Software Requirements Specification (SRS) for SaaS Application

1\. Introduction

1.1 Purpose

This document specifies the functional and non-functional requirements for a new SaaS applica
The purpose of this application is to streamline and automate key business processes for both
Accounts and Human Resources (HR) departments.

1.2 Scope

The system will be an online, multi-tenant SaaS application with distinct modules for managing finar
and human resource operations. It will include a web-based user interface accessible via a brow
The backend will be powered by a RESTful API.

1.3 Definitions and Acronyms

***SRS:** Software Requirements Specification*
SaaS: Software as a Service
***API:** Application Programming Interface*
Voucher: A document representing a financial transaction.
***Livewire:** A full-stack framework for Laravel that simplifies building dynamic interfaces.*
Jetstream: An application starter kit for Laravel.

2\. Overall Description

2.1 Product Perspective

This is a standalone SaaS application intended to be used by multiple organizations. It will not be
of a larger enterprise system.

2.2 User Characteristics

The system will be used by:

Accounts Department Staff: *Professionals who will manage financial records, create vouchers, generate financial statements. They require an intuitive interface for data entry and reporting.*
HR Department Staff: Administrators who will manage employee data, payroll, and leave. T require tools for efficient record-keeping and payroll processing.
General Employees: *Users who may access the system to view their pay slips and apply for leave.*

2.3 Constraints

Technology Stack: The application must be built using Laravel 12, Livewire 3, Tailwind CSS, Alpine.js.
Backend: *The system must use an existing API for data operations.*
Authentication: Authentication will be handled by Laravel Jetstream.

2.4 Assumptions and Dependencies

The existing API is stable, documented, and fully functional.
All necessary endpoints for the specified features exist within the API.
Users will have stable internet connectivity.
The underlying infrastructure (servers, database) will meet performance and scalability requirements

3\. Specific Requirements

3.1 Functional Requirements for Accounts Department

3.1.1 Voucher Management

REQ-AC-001: *The system shall allow authenticated users to create new vouchers of various ty (Sales, Purchase, Salary, Expense).*
REQ-AC-002: The system shall allow users to edit and update existing vouchers.
REQ-AC-003: *The system shall allow users to post vouchers, finalizing them for accounting records*
REQ-AC-004: The system shall support both Sales and Sales Return vouchers.
REQ-AC-005: *The system shall support both Purchase and Purchase Return vouchers.*
REQ-AC-006: The system shall manage salary vouchers.
REQ-AC-007: *The system shall handle expense vouchers.*
REQ-AC-008: The system shall allow for the creation of vouchers for fixed assets.
REQ-AC-009: *The system shall allow for the creation of depreciation vouchers.*

3.1.2 Financial Management

REQ-AC-010: The system shall manage accounts receivable and accounts payable.
REQ-AC-011: *The system shall handle accounts adjustments.*
REQ-AC-012: The system shall manage ledger accounts for customers and vendors.
REQ-AC-013: *The system shall manage bank and cash accounts.*
REQ-AC-014: The system shall generate an advance report.
REQ-AC-015: *The system shall include a comprehensive financial system module.*

3.1.3 Reporting and Statements

REQ-AC-016: The system shall generate and display a Trial Balance report.
REQ-AC-017: *The system shall generate and display a Balance Sheet.*
REQ-AC-018: The system shall generate and display a Profit and Loss statement.
REQ-AC-019: *The system shall generate and display an Income Statement.*
REQ-AC-020: The system shall generate and display an Outstanding Statement.
REQ-AC-021: *The system shall generate and display Bank Statements.*

3.1.4 Accounting Operations

REQ-AC-022: The system shall manage a Chart of Accounts.
REQ-AC-023: *The system shall manage fixed asset depreciation and registration.*
REQ-AC-024: The system shall handle financial year opening and closing procedures.
REQ-AC-025: *The system shall manage inventory costs.*
REQ-AC-026: The system shall manage tax.

