



Presentation of SIP internship

# Stock Management Pro Complete Inventory Management System

Presented by:

Amhidi Hamza

Supervisor :

M.Hmada Yahya

Année universitaire: 2024/2025

# Acknowledgments

---

I would like to express my sincere gratitude to my supervisor, M. Hamada Yahya, for his invaluable guidance, trust, and expertise throughout this project.

His mentorship in software development and web technologies has been crucial to the successful completion of this work.

# Plan



PROJET  
CONTEXT

CORE FEATURES  
&  
PROJECT TIMELINE

STRUCTURS

INTRODUCTION

PROBLEM  
STATEMENT

MODULES

CONCLUSION



# Introduction

- Stock Manager Pro is a comprehensive, web-based inventory management system.
- Built with Flask (Python) for the backend and HTML/CSS/JavaScript for the frontend.
- It simplifies product tracking, delivery control, and employee management.
- Key features include real-time analytics, automated ID generation, multi-format exports, and a scalable, secure architecture.

# Project Context

- In today's competitive market, efficient inventory management is vital for profitability and customer satisfaction.
- Businesses struggle with tracking stock, managing deliveries, and optimizing resources.
- Stock Manager Pro provides a centralized, automated solution to these challenges.

# Problem Statement :

Satisfaction  
of needs



Time-consuming  
manual management of  
products and  
employees.

Absence of integrated analysis  
and reporting tools.

Difficulty in real-time  
stock tracking.

Lack of visibility on  
delivery  
performance

# Key Achievements

Developed a comprehensive, responsive web application with a secure authentication



Provided advanced analytics and reporting tools.

Centralized employee and delivery management.

Ensured real-time data visualization and multi-format exports.

# Business Value



40

40% reduction in manual tracking efforts.



Improved delivery performance tracking.



Comprehensive reporting for informed decision-making.



Scalable architecture to support business growth.

# System Requirements

**Client:** Modern browser, JavaScript enabled.



**Server:** Python 3.8+, Flask 2.0+, 512MB RAM, 1GB storage.

**Hardware:** Client (2GB RAM), Server (2GB RAM, 2-core CPU).

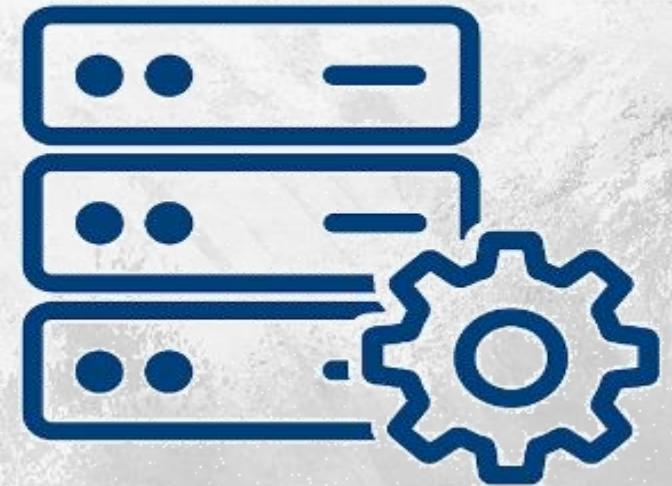
**Dependencies:**  
SQL, SQLAlchemy,  
Chart.js, SheetJS, jsPDF.

# Technical Stack



## Frontend :

- HTML5, CSS3, Vanilla js
- Chart.js
- SheetJS, jsPDF.



## Backend :

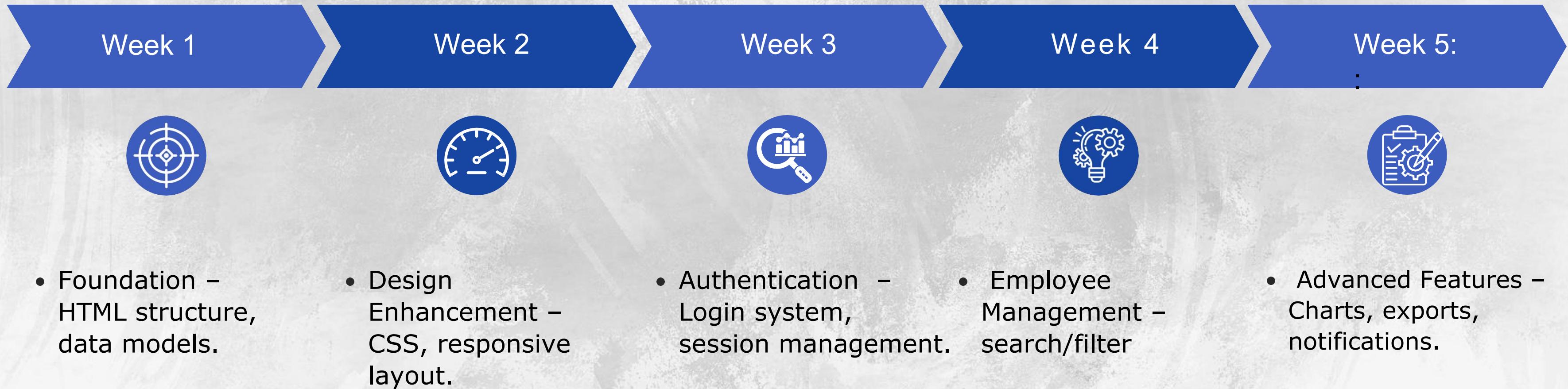
Flask (Python), SQL, Flask-Login, Flask-CORS.



