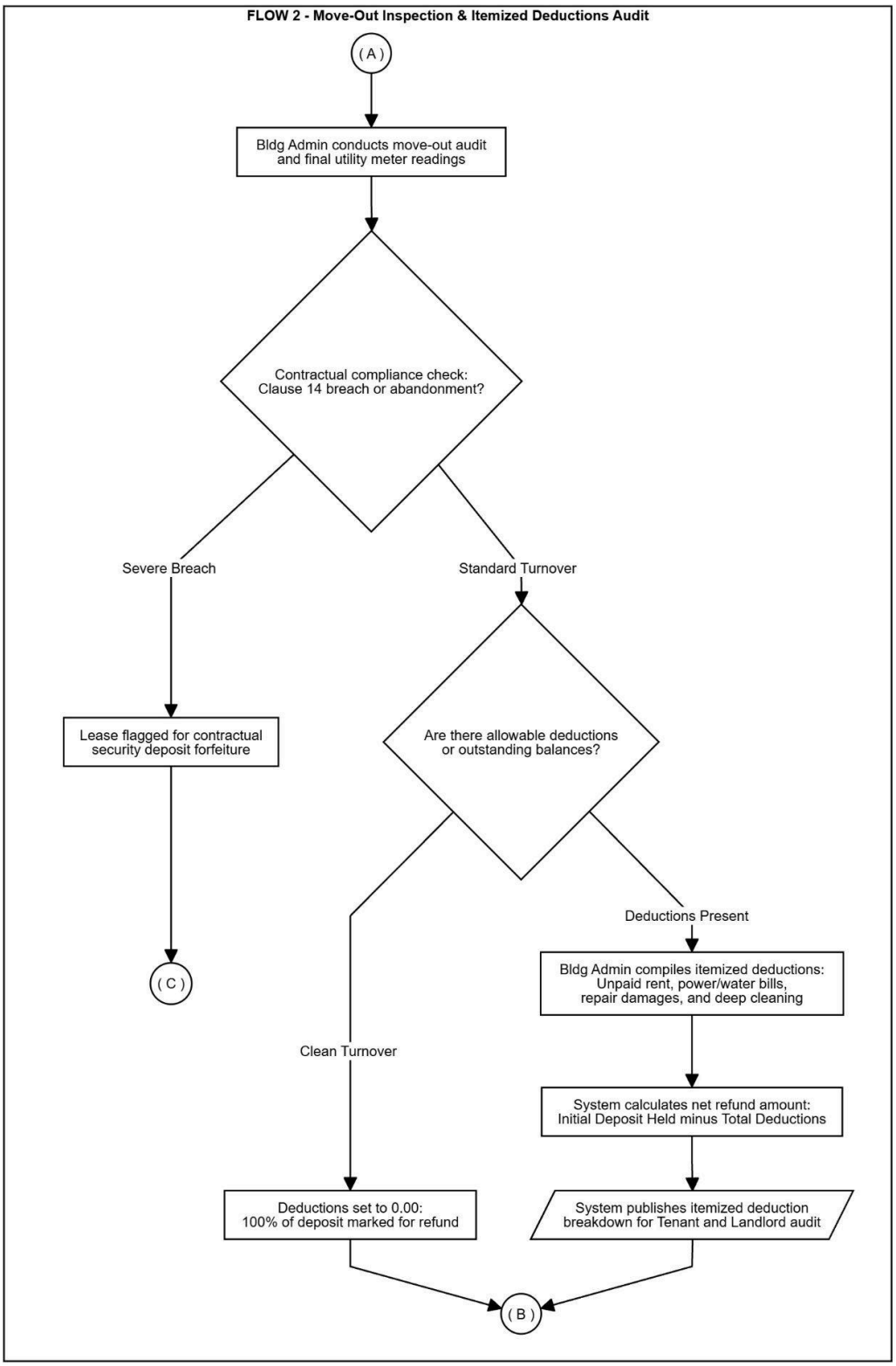

FLOW 2 - Management Notice of Termination & Non-Renewal
Start
Bldg Admin or Landlord reviews active lease in Management Portal
Manager initiates 'Serve Notice of Lease Termination'
Is contract > 30 days before lease expiration?
No
Yes
Notice blocked: Proceed via standard Contract Expiration / Non-Renewal protocol
End
Manager selects termination reason: Lease violation, renovation, or recovery
Manager sets effective turnover date (minimum 7 days advance notice)
Is turnover date at least 7 days in advance?
Yes
No
Manager attaches digital signature and executes formal notice
Validation warning: Minimum 7 days advance notice required
System stores termination notice and logs immutable audit event
System dispatches formal termination notice to Tenant via email, SMS, and in-app alert
(A)

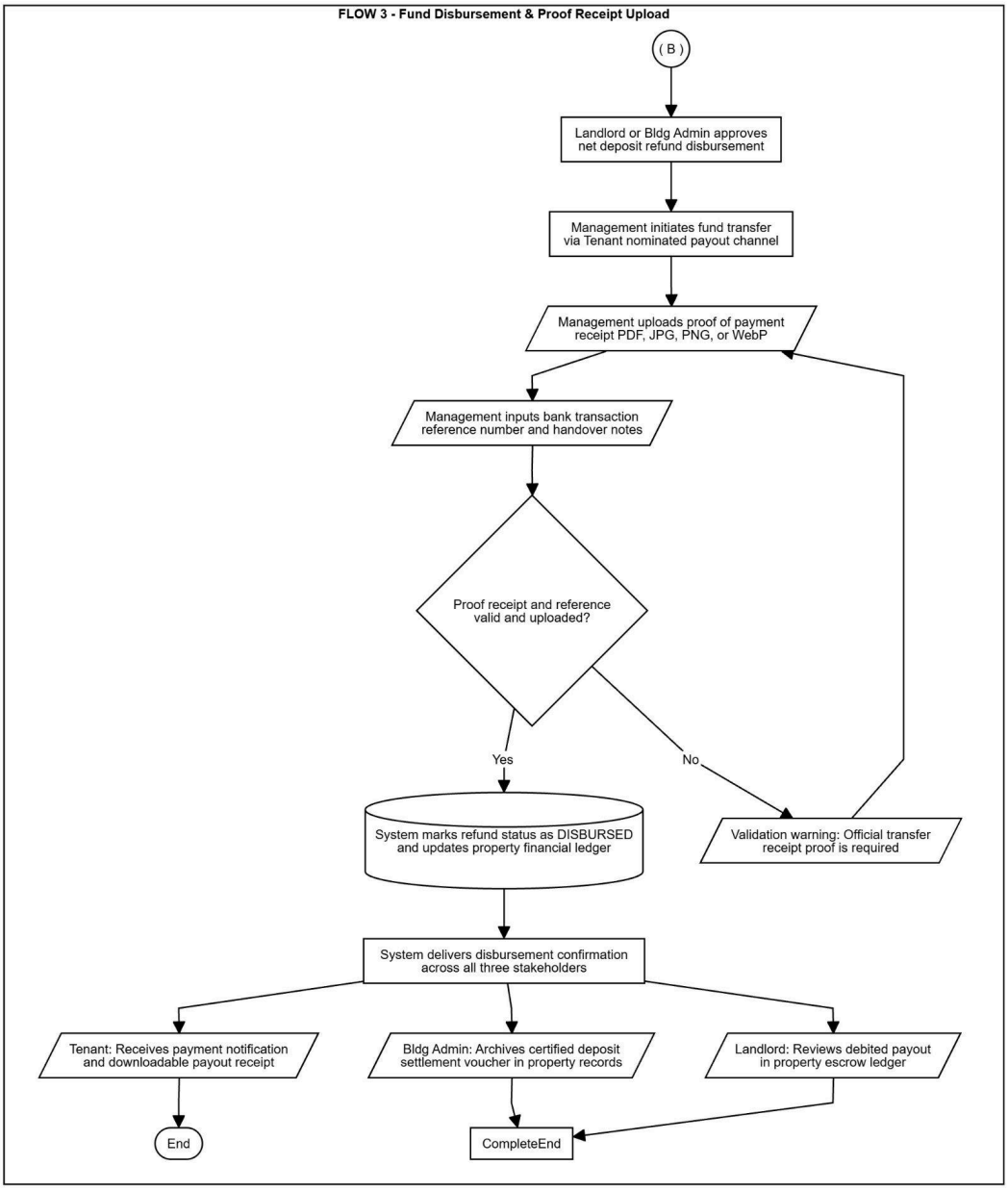

FLOW 3 - Mutual Review, Countersigning & Agreement Compilation
(A)
Responding stakeholder accesses termination document in portal
System computes outstanding dues: Unpaid rent, utilities, and penalty fees up to effective move-out date
Stakeholders review itemized dues and agreed turnover schedule
Responding party executes digital countersignature?
Yes
No
Party submits digital signature with IP and User-Agent capture
Parties enter mediation dialogue via in-app chat or office conference
ContractService compiles legal Termination Agreement PDF with dual signatures
System archives certified agreement and updates lease status to TERMINATION_PENDING
All three stakeholders receive signed agreement copies
(B)

