

# Project Proposal: Inventory Management API Service

## 1. Overview

This project delivers a RESTful API service to centrally manage products, warehouses, and inventory levels. It enables clients to perform create, read, update, and delete operations on core resources while enforcing secure access and data integrity. The API will serve as the backbone for any front-end application or third-party integration requiring real-time inventory data.

## 2. Purpose and Scope

The primary goal is to provide a consistent, scalable interface for:

- Tracking stock levels per warehouse and product
- Automating inventory adjustments based on user actions
- Securing data through token-based authentication
- Delivering clear error messages and validation feedback

This service will cover user management, warehouse and product catalogs, and all inventory transactions.

## 3. Functional Requirements

- User Registration and Authentication
  - Endpoint for new user signup
  - Token-based login and logout
- Warehouse Management
  - CRUD operations for warehouse entities
  - Retrieve all warehouses or a single warehouse by ID
- Product Management
  - CRUD operations for product entities
  - Search and filter products by name or category
- Inventory Management
  - CRUD operations for inventory records
  - Associate inventory entries with warehouses and products
  - Bulk import/export of inventory data
- Reporting and Queries
  - Filter inventory by warehouse, product, or quantity range
  - Paginate large result sets
  - Endpoint to view audit logs of inventory changes

## 4. Non-Functional Requirements

- Data Validation
  - Enforce request payload rules with Form Requests

- Security
    - Protect routes via middleware (e.g., Laravel Sanctum)
  - Documentation
    - Postman collection and OpenAPI/Swagger spec
  - Testing
    - PHPUnit feature tests for all endpoints
    - Automated Postman tests in CI pipeline
  - Performance
- 
- Implement pagination and query optimization
  - Optionally add caching for high-read endpoints

## 5. Deliverables

- A GitHub repository with Laravel project scaffolding
- Well-defined database migrations and Eloquent models
- Controllers, routes, and Form Requests for each feature
- Postman collection and API documentation
- Test suite covering registration, authentication, and CRUD flows
- Readme file detailing setup steps, environment variables, and usage

## 6. Future Extensions

- Supplier and purchase-order management
- Role-based access control and permissions
- Audit logging service for compliance reporting
- Webhooks or event broadcasts for inventory updates
- Front-end demo application consuming the API