3.2 Functional Requirements for Human Resource Department

3.2.1 Employee Management

REQ-HR-001: *The system shall include an administrative system for managing HR functions.*
REQ-HR-002: The system shall maintain a database of all employees.
REQ-HR-003: *The system shall generate an employee list.*

3.2.2 Payroll and Compensation

REQ-HR-004: The system shall include a payroll system for calculating employee salaries.
***REQ-HR-005:** The system shall manage employee increments.*
REQ-HR-006: The system shall manage employee allowances and deductions.
***REQ-HR-007:** The system shall generate pay slips for employees.*
REQ-HR-008: The system shall handle withholding tax.
***REQ-HR-009:** The system shall manage employee loans.*
REQ-HR-010: The system shall include an advance salary system.

3.2.3 Leave Management

***REQ-HR-011:** The system shall include a leave system for tracking and managing employee le requests and balances.*

4\. Non-functional Requirements

Performance: The application's pages and data operations must load within a reasonable time (under 3 seconds).
***Security:** The system shall be secure against common web vulnerabilities. It must ensure that us can only access data relevant to their role and permissions.*
Usability: The user interface must be intuitive and easy to navigate, minimizing the learning curve new users.
***Scalability:** The system must be able to handle an increasing number of users and data witho significant degradation in performance.*
Maintainability: The codebase must be well-structured and documented to facilitate future upd and maintenance.
* **Reliability:** The system shall be available 99.9% of the time, with minimal downtime.

5\. User Interaction and Workflows

This section outlines the key user journeys and the screens and forms associated with the functional requirements.

5.1 Accounts Department Workflows

5.1.1 Voucher Creation and Posting

This workflow describes how an Accounts user creates and finalizes a new financial voucher.

1. **Dashboard:** The user logs in and is presented with a dashboard showing key financial metrics.
2. **Navigate to Vouchers:** The user clicks on a "Vouchers" menu item to view a list of exis vouchers.
3. **Create New Voucher:** The user clicks a "New Voucher" button, which opens a dedicated form.
4. **Voucher Creation Form:** The user fills out the form, selecting a voucher type (e.g., Sa Purchase), entering transaction details (date, amount), and specifying ledger accounts.
5. **Save as Draft:** The user can save the voucher as a draft for later editing.
6. **Post Voucher:** Once all details are correct, the user clicks "Post Voucher" which sends a reques the API.
7. **Confirmation:** The system displays a confirmation message, and the voucher's status is update "Posted."

5.2 Human Resources Department Workflows

5.2.1 Employee Management

This workflow describes the process of adding, editing, and viewing employee records.

1. **Dashboard:** The HR user logs in and views their dashboard with quick links to HR tasks.
2. **Navigate to Employees:** The user clicks on an "Employees" menu item to see a searchable sortable list of all employees.
3. **Add New Employee:** The user clicks a "New Employee" button, which opens the employee crea form.
4. **Employee Profile Form:** The user fills out the form with the employee's personal details, cor information, job role, salary, and other relevant data.
5. **Save Employee:** Upon clicking "Save," the system sends the data to the API and adds the employee to the database.
6. **View/Edit Profile:** From the employee list, the user can click on an employee's name to view or their full profile.

5.2.2 Leave Management

This workflow outlines how an employee requests leave.

1. **User Dashboard:** An employee logs in and navigates to their dashboard, where they can see t leave balances.
2. **Request Leave:** The employee clicks a "Request Leave" button.
3. **Leave Request Form:** The user selects the type of leave, the start and end dates, and provide reason for the request.
4. **Submit Request:** The user submits the request, which is then sent to a manager or HR for appro
5. **Status Tracking:** The user can view the status of their leave request (Pending, Approved, Rejec from their dashboard.

6\. Appendices

6.1 Source Document

The functional requirements were derived from a hand-written list provided by the user.