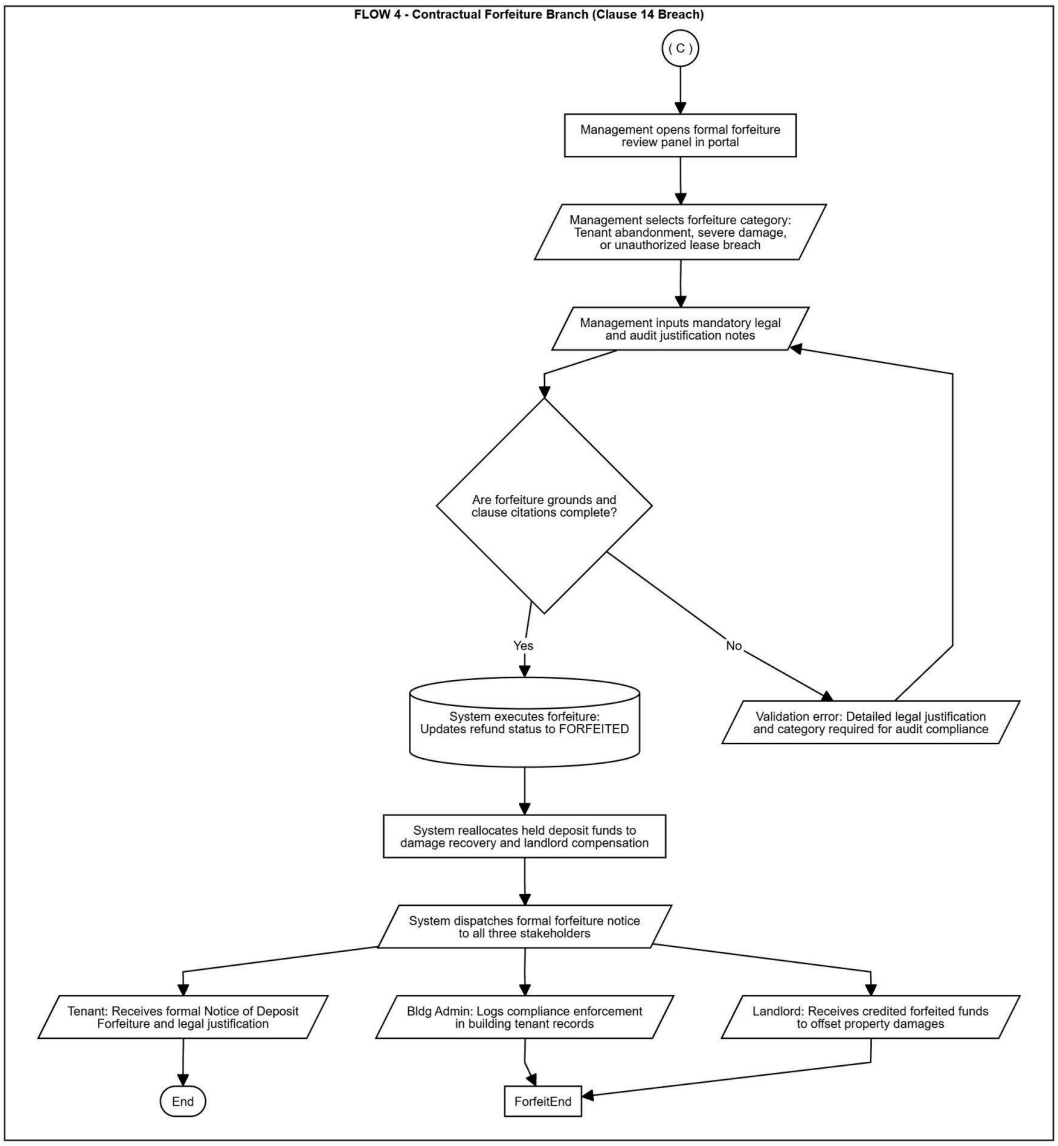

FLOW 4 - Move-Out Turnover, Unit Release & Cancellation Branch
(B)
Lease queued in scheduled move-out turnover calendar
Proceed with scheduled move-out turnover?
Yes
No (Cancelled)
Scheduled turnover date arrives: Tenant vacates premises
Bldg Admin collects physical keys, access RFID badges, and gate passes
Staff and Admin perform move-out inspection and log unit condition
System updates lease status to CLOSED and marks Unit status as AVAILABLE
System automatically initiates Security Deposit Refund lifecycle for lease
Initiator submits voluntary cancellation with mutual management consent
System reverts lease status to ACTIVE and dismisses scheduled turnover
End

Flow 12: Security Deposit Refund Flowchart

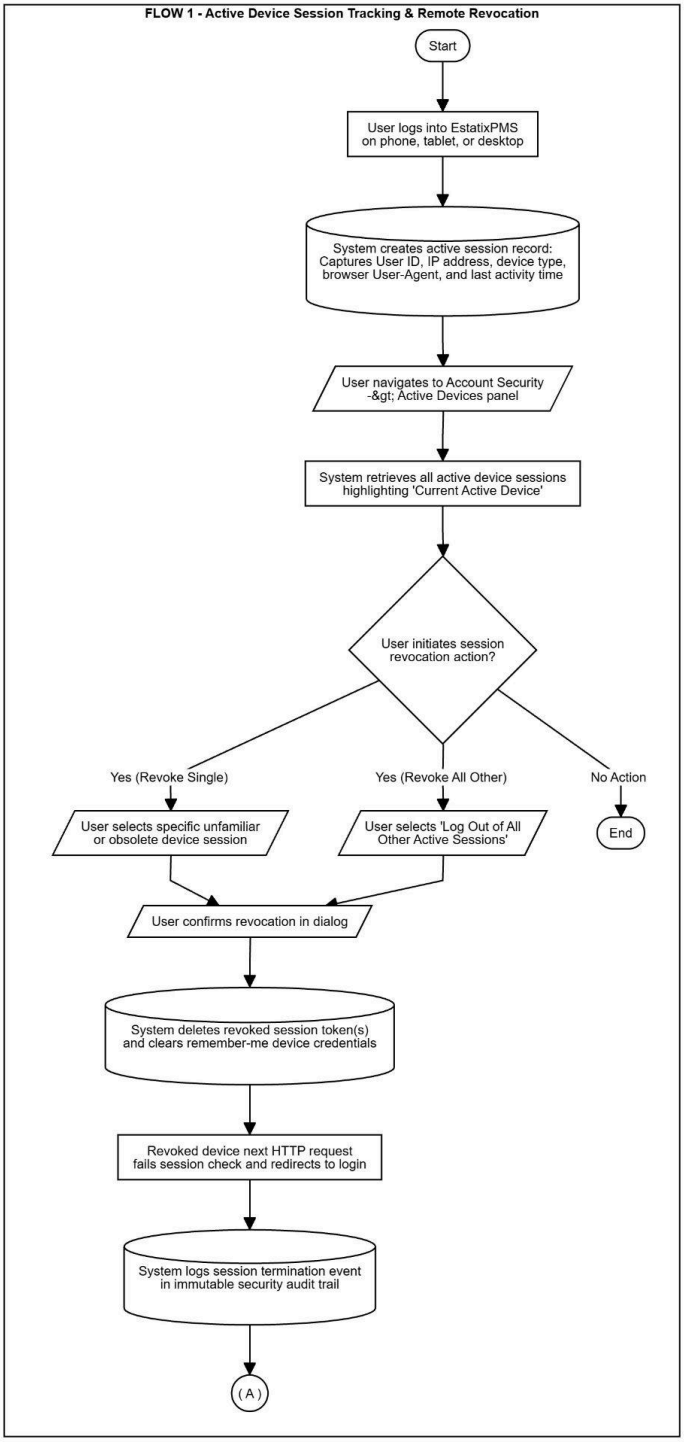


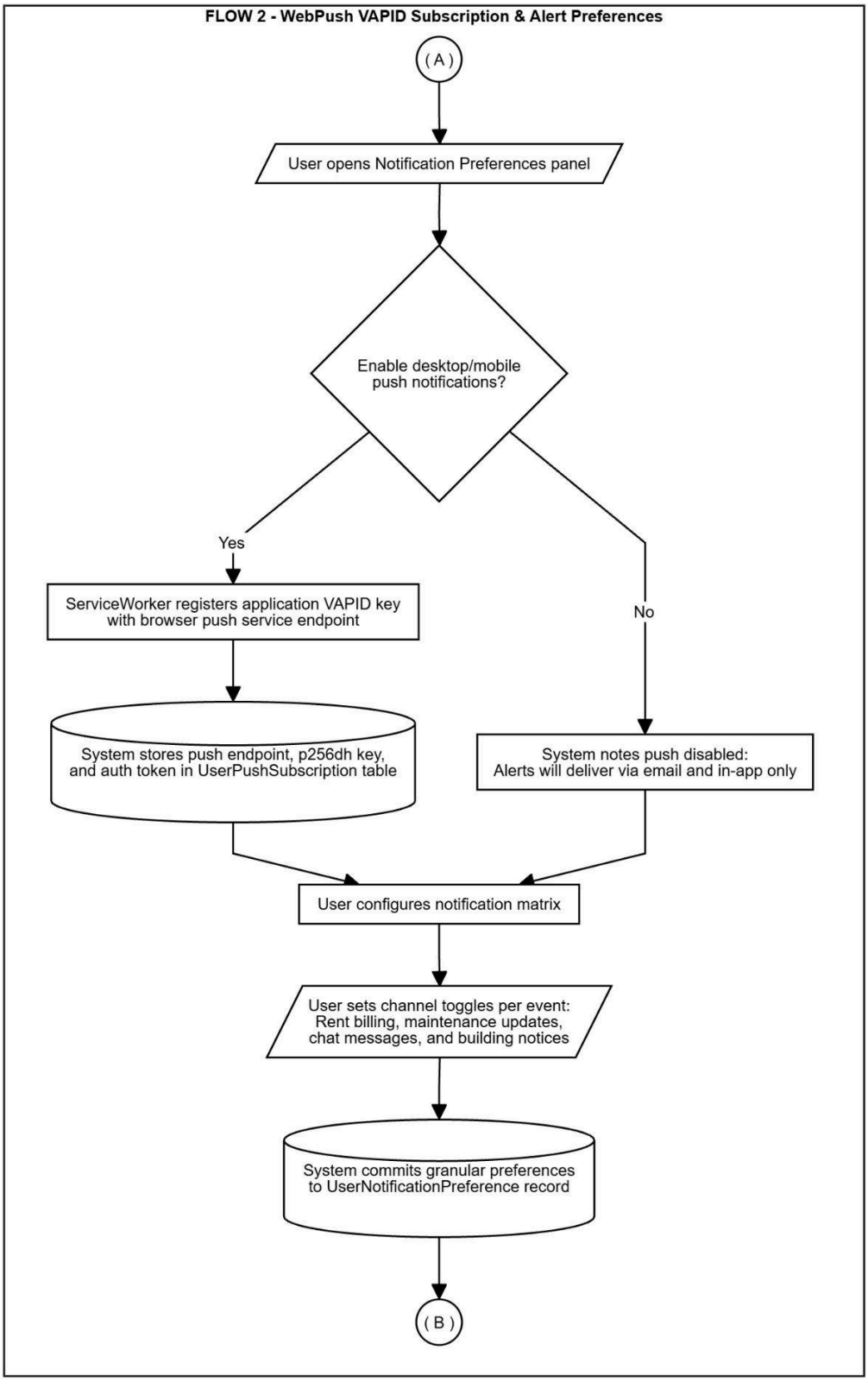

FLOW 2 - Move-Out Inspection & Itemized Deductions Audit
(A)
Bldg Admin conducts move-out audit and final utility meter readings
Contractual compliance check: Clause 14 breach or abandonment?
Severe Breach
Standard Turnover
Lease flagged for contractual security deposit forfeiture
Are there allowable deductions or outstanding balances?
(C)
Deductions Present
Bldg Admin compiles itemized deductions: Unpaid rent, power/water bills, repair damages, and deep cleaning
Clean Turnover
System calculates net refund amount: Initial Deposit Held minus Total Deductions
Deductions set to 0.00: 100% of deposit marked for refund
System publishes itemized deduction breakdown for Tenant and Landlord audit
(B)

