

# Technical Specification and System Proposal: Vendor Payment Tracking System

## 1. Executive Project Overview

The Vendor Payment Tracking System is an enterprise-grade, project-centric solution engineered to bridge the operational gap between centralized administrative oversight and mobile-first field data entry. The primary objective is to provide a unified platform for the meticulous tracking of project-specific vendor invoices, payment schedules, and associated work documentation.

This system is designed to digitize and automate the end-to-end workflow of construction-related financial obligations. By integrating a robust Web Administrative Panel with a high-performance Mobile Application, the solution ensures that every payment request is captured at the source, subjected to hierarchical validation, and processed through a transparent, immutable digital audit trail.

## 2. Core Technology Stack

The architecture leverages a modern, scalable stack optimized for high availability, asynchronous communication, and data persistence.

| Component          | Technology           | Purpose                                              |
|--------------------|----------------------|------------------------------------------------------|
| Web Admin Frontend | HTML, CSS, JS/JQuery | Development of responsive administrative interfaces. |
| Web Framework      | CodeIgniter 4        | Server-side business logic and API orchestration.    |

|                             |                             |                                                                                 |
|-----------------------------|-----------------------------|---------------------------------------------------------------------------------|
| <b>Integration Layer</b>    | RESTful APIs                | Standardized communication bridge between the backend and mobile clients.       |
| <b>Database</b>             | MySQL                       | Relational data persistence for project metadata, user records, and audit logs. |
| <b>Mobile Framework</b>     | Flutter                     | Cross-platform mobile application development for iOS and Android.              |
| <b>Notifications</b>        | Firebase Push Notifications | Asynchronous delivery of real-time workflow alerts and status updates.          |
| <b>Mobile Local Storage</b> | Hive Database               | Lightweight NoSQL persistence for offline operational resilience and caching.   |

### 3. System Architecture and User Roles

The system is bifurcated into two primary interfaces: the Web Admin Panel for governance and the Mobile Application for field operations. User access is managed through a centralized identity framework defined in the Web Admin:

- **Super Admin:** Holds exhaustive control over the environment, including system configuration and the provisioning of all user accounts.
- **Admin/Non-Admin Web Users:** Personnel responsible for back-office management. Access to specific modules is governed by granular "Roles and Permissions" to ensure the principle of least privilege.
- **App Users (Mobile):** This user category is further subdivided into two functional roles:
  - **Project Incharge:** Field personnel responsible for the primary entry of payment records, digital document uploads, and work status updates for their assigned projects.
  - **Approvers:** Personnel assigned to specific hierarchical tiers within a project's approval matrix, responsible for the validation and authorization of requests.

## 4. Web Admin Panel: Functional Specifications

**User & Access Management** This module facilitates the lifecycle management of both web and mobile users. It features a granular authorization engine where "Roles and Permissions" define the functional scope for non-admin users, ensuring secure data segmentation.

**Vendor and Supplier Directory** A master data repository for the maintenance of all supplier and vendor entities. This directory serves as the definitive source for all financial transactions generated within the ecosystem.

**Project Governance** Centralized management of construction projects. In this module, administrators define project parameters and assign a "Project Incharge." It also houses the configuration for project-specific approval hierarchies, mapping authorized app users to their designated validation levels.

**Payment Request Oversight** A comprehensive monitoring dashboard providing real-time visibility into the status of all requests. It displays state logs for each completed validation level and provides direct access to associated digital attachments and evidence.

**Accounting and Payment Processing** This module acts as the final gateway. A payment request only enters the accounting queue once it has successfully cleared the highest defined approval level for its specific project. Upon disbursement confirmation, the final transaction details are persisted for audit and reporting purposes.

## 5. Hierarchical Approval Workflow Engine

The system implements a dynamic "Approval Levels" logic, allowing the workflow to adapt to the unique management requirements of individual projects.

- **Variable Hierarchy:** Approval chains are not static; they are custom-defined during project initialization.
- **Sequential Validation:** Requests must be authorized at lower levels before they are enqueued for higher-level review.
- **Architectural Example:** For the "Rajaji Nagar" project, the system can be configured with a 5-level hierarchical matrix:
  - **Level 1 (Lowest):** Koustubh Chougule
  - **Level 2:** Aparna
  - **Level 3 & 4:** Assigned Personnel
  - **Level 5 (Highest):** Sharan Sir

## 6. Mobile Application: Operational Specifications

The mobile client is optimized for field efficiency and timely hierarchical validation:

1. **Dashboard/Listing:** A project-centric view where the Project Incharge can access cards for all projects under their supervision to view status summaries.
2. **Profile Section:** A dedicated module for users to manage their professional profiles, view assigned roles, and configure application preferences.
3. **Payment Request Entry:** The primary interface for uploading vendor-wise payment records. The Incharge can attach documents and work details, with the ability to perform amendments prior to formal submission.
4. **Asynchronous Notifications:** Powered by Firebase, the system pushes alerts to the relevant Level 1 approver immediately upon submission. Subsequent approvers are notified as the request progresses through the hierarchy.
5. **Offline Persistence:** Utilizing the Hive Database, the app maintains operational capability in low-connectivity environments, ensuring data is captured and queued for synchronization once a connection is re-established.

## 7. The Lifecycle of a Payment Request

The movement of a request from entry to disbursement follows a rigorous, state-controlled workflow:

- **Submission:** The Project Incharge uploads multi-modal data (vendor records and digital attachments) via the mobile interface.
- **Hierarchical Validation:** The request follows a tier-based progression. Each approver evaluates the request and the supporting evidence before promoting it to the next level.
- **Roll-back Mechanism:** If an approver identifies an error, they must provide a mandatory "Remark" for rejection. The system then triggers a roll-back notification to the immediate previous level, allowing for targeted amendments and re-submission without necessitating a full request restart.
- **State Logs:** Every transition, approval, and rejection is recorded in an immutable audit log, complete with user identity and high-precision timestamps.
- **Finalization:** Once the highest-tier approver clears the request, it is enqueued within the "Accounting" module for final processing and disbursement.

## 8. Reporting and Data Analytics

To maintain financial transparency and operational health, the system provides advanced analytical tools:

- **Disbursement Confirmation Logs:** A detailed historical audit of all payments that have successfully completed the hierarchical workflow and been marked as settled.
- **Project-wise Request Analytics:** Comprehensive reports detailing all payment requests categorized by project, enabling stakeholders to identify workflow bottlenecks and manage pending liabilities effectively.