System Requirements Specification

Contents

1. The Purpose of the Project

1a. The User Business or Background of the Project Effort

Individual landlords and small-scale property investors in Egypt currently manage their rental properties using fragmented, non-integrated methods. Property records are maintained in spreadsheets, lease agreements stored in physical files, maintenance issues tracked through notes or memory, and financial transactions logged manually across multiple documents. This disjointed approach creates significant operational challenges.

1b. Goals of the Project

Goal 1: Centralize Property Data
Consolidate all property, unit, tenant, lease, financial, and maintenance information into a single, accessible system, eliminating the need for multiple spreadsheets and physical file storage.

Goal 2: Improve Financial Clarity
Provide property owners with immediate visibility into rental income, operational expenses, and monthly cash flow, simplifying financial decision-making and tax preparation documentation.

Goal 3: Reduce Administrative Time
Decrease the time property owners spend on manual record-keeping, data searching, and event tracking by at least 50% through automation and centralized access.

Goal 4: Prevent Missed Critical Dates
Eliminate missed lease renewals and overdue maintenance requests through automated dashboard alerts for leases expiring within 60 days and unresolved maintenance issues.

Goal 5: Accelerate Vacancy Filling
Reduce average time-to-rent for vacant units by providing a public Property Finder module that allows potential tenants to discover and inquire about available properties without property owner intervention.

Goal 6: Ensure Data Accuracy and Security
Maintain validated, consistent property management data with secure storage, backup capabilities, and role-based access control to protect sensitive tenant and financial information.

Goal 7: Enable Mobile Access
Provide property owners with on-the-go access to all property management functions through a mobile application, allowing them to manage their portfolios from anywhere.

2. Stakeholders

2a. The Client

Client Name: PPMS Founders
Role: The primary sponsor funding the development of the Property Management System (PPMS). The client represents the collective interests of individual property owners and small-scale investors in Egypt who will benefit from the system.

2b. The Customer

Customer Name: Individual Property Owners and Small-Scale Investors
Role: The customers are individual landlords who will purchase or subscribe to use the PPMS for managing their rental properties. They are also the primary decision-makers who will evaluate whether the system meets their needs and choose to adopt it for their property management operations.

2c. Other Stakeholders

Current Tenants

Role: Individuals or businesses currently leasing units from property owners
Interest: Accurate maintenance of their lease records, timely responses to maintenance requests, and proper tracking of their rent payments
Impact on Project: Their data must be securely stored and managed; system accuracy directly affects their rental experience

Tax Authorities / Egyptian Tax Authority

Role: Government officials responsible for tax compliance verification
Interest: Access to standardized financial reports showing rental income and property-related expenses
Impact on Project: System must generate compliant financial reports that meet Egyptian tax documentation requirements

Business Development Team

Role: Team responsible for market expansion and customer acquisition
Interest: Market insights, user adoption metrics, and competitive positioning data
Impact on Project: Requires analytics on system usage and market penetration to guide growth strategy

Compliance and Legal Advisors

Role: Legal professionals ensuring the system meets Egyptian regulatory requirements
Interest: Data privacy compliance, lease documentation standards, and regulatory adherence
Impact on Project: Must validate that system features comply with Egyptian property and tenant laws

2d. The Hands-On Users of the Product

User Category 1: Property Owner

User Role:
Primary system user responsible for managing all aspects of property operations including registering properties and units, recording tenant information and lease agreements, logging rental income and expenses, tracking maintenance requests, and responding to potential tenant inquiries.

Subject Matter Experience: Journeyman
Property owners have practical experience managing their rental properties but may lack formal property management training. They understand lease agreements, tenant relationships, basic accounting for rental income and expenses, and property maintenance needs.

Technological Experience: Journeyman
Users have basic intermediate mobile device proficiency. They are comfortable with common mobile applications (messaging, social media, banking apps) but may not have experience with specialized business software. They can navigate mobile interfaces, complete forms, and understand basic data visualization.

Other User Characteristics:

- Age range: 30-65 years old
- Time availability: Limited time for system training; prefer intuitive interfaces
- Work context: Manage properties as secondary income source alongside primary employment
- Language preference: Comfortable with English interface terms mixed with familiar business terminology
- Motivation: Highly motivated to reduce administrative burden and improve financial visibility

- Technical support needs: May require initial guidance but expect to become self-sufficient quickly
- Mobile dependency: Primarily access system via mobile device during commutes, property visits, or evening hours

---

User Category 2: Potential Tenant / Public User

User Role:
Public-facing users who search for available rental properties using the Property Finder module. They browse vacant units, filter by location and price, review property details, and submit inquiry forms to contact property owners.

Subject Matter Experience: Novice to Journeyman
Users range from first-time renters with limited knowledge of rental markets to experienced tenants familiar with property search processes. They understand basic rental concepts (monthly rent, security deposits, property types) but may need guidance on search and inquiry processes.

Technological Experience: Journeyman
Users are comfortable with mobile web browsing and search interfaces similar to popular real estate or e-commerce platforms. They can use filters, view listings, and complete contact forms.

Other User Characteristics:

- Age range: 20-45 years old (primary rental market demographic)
- Device usage: Access primarily via mobile smartphones
- Search behavior: Quick browsing with expectation of immediate results
- Decision timeline: Compare multiple properties before making inquiry decisions
- Communication preference: Prefer initial contact through online forms rather than phone calls
- Expectations: Expect current, accurate availability information and responsive property owner communication

---

User Category 3: Data Administrator

User Role:
Internal personnel responsible for monitoring data quality, performing validation checks, conducting data analysis to identify trends and patterns, generating analytical reports, and ensuring data backup and integrity across all system modules.

Subject Matter Experience: Master
Data administrators possess comprehensive understanding of property management business processes, financial reporting requirements, and operational metrics. They understand relationships between properties, tenants, leases, and financial data.

Technological Experience: Master
Highly proficient with database concepts, data analysis tools, report generation, and system administration functions. Comfortable with technical interfaces and advanced system features.

Other User Characteristics:

- Technical role requiring understanding of both business and technology
- Responsible for providing insights to property owners and management
- Requires access to comprehensive data across all users and properties
- Works with data exports and integration with external analysis tools
- Focuses on data quality, consistency, and actionable insights

---

User Category 4: System Developer

User Role:
Internal development team members responsible for building, testing, deploying, and maintaining the PPMS. They implement new features, fix bugs, monitor system performance, analyze user interaction patterns, and ensure code quality and system security.

Subject Matter Experience: Journeyman
Developers understand basic property management concepts and workflows but rely on business stakeholders for detailed domain knowledge. They learn business context through requirements and user feedback.

Technological Experience: Master
Expert-level proficiency in mobile application development, database design, API development, security protocols, and development tools. Experienced with Agile/Iterative methodologies and version control.

Other User Characteristics:

- Internal team members with ongoing system involvement
- Require comprehensive technical documentation and system architecture diagrams
- Need access to development, testing, and production environments
- Focus on code maintainability, system performance, and scalability
- Balance feature development with technical debt management

2e. Personas

Persona 1: Ahmed Hassan - The Part-Time Landlord

Demographics:

- Age: 42 years old
- Job: Civil Engineer working for a construction firm in Cairo
- Family: Married with two children (ages 8 and 12)
- Lives in: Nasr City, Cairo

Property Portfolio:

- Owns 3 residential properties (inherited one from parents, purchased two as investments)
- Total of 8 apartments across the three buildings
- Properties located in Maadi, Heliopolis, and downtown Cairo

Personality & Lifestyle:

- Hobbies: Enjoys football, spending time with family, occasional fishing trips
- Favorite food: Koshari and grilled meats

- Favorite music: Classical Arabic music (Abdel Halim, Um Kulthum)
- Holiday preference: Family trips to Alexandria or Hurghada during summer
- Likes: Organized systems, clear financial records, efficiency
- Dislikes: Paperwork, wasting time searching for documents, unexpected expenses

Technology & Money Attitudes:

- **Technology:** Comfortable with smartphones and uses banking apps regularly. Somewhat hesitant about new software but willing to learn if it saves time. Prefers simple, intuitive interfaces.
- **Money:** Conservative with finances, tracks income carefully for tax purposes. Views rental properties as long-term investment and retirement security.

Current Challenges:

- Struggles to remember which tenant's lease expires when
- Keeps rent payment records in Excel but finds it tedious
- Once forgot about a maintenance request for two weeks, leading to tenant complaint
- Takes 30-45 minutes each evening to update his various spreadsheets
- Recently missed a lease renewal opportunity, resulting in unexpected vacancy

Product Expectations:

- Wants quick access to see which apartments are vacant
- Needs automatic reminders for lease expirations
- Expects to record rent payments in under 2 minutes
- Must be able to check property status during lunch breaks or commute
- Values reliability over fancy features

2f. Priorities Assigned to Users

User Category	Priority Level	Rationale
Property Owner	**Critical (Priority 1)**	Primary revenue-generating users; system success depends on their adoption and satisfaction
Potential Tenant	**High (Priority 2)**	Essential for achieving vacancy reduction goal; directly impacts property owner ROI
Data Administrator	**Medium (Priority 3)**	Important for system insights and data quality but not blocking for core functionality
System Developer	**Medium (Priority 3)**	Internal stakeholders necessary for ongoing maintenance and enhancement
Current Tenants	**Low (Priority 4)**	Indirect users whose data is managed but who don't interact directly with system
Tax Authorities	**Low (Priority 4)**	Compliance requirement but interaction is periodic and export-based
Municipal Officers	**Low (Priority 4)**	Secondary users with infrequent, read-only data access needs

2g. User Participation

Property Owner

Required Participation:

- **Business knowledge contribution:** 15-20 hours across project lifecycle
    - Initial requirements validation sessions (4-6 hours)
    - User interface prototype feedback (4-6 hours)
    - User acceptance testing (6-8 hours)
- **Ongoing feedback:** Participation in bi-weekly sprint reviews (1 hour per sprint)

- **Critical input areas:** Dashboard design, financial reporting formats, alert preferences, Egyptian tax compliance requirements

Potential Tenant / Public User

Required Participation:

- **Usability testing:** 8-10 hours total
    - Property Finder interface testing (4-5 hours)
    - Search and filter functionality feedback (2-3 hours)
    - Mobile experience evaluation (2 hours)
- **Critical input areas:** Search behavior patterns, property information needs, inquiry form design

Data Administrator

Required Participation:

- **Technical requirements definition:** 10-12 hours
    - Data model validation (3-4 hours)
    - Analytics requirements specification (4-5 hours)
    - Report design review (3 hours)
- **Critical input areas:** Data integrity rules, analytical metrics, report formats

System Developer

Continuous Participation:

- Full-time involvement throughout development lifecycle
- Daily development activities, code reviews, and technical decision-making
- Architecture design, implementation, testing, and deployment

2h. Maintenance Users and Service Technicians

System Administrator / DevOps Engineer

Role:
Internal technical personnel responsible for system deployment, monitoring, performance optimization, security updates, database backup and recovery, and troubleshooting production issues.

Maintenance Requirements:

- Access to system logs and performance monitoring dashboards
- Ability to perform database backups and restore operations
- Tools for monitoring system uptime, response times, and error rates
- Access to deployment pipelines for applying updates and patches
- Documentation for troubleshooting common issues
- Alert mechanisms for system failures or performance degradation

Technical Characteristics:

- Expert-level system administration and database management skills
- Experience with mobile application deployment and monitoring
- Knowledge of security protocols and encryption mechanisms
- Ability to work during off-hours for maintenance windows

Participation:

- Ongoing monitoring and maintenance activities
- Immediate response to critical system issues
- Scheduled maintenance windows for updates and backups
- Regular security audits and performance optimization reviews

3.Mandated Constraints

3a. Solution Constraints

Description of Constraint: The system must be developed as a mobile application.

Rationale: Property owners require on-the-go access to manage their properties from any location at any time.

3b. Implementation Environment of the Current System

Description of Constraint: There is no existing automated system. Property owners currently use manual methods including spreadsheets, physical files, and handwritten notes.

Rationale: This is a new system implementation replacing entirely manual processes.

3c. Partner or Collaborative Applications

Description of Constraint: The system shall operate as a standalone application with no required integrations to external partner systems or collaborative applications.

Rationale: Simplifies development and maintains full control over system functionality and data.

3d. Off-the-Shelf Software

Description of Constraint: No off-the-shelf software components shall be incorporated into the system.

Rationale: The system must be custom-built to meet specific requirements and ensure complete ownership of all functionality.

3e. Anticipated Workplace Environment

Description of Constraint: Property owners will use the system in mobile environments including:

- While traveling between properties
- During property site visits
- At home during evening hours
- During work breaks throughout the day

Rationale: Users need flexibility to manage properties whenever and wherever needed, not tied to office environments.

3f. Schedule Constraints

Description of Constraint: Project timeline and delivery schedule to be determined by investors.

Rationale: Project phases and release dates depend on investment funding approval and strategic business priorities.

Status: Pending investor decision.

3g. Budget Constraints

Description of Constraint: Project budget to be determined by investors.

Rationale: Available funding determines scope, resources, and timeline feasibility.

Status: Pending investor approval.

3h. Enterprise Constraints

Constraint 1 - Development Resources:
The system must be developed and maintained exclusively by an internal development team.

Rationale: Ensures knowledge retention, intellectual property protection, and long-term maintainability.

Constraint 2 - Development Methodology:
The project must follow Agile/Iterative development methodology.

Rationale: Enables incremental delivery, continuous stakeholder feedback, and adaptive planning throughout the development lifecycle.