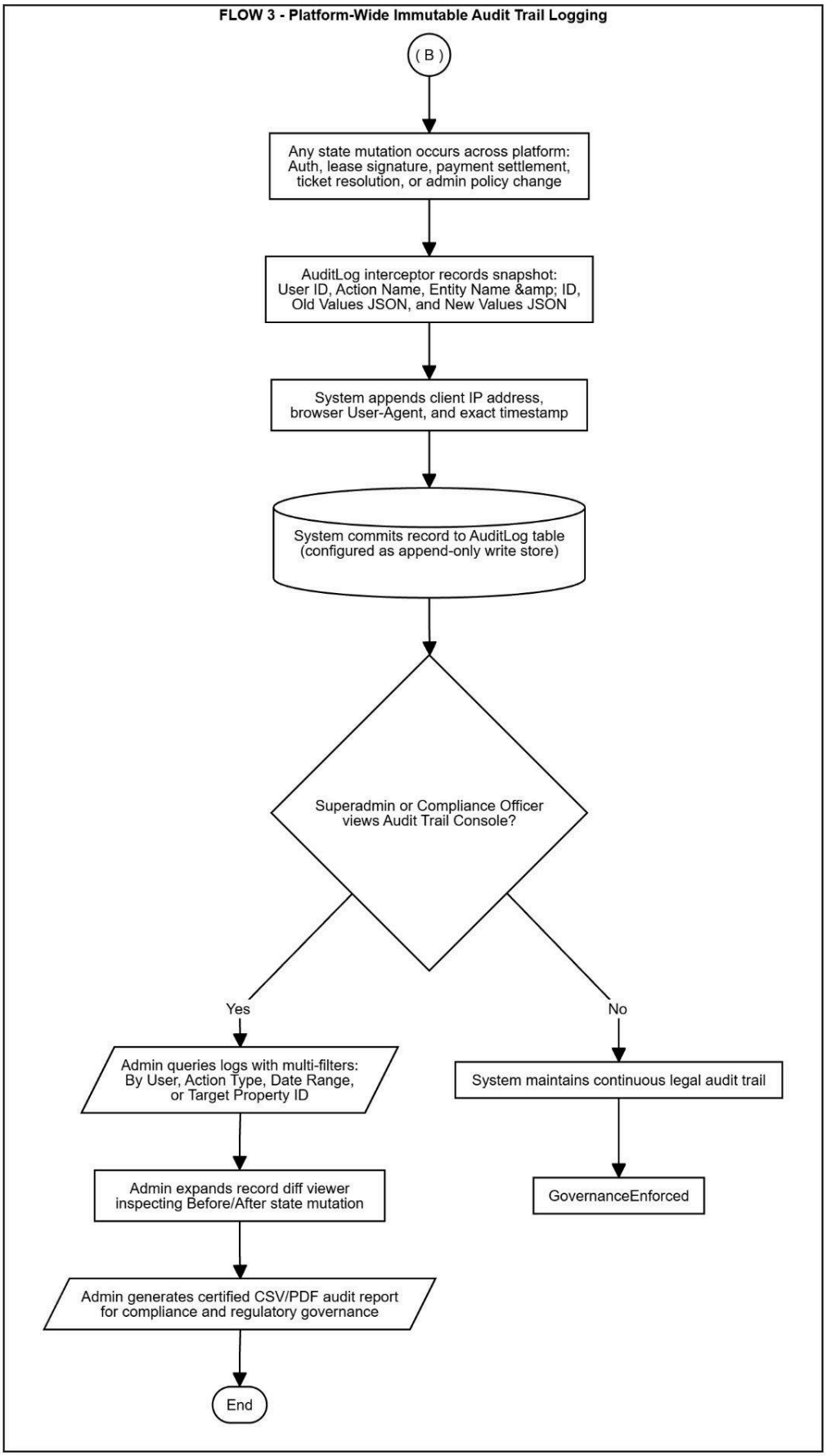

FLOW 3 - Fund Disbursement & Proof Receipt Upload
(B)
Landlord or Bldg Admin approves net deposit refund disbursement
Management initiates fund transfer via Tenant nominated payout channel
Management uploads proof of payment receipt PDF, JPG, PNG, or WebP
Management inputs bank transaction reference number and handover notes
Proof receipt and reference valid and uploaded?
Yes
No
System marks refund status as DISBURSED and updates property financial ledger
Validation warning: Official transfer receipt proof is required
System delivers disbursement confirmation across all three stakeholders
Tenant: Receives payment notification and downloadable payout receipt
Bldg Admin: Archives certified deposit settlement voucher in property records
Landlord: Reviews debited payout in property escrow ledger
End
CompleteEnd


FLOW 4 - Contractual Forfeiture Branch (Clause 14 Breach)
(C)
Management opens formal forfeiture review panel in portal
Management selects forfeiture category: Tenant abandonment, severe damage, or unauthorized lease breach
Management inputs mandatory legal and audit justification notes
Are forfeiture grounds and clause citations complete?
Yes
No
System executes forfeiture: Updates refund status to FORFEITED
Validation error: Detailed legal justification and category required for audit compliance
System reallocates held deposit funds to damage recovery and landlord compensation
System dispatches formal forfeiture notice to all three stakeholders
Tenant: Receives formal Notice of Deposit Forfeiture and legal justification
Bldg Admin: Logs compliance enforcement in building tenant records
Landlord: Receives credited forfeited funds to offset property damages
End
ForfeitEnd