## Tools :

- Git, VS Code, Chrome
- DevTools, Postman.

# Project Timeline



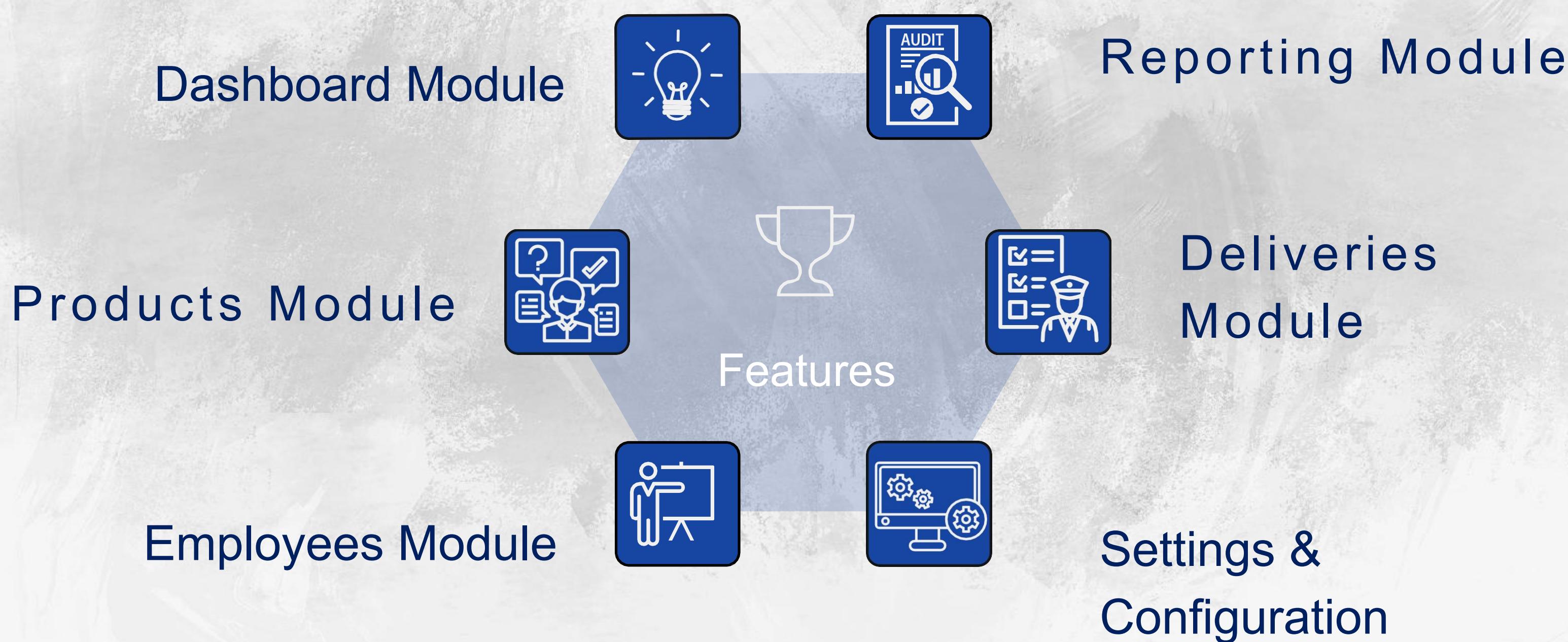
# Project Timeline

Week 6

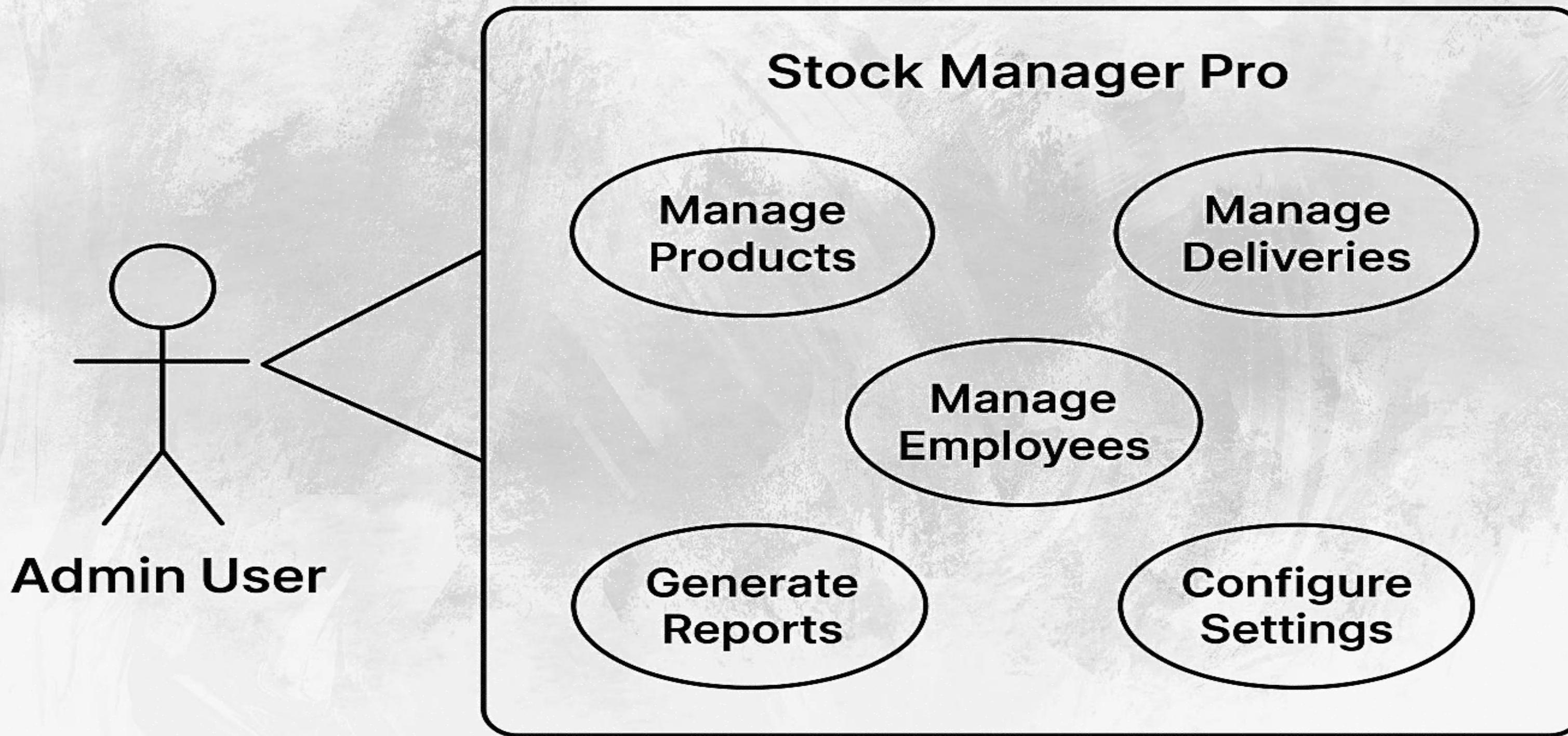


- Backend Integration – Flask API, database, deployment.