Constraint 3 - Egyptian Legal Compliance:
The system must comply with all Egyptian laws regarding:

- Tax reporting and documentation requirements
- Property management regulations
- Data protection and tenant privacy
- Lease agreement standards

4. Naming Conventions and Terminology

Term / Acronym	Definition
PPMS	Property Management System - The complete software application being developed
Property	A physical real estate asset registered in the system, characterized by a unique address and property type, containing one or more units available for rental
Unit	An individual rentable space within a property, characterized by unit number, area, rental price, and status
Property Owner	Individual or entity that owns and manages properties within the system; the primary system user with full administrative access rights
Administrator	System role assigned to property owners, granting full access to all management functions
Tenant	Individual or business entity that leases a unit from a property owner; current occupant of a unit under an active lease agreement
Public User	Unauthenticated visitor accessing the Property Finder module; potential tenant searching for available units
Vacant	Unit status indicating the unit is available for rent and currently unoccupied
Occupied	Unit status indicating the unit is currently leased to a tenant
Under Maintenance	Unit status indicating the unit is temporarily unavailable due to repairs or improvements
Lease Agreement	Legal contract between property owner and tenant specifying rental terms, including start date, end date, rent amount, and security deposit
Lease Renewal	Extension of an existing lease agreement for an additional term
Rent Payment	A recorded transaction of rent received from a tenant for a specific period
Income	Total rental payments received within a specified period
Expense	Any cost incurred related to property operations
Monthly Summary	Aggregated view of total income and expenses for a calendar month
Maintenance Status	Current state of a maintenance request: New (recently reported), In Progress (being worked on), or Completed (issue resolved)
Dashboard	Administrative interface displaying key metrics, alerts, and summary information for property owners
Property Finder	Public-facing module allowing potential tenants to search and view available units without authentication

5. Relevant Facts and Assumptions

5a. Relevant Facts

Fact 1: Current Manual Operations

Property owners currently manage properties using spreadsheets, physical files, and handwritten notes without any integrated system.

Fact 2: Target User Profile

The system is designed for individual property owners and small-scale investors, not large property management companies.

Fact 3: Mobile Platform Requirement

The system must be delivered as a mobile application to support on-the-go property management.

5b. Business Facts

Business Fact 1: Administrative Inefficiency

Property owners spend excessive time on manual record-keeping and frequently miss critical deadlines such as lease renewals.

Business Fact 2: Financial Tracking Need

Property owners require structured tracking of rental income and expenses for tax preparation and financial planning.

Business Fact 3: Vacancy Marketing Gap

Property owners lack an efficient method to advertise vacant units to potential tenants, resulting in extended vacancy periods.

5c. Assumptions

Assumption 1: User Adoption

Property owners are willing to transition from manual methods to a digital system if it provides clear value and time savings.

Assumption 2: Technical Capability

Users have basic mobile device literacy and can learn to navigate the application with minimal training.

6. The Scope of the Work

6a. The Current Situation

Property owners currently manage their rental properties through fragmented manual processes. Financial records are maintained in separate spreadsheets with income and expenses tracked independently. Lease agreements are stored in physical file folders or scattered digital documents. Tenant contact information exists in phone contacts or handwritten address books. Maintenance issues are noted informally through phone reminders, sticky notes, or memory.

6b. The Context of the Work

The Property Management System (PPMS) will operate within the Egyptian property rental market, serving individual property owners who manage residential and commercial rental properties.

System Boundaries:

The system **includes**:

- Property and unit registration and management
- Tenant information storage and lease agreement tracking
- Rental income and expense recording with categorization
- Maintenance request logging and status tracking
- Public-facing Property Finder for vacant unit advertising
- Administrative dashboard with alerts and financial summaries
- Report generation and data export capabilities
- User authentication and role-based access control

The system **excludes**:

- Tenant screening or background check services
- Automated rent collection or payment processing
- Bank account integration or automated financial transactions
- Legal document generation or e-signature capabilities
- Vendor/contractor management or bidding systems

- Eviction processing or legal dispute management
- Accounting software integration (QuickBooks, etc.)
- Property valuation or market analysis tools
- Insurance claim processing
- Utility bill payment processing

Adjacent Systems and Entities:

- **Egyptian Tax Authority**: Receives exported financial reports for tax compliance verification (one-way data flow out)
- **Potential Tenants**: Access Property Finder to view vacant units and submit inquiries (limited read access)

c. Work Partitioning

Business Event List

Event Number	Event Name	Input/Trigger	Output/Response
1	Property owner registers new property	Property owner enters property details (address, type, number of units)	Property record created with unique ID; confirmation displayed
2	Property owner adds unit to property	Property owner enters unit details (number, area, price, status)	Unit record created and linked to property; confirmation displayed
3	Property owner updates property information	Property owner modifies existing property details	Property record updated; confirmation displayed
4	Property owner deletes property	Property owner requests property deletion	Property and associated units removed; confirmation displayed

5	Property owner registers new tenant	Property owner enters tenant details (name, contact info, associated unit)	Tenant record created; confirmation displayed
6	Property owner creates lease agreement	Property owner enters lease terms (start date, end date, rent amount, deposit)	Lease record created and linked to tenant and unit; unit status changed to Occupied
7	Property owner renews lease	Property owner extends existing lease with new end date	Lease record updated with new term; confirmation displayed
8	Property owner terminates lease	Property owner ends lease agreement	Lease marked as terminated; unit status changed to Vacant
9	Property owner records rent payment	Property owner enters payment details (amount, date, tenant)	Payment record created; displayed in financial summary
10	Property owner records expense	Property owner enters expense details (amount, date, category, property)	Expense record created; displayed in financial summary
11	Property owner logs maintenance request	Property owner enters maintenance details (unit, issue description, date)	Maintenance request created with "New" status; confirmation displayed
12	Property owner updates maintenance status	Property owner changes status to In Progress or Completed	Maintenance record updated; confirmation displayed
13	Property owner changes unit status	Property owner updates unit to Vacant, Occupied, or Under Maintenance	Unit status updated; Property Finder list automatically refreshed

14	System identifies expiring leases	System checks daily for leases ending within 60 days	Dashboard displays list of expiring leases with tenant and property details
15	System calculates monthly financial summary	End of month or property owner requests summary	Dashboard displays total income, total expenses, and net income for selected month
16	Property owner views dashboard	Property owner logs in and accesses dashboard	System displays key metrics, alerts, and summaries
17	Property owner exports financial report	Property owner requests report export for date range	System generates CSV or PDF file with income and expense details
18	Potential tenant searches for properties	Public user accesses Property Finder and applies filters	System displays list of vacant units matching search criteria
19	Potential tenant views property details	Public user selects a property from search results	System displays property address, unit details, rental price in read-only format
20	Potential tenant submits inquiry	Public user completes and submits contact form	Inquiry sent to property owner; confirmation displayed to user
21	Data administrator performs backup	Administrator initiates backup operation	System creates complete data backup with timestamp
22	Data administrator runs validation check	Administrator requests data integrity check	System reports any incomplete or inconsistent records
23	System alerts for unresolved maintenance	System checks for maintenance requests exceeding threshold	Dashboard displays alert for overdue maintenance items

6d. Specifying a Business Use Case (BUC)

Business Use Case: User Adds a New Property

BUC Number: BUC-001

BUC Name: Property Registration

Triggering Event: Property owner acquires a new rental property and needs to register it in the system (Event #1 from Work Partitioning)

Preconditions:

- Property owner is logged into the system as an administrator
- Property owner has property details available (address, type, number of units)

Main Success Scenario:

1. Property owner navigates to the Property Management module
2. Property owner selects "Add New Property" option
3. System displays property registration form
4. Property owner enters property address (street, city, district)
5. Property owner selects property type (Residential or Commercial)
6. Property owner enters total number of units in the property
7. Property owner submits the form
8. System validates all required fields are complete and correctly formatted
9. System generates unique property ID
10. System saves property record to database
11. System displays confirmation message with property ID
12. System returns property owner to property list showing newly added property

Alternative Paths:

8a. Validation fails - missing required field:

- System highlights incomplete field(s) in red
- System displays error message: "Please complete all required fields"

- Property owner corrects missing information
- Return to step 7

8b. Validation fails - invalid data format:

- System highlights field with invalid format
- System displays specific error message (e.g., "Number of units must be a positive number")
- Property owner corrects the data
- Return to step 7

8c. Property owner cancels operation:

- Property owner selects "Cancel" button at any point before submission
- System discards entered data without saving
- System returns property owner to property list

Postconditions:

- New property record exists in database with unique ID
- Property appears in property owner's property list
- Property is available for unit registration
- System logs property creation event with timestamp

Business Rules:

- BR1: Property address must be unique within the system
- BR2: Number of units must be a positive integer (minimum 1)
- BR3: Property type must be either "Residential" or "Commercial"
- BR4: All fields (address, type, number of units) are mandatory

Acceptance Criteria:

- Property owner can successfully register a property in less than 2 minutes
- System prevents duplicate property addresses
- System generates unique property ID automatically

- Confirmation message clearly indicates successful registration
- New property immediately appears in property list

Priority: High (Critical for system functionality)

Frequency of Use: Low to Medium (property owners add properties infrequently, typically 1-5 times per year)

Related Business Use Cases:

- BUC-002: Add Unit to Property (follows property registration)
- BUC-003: Update Property Information
- BUC-004: Delete Property

7. Business Data Model and Data Dictionary

7a. Business Data Model

The business data model illustrated in the diagram shows the core entities and their relationships within the Property Management System (PPMS).

Entity Relationships Overview:

- **User** is the parent entity with two specialized types: **Tenant** and **Property Owner**
- **Property Owner** owns multiple **Properties** (1 to many relationship)
- **Property** contains multiple **Units** (1 to many relationship)
- **Unit** can have multiple **Maintenance Requests** (1 to many relationship)
- **Tenant** can have multiple **Leases** (1 to many relationship)
- **Lease** is associated with one **Unit** (many to 1 relationship)
- **Lease** can have multiple **Payments** (1 to many relationship)
- **Property** can have multiple **Expenses** (1 to many relationship)

7b. Data Dictionary

Entity: User

Description: Base entity representing all system users including property owners and tenants.

Attribute	Type	Length	Required	Description	Constraints/Rules
UserID	Integer	-	Yes	Unique identifier for each user	Primary Key, Auto-generated
UserName	String	50	Yes	User's login username	Unique, Alphanumeric
Password	String	255	Yes	Encrypted password for authentication	Minimum 8 characters, encrypted storage

Entity: Property Owner

Description: Specialized user type representing property owners who manage rental properties.

Attribute	Type	Length	Required	Description	Constraints/Rules
UserID	Integer	-	Yes	Inherited from User entity	Foreign Key references User(UserID)
PortfolioID	String	20	Yes	Unique identifier for property owner's portfolio	Unique, Auto-generated

Relationship: Inherits from User (IS-A relationship)

Entity: Tenant

Description: Specialized user type representing individuals or businesses renting units.

Attribute	Type	Length	Required	Description	Constraints/Rules
UserID	Integer	-	Yes	Inherited from User entity	Foreign Key references User(UserID)
ContactInformation	String	255	Yes	Tenant's phone number, email, or other contact details	Must include at least phone or email
FullName	String	100	Yes	Complete legal name of the tenant	Minimum 2 characters

Relationship: Inherits from User (IS-A relationship)

Entity: Property

Description: Represents a real estate asset owned by a property owner, containing one or more rental units.

Attribute	Type	Length	Required	Description	Constraints/Rules
PropertyID	Integer	-	Yes	Unique identifier for each property	Primary Key, Auto-generated
PropertyType	String	20	Yes	Classification of property	Must be "Residential" or "Commercial"
TotalUnits	Integer	-	Yes	Total number of units in the property	Positive integer, minimum 1

Address	String	255	Yes	Complete physical address of the property	Must be unique within system

Relationships:

- Owned by one Property Owner (Many-to-One)
- Contains one or many Units (One-to-Many)
- Has zero or many Expenses (One-to-Many)

Entity: Unit

Description: Individual rentable space within a property.

Attribute	Type	Length	Required	Description	Constraints/Rules
UnitID	Integer	-	Yes	Unique identifier for each unit	Primary Key, Auto-generated
UnitNumber	String	20	Yes	Unit identifier within the property (e.g., "Apt 101", "Suite A")	Unique within property
RentalPrice	Decimal	10,2	Yes	Monthly rental amount	Positive number, two decimal places
Area	Decimal	10,2	No	Unit size in square meters	Positive number if provided
Status	String	20	Yes	Current availability status of the unit	Must be "Vacant", "Occupied", or "Under Maintenance"

Relationships:

- Belongs to one Property (Many-to-One)
- Can have zero or many Maintenance Requests (One-to-Many)
- Can be associated with zero or one Lease at a time (One-to-One for active leases)

Entity: Lease

Description: Rental agreement between property owner and tenant for a specific unit.

Attribute	Type	Length	Required	Description	Constraints/Rules
LeaseID	Integer	-	Yes	Unique identifier for each lease	Primary Key, Auto-generated
StartDate	Date	-	Yes	Date when tenant occupancy begins	Cannot be in the distant past
EndDate	Date	-	Yes	Date when lease agreement terminates	Must be after StartDate
RentAmount	Decimal	10,2	Yes	Monthly rent specified in the lease	Positive number, two decimal places
SecurityDeposit	Decimal	10,2	Yes	Refundable deposit amount	Positive number or zero
Status	String	20	Yes	Current state of the lease	"Active", "Expired", or "Terminated"

Relationships:

- Associated with one Tenant (Many-to-One)
- Associated with one Unit (Many-to-One)
- Can have zero or many Payments (One-to-Many)

Business Rules:

- EndDate must be after StartDate
- System flags lease as "Expiring" when EndDate is within 60 days
- When lease Status is "Active", associated Unit Status should be "Occupied"

Entity: Payment

Description: Record of rent payment received from tenant.