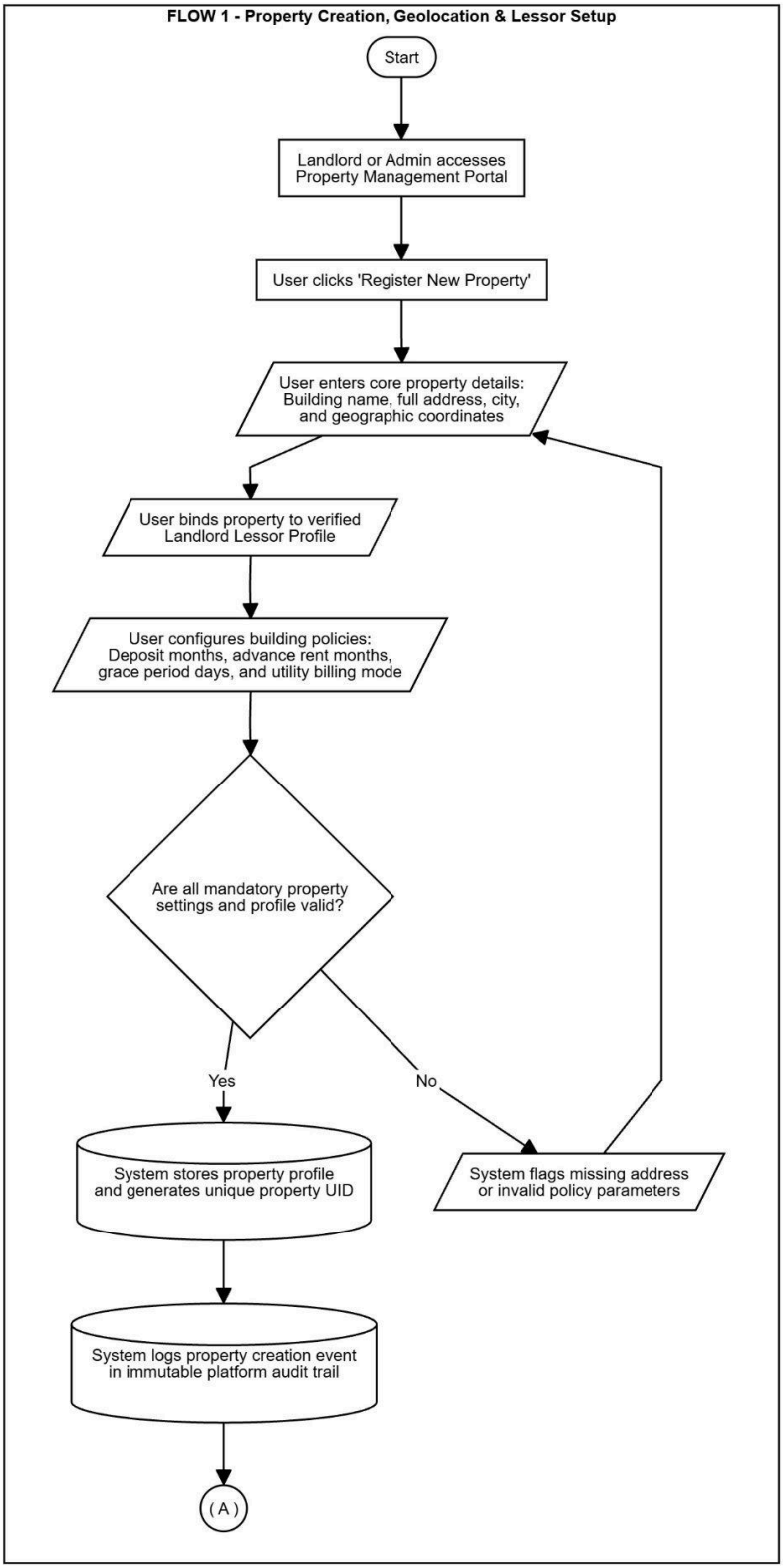
Flow 13: Session Governance Flowchart

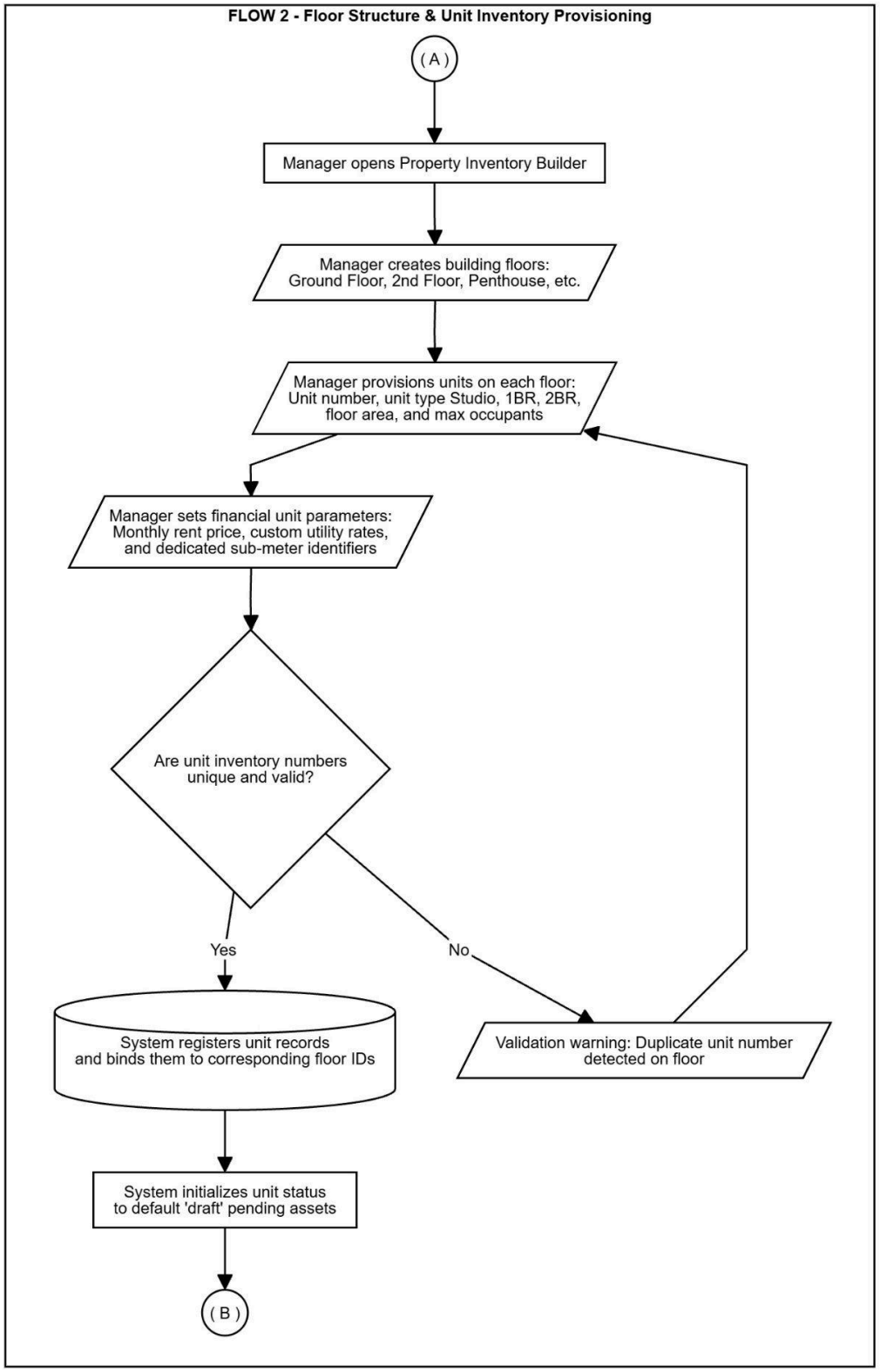


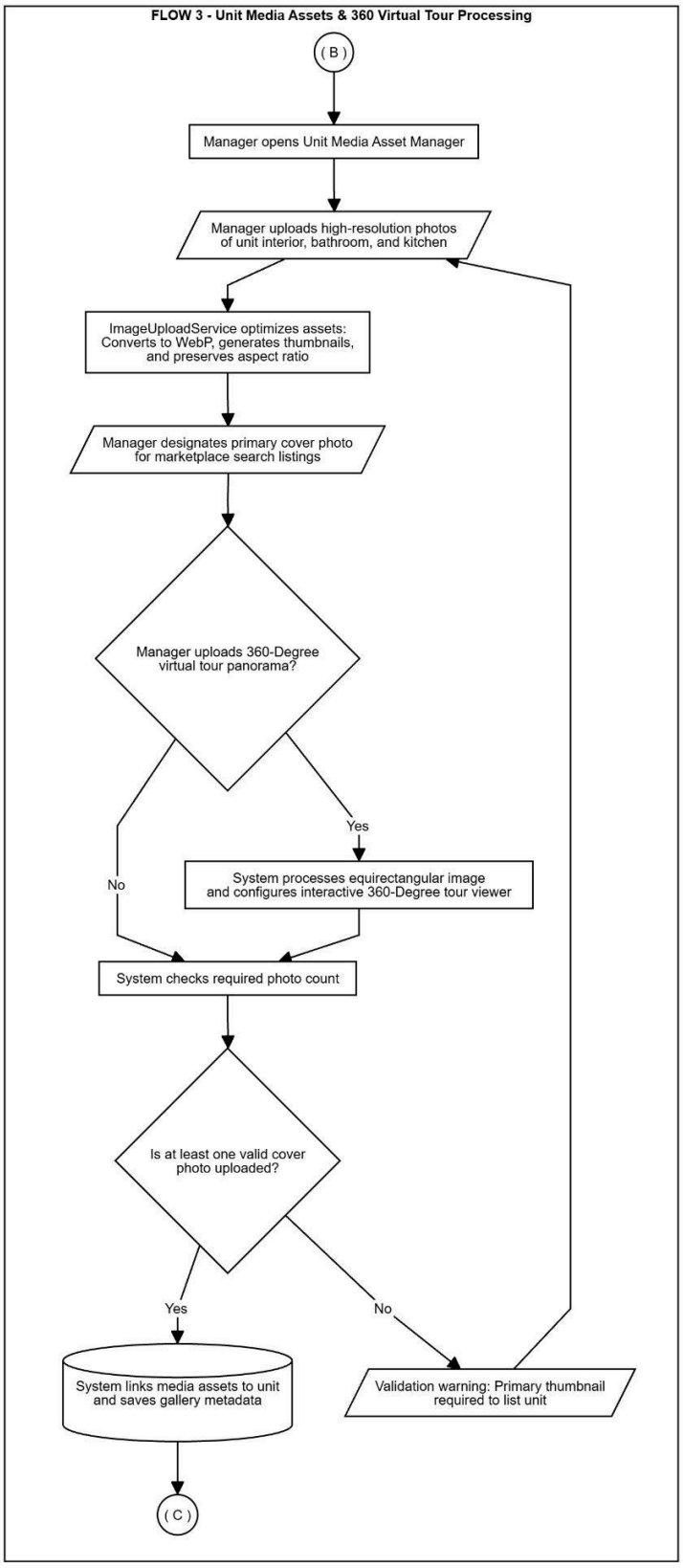

FLOW 2 - WebPush VAPID Subscription & Alert Preferences
(A)
User opens Notification Preferences panel
Enable desktop/mobile push notifications?
Yes
No
ServiceWorker registers application VAPID key with browser push service endpoint
System stores push endpoint, p256dh key, and auth token in UserPushSubscription table
System notes push disabled: Alerts will deliver via email and in-app only
User configures notification matrix
User sets channel toggles per event: Rent billing, maintenance updates, chat messages, and building notices
System commits granular preferences to UserNotificationPreference record
(B)

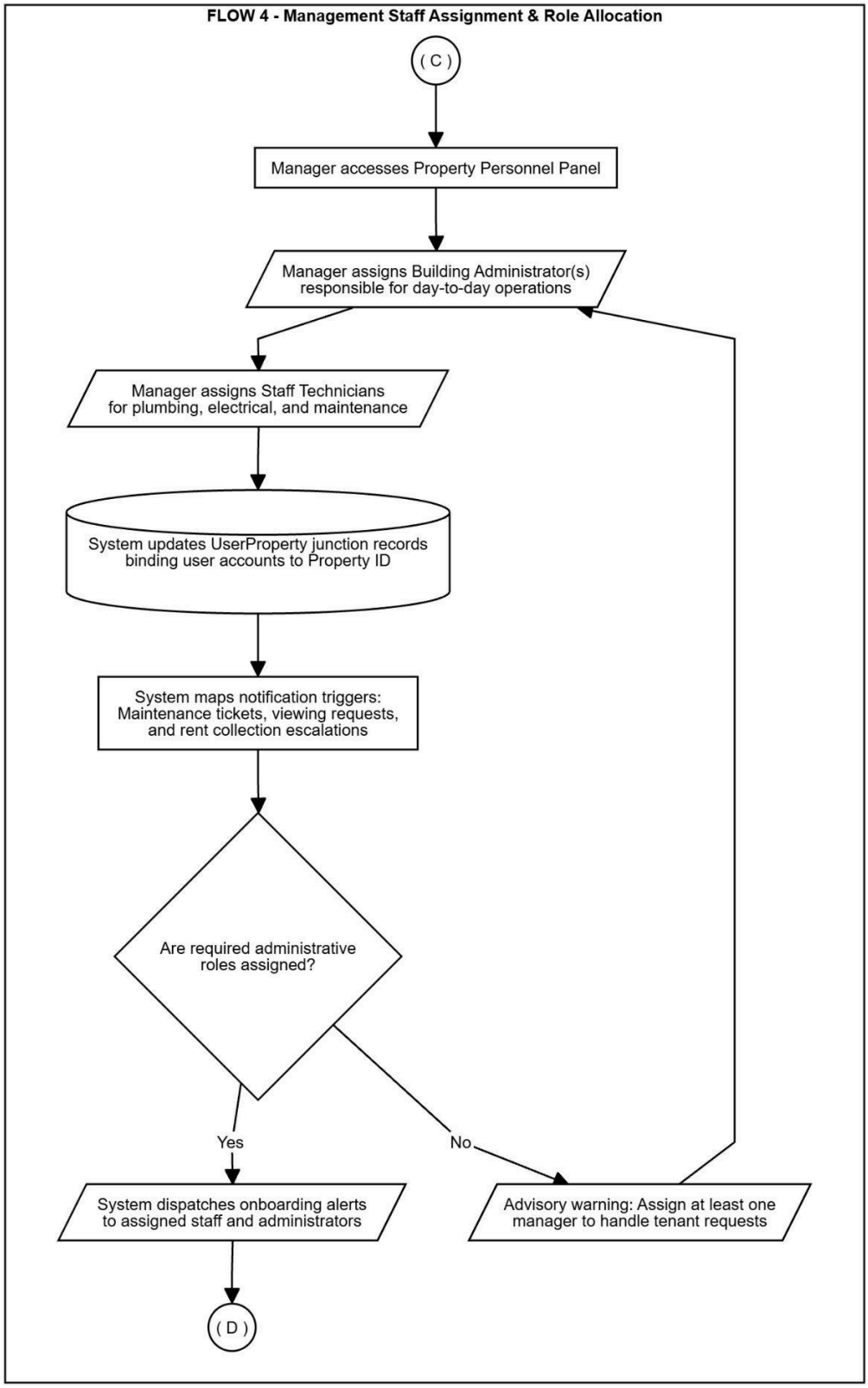

FLOW 3 - Platform-Wide Immutable Audit Trail Logging
(B)
Any state mutation occurs across platform: Auth, lease signature, payment settlement, ticket resolution, or admin policy change
AuditLog interceptor records snapshot: User ID, Action Name, Entity Name & ID, Old Values JSON, and New Values JSON
System appends client IP address, browser User-Agent, and exact timestamp
System commits record to AuditLog table (configured as append-only write store)
Superadmin or Compliance Officer views Audit Trail Console?
Yes
No
Admin queries logs with multi-filters: By User, Action Type, Date Range, or Target Property ID
Admin expands record diff viewer inspecting Before/After state mutation
Admin generates certified CSV/PDF audit report for compliance and regulatory governance
End
System maintains continuous legal audit trail
GovernanceEnforced