# Core Features Overview



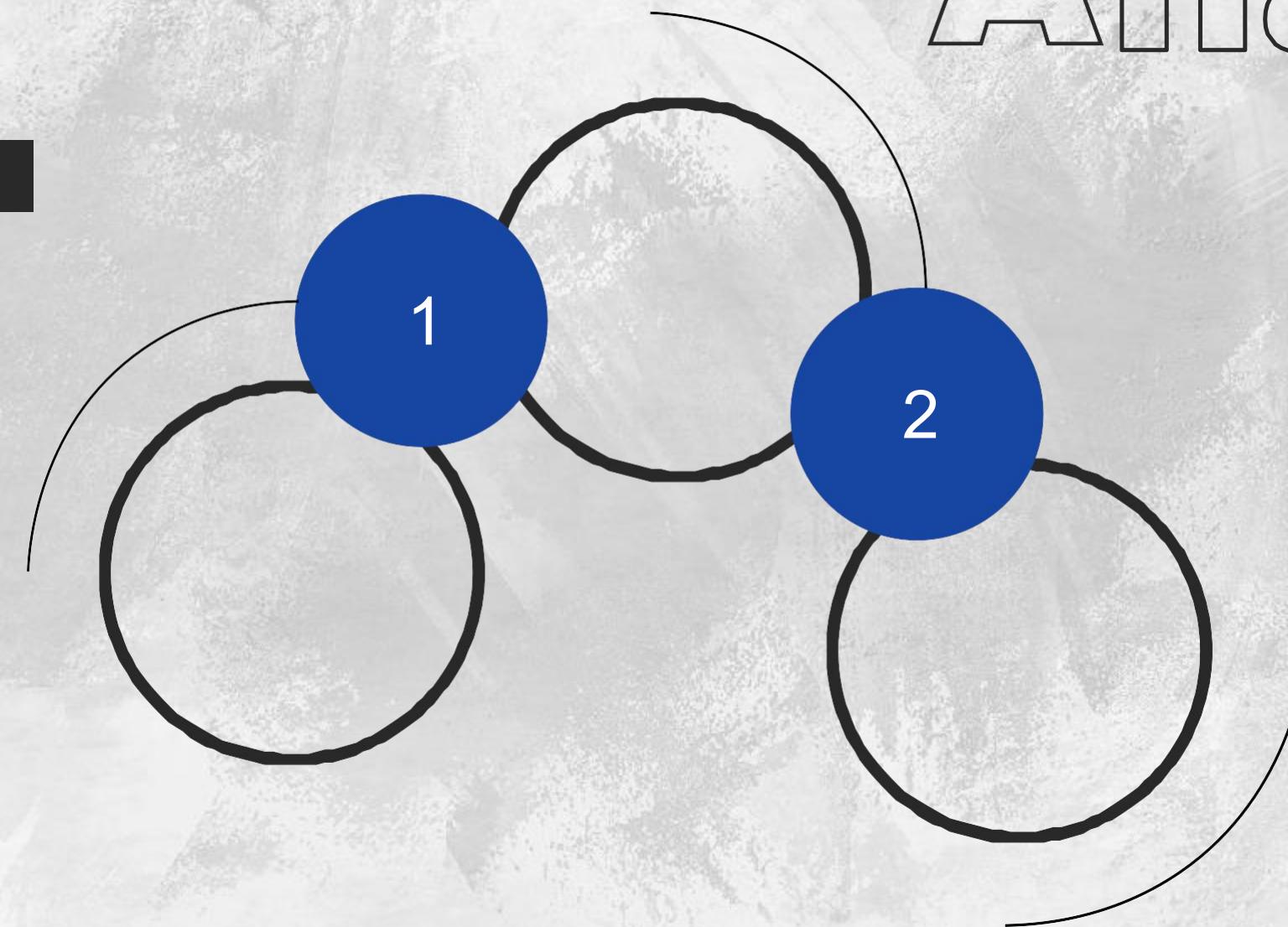
# Use case diagram



# Modules Analyse

## Dashboard

- Summary cards recent
- Activity
- stock overview
- Performance metrics



## Products

- Automatic ID generation-CRUD
- Validation input
- Search/filter
- Real-time updates.

# Modules Analyse

## Employees

- Automatic ID generation - CRUD
- Professional and contact info
- HR export features

3

4

## Deliveries

- Automatic ID generation - CRUD
- stock integration
- status filtering
- real-time updates

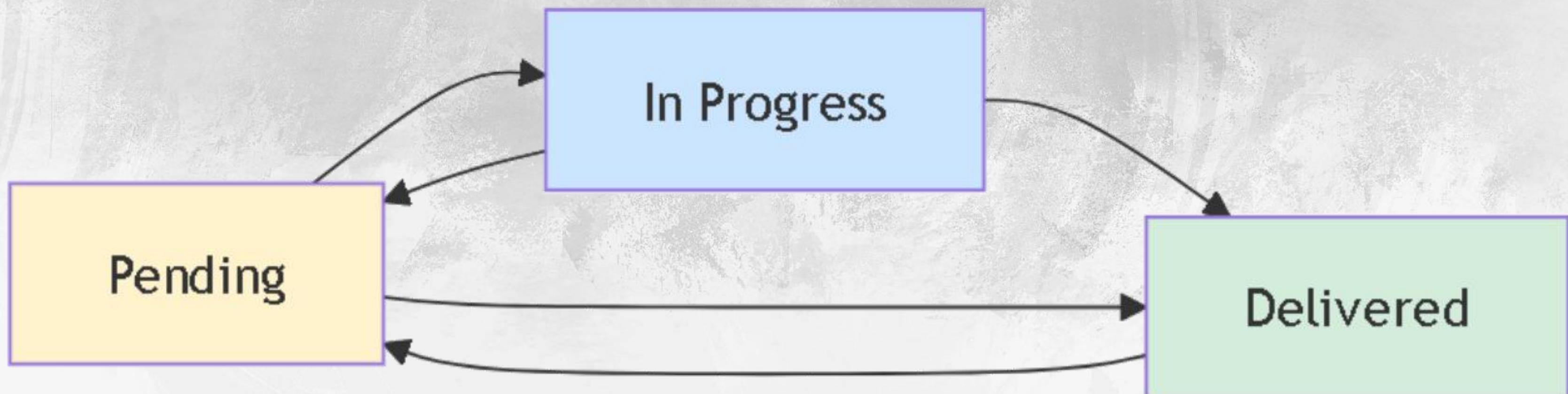
# Real-time Status Updates :

The application automatically calculates delivery statuses based on current dates:

**Pending**: Current date before delivery date

**In Progress**: Current date between delivery and reception dates

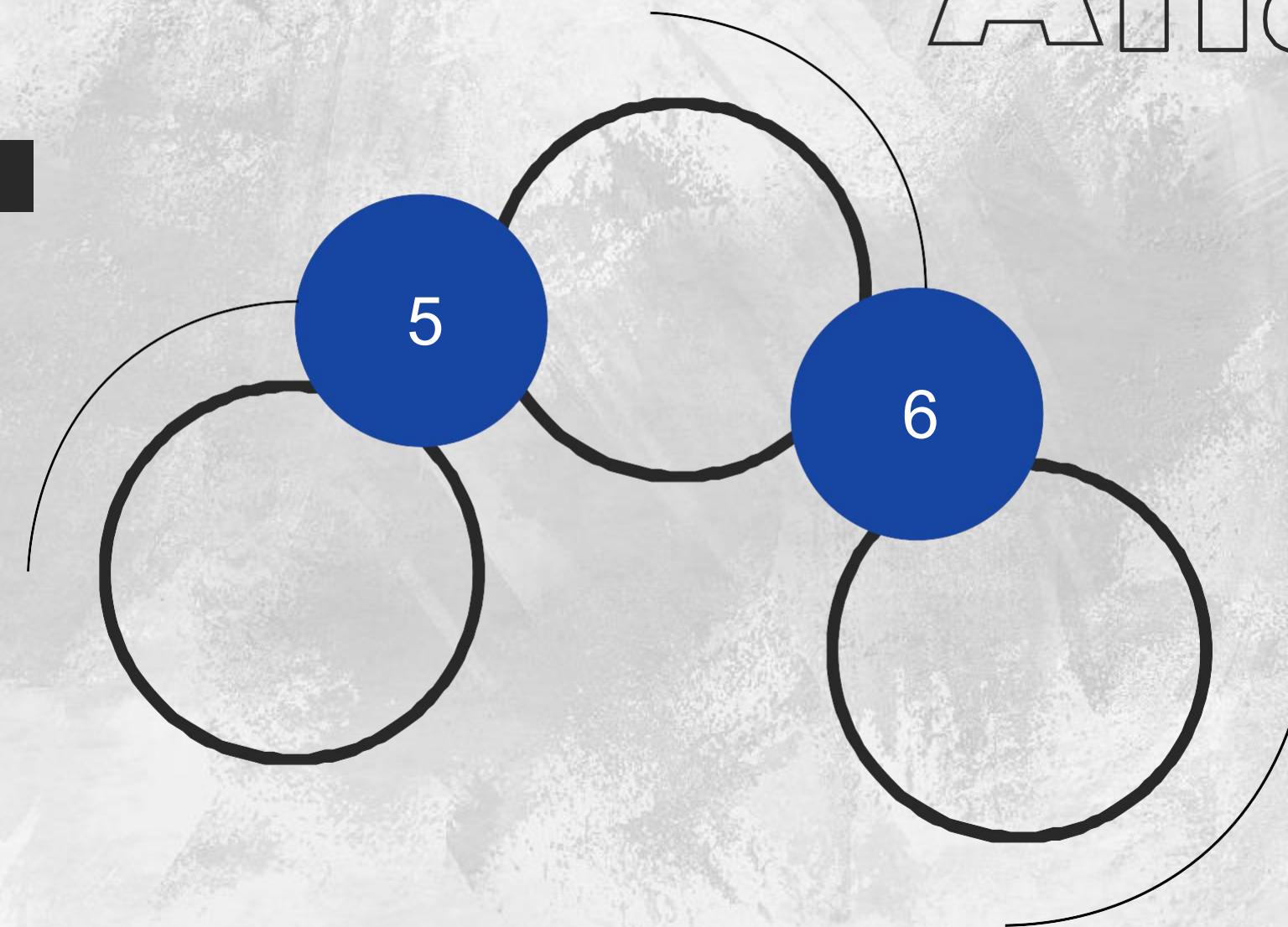
**Delivered**: Current date after reception date