Attribute	Type	Length	Required	Description	Constraints/Rules
PaymentID	Integer	-	Yes	Unique identifier for each payment	Primary Key, Auto-generated
PaymentDate	Date	-	Yes	Date when payment was received	Cannot be future date
PaymentMethod	String	50	No	Method used for payment (e.g., Cash, Bank Transfer, Check)	Optional field
Date	Decimal	10,2	Yes	Amount of payment received	Positive number, two decimal places

Relationships:

- Associated with one Lease (Many-to-One)

Note: The attribute "Date" in the diagram appears to be the payment amount; this should be clarified as "Amount" for clarity.

Entity: Expense

Description: Cost incurred related to property operations.

Attribute	Type	Length	Required	Description	Constraints/Rules
ExpenseID	Integer	-	Yes	Unique identifier for each expense	Primary Key, Auto-generated
Amount	Decimal	10,2	Yes	Cost of the expense	Positive number, two decimal places
Category	String	50	Yes	Type of expense	Must be "Maintenance", "Utilities", "Taxes", or other defined category
Date	Date	-	Yes	Date when expense was incurred	Cannot be future date

Relationships:

- Associated with one Property (Many-to-One)

Entity: Maintenance Request

Description: Report of repair or maintenance issue for a specific unit.

Attribute	Type	Length	Required	Description	Constraints/Rules
RequestID	Integer	-	Yes	Unique identifier for each maintenance request	Primary Key, Auto-generated
IssueDescription	Text	1000	Yes	Detailed description of the maintenance problem	Minimum 10 characters

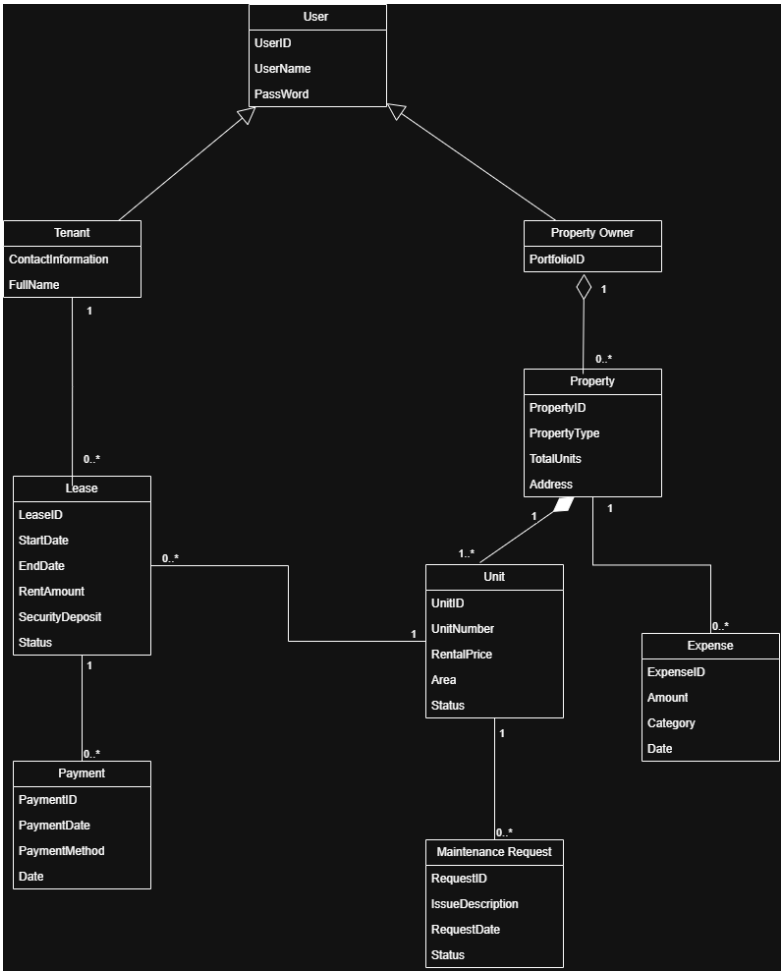
RequestDate	Date	-	Yes	Date when issue was reported	Defaults to current date
Status	String	20	Yes	Current state of the maintenance request	Must be "New", "In Progress", or "Completed"

Relationships:

- Associated with one Unit (Many-to-One)

Business Rules:

- System alerts if Status remains "New" or "In Progress" beyond defined threshold (e.g., 30 days)
- When Unit has active maintenance with Status "In Progress", Unit Status may be set to "Under Maintenance"



8. The Scope of the Product

8a. Product Boundary

The Product Boundary defines what functionality is included within the Property Management System (PPMS) and identifies the actors who interact with the system.

System Context Diagram Description:

The PPMS consists of four main functional areas as illustrated in the use case diagrams:

1. Searching for Properties (Public Module)

- **Actor:** Tenant (Potential Tenant/Public User)
- **System Boundary:** Property Finder module accessible without authentication
- **Interactions:** View vacant units, search/filter units, inquire about units/property

2. Viewing Dashboard (Administrative Module)

- **Actor:** Property Owner
- **System Boundary:** Administrative dashboard and core management functions
- **Interactions:** View property/lease status, record payments (includes payment use case), manage tenants, manage property maintenance

3. Reporting (Analytics Module)

- **Actors:** Data Admin, Development Team, CEO, Technical Lead
- **System Boundary:** Analytics, reporting, and system monitoring capabilities
- **Interactions:** Generate analytics reports, provide usage metrics, view executive dashboard, view technical metrics

4. Compliance and Law (Regulatory Module)

- **Actors:** Tax Authority, Compliance Officer
- **System Boundary:** Regulatory reporting and compliance documentation
- **Interactions:** Provide financial reports, provide compliance reports

8b. Product Use Case Table

Use Case ID	Use Case Name	Actor(s)	Priority	Summary
PUC-01	View vacant units	Tenant	High	Potential tenant views list of available properties without logging in
PUC-02	Search/filter units	Tenant	High	Potential tenant applies search criteria to find properties matching needs
PUC-03	Inquire about unit/property	Tenant	High	Potential tenant submits inquiry form to contact property owner
PUC-04	Add/update/delete properties	Property Owner	Critical	Property owner manages property records in the system
PUC-05	Add/remove tenants	Property Owner	Critical	Property owner creates, updates, or deletes tenant information
PUC-06	Add/update/delete leases	Property Owner	Critical	Property owner manages lease agreements and terms
PUC-07	Record payments	Property Owner	Critical	Property owner logs rent payments received from tenants
PUC-08	Add/update/delete property maintenance	Property Owner	High	Property owner logs and tracks maintenance requests and status
PUC-09	View dashboard	Property Owner	Critical	Property owner accesses summary of key metrics and alerts

PUC-10	Generate analytics report	Data Admin	Medium	Data administrator creates analytical reports on property performance
PUC-11	Provide usage metrics	Development Team	Medium	Development team accesses system usage and interaction data
PUC-12	View executive dashboard	CEO	Medium	CEO views high-level strategic business metrics
PUC-13	View technical metrics	Technical Lead	Medium	Technical lead monitors system performance and code quality
PUC-14	Provide financial report	Tax Authority	High	System generates financial reports for tax compliance verification
PUC-15	Provide compliance report	Compliance Officer	Medium	System generates reports demonstrating regulatory compliance
PUC-16	User login/logout	Property Owner	Critical	Property owner authenticates to access administrative functions
PUC-17	Manage user accounts	Property Owner	Medium	Property owner creates, updates, or deletes user accounts
PUC-18	Export data	Property Owner, Tax Authority	High	Users export financial or property data to external formats
PUC-19	Backup/restore data	Data Admin	Critical	System administrator performs data backup and recovery operations

PUC-20	View expiring leases alert	Property Owner	Critical	System automatically displays leases expiring within 60 days

---

8c. Individual Product Use Cases

Product Use Case PUC-01: View Vacant Units

Primary Actor: Tenant (Potential Tenant/Public User)

Goal: Allow potential tenants to browse all currently available rental units

Trigger: Potential tenant accesses the Property Finder webpage

Main Success Scenario:

1. Potential tenant opens Property Finder module via web browser or mobile app
2. System retrieves all units with status "Vacant" from database
3. System displays list of available units with basic information (property address, unit number, rental price)
4. Potential tenant scrolls through the list of vacant units
5. Potential tenant can select a unit to view detailed information

Alternative Paths:

- 2a. No vacant units exist in system
    - System displays message: "No properties currently available. Please check back soon."
    - Use case ends

Product Use Case PUC-02: Search/Filter Units

Primary Actor: Tenant (Potential Tenant/Public User)

Goal: Enable potential tenants to narrow available units based on specific criteria

Trigger: Potential tenant applies search filters

Main Success Scenario:

1. Potential tenant is viewing list of vacant units
2. Potential tenant selects filter criteria:
    - Location/address (city, district, street)
    - Maximum rental price
    - Property type (Residential or Commercial)
3. System applies selected filters to vacant units list
4. System displays filtered results matching all selected criteria
5. Potential tenant reviews filtered results

Alternative Paths:

- 4a. No units match filter criteria
    - System displays message: "No properties match your search criteria. Try adjusting your filters."
    - Potential tenant can modify filters
    - Return to step 2
- 2a. Potential tenant clears all filters
    - System displays complete list of all vacant units
    - Return to main success scenario

Product Use Case PUC-03: Inquire About Unit/Property

Primary Actor: Tenant (Potential Tenant/Public User)

Goal: Allow potential tenant to contact property owner regarding a specific unit

Trigger: Potential tenant clicks "Inquire" or "Contact" button on property listing

Main Success Scenario:

1. Potential tenant selects specific unit from search results
2. Potential tenant clicks inquiry/contact option
3. System displays inquiry form with fields:
    - Name (required)
    - Contact information (phone/email, required)
    - Message/inquiry details (required)
    - Unit reference (auto-filled)
4. Potential tenant completes form fields
5. Potential tenant submits inquiry
6. System validates all required fields are complete
7. System records inquiry with timestamp and associated property/unit
8. System sends inquiry notification to property owner
9. System displays confirmation message to potential tenant

Alternative Paths:

- 6a. Required fields missing or invalid
    - System highlights incomplete/invalid fields
    - System displays error message
    - Potential tenant corrects information
    - Return to step 5
- 5a. Potential tenant cancels inquiry
    - System discards form data

- System returns to property listing
- Use case ends

Product Use Case PUC-04: Add/Update/Delete Properties

Primary Actor: Property Owner

Goal: Manage property records in the system

Trigger: Property owner accesses Property Management module

Main Success Scenario (Add Property):

1. Property owner navigates to Property Management module
2. Property owner selects "Add New Property"
3. System displays property registration form
4. Property owner enters:
    - Property address (street, city, district)
    - Property type (Residential or Commercial)
    - Total number of units
5. Property owner submits form
6. System validates all required fields
7. System checks for duplicate address
8. System generates unique PropertyID
9. System saves property record to database
10. System displays confirmation with PropertyID
11. System updates property owner's property list

Alternative Paths (Update Property):

- 2a. Property owner selects existing property and chooses "Edit"
    - System loads current property information into form
    - Property owner modifies desired fields
    - Continue to step 5

Alternative Paths (Delete Property):

- 2b. Property owner selects existing property and chooses "Delete"
    - System checks if property has associated units or active leases
    - If associations exist, system displays warning
    - Property owner confirms deletion
    - System removes property and associated records
    - System displays confirmation
    - Use case ends

Product Use Case PUC-07: Record Payments

Primary Actor: Property Owner

Goal: Log rent payment received from tenant

Trigger: Property owner accesses Financial Management module to record payment

Main Success Scenario:

1. Property owner navigates to Financial Management module
2. Property owner selects "Record Payment"
3. System displays payment entry form
4. Property owner selects tenant from dropdown list
5. System auto-fills associated lease and rental amount
6. Property owner enters:
    - Payment date (defaults to current date)
    - Payment amount
    - Payment method (optional: Cash, Transfer, Check, etc.)
7. Property owner submits payment record
8. System validates required fields
9. System checks payment amount against lease rent amount
10. System saves payment record linked to lease

11. System updates financial summary (monthly income)
12. System displays confirmation message
13. System updates dashboard financial metrics

Alternative Paths:

- 9a. Payment amount significantly differs from lease amount
    - System displays warning: "Payment amount differs from lease rent. Proceed?"
    - Property owner confirms or adjusts amount
    - Continue to step 10
- 5a. Selected tenant has multiple active leases
    - System prompts property owner to select specific lease
    - Property owner selects correct lease
    - Continue to step 6
- 7a. Property owner cancels operation
    - System discards entered data
    - Return to Financial Management module
    - Use case ends

Product Use Case PUC-09: View Dashboard

Primary Actor: Property Owner

Goal: Access summary view of key property management metrics and alerts

Preconditions:

- Property owner is logged in as administrator
- Property owner has at least one property in system

Trigger: Property owner logs in or navigates to dashboard

Main Success Scenario:

1. Property owner accesses dashboard (default landing page after login)

2. System retrieves property owner's portfolio data
3. System calculates key metrics:
    - Total number of properties
    - Total number of units
    - Units by status (Vacant, Occupied, Under Maintenance)
    - Leases expiring within 60 days
    - Monthly income vs. expenses summary
4. System identifies alerts:
    - Expiring leases (within 60 days)
    - Unresolved maintenance requests (exceeding threshold)
5. System displays dashboard with:
    - Summary metrics in visual cards/widgets
    - Expiring leases list with tenant and property details
    - Financial summary (current month income vs. expenses)
    - Maintenance alerts
    - Quick action buttons (Add Property, Record Payment, etc.)
6. Property owner reviews dashboard information
7. Property owner can click on any metric or alert to navigate to detailed view