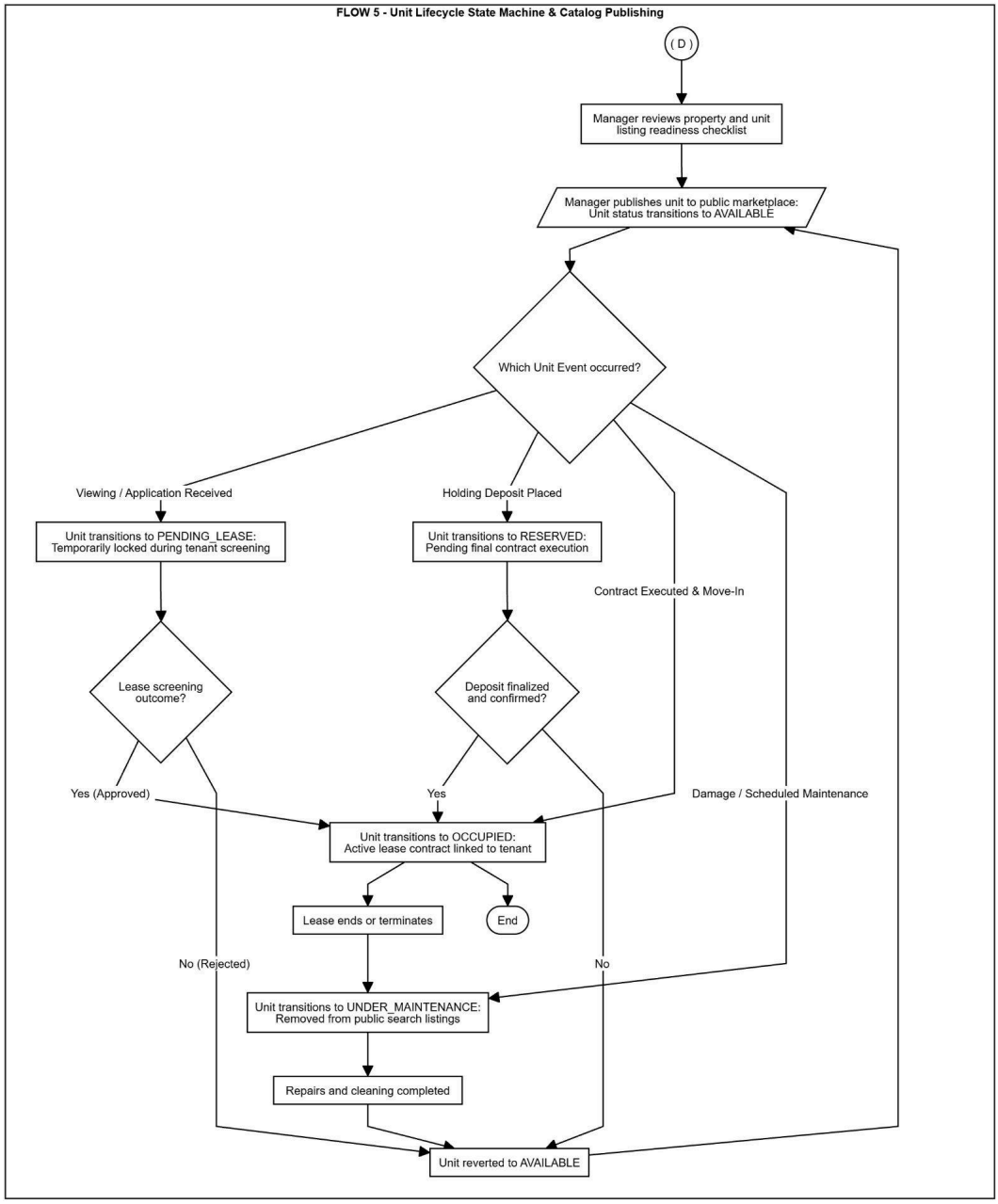
Flow 14: Property Management Flowchart



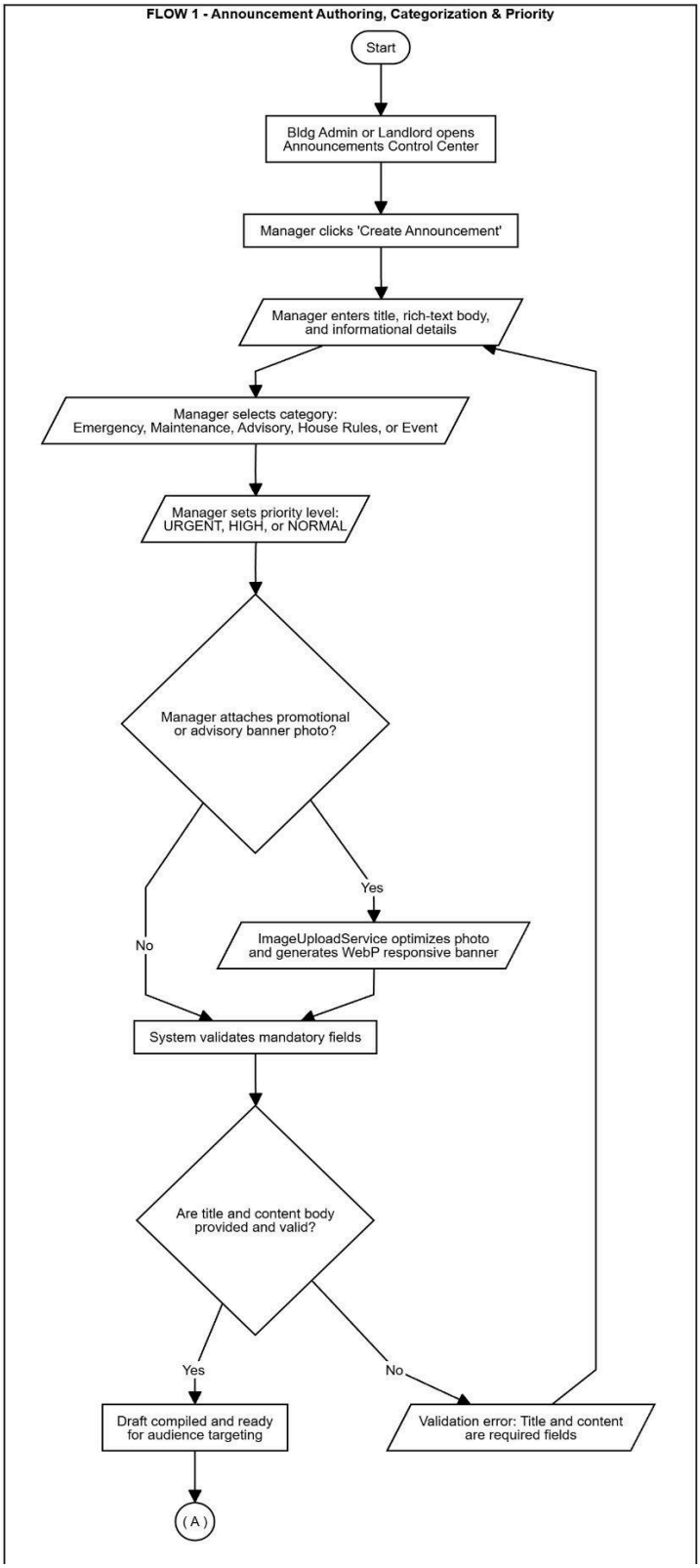

FLOW 2 - Floor Structure & Unit Inventory Provisioning
(A)
Manager opens Property Inventory Builder
Manager creates building floors:
Ground Floor, 2nd Floor, Penthouse, etc.
Manager provisions units on each floor:
Unit number, unit type Studio, 1BR, 2BR,
floor area, and max occupants
Manager sets financial unit parameters:
Monthly rent price, custom utility rates,
and dedicated sub-meter identifiers
Are unit inventory numbers
unique and valid?
Yes
No
System registers unit records
and binds them to corresponding floor IDs
Validation warning: Duplicate unit number
detected on floor
System initializes unit status
to default 'draft' pending assets
(B)

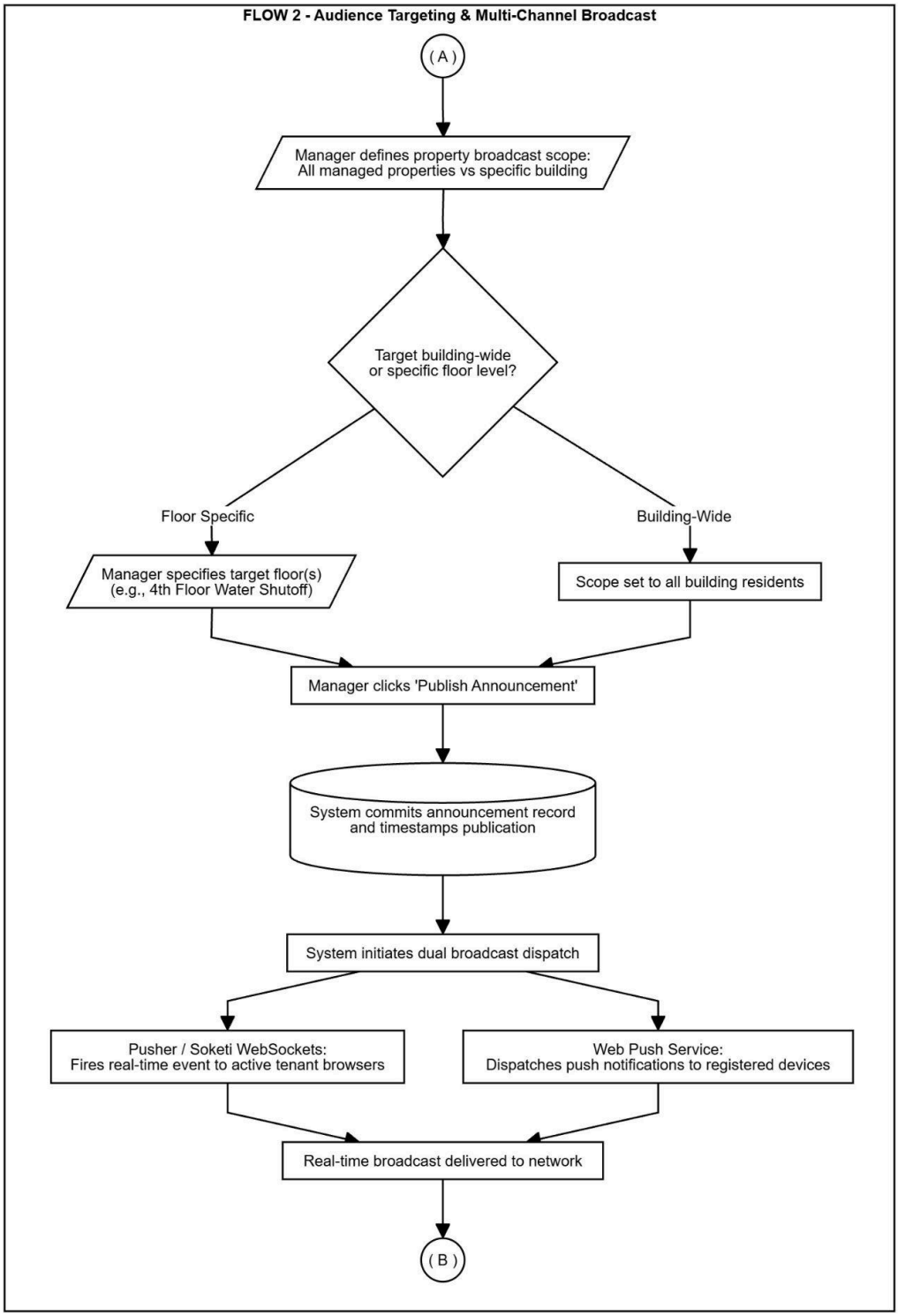

FLOW 3 - Unit Media Assets & 360 Virtual Tour Processing
(B)
Manager opens Unit Media Asset Manager
Manager uploads high-resolution photos of unit interior, bathroom, and kitchen
ImageUploadService optimizes assets: Converts to WebP, generates thumbnails, and preserves aspect ratio
Manager designates primary cover photo for marketplace search listings
Manager uploads 360-Degree virtual tour panorama?
Yes
No
System processes equirectangular image and configures interactive 360-Degree tour viewer
System checks required photo count
Is at least one valid cover photo uploaded?
Yes
No
System links media assets to unit and saves gallery metadata
Validation warning: Primary thumbnail required to list unit
(C)