# Modules Analyse

## Reporting

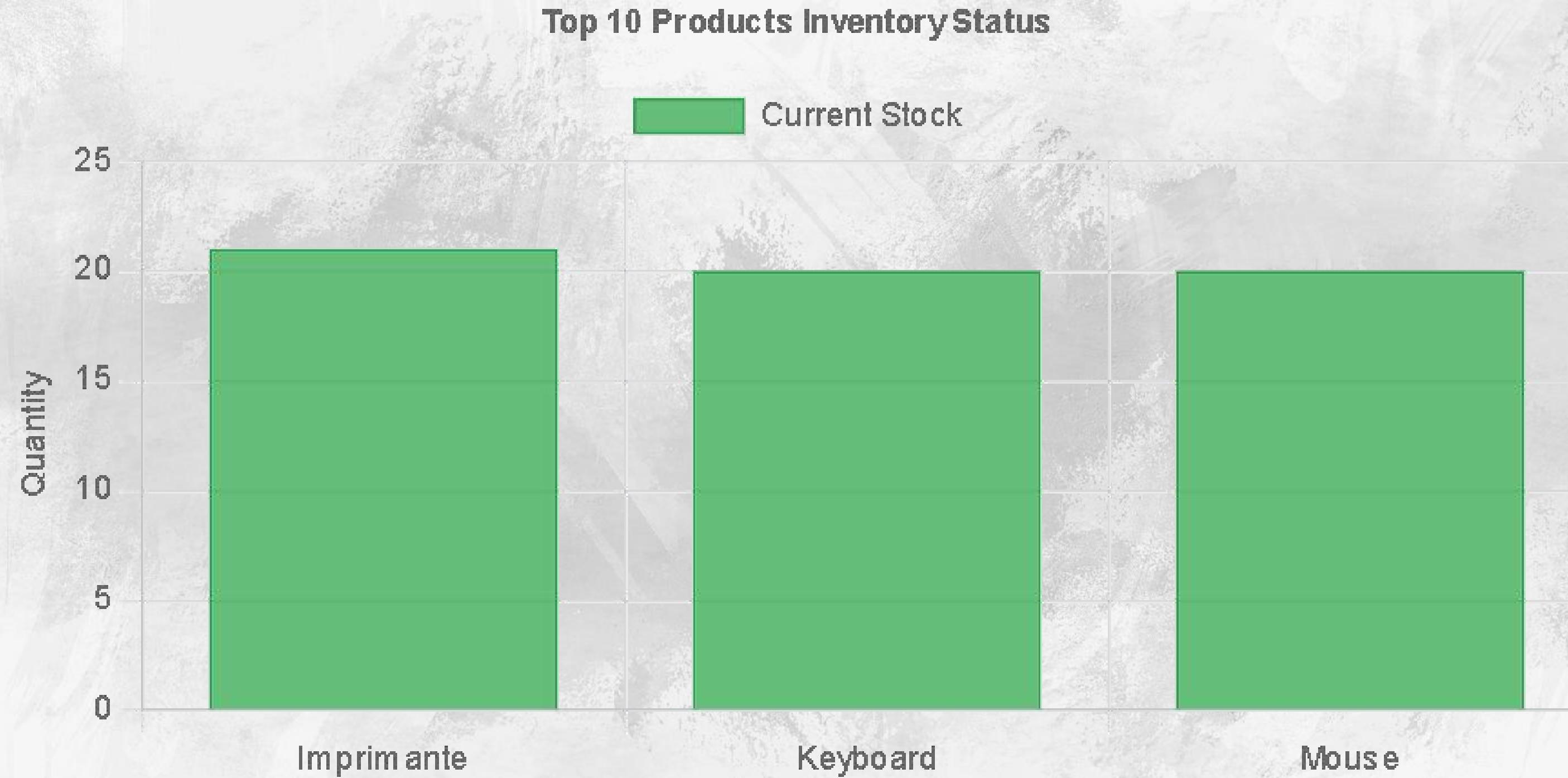
- Interactive charts
- lazy loading
- multi-format exports (PDF, Excel, CSV).