Alternative Paths:

- 2a. Property owner has no properties registered
    - System displays welcome message with prompt to add first property
    - Dashboard shows empty state with "Add Property" call-to-action
    - Use case ends
- 7a. Property owner clicks on expiring lease
    - System navigates to Lease Management module with specific lease displayed

    - Use case ends
- 7b. Property owner clicks on financial summary
    - System navigates to Financial Management module with current month report
    - Use case ends
- 7c. Property owner clicks on maintenance alert
    - System navigates to Maintenance Management module with filtered view of pending requests
    - Use case ends

Product Use Case PUC-14: Provide Financial Report

Primary Actor: Tax Authority

Secondary Actor: Property Owner (initiates export)

Goal: Generate and export financial report for tax compliance purposes

Trigger: Tax authority requests financial documentation, or property owner proactively exports for tax filing

Main Success Scenario:

1. Property owner navigates to Financial Management module
2. Property owner selects "Export Financial Report"
3. System displays export options form
4. Property owner specifies:
    - Date range (start date, end date)
    - Report format (CSV or PDF)
    - Data to include (Income only, Expenses only, or Both)
5. Property owner submits export request
6. System validates date range
7. System retrieves all income and expense records within specified period
8. System calculates totals and subtotals by category

9. System generates report in requested format with:
    - Property owner information
    - Report period
    - Income records (date, tenant, amount, property)
    - Expense records (date, category, amount, property)
    - Category subtotals (Maintenance, Utilities, Taxes)
    - Total income and total expenses
    - Net income (Income - Expenses)
10. System provides download link or file for export
11. Property owner downloads report
12. Property owner provides report to tax authority

Alternative Paths:

- 6a. Invalid date range (end date before start date)
    - System displays error message
    - Property owner corrects dates
    - Return to step 5
- 7a. No financial data exists for specified period
    - System displays message: "No financial records found for selected period"
    - Property owner adjusts date range or cancels
    - Return to step 4 or use case ends
- 10a. Property owner selects "Email Report"
    - System sends report to property owner's registered email
    - System displays confirmation
    - Use case ends

Product Use Case PUC-20: View Expiring Leases Alert

Primary Actor: Property Owner

Goal: Automatically identify and display leases approaching expiration

Trigger: System performs daily automated check or property owner accesses dashboard

Main Success Scenario:

1. System runs automated daily check (scheduled process)
2. System retrieves all lease records with status "Active"
3. For each active lease, system calculates: Days Until Expiration = Lease.EndDate - Current Date
4. System identifies leases where Days Until Expiration ≤ 60 days
5. System flags these leases as "Expiring"
6. When property owner accesses dashboard:
7. System displays "Expiring Leases" section with:
    - Number of expiring leases
    - List showing: Tenant name, Property address, Unit number, End date, Days remaining
8. Property owner reviews expiring leases list
9. Property owner can click on lease to view details or initiate renewal

Alternative Paths:

- 4a. No leases expiring within 60 days
    - System displays message: "No leases expiring soon"
    - Dashboard shows green status indicator
    - Use case ends
- 4b. Lease has already expired (End Date < Current Date)
    - System marks lease as "Expired" status
    - System changes associated unit status to "Vacant"
    - Lease appears in "Expired Leases" section, not "Expiring"
- 9a. Property owner initiates lease renewal
    - System navigates to Lease Management with renewal option

- Property owner creates new lease or extends existing
- Use case transitions to PUC-06 (Manage Leases)

9. Functional Requirements

9a. Functional Requirements for Tenants (Potential Tenants/Public Users)

FR-T1: Property Search and Filtering

Description: The system shall allow potential tenants to search and filter available properties based on location, rental price range, and property type without requiring authentication.

Rationale: Potential tenants need an efficient way to narrow down available units that match their specific needs and budget before making contact with property owners.

Fit Criterion:

- System displays all vacant units within 2 seconds of accessing Property Finder
- Filter results update within 1 second of applying search criteria
- System successfully filters by location (city/district), maximum rental price, and property type (Residential/Commercial)
- Filtered results show only units matching ALL selected criteria (AND logic)

Priority: High

Related Use Cases: PUC-01, PUC-02

FR-T2: Property Inquiry Submission

Description: The system shall provide an inquiry form that allows potential tenants to submit their contact information and message regarding a specific property, and shall deliver this inquiry to the property owner.

Rationale: Potential tenants need a simple, direct method to express interest and contact property owners about available units without requiring phone calls.

Fit Criterion:

- Inquiry form accepts name, contact information (phone/email), and message (minimum 10 characters)

- System validates all required fields before submission
- System successfully records inquiry with timestamp and associated property/unit reference
- Property owner receives inquiry notification within their dashboard immediately after submission
- System displays confirmation message to tenant within 2 seconds of successful submission
- 95% of inquiry submissions complete successfully without errors

Priority: High

Related Use Cases: PUC-03

FR-T3: Vacant Unit Display

Description: The system shall display all properties with units marked as "Vacant" status in the public Property Finder module, showing property address, unit specifications, and rental price in a read-only format.

Rationale: Potential tenants need accurate, current information about available rental units to make informed decisions about which properties to inquire about.

Fit Criterion:

- System displays ONLY units with status = "Vacant"
- Each listing shows: property address, unit number, unit area, rental price, property type
- Property information updates automatically when unit status changes to "Vacant"
- Unit information is displayed in read-only format (no editing capability for public users)
- System removes units from display within 5 seconds when status changes from "Vacant" to "Occupied" or "Under Maintenance"
- Display layout is mobile-responsive and readable on smartphone screens

Priority: High

Related Use Cases: PUC-01

9b. Functional Requirements for Property Owners

FR-PO1: User Authentication and Authorization

Description: The system shall allow property owners to securely log in using valid credentials (username and password) and shall provide role-based access control restricting administrative functions to authorized users only.

Rationale: Property owners need secure access to protect sensitive tenant and financial data, while preventing unauthorized access to administrative functions.

Fit Criterion:

- System authenticates user credentials against stored encrypted passwords
- Login fails with invalid credentials and displays error message "Invalid username or password"
- Successful login grants access to administrative dashboard within 3 seconds
- System restricts Property Finder public access from administrative functions
- System provides secure logout function that terminates session completely
- Session expires after 30 minutes of inactivity
- System prevents access to administrative functions without valid authentication

Priority: Critical

Related Use Cases: PUC-16

FR-PO2: Property and Unit Management

Description: The system shall allow property owners to create, read, update, and delete property records with essential details (address, property type, number of units) and individual unit records (unit number, area, rental price, status).

Rationale: Property owners need comprehensive control over their property portfolio data to maintain accurate records of all rental assets.

Fit Criterion:

- System successfully creates property record with unique PropertyID within 2 seconds
- System validates required fields: address (not empty, unique), property type (Residential or Commercial), total units (positive integer ≥ 1)

- System allows modification of existing property details except PropertyID
- System successfully creates unit record with unique UnitID linked to property
- System validates unit fields: unit number (unique within property), rental price (positive decimal), status (Vacant/Occupied/Under Maintenance)
- System prevents deletion of property with active leases without confirmation warning
- System displays updated property list immediately after create/update/delete operations
- 100% of property addresses must be unique (no duplicates allowed)

Priority: Critical

Related Use Cases: PUC-04

FR-PO3: Dashboard with Alerts and Summaries

Description: The system shall provide an administrative dashboard that displays key portfolio metrics, automatically identifies and displays leases expiring within 60 days, and presents a monthly financial summary comparing income versus expenses.

Rationale: Property owners need immediate visibility into critical business metrics and alerts to make timely decisions and prevent missed opportunities or deadlines.

Fit Criterion:

- Dashboard loads within 3 seconds of property owner login
- Dashboard displays: total properties, total units, unit count by status (Vacant/Occupied/Under Maintenance)
- System automatically identifies leases where (EndDate - CurrentDate) ≤ 60 days
- Expiring leases section displays: tenant name, property address, unit number, end date, days remaining
- Financial summary shows current month: total income, total expenses, net income (income - expenses)
- Dashboard data refreshes with real-time information on each access
- System updates dashboard within 5 seconds after any data change (payment recorded, lease created, unit status changed)

- Maintenance alerts display requests with status "New" or "In Progress" exceeding 30 days
- Dashboard provides clickable links to navigate to detailed views of each metric

Priority: Critical

Related Use Cases: PUC-09, PUC-20

10. Look and Feel Requirements

10a. Appearance Requirements

AR-01: Professional and Trustworthy Aesthetic
Description: The application shall present a clean, professional, and trustworthy appearance that instills confidence in property owners managing financial and legal data.
Rationale: Property owners are dealing with significant financial assets and sensitive tenant information; the visual design must reflect reliability and competence.
Fit Criterion:

- 90% of property owner test users agree with the statement "The application's design appears professional and trustworthy" in post-usability surveys.
- The color scheme and typography are consistent with established design guidelines for financial and business applications.

AR-02: Clear Visual Hierarchy and Readability
Description: The interface shall use a clear visual hierarchy to distinguish between different types of information (headings, data, actions, alerts) and ensure all text is legible on mobile screens.
Rationale: Users with limited time need to scan and find information quickly without confusion.
Fit Criterion:

- All text maintains a minimum contrast ratio of 4.5:1 against its background.
- Key metrics and alerts on the dashboard are visually prominent and identifiable within 3 seconds by test users.
- Font size for body text is no smaller than 16px on standard mobile displays.

AR-03: Egyptian Cultural Resonance
Description: The visual design shall incorporate subtle elements that resonate with the Egyptian user base, such as culturally appropriate color palettes and imagery, without

compromising usability.
Rationale: Creating a sense of familiarity and local relevance can improve user adoption and comfort.
Fit Criterion:

- A palette of colors commonly associated with Egyptian aesthetics (e.g., blues, sand tones, gold accents) is used appropriately in non-critical UI elements.
- 80% of Egyptian user testers agree with the statement "The application feels relevant and designed for my market."

10b. Style Requirements

SR-01: Consistent and Intuitive Navigation
Description: The application shall maintain a consistent navigation structure and placement of common action buttons (e.g., Save, Cancel, Add New) across all modules.
Rationale: Consistency reduces the learning curve and allows users to build usage patterns, increasing efficiency.
Fit Criterion:

- The primary navigation menu is accessible from all screens within the application with identical layout and labeling.
- Users can successfully complete core tasks (e.g., adding a property, recording a payment) without training on their second attempt.

SR-02: Mobile-First, Touch-Optimized Design
Description: All interactive elements (buttons, form fields, links) shall be designed for touch interaction on mobile devices, with adequate size and spacing.
Rationale: The primary usage environment is mobile; the interface must prevent input errors and be easy to use on-the-go.
Fit Criterion:

- All touch targets (buttons, icons) have a minimum size of 44x44 pixels.
- There is sufficient spacing (minimum 8 pixels) between interactive elements to prevent mis-taps.
- Forms are optimized for vertical scrolling and single-column layout on mobile screens.

SR-03: Contextual and Informative Feedback
Description: The system shall provide immediate and clear visual feedback for all user actions, including success confirmations, error states, and loading indicators.

Rationale: Users need to understand the result of their actions, especially when managing important property and financial data.
Fit Criterion:

- After any data submission (e.g., saving a property), a clear confirmation message is displayed for at least 2 seconds.
- Form validation errors are highlighted immediately upon submission, with a clear text explanation.
- For operations taking longer than 1 second, a loading indicator is displayed.

SR-04: Status-Driven Visual Coding
Description: The system shall use a consistent color and icon scheme to represent the status of key entities (e.g., Units, Leases, Maintenance Requests).
Rationale: Visual coding allows for rapid assessment of portfolio health without reading detailed text.
Fit Criterion:

- Unit Status: "Vacant" (e.g., Green), "Occupied" (e.g., Blue), "Under Maintenance" (e.g., Orange).
- Lease Status: "Active" (e.g., Blue), "Expiring" (e.g., Orange), "Expired" (e.g., Red).
- Maintenance Status: "New" (e.g., Red), "In Progress" (e.g., Orange), "Completed" (e.g., Green).
- 95% of users can correctly identify the status of 10 sample units/leases based solely on their color/icon in usability tests.

11. Usability and Humanity Requirements

11a. Ease of Use Requirements

EUR-01: Efficient Task Completion for Primary Users
Description: The system shall enable property owners to complete their most frequent tasks (recording a payment, checking lease status, logging maintenance) in minimal steps and time.
Rationale: Property owners have limited time and manage properties as a secondary activity; efficiency is critical for adoption.
Fit Criterion:

- Recording a rent payment shall take a proficient user no more than 3 taps and 30 seconds from the dashboard.

- Checking the status of all leases shall be possible from the main dashboard without navigation.
- 90% of test users can successfully record a payment and log a maintenance request on their first attempt without assistance.

EUR-02: Minimal Data Entry
Description: The system shall minimize manual data entry through auto-population, default values, and selection from predefined lists wherever possible.
Rationale: Reducing typing on mobile devices decreases time-on-task and input errors.
Fit Criterion:

- When recording a payment, the tenant name, associated lease, and standard rent amount are auto-populated upon tenant selection.
- Date fields default to the current date.
- At least 70% of form fields in core workflows use selection (dropdowns, toggles) rather than free-text entry.