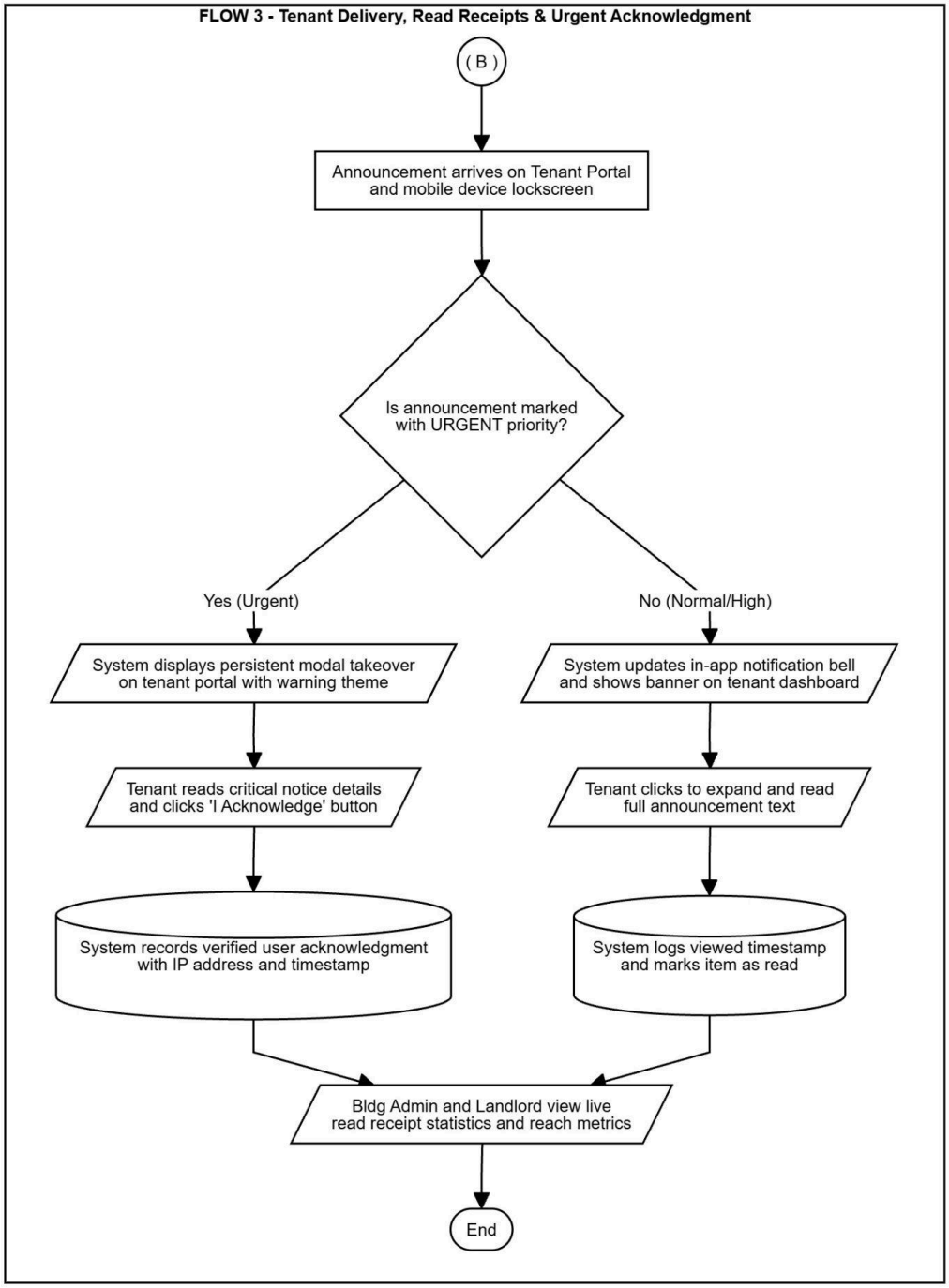

FLOW 4 - Management Staff Assignment & Role Allocation
(C)
Manager accesses Property Personnel Panel
Manager assigns Building Administrator(s) responsible for day-to-day operations
Manager assigns Staff Technicians for plumbing, electrical, and maintenance
System updates UserProperty junction records binding user accounts to Property ID
System maps notification triggers: Maintenance tickets, viewing requests, and rent collection escalations
Are required administrative roles assigned?
Yes
No
System dispatches onboarding alerts to assigned staff and administrators
Advisory warning: Assign at least one manager to handle tenant requests
(D)


FLOW 5 - Unit Lifecycle State Machine & Catalog Publishing
(D)
Manager reviews property and unit listing readiness checklist
Manager publishes unit to public marketplace: Unit status transitions to AVAILABLE
Which Unit Event occurred?
Viewing / Application Received
Holding Deposit Placed
Contract Executed & Move-In
Unit transitions to PENDING_LEASE: Temporarily locked during tenant screening
Unit transitions to RESERVED: Pending final contract execution
Lease screening outcome?
Deposit finalized and confirmed?
Yes (Approved)
Yes
Damage / Scheduled Maintenance
Unit transitions to OCCUPIED: Active lease contract linked to tenant
Lease ends or terminates
End
No (Rejected)
No
Unit transitions to UNDER_MAINTENANCE: Removed from public search listings
Repairs and cleaning completed
Unit reverted to AVAILABLE