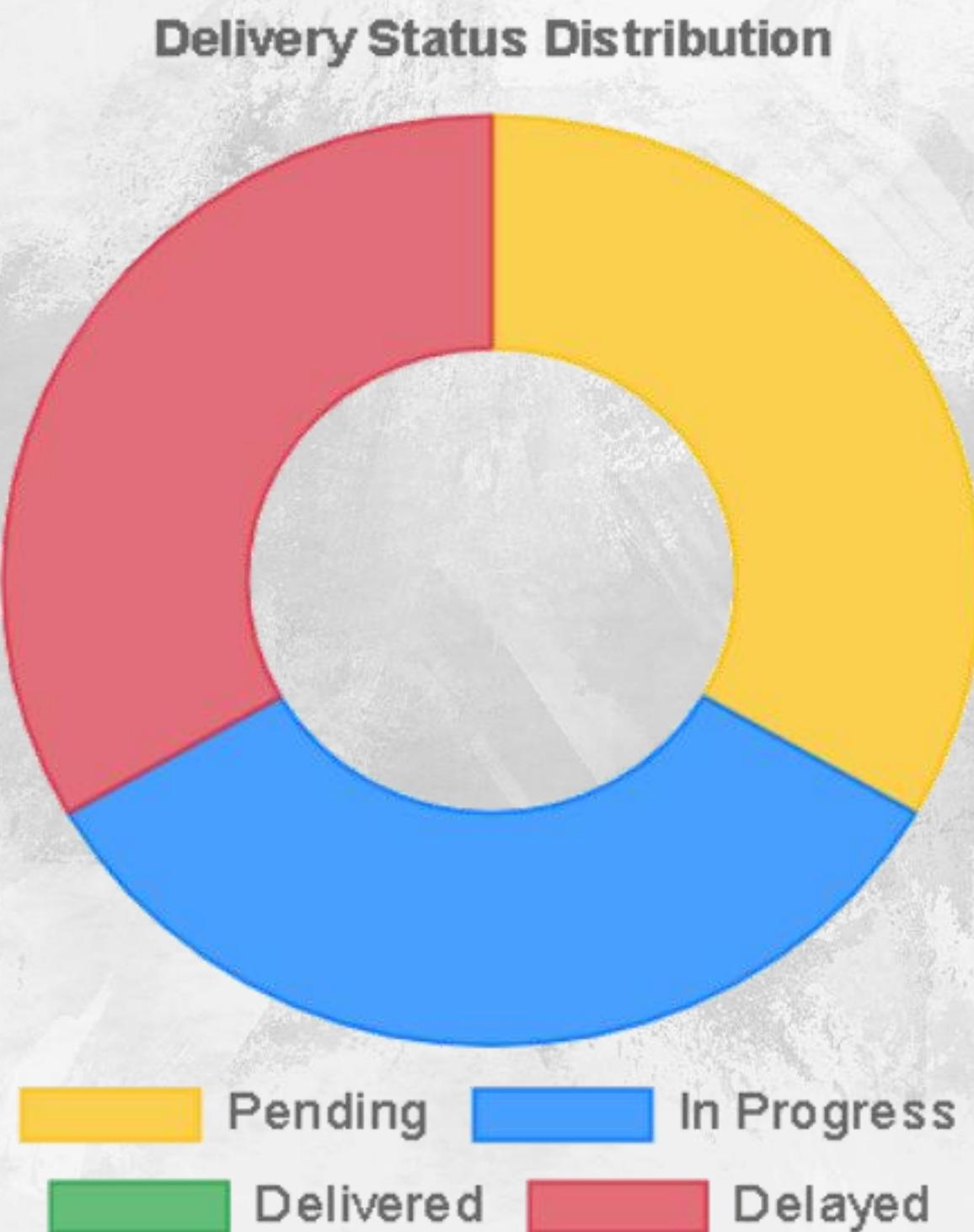
## Settings

- Language
- notifications
- change date/time format
- security /backups

# Charts : Inventory Chart :



# Charts : Delivery Chart :



# Modules Analyse

## Authentication & Security

- Secure login system with session management.
- Route protection and user role management (extensible).