11b. Personalization and Internationalization Requirements

PIR-01: Egyptian Arabic Language Support
Description: The system shall support Egyptian Arabic as the primary language for all user-facing text, including labels, messages, and help content.
Rationale: The primary user base is located in Egypt and is more comfortable conducting business in Arabic.
Fit Criterion:

- 100% of the application's user interface text is available and accurately translated into Egyptian Arabic.
- Users can select Arabic as their preferred language upon first launch and within the application settings.

PIR-02: Date and Currency Formatting
Description: The system shall display dates and currency values using formats standard for Egypt.
Rationale: To prevent confusion and ensure clarity for financial and contractual data.
Fit Criterion:

- Dates are displayed in the DD/MM/YYYY format.

- Currency is displayed using the Egyptian Pound (EGP) symbol (ج.م. or EGP) with appropriate decimal formatting.

11c. Learning Requirements

LR-01: Zero-Training Startup
Description: A property owner with journeyman-level technology experience shall be able to perform basic operations without formal training.
Rationale: Property owners have limited time and availability for training sessions.
Fit Criterion:

- 80% of test users can successfully add a property and record their first payment within 10 minutes of first launching the application, using only in-app guidance.
- A contextual "What is this?" help icon is available on all screens explaining key terms and functions.

LR-02: Contextual Onboarding
Description: The system shall provide a brief, interactive onboarding tour upon first use, highlighting key features and navigation.
Rationale: To accelerate user proficiency and demonstrate immediate value.
Fit Criterion:

- The onboarding process is completed in less than 2 minutes.
- Users who complete onboarding show a 50% reduction in time to complete core tasks on their first attempt compared to those who do not.

11d. Understandability and Politeness Requirements

UPR-01: Clear, Non-Technical Language
Description: All system messages, labels, and instructions shall use clear, simple language familiar to property owners, avoiding technical jargon.
Rationale: Users have varying levels of technical expertise but share common property management domain knowledge.
Fit Criterion:

- 100% of error messages and instructions are phrased in terms of the business problem (e.g., "Please select a tenant for this payment" instead of "Null value in TenantID field").
- A panel of target users identifies zero instances of confusing or overly technical terminology during a review of all interface text.

UPR-02: Constructive and Respectful Communication
Description: The system shall communicate errors and alerts in a constructive manner that guides the user toward a solution without assigning blame.
Rationale: To maintain a positive user experience and reduce frustration.
Fit Criterion:

- All error messages shall suggest a corrective action (e.g., "The end date must be after the start date. Please adjust the dates.").
- Warnings for destructive actions (e.g., deleting a property) shall be phrased respectfully and require explicit confirmation (e.g., "Are you sure? This property and all its unit data will be permanently deleted.").

11e. Accessibility Requirements

ACR-01: Screen Reader Compatibility
Description: The application shall be compatible with built-in mobile screen readers (e.g., VoiceOver on iOS, TalkBack on Android) to ensure access for users with visual impairments.
Rationale: To ensure the product is usable by property owners and potential tenants with disabilities.
Fit Criterion:

- All interactive elements have descriptive labels readable by screen readers.
- All informative images and status icons have appropriate alternative text (alt text).
- The application achieves a minimum of WCAG 2.1 Level AA compliance for mobile applications.

ACR-02: Scalable Text and Sufficient Contrast
Description: Text content shall be scalable up to 200% without loss of content or functionality, and maintain sufficient color contrast.
Rationale: To support users with low vision.
Fit Criterion:

- Users can use the device's native text scaling settings to increase the application's text size without breaking the layout.
- All text maintains a contrast ratio of at least 4.5:1, and large text (over 18pt) maintains a ratio of 3:1, as verified by automated testing tools.

12. Performance Requirements

12a. Speed and Latency Requirements

SLR-01: Application Response Time
Description: The system shall provide rapid feedback to user interactions to maintain a fluid and responsive experience.
Rationale: Slow response times disrupt workflow and reduce user confidence, especially for on-the-go property management.
Fit Criterion:

- The dashboard shall load and display all key metrics within 3 seconds of user login on a standard 4G mobile connection.
- All user interface transitions (screen changes) shall complete within 1 second.
- Search results in the Property Finder shall be displayed within 2 seconds of initiating a search.

SLR-02: Data Submission Confirmation
Description: The system shall provide confirmation of successful data submission (e.g., saving a property, recording a payment) with minimal delay.
Rationale: Users need immediate assurance that their critical business data has been saved.
Fit Criterion:

- A visual confirmation for successful create, update, or delete operations shall be displayed to the user within 2 seconds of initiating the action over a standard 4G connection.

12b. Safety-Critical Requirements

Not applicable for this business application. The PPMS does not control any physical systems or processes where failure could lead to injury, loss of life, or catastrophic environmental damage.

12c. Precision or Accuracy Requirements

PAR-01: Financial Calculation Accuracy
Description: All financial calculations (totals, summaries, net income) shall be mathematically precise and consistent.
Rationale: Inaccurate financial data can lead to poor business decisions, tax filing errors,

and legal disputes.
Fit Criterion:

- All financial calculations (e.g., monthly net income, category subtotals) shall be 100% accurate when compared to a manual audit of the underlying transaction data.
- Currency values shall be stored and calculated with exact decimal precision without floating-point rounding errors.

PAR-02: Data Integrity and Validation
Description: The system shall enforce data validation rules to ensure the accuracy and consistency of all stored information.
Rationale: Preventing invalid data at the point of entry maintains the overall quality and reliability of the system.
Fit Criterion:

- The system shall prevent the saving of any record with missing mandatory fields.
- The system shall enforce business rules (e.g., lease end date after start date, unique property addresses) with 100% effectiveness.

12d. Reliability and Availability Requirements

RAR-01: System Uptime
Description: The core application services and database shall be highly available to support user access at any time.
Rationale: Property management needs can arise outside of business hours; users expect constant access.
Fit Criterion:

- The system shall achieve 99.5% uptime during core local business hours (8:00 AM - 10:00 PM Egypt Time), excluding scheduled maintenance.

RAR-02: Scheduled Maintenance Communication
Description: Any planned system downtime for maintenance shall be communicated to users at least 24 hours in advance.
Rationale: To allow users to plan their activities and manage expectations.
Fit Criterion:

- 100% of planned maintenance windows are announced via an in-app notification system at least 24 hours prior to commencement.

12e. Robustness or Fault-Tolerance Requirements

FTR-01: Graceful Handling of Network Failure
Description: The application shall handle loss of network connectivity gracefully without losing user data or crashing.
Rationale: Users operate in mobile environments with potentially unstable internet connections.
Fit Criterion:

- If a network connection is lost during a data submission, the application shall save the data locally and automatically retry the synchronization when the connection is restored.
- The user shall receive a clear notification that they are working offline and which data is pending synchronization.

FTR-02: Error Recovery
Description: The system shall recover cleanly from application errors and allow the user to continue their work.
Rationale: To prevent data loss and user frustration from unexpected crashes.
Fit Criterion:

- In the event of an application crash, upon restart, the user shall be returned to the screen they were previously on, with any unsaved data in forms recovered where possible.

12f. Capacity Requirements

CR-01: User and Data Volume
Description: The system shall support the projected number of concurrent users and the volume of property data.
Rationale: The system must scale to accommodate the target market of individual landlords in Egypt.
Fit Criterion:

- The system shall support up to 5,000 concurrent active property owner users.
- The database shall efficiently handle portfolios comprising up to 50 properties and 200 units per single property owner.

12g. Scalability or Extensibility Requirements

SER-01: Architectural Scalability
Description: The system architecture shall be designed to allow for scaling to accommodate increased user load and data volume without major re-engineering.

Rationale: To support business growth and market expansion.
Fit Criterion:

- The system shall be deployable on cloud infrastructure that allows for horizontal scaling of application servers and database resources.
- A 50% increase in concurrent users shall not require a change to the application's core architecture.

SER-02: Feature Extensibility
Description: The system shall be designed to facilitate the future addition of new modules and features.
Rationale: To accommodate evolving user needs and potential integration with external services (e.g., payment gateways, accounting software) in future versions.
Fit Criterion:

- The codebase shall be modular, with clear separation between the core data model, business logic, and user interface layers.
- A new, simple feature (e.g., a new expense category) can be added with changes isolated to a single module.

12h. Longevity Requirements

LR-01: Technology Viability
Description: The technological stack (programming languages, frameworks, databases) shall be based on currently supported and widely-adopted platforms with a foreseeable long-term support horizon.
Rationale: To ensure the system remains maintainable and secure for its expected operational lifespan.
Fit Criterion:

- All core technologies used in development shall be from vendors/projects that provide public support and update roadmaps, with no core technology being in a legacy or end-of-life state at the time of launch.

13. Operational and Environmental Requirements

13a. Expected Physical Environment

EPE-01: Mobile-First Operating Environment
Description: The product shall be designed to operate primarily on smartphones (iOS and Android) used in varied mobile environments typical for individual property owners in Egypt.
Rationale: The primary users manage properties as a secondary activity and require access outside of a fixed office location.
Fit Criterion:

- The application shall function correctly on mobile devices with screen sizes ranging from 4.7 inches to 6.7 inches.
- The application shall be usable in environments with variable network connectivity, from high-speed Wi-Fi to standard 4G/5G mobile data networks.
- Core functionality (viewing data, recording payments) shall remain accessible during intermittent network connectivity.

EPE-02: Support for Egyptian Mobile Infrastructure
Description: The system shall be optimized for the performance characteristics and data plans commonly available in the Egyptian market.
Rationale: To ensure a cost-effective and performant user experience for the target audience.
Fit Criterion:

- The application's initial download size shall not exceed 50 MB.
- The average data usage for a typical 10-minute session of active use (excluding initial download) shall be less than 5 MB.

13b. Requirements for Interfacing with Adjacent Systems

IAS-01: One-Way Data Export to Tax Authority
Description: The system shall provide a capability to export standardized financial reports in PDF and CSV formats for manual submission to the Egyptian Tax Authority. This is a one-way data flow out of the system.
Rationale: To fulfill the compliance requirement without requiring a direct, real-time system integration.
Fit Criterion:

- The exported financial report shall include all data fields required by the Egyptian Tax Authority for rental income documentation for a user-selected date range.
- The export function shall generate a valid file within 30 seconds of the user's request.

IAS-02: Read-Only Access for Potential Tenants
Description: The public Property Finder module shall provide unauthenticated, read-only access to vacant unit listings.
Rationale: To allow potential tenants to discover available properties without granting them access to the administrative system.
Fit Criterion:

- Public users can view all data within the Property Finder without providing login credentials.
- The system shall enforce data access rules that prevent public users from viewing any data marked as non-public (e.g., owner contact info, tenant data, financial records).

13c. Productization Requirements

PR-01: Self-Contained Deployment
Description: The delivered product shall be a self-contained application that does not require the installation of additional third-party software or components for the end-user to operate.
Rationale: To simplify the installation process and ensure a consistent user experience.
Fit Criterion:

- The application is downloadable and installable directly from the Apple App Store and Google Play Store without supplementary downloads.
- All necessary libraries and frameworks are bundled within the application package.

PR-02: Branding and Configuration
Description: The application shall display consistent branding (logo, color scheme) for the PPMS product and shall be configurable for different deployment environments (e.g., development, staging, production).
Rationale: To present a professional, market-ready product and support a standard software development lifecycle.
Fit Criterion:

- The application name and logo displayed to users are "PPMS" and its associated branding.
- Connection endpoints (e.g., for database, APIs) are configurable without requiring a rebuild of the application binary.

13d. Release Requirements

RR-01: Mobile Platform Store Compliance
Description: The application release packages shall comply with all current submission guidelines and technical requirements of the Apple App Store and Google Play Store.
Rationale: To ensure the application can be distributed through the primary mobile software marketplaces.
Fit Criterion:

- Each release candidate passes automated validation checks for the respective app store (e.g., Apple's App Store Connect validation, Google Play's pre-launch report) without critical errors.

RR-02: Versioning and Update Management
Description: The application shall support a clear versioning scheme and allow users to receive update notifications and install new versions seamlessly.
Rationale: To facilitate the delivery of new features, bug fixes, and security patches.
Fit Criterion:

- The application version is displayed in the settings menu.
- The application checks for updates on startup and can notify the user when a new version is available in the respective app store.

14. Maintainability and Support Requirements

14a. Maintenance Requirements

MR-01: Modular System Architecture
Description: The system shall be designed with a modular architecture where components are loosely coupled and have well-defined interfaces.
Rationale: To enable efficient debugging, updating, and replacement of individual system components without impacting the entire system.
Fit Criterion:

- A change to one business module (e.g., Lease Management) shall not require changes to unrelated modules (e.g., Property Finder).
- 90% of code dependencies shall flow inward toward core business logic, with no cyclic dependencies between modules.

MR-02: Comprehensive Logging
Description: The system shall maintain detailed logs of user actions, system errors, and performance metrics.
Rationale: To facilitate troubleshooting, audit trails, and understanding of system behavior in production.
Fit Criterion:

- All API endpoints log request and response summaries.
- All database update operations (Create, Update, Delete) are logged with the user ID, timestamp, and record identifier.
- System errors are logged with sufficient detail (stack trace, context) to diagnose the root cause without needing to reproduce the issue.

MR-03: Database Schema Management
Description: All changes to the database schema shall be applied through version-controlled migration scripts.
Rationale: To ensure schema changes are reproducible, reversible, and can be applied consistently across development, testing, and production environments.
Fit Criterion:

- 100% of database structure changes are defined in incremental, versioned SQL migration scripts.

- The database can be recreated from an empty state to the current version by running the migration script sequence.