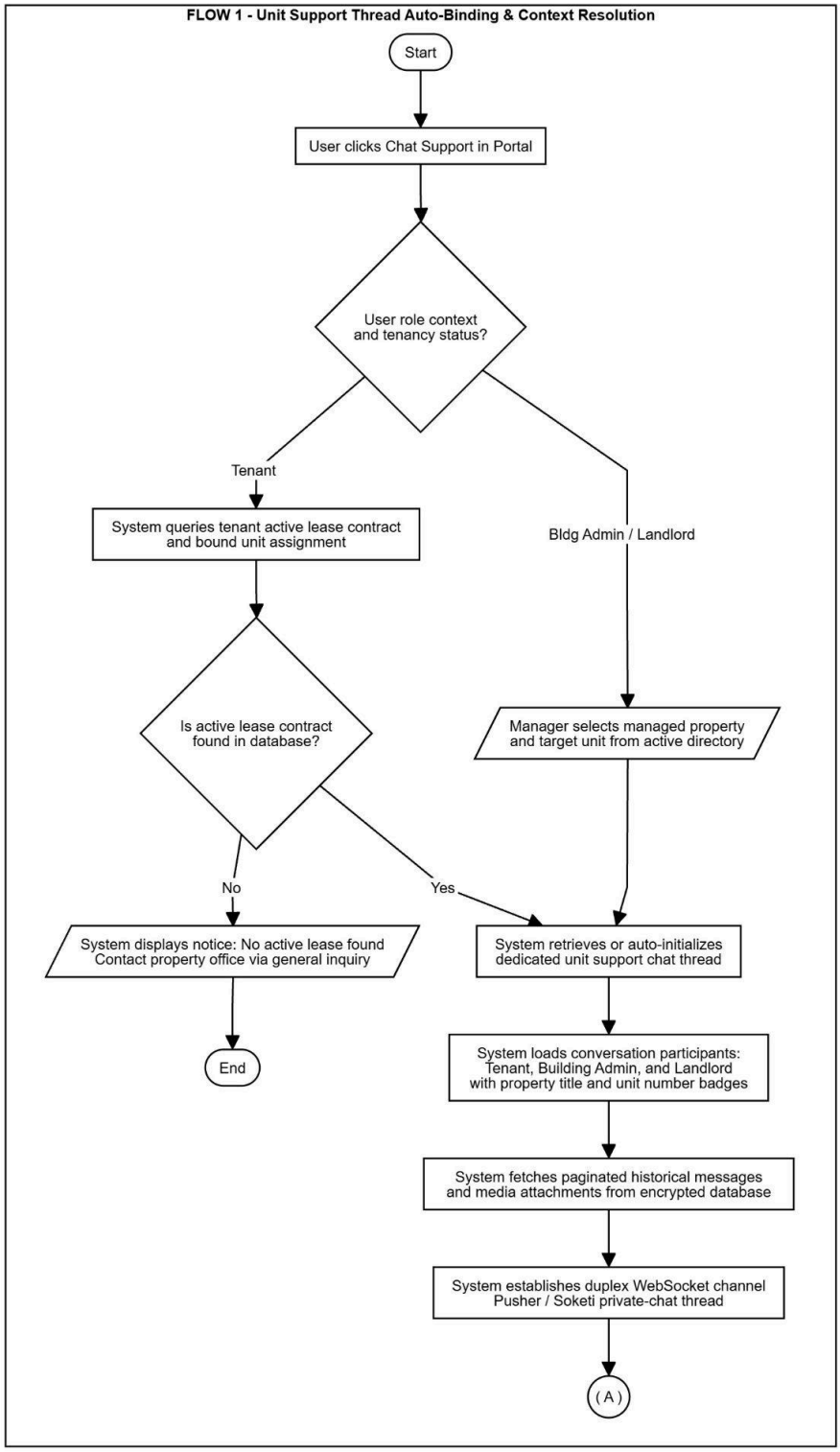
Flow 15: Announcement Flowchart

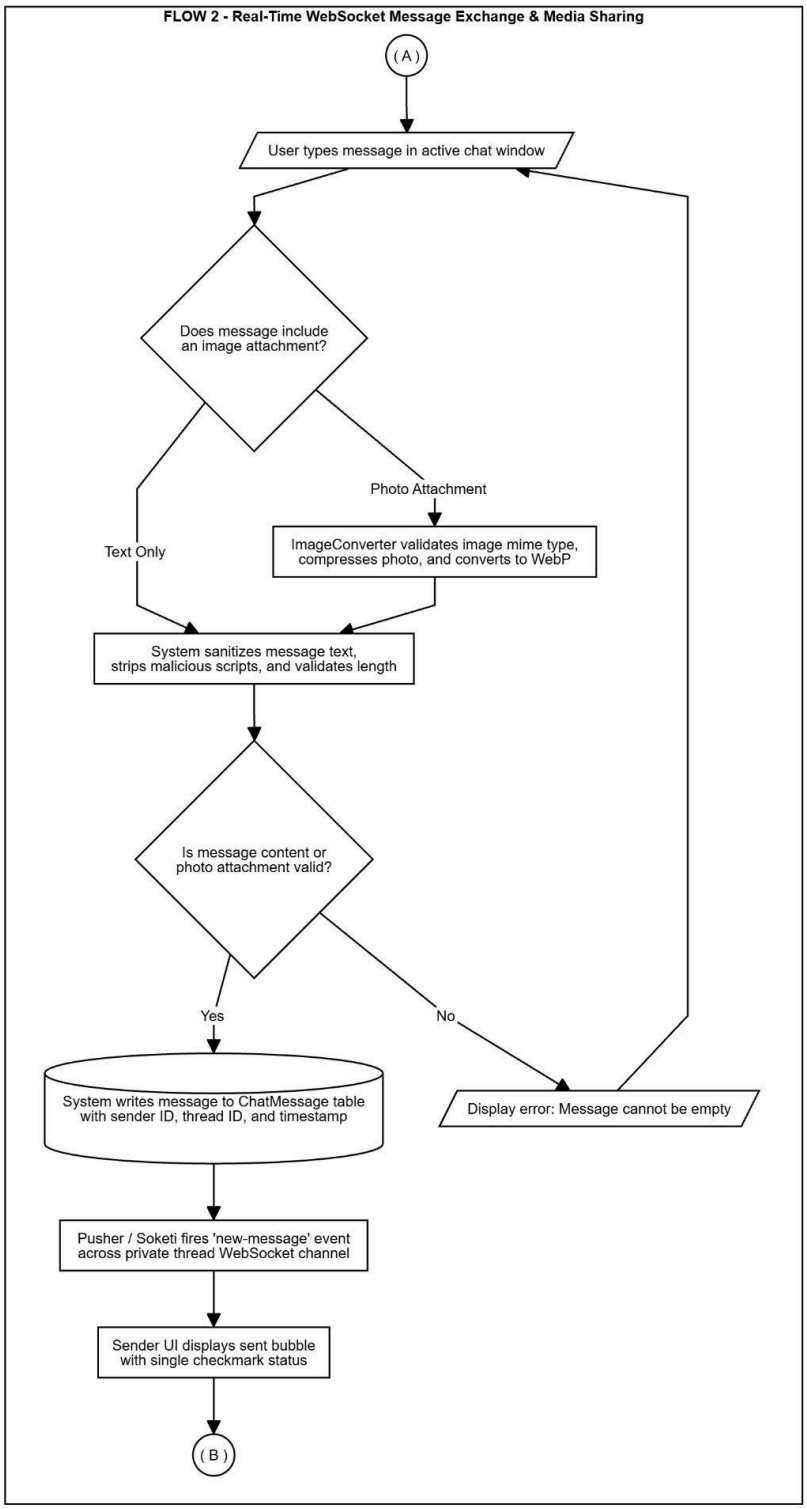


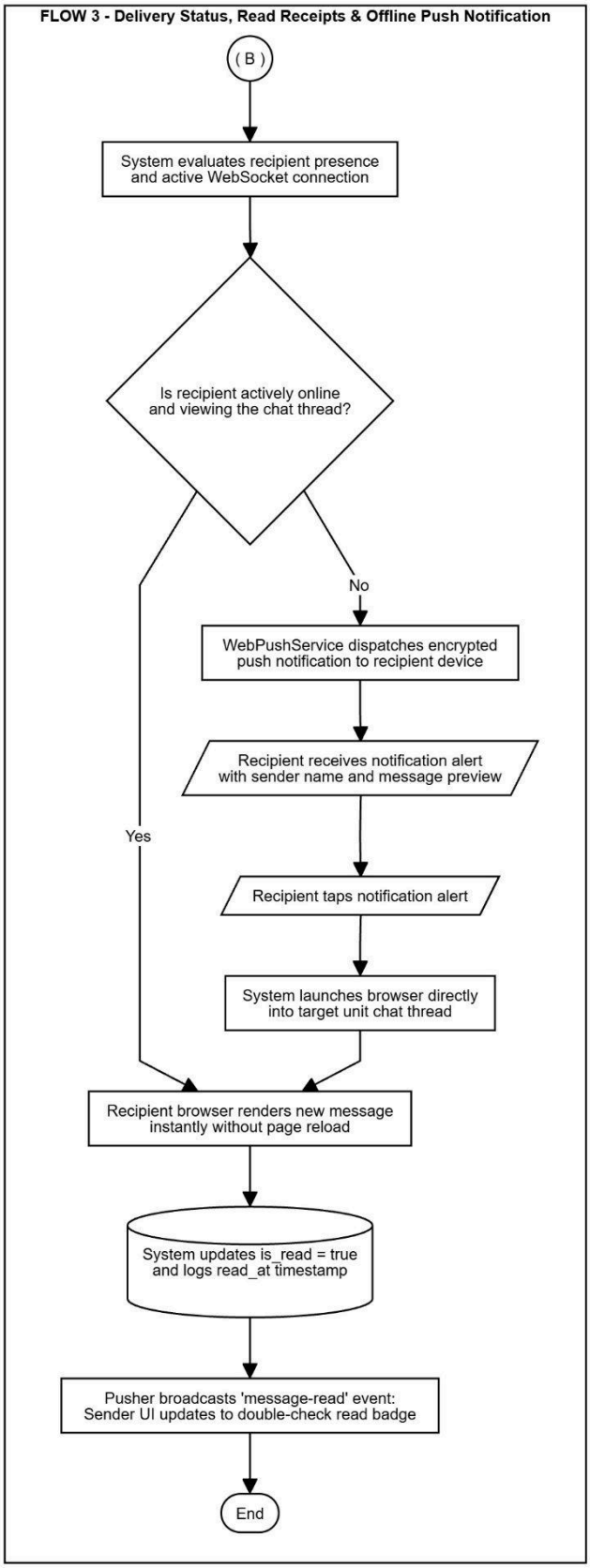

FLOW 2 - Audience Targeting & Multi-Channel Broadcast
(A)
Manager defines property broadcast scope:
All managed properties vs specific building
Target building-wide
or specific floor level?
Floor Specific
Building-Wide
Manager specifies target floor(s)
(e.g., 4th Floor Water Shutoff)
Scope set to all building residents
Manager clicks 'Publish Announcement'
System commits announcement record
and timestamps publication
System initiates dual broadcast dispatch
Pusher / Soketi WebSockets:
Fires real-time event to active tenant browsers
Web Push Service:
Dispatches push notifications to registered devices
Real-time broadcast delivered to network
(B)


FLOW 3 - Tenant Delivery, Read Receipts & Urgent Acknowledgment
(B)
Announcement arrives on Tenant Portal and mobile device lockscreen
Is announcement marked with URGENT priority?
Yes (Urgent)
No (Normal/High)
System displays persistent modal takeover on tenant portal with warning theme
System updates in-app notification bell and shows banner on tenant dashboard
Tenant reads critical notice details and clicks 'I Acknowledge' button
Tenant clicks to expand and read full announcement text
System records verified user acknowledgment with IP address and timestamp
System logs viewed timestamp and marks item as read
Bldg Admin and Landlord view live read receipt statistics and reach metrics
End