7

# Recap :

## STOCKMASTER PROFESSIONAL

### AUTH MODULE

- Login
- Session
- Security

### PRODUCT MODULE

- CRUD
- Search
- Export

### DELIVERY MODULE

- Create
- Track
- Status

### EMPLOYEE MODULE

- Manage
- Assign
- Export

### REPORTING MODULE

- Analytics
- Charts
- Export

### SETTINGS MODULE

- Theme
- Config
- Backup

# Frontend Architecture

01

**Mobile-first, responsive design.**

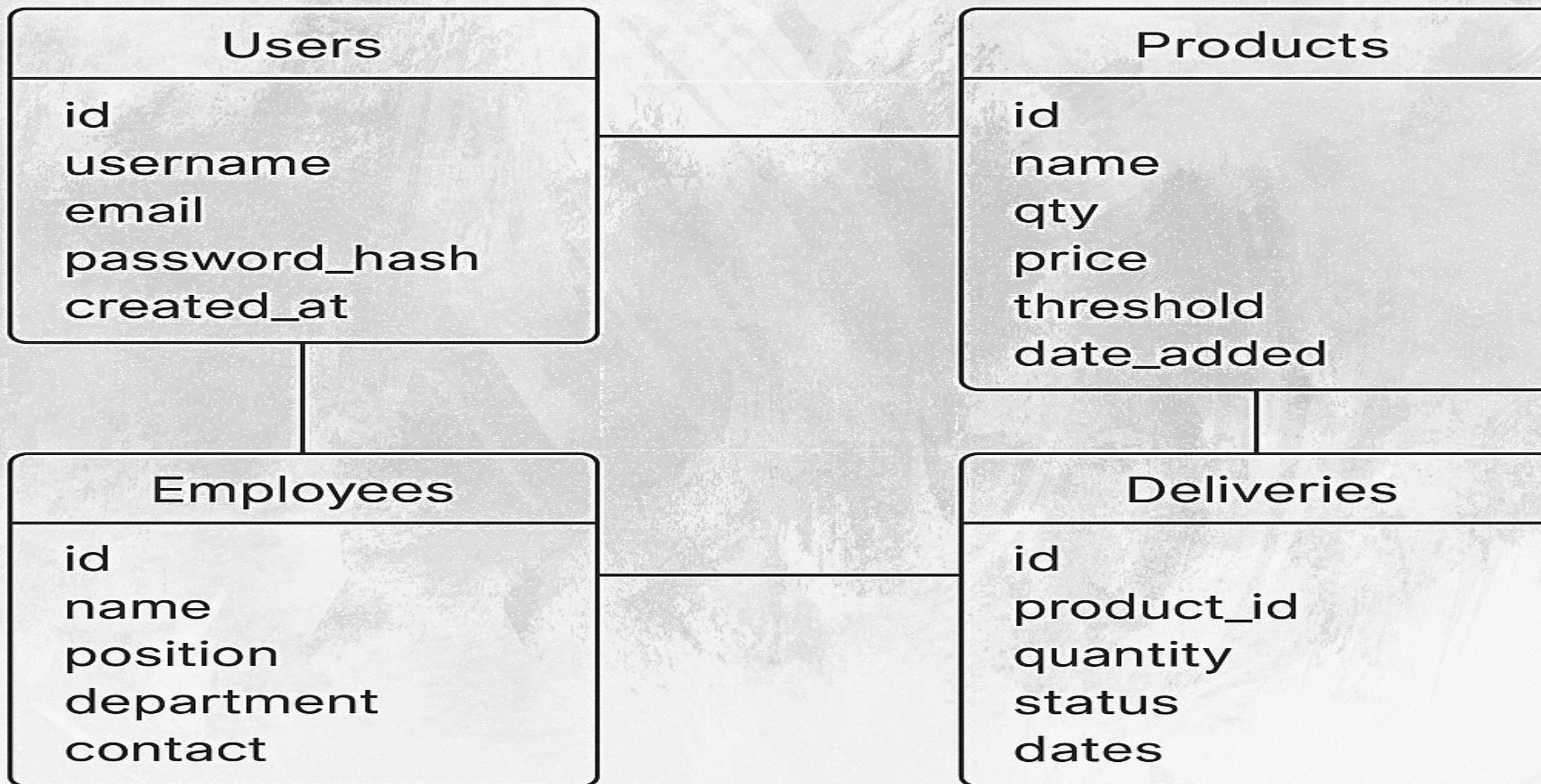
02

**CSS variables for theme support  
(light/dark mode).**

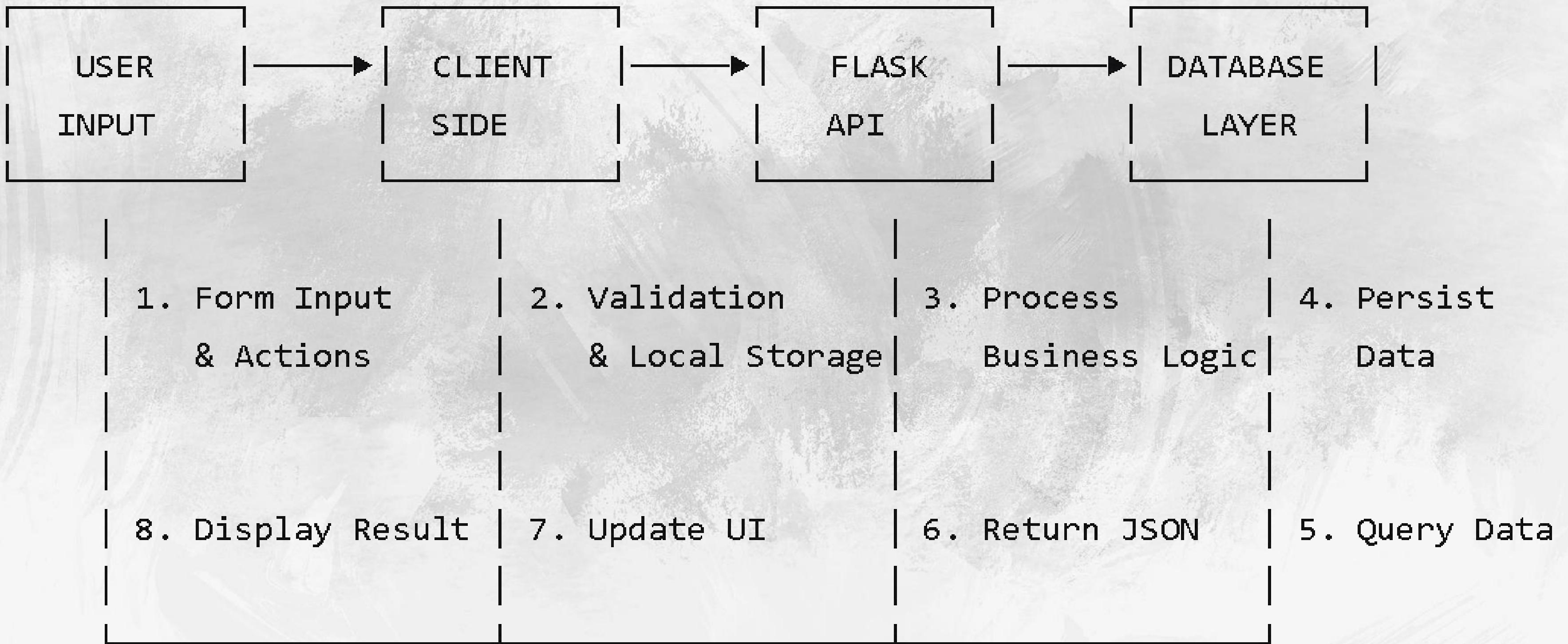
03

**Touch-friendly components and optimized  
tables for mobile.**

# Backend Structure :Database Models:



# Backend Structure :Data Flow Diagram :



# Backend Structure : RESTful Endpoints :

- Products: **GET/POST/PUT/DELETE /api/products**
- Employees: **GET/POST /api/employees**
- Deliveries: **GET/POST /api/deliveries**
- Auth: **POST /api/login, /api/logout, GET /api/check-auth**

# Innovation Highlights :

- Intelligent status management (auto-calculated delivery status).
- Comprehensive multi-format export system.
- Advanced UX with persistent themes and responsive design.
- Scalable architecture with modular backend and reusable frontend

# All planned objectives were successfully achieved:



**Responsive UI**



**Real-time monitoring**



**Delivery tracking**



**Employee management**



**Multi-format reports**



**Security & scalability**



**Performance targets met**



# Summary :



Real-time inventory tracking with automated low-stock alerts.



Centralized management for products, employees, and deliveries.



Multi-format reporting and data export (PDF, Excel, CSV).



Automated ID generation and status calculation for all entities.



Secure, responsive web interface with a scalable Flask backend.



Thank U

For u attention