14b. Supportability Requirements

SR-01: Remote Diagnostics and Monitoring
Description: The system shall provide health monitoring endpoints and expose key performance indicators for support personnel.
Rationale: To allow the support team to proactively identify issues and quickly diagnose problems reported by users.
Fit Criterion:

- A dedicated /health API endpoint returns system status (database connectivity, etc.).
- A monitoring dashboard is available for support staff showing system uptime, average response times, and error rates.
- Anonymous usage statistics (feature usage frequency, performance metrics) can be collected to inform support and development priorities.

SR-02: User Data Management Tools
Description: Authorized support administrators shall have controlled tools to view and, if necessary, correct user data to resolve support tickets.
Rationale: To efficiently assist users with data-related issues without granting full database access.
Fit Criterion:

- A secure administrative interface allows support staff to search for users and view their associated data (properties, tenants, leases) in a read-only format.
- With elevated privileges and a documented approval process, support staff can execute predefined data correction scripts for common issues.

14c. Adaptability Requirements

AR-01: Configurable Business Rules
Description: Key business parameters that may change over time shall be configurable without requiring code changes or a full application release.
Rationale: To allow the business to adapt to new market or regulatory conditions quickly.
Fit Criterion:

- The threshold for "expiring lease" alerts (currently 60 days) is stored in a configuration file or database table and can be modified by an administrator.

- The list of available expense categories is maintainable through an administrative interface.

AR-02: Localization Readiness
Description: The system architecture shall support the addition of new languages and regional settings without structural changes to the codebase.
Rationale: To facilitate future expansion into other markets in the region.
Fit Criterion:

- All user-facing text strings are externalized into resource files.
- The data model supports storing multiple address formats and contact number international codes.

AR-03: Adaptable Reporting
Description: The system shall allow for the creation of new report templates and data export formats through configuration.
Rationale: To meet evolving compliance and stakeholder reporting needs without developer intervention for each new report.
Fit Criterion:

- A reporting engine can generate reports based on user-defined templates (e.g., using a markup language) that specify data fields, filters, and layout.
- New CSV export formats can be defined by specifying a list of data fields and their corresponding database columns.

15. Security Requirements

15a. Access Requirements

ACR-01: Role-Based Access Control (RBAC)
Description: The system shall enforce access to functions and data based on predefined user roles (Property Owner, Public User, Data Administrator, System Administrator).
Rationale: To ensure users can only access the information and perform the actions appropriate to their role, protecting sensitive data.
Fit Criterion:

- A user with the "Public User" role cannot access any administrative functions or data.
- A "Property Owner" can only view and manage data associated with their own portfolio.
- Attempts to access unauthorized API endpoints or data records return a "403 Forbidden" error.

ACR-02: Secure Authentication
Description: The system shall require secure authentication using a username and strong password before granting access to administrative functions.
Rationale: To prevent unauthorized access to property and financial data.
Fit Criterion:

- Passwords must be a minimum of 8 characters, containing at least one uppercase letter, one lowercase letter, one number, and one special character.
- User passwords are stored using a strong, salted, and hashed encryption algorithm (e.g., bcrypt).
- After 5 consecutive failed login attempts, the user account is temporarily locked for 15 minutes.

15b. Integrity Requirements

IR-01: Data Validation and Sanitization
Description: All user input shall be validated for type, length, format, and range on both the client and server side to prevent malformed or malicious data from being processed or stored.
Rationale: To maintain data integrity and prevent injection attacks.
Fit Criterion:

- All form inputs are trimmed and sanitized of potentially harmful scripts or SQL commands before processing.
- API endpoints reject requests with parameters that do not conform to the expected data schema.
- No successful SQL injection or Cross-Site Scripting (XSS) attacks are possible as verified by automated security testing tools.

IR-02: Prevention of Concurrent Data Corruption
Description: The system shall implement mechanisms to prevent data corruption when multiple users attempt to modify the same record simultaneously.
Rationale: To ensure financial and lease data remains accurate and consistent.
Fit Criterion:

- The system uses optimistic locking or a similar mechanism to detect edit conflicts and prompt the user to resolve them.
- Financial transactions are processed in a way that maintains atomicity, consistency, isolation, and durability (ACID properties).

15c. Privacy Requirements

PRR-01: Protection of Personal Identifiable Information (PII)
Description: The system shall protect tenant PII (full name, contact information, payment history) and property owner data from unauthorized access or disclosure.
Rationale: To comply with data protection regulations and maintain tenant and landlord confidentiality.
Fit Criterion:

- Tenant PII is never exposed to Public Users through the Property Finder module.
- All PII is encrypted in transit using TLS 1.2 or higher and encrypted at rest in the database.
- Data is stored on servers located in jurisdictions compliant with Egyptian data protection laws.

PRR-02: Data Minimization
Description: The system shall collect and retain only the personal data strictly necessary for the specified business purposes.
Rationale: To reduce privacy risk and comply with the principle of data minimization.
Fit Criterion:

- The data collection forms do not request unnecessary personal information (e.g., national ID numbers) unless required for a specific, justified business need.
- A data retention policy is defined and implemented, automatically archiving or deleting tenant data a specified period after lease termination.

15d. Audit Requirements

AUR-01: Security Event Logging
Description: The system shall log all security-relevant events, including successful and failed login attempts, password changes, and data access by administrators.
Rationale: To enable security monitoring, forensic analysis, and compliance auditing.
Fit Criterion:

- The audit log captures the user ID, timestamp, event type, and IP address for each security event.
- Audit logs are stored in a secure, append-only manner and are accessible only to users with the System Administrator role.
- Audit records are retained for a minimum of one year.

AUR-02: Financial Transaction Audit Trail
Description: All financial transactions (rent payments, expense entries) shall create an immutable audit trail, recording the amount, date, associated entities, and the user who created the record.
Rationale: To ensure financial data can be traced and verified for accuracy and compliance.
Fit Criterion:

- Once a payment or expense record is saved, its core attributes (amount, date) cannot be modified, only annotated or voided, with the voiding action itself being logged.
- Every financial record includes the UserID of the person who created it and the timestamp of creation.

15e. Immunity Requirements

IMR-01: Protection Against Common Exploits
Description: The system shall be immune to common web and mobile application security vulnerabilities.
Rationale: To protect the system and its data from being compromised by standard attack

vectors.
Fit Criterion:

- The application demonstrates no critical or high-level vulnerabilities when scanned by automated static (SAST) and dynamic (DAST) application security testing tools.
- The system is not vulnerable to OWASP Top 10 security risks for the applicable year.

IMR-02: Regular Security Patching
Description: The system's dependencies (libraries, frameworks, operating system) shall be regularly updated to address known security vulnerabilities.
Rationale: To maintain a strong security posture over the application's lifespan.
Fit Criterion:

- A process is established to review and apply security patches to all third-party dependencies within 30 days of a patch being released for a critical or high-severity vulnerability.
- The development team subscribes to security advisories for all technologies used in the stack.

16. Cultural and Political Requirements

16a. Cultural Requirements

CR-01: Egyptian Arabic Language Primacy
Description: The user interface, help text, notifications, and support documentation shall be available in Egyptian Arabic as the primary language, with clear and culturally appropriate translations.
Rationale: The primary user base conducts business in Arabic; using their native language is essential for usability, trust, and adoption.
Fit Criterion:

- 100% of the application's user-facing text is professionally translated into Egyptian Arabic, ensuring colloquial terms are used where appropriate (e.g., "شقة" for apartment, "عقد إيجار" for lease).
- The Arabic interface is the default for users whose device language is set to Arabic.

CR-02: Cultural Norms in Communication and Design
Description: The application's communication style, imagery, and color scheme shall respect and reflect Egyptian cultural norms and sensibilities.
Rationale: To create a comfortable and familiar user experience that resonates with the

target audience.
Fit Criterion:

- All system messages and notifications use a formal and respectful tone consistent with Egyptian business communication.
- Imagery used in the application (icons, placeholders) reflects common Egyptian architectural styles and living environments where appropriate.
- A panel of Egyptian test users confirms that the application's design and communication style are culturally appropriate.

CR-03: Support for Islamic Calendar (Hijri)
Description: The system shall provide the option to display and, where relevant, record dates in the Hijri calendar alongside the Gregorian calendar.
Rationale: A significant portion of the user base may reference the Hijri calendar for religious or personal record-keeping.
Fit Criterion:

- Users can select a preference to view dates in the Gregorian, Hijri, or both calendars in their application settings.
- Financial reports and lease agreements can be configured to print with Hijri dates.

16b. Political Requirements

PR-01: Compliance with Egyptian Property and Rental Laws
Description: The system's functionality, particularly regarding lease agreements and financial tracking, shall align with the legal framework governing property rentals in Egypt.
Rationale: To ensure that the system supports legally compliant practices and protects both the property owner and the tenant.
Fit Criterion:

- The data model for lease agreements includes fields for all elements required by Egyptian tenancy law (e.g., specific clauses, deposit handling rules).
- The system's standard lease agreement template is reviewed and approved by a qualified Egyptian legal professional.

PR-02: Adherence to Egyptian Tax Authority Regulations
Description: The financial reporting and data export capabilities shall produce outputs that meet the specific documentation requirements of the Egyptian Tax Authority for declaring rental income.

Rationale: To enable property owners to fulfill their tax obligations correctly and efficiently.
Fit Criterion:

- The exported financial report includes all data points and follows the format specified in the latest official guidelines from the Egyptian Tax Authority for rental income reporting.
- The report generation logic is validated against a checklist provided by a tax compliance expert.

PR-03: Data Sovereignty and Local Hosting
Description: All system data, particularly PII (Personally Identifiable Information) of Egyptian citizens, shall be stored on servers physically located within Egypt.
Rationale: To comply with Egyptian data protection regulations and data sovereignty laws that may require citizen data to be stored within the country.
Fit Criterion:

- The production database and file storage servers are hosted with a cloud provider or in a data center with a verified physical presence within Egypt.
- A system configuration report confirms that no data is replicated or stored in data centers outside Egyptian borders.

17. Legal Requirements

17a. Compliance Requirements

CLR-01: Egyptian Data Protection Law Compliance
Description: The system shall comply with all applicable Egyptian data protection and privacy laws, including the Egyptian Data Protection Law (Law No. 151 of 2020).
Rationale: To ensure the legal collection, processing, and storage of personal data, avoiding significant legal penalties and protecting user rights.
Fit Criterion:

- A legal review confirms that the system's data collection forms, privacy policy, and data processing activities are fully compliant with Law No. 151 of 2020.
- Users provide explicit consent for the collection and use of their personal data upon registration, as required by law.
- Mechanisms exist for users to access, correct, and request deletion of their personal data as mandated.

CLR-02: Tenancy Law Adherence
Description: The system shall facilitate compliance with the Egyptian Tenancy Law (Law No. 4 of 1996 and its amendments), particularly in structuring lease agreements and managing security deposits.
Rationale: To ensure that the business processes supported by the system are legally sound and enforceable.
Fit Criterion:

- The lease agreement data structure includes fields for all mandatory clauses stipulated by Egyptian tenancy law.
- The system logs the receipt and return of security deposits separately from rent payments, as legally required.
- A legal expert certifies that the standard lease agreement template generated by the system is legally valid in Egypt.

CLR-03: Electronic Signature Legality
Description: If the system implements electronic signature capabilities for leases in the future, it shall comply with the Egyptian Law on Electronic Signatures (Law No. 15 of 2004).
Rationale: To ensure that electronically signed agreements are legally binding.
Fit Criterion:

- Any e-signature implementation uses a technology and process certified by the Egyptian National Postal Organization (the designated regulatory body for e-signatures).
- The system maintains a secure audit trail for e-signed documents that meets the non-repudiation requirements of the law.

17b. Standards Requirements

SR-01: Financial Data Reporting Standards
Description: Financial reports generated for tax purposes shall adhere to the standardized formats and accounting principles recognized by the Egyptian Tax Authority.
Rationale: To ensure reports are accepted by regulatory bodies and facilitate accurate tax filing for users.
Fit Criterion:

- The financial report export function generates documents that pass validation checks against the latest Egyptian Tax Authority guidelines for rental income documentation.

- Income and expense categorization aligns with standard chart of accounts used for property management in Egypt.

SR-02: Technical Security Standards
Description: The system shall implement security controls aligned with internationally recognized standards such as the OWASP Application Security Verification Standard (ASVS) and relevant aspects of ISO/IEC 27001.
Rationale: To provide a robust, verifiable level of security for protecting sensitive user data.
Fit Criterion:

- The application achieves Level 1 verification for all applicable requirements in the OWASP ASVS for the application type.
- Security policies and procedures for development and operations are documented according to ISO/IEC 27001 guidelines.

SR-03: Mobile Application Development Standards
Description: The application's development, packaging, and distribution shall comply with the official standards and guidelines set by Apple (App Store Review Guidelines) and Google (Google Play Developer Policy).
Rationale: To ensure the application is accepted and remains available on the primary mobile application distribution platforms.
Fit Criterion:

- Each application release is tested against the current Apple App Store Review Guidelines and Google Play Developer Policy checklists before submission.
- The application does not use any deprecated APIs or violate any design guidelines stipulated by the platform owners.

SR-04: Accessibility Standards
Description: The application shall strive to meet international accessibility standards to ensure usability by people with disabilities.
Rationale: To promote inclusivity and meet potential legal obligations for digital accessibility.
Fit Criterion:

- The mobile application meets the Web Content Accessibility Guidelines (WCAG) 2.1 at Level AA for mobile applications, as verified by automated and manual testing.