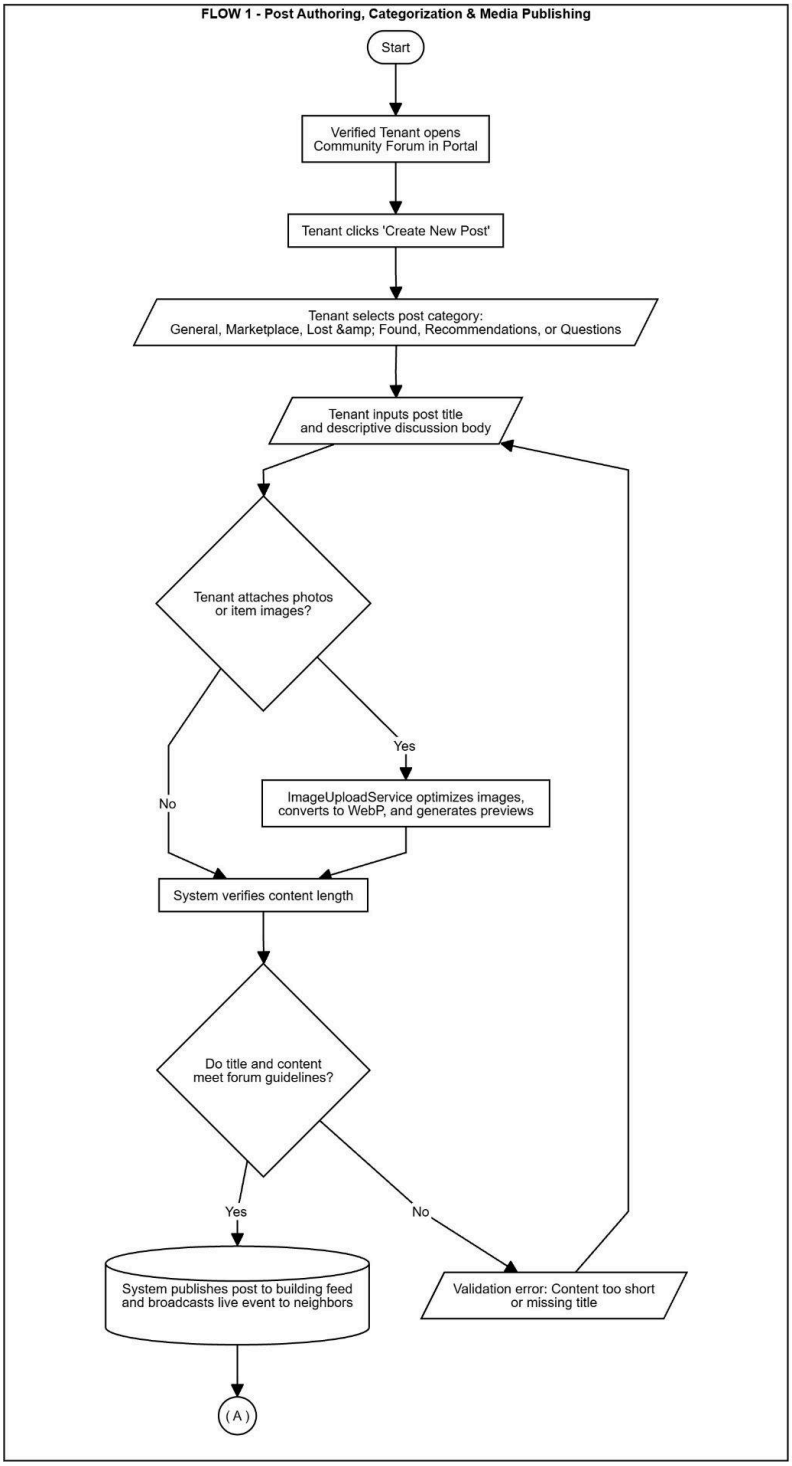
Flow 16: Chat Support Flowchart

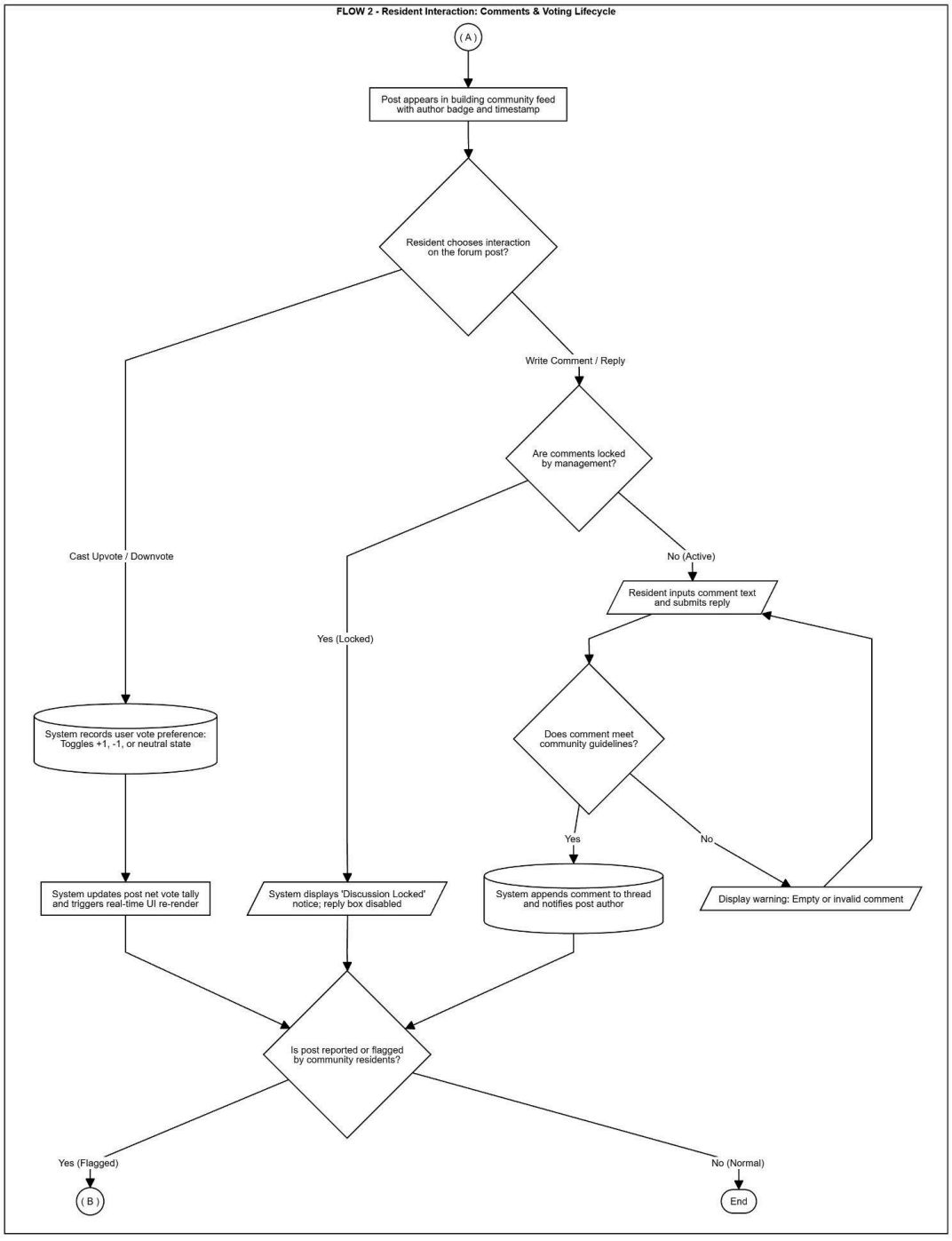


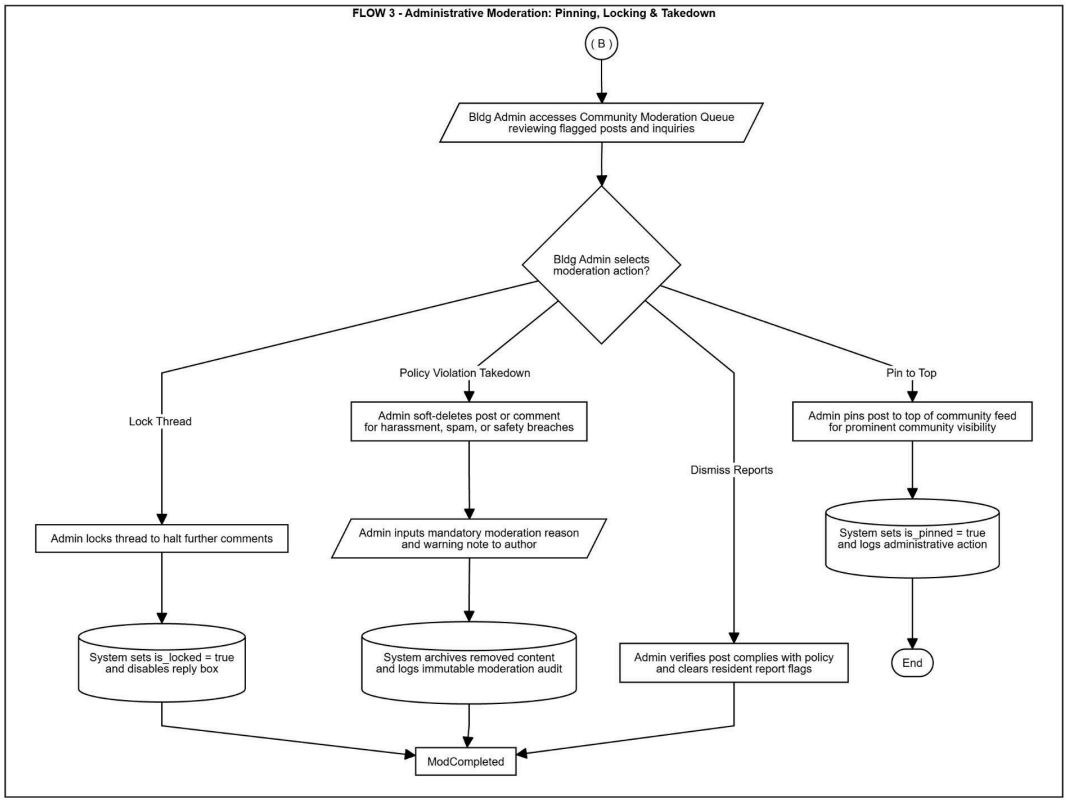

FLOW 2 - Real-Time WebSocket Message Exchange & Media Sharing
(A)
User types message in active chat window
Does message include an image attachment?
Photo Attachment
Text Only
ImageConverter validates image mime type, compresses photo, and converts to WebP
System sanitizes message text, strips malicious scripts, and validates length
Is message content or photo attachment valid?
Yes
No
System writes message to ChatMessage table with sender ID, thread ID, and timestamp
Display error: Message cannot be empty
Pusher / Soketi fires 'new-message' event across private thread WebSocket channel
Sender UI displays sent bubble with single checkmark status
(B)


FLOW 3 - Delivery Status, Read Receipts & Offline Push Notification
(B)
System evaluates recipient presence and active WebSocket connection
Is recipient actively online and viewing the chat thread?
No
Yes
WebPushService dispatches encrypted push notification to recipient device
Recipient receives notification alert with sender name and message preview
Recipient taps notification alert
System launches browser directly into target unit chat thread
Recipient browser renders new message instantly without page reload
System updates is_read = true and logs read_at timestamp
Pusher broadcasts 'message-read' event: Sender UI updates to double-check read badge
End

Flow 17: Community Forum Flowchart




FLOW 2 - Resident Interaction: Comments & Voting Lifecycle
(A)
Post appears in building community feed with author badge and timestamp
Resident chooses interaction on the forum post?
Cast Upvote / Downvote
Write Comment / Reply
Are comments locked by management?
Yes (Locked)
No (Active)
Resident inputs comment text and submits reply
Does comment meet community guidelines?
Yes
No
System records user vote preference: Toggles +1, -1, or neutral state
System updates post net vote tally and triggers real-time UI re-render
System displays 'Discussion Locked' notice; reply box disabled
System appends comment to thread and notifies post author
Display warning: Empty or invalid comment
Is post reported or flagged by community residents?
Yes (Flagged)
No (Normal)
(B)
End


FLOW 3 - Administrative Moderation: Pinning, Locking & Takedown
(B)
Bldg Admin accesses Community Moderation Queue reviewing flagged posts and inquiries
Bldg Admin selects moderation action?
Lock Thread
Policy Violation Takedown
Dismiss Reports
Pin to Top
Admin locks thread to halt further comments
Admin soft-deletes post or comment for harassment, spam, or safety breaches
Admin inputs mandatory moderation reason and warning note to author
Admin verifies post complies with policy and clears resident report flags
Admin pins post to top of community feed for prominent community visibility
System sets is_locked = true and disables reply box
System archives removed content and logs immutable moderation audit
System sets is_pinned = true and logs administrative action
End
ModCompleted

RESEARCHERS' PROFILE



Kivz M. Aragon is responsible for overseeing the overall planning, coordination, and technical implementation of Estatix PMS. As the Project Manager, he will lead sprint planning, manage the project backlog, monitor development progress, assign tasks, and ensure that project objectives and deadlines are met. As the System Architect, he will design the system architecture, workflows, and logical diagrams that define the structure and operation of the application. He will also lead the UI/UX design and prototyping, frontend and backend development, and database administration. Furthermore, he will handle the integration of cryptographic security mechanisms and oversee quality assurance, functional testing, and security testing to ensure that the system meets its technical and security requirements. In addition to his technical and management responsibilities, he will serve as the final reviewer of the research manuscript. He will carefully examine all chapters and sections prepared by the research team to ensure that the content is accurate, complete, consistent, properly organized, and aligned with the study's objectives and requirements. He will also check grammar, sentence construction, formatting, citations, references, and the consistency of information throughout the paper. Before submission, he will review the final revisions, identify any missing or unclear discussions, and ensure that the manuscript meets the required academic standards and is ready for presentation and evaluation.



Edrian Beronia is responsible for contributing to the writing, revision, and organization of the research manuscript. His tasks include developing and refining the research discussions, reviewing related literature and studies, and ensuring that the documentation accurately reflects the objectives, features, and implementation of Estatix PMS. He will also assist in preparing research instruments, administering questionnaires, gathering evaluation responses, and organizing collected data for analysis. Alongside these responsibilities, he will participate in the testing and validation of system features throughout the development process. Whenever a feature or module is completed, he will conduct functional checks to determine whether it works according to its intended purpose, test different user interactions and possible scenarios, identify bugs or inconsistencies, and document the results for the development team. He will also assist in retesting corrected features to verify that reported issues have been resolved and that updates do not negatively affect existing system functions. Furthermore, he will contribute to the interpretation and documentation of evaluation results in accordance with ISO/IEC 25010:2023, ISO/IEC 27001:2022 Annex A security controls, and the Technology Acceptance Model (TAM).



Jimboy A. Cajefe is responsible for preparing and organizing the academic documentation of Estatix PMS. His responsibilities include contributing to the development and refinement of the research chapters, gathering and synthesizing related literature and studies, and ensuring that the manuscript follows the required academic format. He will also assist in documenting the system's development process, recording relevant findings, and preparing research materials. In addition to his documentation responsibilities, he will participate in testing the system as individual features and modules are completed. This includes checking the functionality of implemented features, verifying whether system operations produce the expected results, identifying errors or unexpected behavior, and reporting issues to the development lead for correction. He will also provide feedback on the usability, functionality, and overall performance of the completed features to help ensure that the system meets its intended requirements.



Rhenz P. Culaban is responsible for assisting in the preparation and completion of the academic research documentation. His responsibilities include writing and editing assigned sections of the manuscript, reviewing and organizing related literature, and ensuring consistency and clarity throughout the research. He will also assist in documenting the system's features, development procedures, and evaluation results. During the data collection and evaluation phase, he will help distribute and administer survey questionnaires, compile respondents' feedback, and prepare written reports that support the assessment of Estatix PMS in terms of software quality, security controls, perceived usefulness, and perceived ease of use.