18. Open Issues

OI-01: Integration with Egyptian Tax Authority Systems
Description: The potential for a future direct, automated integration with the Egyptian Tax Authority's online portal for submitting financial reports has been identified. The feasibility, technical requirements, and legal approvals for such an integration have not yet been investigated.
Impact: A direct integration could significantly streamline the tax filing process for property owners but would introduce complexity and dependency on an external government system.
Resolution Needed By: Phase 2 Planning

OI-02: Automated Payment and Rent Collection
Description: Stakeholders have expressed interest in integrating with local Egyptian banking systems or payment gateways (e.g., Fawry, Vodafone Cash) to enable automated rent collection. This was explicitly excluded from the initial scope, but its technical and regulatory feasibility for a future release is an open question.
Impact: This would be a major value-add feature but involves significant complexity regarding financial regulations, security, and partnership agreements.
Resolution Needed By: Phase 2 Feasibility Study

OI-03: Handling of Digital Lease Agreements and E-Signatures
Description: While the system stores lease data, the legal standing and technical implementation of generating a full legal document and obtaining legally binding e-signatures within the app, per Egyptian Law No. 15 of 2004, is an open issue.
Impact: This feature would move the system from a record-keeping tool to a transaction-enabling platform, but requires legal validation and a certified e-signature provider.
Resolution Needed By: Phase 2 / Legal Review

OI-04: Data Retention and Archiving Policy
Description: The specific durations for which different types of data (e.g., active tenant PII, financial records of terminated leases, system audit logs) must be retained to comply with Egyptian law have not been definitively established and codified into system rules.
Impact: Without a clear policy, the system cannot automatically manage data lifecycle, potentially leading to non-compliance or unnecessary data storage.
Resolution Needed By: Development Phase (before launch)

OI-05: Offline-First Functionality Scope
Description: The precise scope of which core functions must be available during extended network outages, and the strategy for conflict resolution upon re-synchronization, requires

further detailed analysis.
Impact: Affects the complexity of the local data storage and synchronization logic. A limited scope reduces development time but may impact usability in areas with poor connectivity.
Resolution Needed By: Architectural Design Phase

OI-06: Finalization of Alert and Notification Thresholds
Description: While a 60-day lease expiration alert is specified, the optimal thresholds for other alerts (e.g., unresolved maintenance requests) and the configuration of push notifications versus in-app alerts need to be finalized based on further user research.
Impact: Impacts user satisfaction and the system's effectiveness in preventing oversights. Incorrect thresholds could lead to alert fatigue or missed issues.
Resolution Needed By: Detailed Design Phase

OI-07: Vendor and Contractor Management
Description: The potential for a future module to manage relationships with maintenance vendors and contractors has been discussed. The requirements and data model implications for such a module are currently undefined.
Impact: This is a logical extension of the maintenance tracking feature but would expand the scope significantly.
Resolution Needed By: Phase 2 Planning

19. Off-the-Shelf Solutions

19a. Ready-Made Products

RMP-01: Evaluation of Existing Property Management Software
Description: An analysis shall be conducted to determine if any existing off-the-shelf (OTS) property management software can be purchased and configured to meet the core requirements specified in this document, particularly for the Egyptian market.
Rationale: To evaluate if a commercial OTS solution could fulfill the business needs more quickly or cost-effectively than custom development.
Fit Criterion:

- A formal evaluation report is produced, comparing at least three leading OTS property management solutions against the critical requirements (e.g., mobile-first design, Egyptian Arabic support, compliance with Egyptian tax reporting).

- The evaluation must conclusively demonstrate why an OTS solution is or is not viable.

19b. Reusable Components

RC-01: Backend Application Framework
Description: The system shall be built using a established, well-supported open-source backend framework (e.g., Django, Spring Boot, Laravel) to accelerate development and ensure maintainability.
Rationale: Using a standard framework provides built-in solutions for common tasks (routing, security, ORM) and access to a large ecosystem of libraries.
Fit Criterion:

- The selected framework is publicly available under an open-source license and has an active community.
- The framework provides native or well-integrated support for core non-functional requirements like security, database abstraction, and API development.

RC-02: Frontend Mobile Framework
Description: The mobile application shall be developed using a modern, cross-platform framework (e.g., React Native, Flutter) or native platform SDKs (Swift for iOS, Kotlin for Android) that are standard and reusable.
Rationale: To leverage pre-built UI components, development tools, and ensure compatibility with app store requirements.
Fit Criterion:

- The chosen technology is a current, industry-standard platform for mobile development.
- The technology provides reusable UI components for standard mobile patterns (navigation, forms, lists).

RC-03: Database Management System
Description: The system shall use a standard, relational database management system (e.g., PostgreSQL, MySQL) for persistent data storage.
Rationale: A standard RDBMS provides reliability, ACID compliance for financial data, and a well-understood query language.
Fit Criterion:

- The selected database is a widely-used, open-source or commercially licensed RDBMS.

- The database supports the transaction volume and data integrity requirements outlined in the Performance and Security sections.

RC-04: Authentication and Authorization Library
Description: The system shall integrate a robust, standards-compliant library or service for handling user authentication (e.g., OAuth 2.0, JWT) and authorization.
Rationale: Security implementations are complex and error-prone; using a battle-tested library reduces risk.
Fit Criterion:

- The chosen library is a well-maintained, open-source project or a reputable commercial identity service.
- The library supports the encryption standards and session management requirements specified in the Security section.

19c. Products That Can Be Copied

PCC-01: User Interface Patterns from Leading Applications
Description: The user interface design shall adopt established, user-friendly interaction patterns from successful mobile applications in adjacent domains (e.g., banking apps for financial data presentation, e-commerce apps for search and filtering).
Rationale: Users are already familiar with these patterns, which reduces the learning curve and increases usability.
Fit Criterion:

- The dashboard design incorporates best practices for data visualization and summary metrics from leading financial and analytics applications.
- The Property Finder search and filter interface follows conventions used by major real estate and e-commerce platforms.

PCC-02: Public Domain Legal Document Templates
Description: For any auto-generated lease agreements or terms of service, the initial templates shall be based on standardized, public domain, or professionally reviewed templates common for Egyptian rental agreements.
Rationale: To provide a legally sound starting point without initial investment in custom legal drafting, while clearly stating the need for professional legal review by the user.
Fit Criterion:

- The lease agreement template is sourced from a reputable Egyptian legal resource or property management association and is reviewed by a legal professional for basic suitability.

- The application includes a clear disclaimer that the template is a starting point and professional legal advice is recommended.

20. New Problems

20a. Effects on the Current Environment

ECE-01: Transition from Manual to Digital Processes
Description: The introduction of the PPMS will disrupt the current manual workflows (spreadsheets, physical files), requiring property owners to change established habits and adopt new digital procedures.
Impact: This may cause initial resistance, a temporary decrease in productivity during the transition period, and a need for user support and training resources.
Solution:

- Provide a simple data import template for initial migration of property/tenant data from spreadsheets.
- Design an intuitive user interface that closely mirrors the mental model of existing manual processes.
- Develop a comprehensive onboarding tutorial and context-sensitive help within the application.

ECE-02: Increased Dependency on Technology and Connectivity
Description: Property owners will become dependent on their mobile devices and a stable internet connection to perform core management tasks that were previously possible with pen and paper.
Impact: Property owners in areas with poor connectivity or those who are less technologically adept may face challenges. System downtime directly halts management activities.
Solution:

- Implement robust offline functionality for critical tasks like recording payments and logging maintenance requests.
- Ensure the application provides clear status indicators for connectivity and sync status.
- Maintain a simple, printable summary report that can be used as a temporary backup during extended outages.

20b. Effects on the Installed Systems

Not applicable. The specification states there is no existing automated system to be replaced or integrated with (Constraint 3b).

20c. Potential User Problems

UPR-01: Data Security and Privacy Concerns
Description: Users may be hesitant to store sensitive financial and tenant data in a digital system due to fears of data breaches or unauthorized access.
Impact: This could be a significant barrier to adoption, especially for users who currently keep all records offline.
Solution:

- Implement and clearly communicate the robust security measures outlined in the Security Requirements (Section 15).
- Provide a transparent privacy policy that explains how data is stored, used, and protected.
- Offer features that give users a sense of control, such as the ability to export their data at any time.

UPR-02: Accuracy and Reliability Anxiety
Description: Users might not trust the system's automated calculations and alerts initially, leading to double-work where they maintain both the digital system and their old manual records.
Impact: Reduces the time-saving benefits of the system and can lead to user frustration.
Solution:

- Ensure all calculations are transparent and allow users to drill down from summaries to individual transactions.
- Provide clear audit trails for financial data.
- Design the system for high reliability and uptime to build user confidence over time.

20d. Limitations in the Anticipated Implementation Environment

LIE-01: Variable Mobile Network Quality in Egypt
Description: The quality and availability of mobile data networks can be inconsistent across different regions of Egypt, potentially affecting the real-time functionality of the application.
Impact: Users in areas with poor connectivity may experience slow performance or inability to use online-only features, leading to a poor user experience.
Solution:

- The application must be designed with a strong offline-first capability, as outlined in performance requirements.
- Optimize data payloads and use efficient synchronization protocols to minimize data usage and improve performance on slow networks.

20e. Environment That May Inhibit the New Product

EIN-01: Low Digital Literacy Among Target User Segment
Description: A portion of the target market (individual property owners, age 30-65) may have limited experience with sophisticated mobile applications beyond basic messaging and social media.
Impact: This could lead to low adoption rates, high support costs, and users failing to utilize the system to its full potential, thus not realizing the intended benefits.
Solution:

- Adhere strictly to the Usability and Humanity Requirements (Section 11), focusing on simplicity, intuitive design, and clear language.
- Provide multiple support channels, including phone support and detailed visual guides.
- Implement progressive disclosure, showing basic functions first and advanced features only when needed.

EIN-02: Competitive and Price-Sensitive Market
Description: The market may contain low-cost or free alternative solutions (simple spreadsheets, basic listing sites) against which the value proposition of a dedicated paid system must be clearly demonstrated.
Impact: Could limit market penetration and require a stronger focus on marketing and demonstrating clear ROI (Return on Investment).
Solution:

- Ensure the product delivers undeniable value on the core goals: time savings, financial clarity, and reduced vacancies.
- Consider a freemium model or a free trial period to lower the barrier to initial adoption.

20f. Follow-Up Problems

FUP-01: Increased Expectation for Feature Expansion
Description: After successful adoption, users will likely request additional complex features that were explicitly excluded from the initial scope (e.g., automated payments,

vendor management, accounting software integration).
Impact: Places ongoing pressure on the development team, potentially leading to "scope creep" in future versions and increasing the long-term complexity and cost of the product.
Solution:

- Manage user expectations clearly from the outset regarding the V1.0 feature set.
- Establish a transparent and structured process for collecting and prioritizing feature requests for future versions.
- Maintain a modular architecture to facilitate the easier addition of new features later.

FUP-02: Data Portability and Vendor Lock-in Concerns
Description: As users accumulate more data within the PPMS, they may become concerned about the difficulty of switching to a different system in the future, leading to reluctance in fully committing.
Impact: May discourage users from fully migrating their historical data or using the system as their "system of record."
Solution:

- Provide robust, easy-to-use data export functionalities for all major data entities (properties, tenants, financial records) in standard, non-proprietary formats (CSV, PDF).
- Clearly state the data export policy in the terms of service.

21. Tasks

21a. Project Planning

T-PP01: Finalize Project Charter and Secure Funding
Description: Formalize the project objectives, scope, high-level timeline, and budget based on this SRS. Present to investors for final approval and release of initial funding.
Deliverable: Signed Project Charter; Approved initial project funding.
Prerequisites: Completed System Requirements Specification (this document).

T-PP02: Assemble Project Team
Description: Recruit and onboard the core project team, including Project Manager, Technical Lead, Mobile Developers (iOS/Android), Backend Developer, UI/UX Designer, and QA Engineer.

Deliverable: Fully staffed project team with defined roles and responsibilities.
Prerequisites: Project Charter approval.

T-PP03: Establish Project Infrastructure and Environments
Description: Set up the necessary development, testing, staging, and production environments. Establish version control, project management, and communication tools.
Deliverable: Fully configured and operational development infrastructure.
Prerequisites: Project team assembled.

T-PP04: Develop Detailed Project Schedule
Description: Create a detailed work breakdown structure (WBS) and project schedule using an Agile methodology, defining sprints, milestones, and release dates.
Deliverable: Detailed project schedule in the chosen project management tool (e.g., Jira, Azure DevOps).
Prerequisites: Finalized SRS; Assembled team.

T-PP05: Conduct Legal and Compliance Review
Description: Engage legal counsel to review the SRS, data model, and planned workflows for compliance with Egyptian data protection, tenancy, and e-signature laws.
Deliverable: Legal compliance sign-off report; List of required changes to ensure compliance.
Prerequisites: Completed SRS.

21b. Planning of the Development Phases

T-DP01: Phase 1 - Foundation & Core MVP
Objective: Develop and release a Minimum Viable Product (MVP) with the most critical functionality for property owners.

- **Sprint 0: Inception & Setup**
    - Finalize technology stack selection.
    - Create initial UI/UX wireframes and mockups for core flows (dashboard, property/tenant/lease management, payment recording).
    - Set up CI/CD pipeline.
- **Sprint 1-n: Core Development**
    - Implement backend data model and core APIs (User, Property, Unit, Tenant, Lease, Payment).
    - Develop mobile app core structure and navigation.

    - Implement user authentication and authorization.
    - Develop Property & Unit Management module (PUC-04).
    - Develop Tenant & Lease Management module (PUC-05, PUC-06).
    - Develop Financial Management - Record Payments (PUC-07).
    - Develop the core Dashboard with expiring lease alerts (PUC-09, PUC-20).
- **Sprint n+1: Polish & Release**
    - Security testing and penetration testing.
    - User Acceptance Testing (UAT) with target user group.
    - Address critical bugs from UAT.
    - Prepare for and submit to Apple App Store and Google Play Store.
      Deliverable: Publicly released V1.0 (MVP) of the PPMS mobile application.

T-DP02: Phase 2 - Enhanced Management & Public Finder
Objective: Expand functionality with maintenance tracking and the public-facing Property Finder to directly address vacancy reduction.

- **Development Sprints:**
    - Develop Maintenance Request module (PUC-08).
    - Develop Public Property Finder module (PUC-01, PUC-02, PUC-03).
    - Enhance Dashboard with maintenance alerts.
    - Implement data export for financial reports (PUC-14).
    - Implement basic data analytics for Data Administrators.
      Deliverable: Released V2.0 with Property Finder and Maintenance Tracking.

T-DP03: Phase 3 - Advanced Features & Scalability
Objective: Address open issues and add advanced features based on user feedback and business strategy.

- **Planning & Development:**

- Feasibility study and implementation for automated payment integration (Addresses OI-02).
- Research and potential implementation of e-signatures for leases (Addresses OI-03).
- Advanced reporting and analytics dashboard.
- Performance and scalability optimizations.
  Deliverable: V3.0 with advanced financial and operational features.

22. Migration to the New Product

22a. Requirements for Migration to the New Product

RM-01: User-Initiated Data Migration
Description: The migration of existing property, unit, tenant, and lease data from manual records (spreadsheets, physical files) into the PPMS shall be performed by the property owner (user). The system shall provide guided tools and templates to facilitate this process.
Rationale: There is no legacy digital system to automate migration from. The variety and inconsistency of manual records make automated migration infeasible. User-initiated entry ensures data accuracy and provides an opportunity for data cleansing.
Fit Criterion:

- The system provides downloadable CSV template files for bulk import of Properties, Units, and Tenants.
- The import function validates data upon upload and provides a clear report of errors (e.g., missing required fields, duplicate addresses) for the user to correct.
- A user can successfully migrate data for a portfolio of 5 properties with 15 units and 10 active leases within 2 hours using the provided tools.

RM-02: Phased User Onboarding
Description: The system shall allow a new property owner to begin using the application with a minimal initial dataset, adding more data over time.
Rationale: To lower the barrier to entry by not requiring a complete historical data migration before realizing value from the system's core functions.
Fit Criterion:

- A user can register, add a single property and unit, and record a payment without being forced to import their entire portfolio first.
- The application provides clear guidance on how and when to add different types of data (e.g., "Start by adding your properties," "Now let's add your current tenants").

RM-03: Historical Data Entry Guidance
Description: The system shall provide clear guidance on what historical data is necessary versus optional for effective use, focusing on current active leases and recent financial records.
Rationale: To prevent users from being overwhelmed by the prospect of entering years of historical data and to focus effort on data that provides immediate value (active management).
Fit Criterion:

- The onboarding process clearly states that only current tenants and active leases are required to start.
- The financial recording module suggests starting with the current month's income and expenses, not past years.

22b. Data That Has to Be Modified or Translated for the New System

DMT-01: Standardization of Property Addresses
Description: Free-form property addresses from various sources (spreadsheets, handwritten notes) must be standardized into the structured format required by the PPMS (e.g., separating street, district, city).
Action: The data import template and manual entry forms will guide users to enter addresses in a structured manner. This translation is the user's responsibility during data entry.

DMT-02: Categorization of Financial Records
Description: Income and expenses logged in simple lists or various spreadsheet columns must be mapped to the standardized categories defined in the PPMS data model (e.g., 'Maintenance', 'Utilities', 'Taxes').
Action: During data entry or import, users will be required to assign each financial transaction to one of the pre-defined system categories.

DMT-03: Definition of Lease Status
Description: The status of lease agreements (e.g., "active", "month-to-month", "expired") from manual records must be translated to the system's defined statuses: "Active", "Expired", or "Terminated". The system will automatically manage the "Expiring" alert status.

Action: Users will set the initial status during lease entry. The system will automatically change "Active" leases to "Expired" after the end date passes.

DMT-04: Consolidation of Fragmented Data
Description: Information about a single entity (e.g., a tenant) that is currently stored across multiple disconnected records (phone contact, email in a different list, lease in a file) must be consolidated into a single, comprehensive digital record within the PPMS.
Action: This is a key manual activity for the user during migration, facilitated by the system's unified data model.

23.Risks

RK-B01: Inaccurate Assumption of User Technical Capability
Description: The project assumes property owners have journeyman-level technological experience (Assumption 2). If a significant portion of the target market has lower technical literacy, they may struggle with the initial setup, data migration, and navigation, leading to frustration and abandonment of the product.
Probability: Medium
Impact: High
Mitigation:

- Conduct usability testing with users who have lower technical confidence during the design phase.
- Enhance in-app guidance, tutorials, and tooltips beyond the initial onboarding.
- Provide readily accessible and patient customer support channels, including phone support.

24. Costs

C-01: Initial Development & Launch Investment
Description: The one-time capital expenditure required to fund the project team, technology infrastructure, and operational activities through to the successful launch of the MVP (Version 1.0) on both app stores.
Rationale: Covers the creation of the product before it generates revenue. This is the

funding being sought from investors.
Cost Components:

- **Personnel Costs:** Salaries and benefits for the project team (Project Manager, Technical Lead, Developers, Designer, QA) for the duration of Phase 1.
- **Software & Tools:** Licenses for development tools, design software, project management, and communication platforms.
- **Infrastructure & Hosting:** Initial setup and first-year costs for cloud hosting, databases, and backend services for development, testing, and production environments.
- **Third-Party Services:** Costs for app store developer accounts, security penetration testing, and legal/compliance review.
- **Contingency:** A buffer (typically 15-20%) for unforeseen expenses during development.

Estimated Budget Range: Pending detailed estimation based on finalized project schedule and team composition.

25. User Documentation and Training

25a. User Documentation Requirements

UDR-01: Context-Sensitive In-App Help
Description: The system shall provide brief, accessible help text and explanations for all major screens and form fields, accessible via a help icon (?), without requiring the user to leave the current context.
Rationale: To provide immediate assistance to users when and where they need it, reducing frustration and support calls.
Fit Criterion:

- 100% of screens containing form fields or complex data have a help icon that displays a relevant explanation when tapped.
- 90% of test users can find an explanation for "security deposit tracking" without leaving the lease management screen.

UDR-02: Online Knowledge Base / User Guide
Description: A comprehensive, searchable online knowledge base shall be provided, covering setup, core workflows, troubleshooting, and FAQs.

Rationale: To provide a self-service resource for users who prefer to find answers themselves or need help outside of support hours.
Fit Criterion:

- The knowledge base contains a step-by-step article with screenshots for each primary use case (e.g., "How to Add Your First Property").
- Users can find an answer to a common question (e.g., "How do I export a tax report?") using the search function within 3 clicks.

UDR-03: Video Tutorial Library
Description: A library of short (under 2-minute) video tutorials shall be created, demonstrating key tasks within the actual application interface.
Rationale: Visual demonstrations are often the easiest way for users to learn a new process.
Fit Criterion:

- Videos are available for the top 10 most critical user tasks (e.g., recording a payment, renewing a lease, responding to an inquiry).
- The video library is accessible from within the mobile application and linked from the knowledge base.

25b. Training Requirements

TR-01: Interactive Onboarding Tutorial
Description: The application shall include an interactive, step-by-step guided tutorial upon first launch, walking the user through the basic setup and core functions.
Rationale: To accelerate time-to-value and ensure users understand the fundamental capabilities of the system from the very beginning.
Fit Criterion:

- 100% of new users are presented with the onboarding tutorial after initial registration.
- Users who complete the tutorial are able to add a test property and tenant without external help.

TR-02: Availability of Live Webinar Training
Description: Regularly scheduled, live online training webinars shall be offered to both new and existing users, covering basic usage and advanced features.
Rationale: To provide structured learning and allow users to ask questions in real-time.
Fit Criterion:

- A "Getting Started" webinar is offered at least twice per month.
- Attendance is available via registration, and recordings are posted to the knowledge base afterward.

TR-03: Support for Training by Property Owner Associations
Description: Provide training materials (slide decks, facilitator guides) to Egyptian property owner associations to enable them to conduct their own training sessions for members.
Rationale: To leverage community networks and provide training in a trusted, local context, potentially increasing adoption.
Fit Criterion:

- A standardized training package is developed and offered to at least three major property owner associations in Cairo and Alexandria within 6 months of launch.

26.Waiting Room

The following items have been identified as valuable but are explicitly excluded from the initial release (V1.0 / MVP) scope. They are documented here for future consideration and prioritization.

WR-01: Automated Rent Collection & Payment Gateway Integration
Description: Integration with Egyptian payment gateways (e.g., Fawry, Vodafone Cash) or bank APIs to enable tenants to pay rent electronically, with automatic recording of payments in the PPMS.
Reason for Postponement: High complexity, regulatory hurdles, security concerns, and partnership requirements. The manual "Record Payment" function provides the core value for V1.

WR-02: Digital Lease Signing (E-Signatures)
Description: Functionality to generate a complete lease agreement document from system data and obtain legally binding e-signatures from owners and tenants within the application.
Reason for Postponement: Requires legal validation under Egyptian Law No. 15 of 2004 and integration with a certified e-signature provider. Manual lease handling is acceptable for MVP.

WR-03: Vendor & Contractor Management Module
Description: A module to maintain a directory of maintenance vendors/contractors, track assignments, request quotes, and record service histories.

Reason for Postponement: Expands scope beyond core property-tenant-lease-finance data model. Can be added as a logical extension after the core system is stable.

WR-04: Advanced Analytics and Forecasting
Description: Predictive analytics for cash flow forecasting, rental price optimization suggestions based on market data, and detailed portfolio performance trends over time.
Reason for Postponement: Requires significant data history to be valuable. Basic financial summaries and alerts are sufficient for initial financial clarity goals.

27. Ideas for Solutions

This section captures potential solution approaches and technologies that could be employed to address the defined requirements. These are ideas for the development team to evaluate, not commitments.

IS-01: Offline-First Data Synchronization
Idea: Use a local database on the mobile device (e.g., SQLite) paired with a conflict-resolution strategy (like timestamp-based "last write wins" for simple records or manual merge for critical data like payments). A background sync service would push/pull changes when the network is available.
Applicable To: Performance Requirements (FTR-01), Operational Environment (EPE-01).

IS-02: Pre-Signed URL for Secure File Upload
Idea: For the public Property Finder inquiry form, use a server-generated pre-signed URL that allows the unauthenticated user to submit their inquiry data directly to a secure endpoint, avoiding the need for a complex public API.
Applicable To: Security Requirements (ACR-01), Product Use Case PUC-03.

IS-03: Cached Public Data for Property Finder
Idea: Serve the public Property Finder module from a static site generator or a highly cached CDN (Content Delivery Network) that is periodically updated (e.g., every 5 minutes) from the main database. This improves performance and security by isolating the public-facing part from the core application.
Applicable To: Performance Requirements (SLR-01), Security Requirements (IMR-01).

IS-04: Configuration-Driven Reporting Engine
Idea: Build the reporting module around a template engine (e.g., JasperReports, a custom JSON-to-PDF/CSV engine). Report structures (fields, filters, layout) are defined in

configuration files, allowing new reports to be added without code changes.
Applicable To: Adaptability Requirements (AR-03), Compliance Requirements (CLR-02).

IS-05: Modular Microservices Architecture
Idea: Structure the backend as a set of loosely coupled microservices (e.g., User Service, Property Service, Payment Service). This aligns with the business modules, improves scalability, and allows independent deployment and scaling of high-load services.
Applicable To: Maintainability Requirements (MR-01), Scalability Requirements (SER-01).

IS-06: Feature Toggle Framework
Idea: Implement a feature toggle system to control the release of new features. This allows features to be deployed to production in a disabled state and activated later, enabling safer rollouts and A/B testing.
Applicable To: Reduced deployment risk, easier management of the Waiting Room features.

IS-07: Reactive Dashboard with WebSockets
Idea: Implement the owner's dashboard using a reactive approach. When backend data changes (e.g., a payment is recorded), a push notification (via WebSockets) is sent to update the dashboard in near real-time for any user viewing it.
Applicable To: Functional Requirement FR-PO3, enhancing user perception of a live system.

IS-08: Bulk Data Import via CSV Template Validation
Idea: Provide users with downloadable CSV templates containing validation rules. The mobile app or a companion web portal would parse and validate the CSV file upon upload, providing a clear error report before committing any data to the database.
Applicable To: Migration Requirements (RM-01), Data Integrity (IR-01).

IS-09: "Data Migration Assistant" Wizard
Idea: Create a guided, step-by-step wizard within the app that helps users migrate their data. It would prompt for one type of data at a time (e.g., "Let's start with your properties"), provide the template, and handle the upload and error correction process interactively.
Applicable To: Migration Requirements (RM-01), Ease of Use (EUR-01).

IS-10: Third-Party Authentication Providers
Idea: In addition to username/password, allow users to authenticate using popular social providers (e.g., Google, Apple) which are widely used in Egypt. This can simplify the sign-up process and improve security by reducing reliance on user-created passwords.
Applicable To: Security Requirements (ACR-02), Ease of Use (EUR-01).