

Infosys Springboard Virtual Internship 6.0

Project Completion Report

1. Internship Details

- **Internship Program:** Infosys Springboard Virtual Internship 6.0
- **Name:** Christina J
- **Team:** Team 2
- **Internship Duration:** 8 Weeks
- **Start Date:** 26-NOV-2025

2. Student Details:

S. No	Name
1	Christina J

3. Project Title

Inventra – Intelligent Inventory Management System

4. Project Objective

- **Enable Intelligent Inventory Tracking** – Accurately monitor product quantities, stock movement, and availability in real time using a centralized database system.
- **Provide Efficient Product & Stock Management** – Support seamless addition, update, and deletion of product and inventory records to maintain accurate stock information.
- **Implement Automated Inventory Alerts** – Generate timely notifications for low-stock situations to prevent shortages and ensure uninterrupted operations.
- **Support Data-Driven Inventory Decisions** – Deliver clear insights through reports and analytics such as stock summaries, usage trends, and inventory status reports.
- **Ensure Secure and Scalable System Operation** – Protect user access

through authentication mechanisms while maintaining a modular architecture that supports future scalability and enhancements.

5. Project Description

Inventra – Intelligent Inventory Management System is a web-based application developed to automate inventory operations and overcome the limitations of traditional manual or semi-automated inventory systems. The system provides a centralized platform to manage products, monitor stock levels, and track inventory transactions in real time.

The application allows authenticated users to perform product management, stock-in and stock-out operations, and automatically records every inventory transaction to ensure accuracy and transparency. An automated alert mechanism is implemented to notify users when inventory levels fall below a predefined threshold, helping prevent stock shortages.

Inventra follows a modular and layered architecture using Spring Boot and REST APIs, ensuring secure data handling, scalability, and ease of maintenance. By integrating inventory tracking, alerts, and reporting features into a single system, the project provides an efficient, reliable, and user-friendly solution for effective inventory management.

6. Technology Stack Used

Backend: Java, Spring Boot, REST APIs

Security: JWT, BCrypt, Role-Based Access Control

Frontend: HTML, CSS, JavaScript

Database: MySQL

Tools: IntelliJ IDEA, Maven

7. Modules Implemented

Module 1: Authentication

- Secure login and registration
- JWT-based authentication
- Role-based access control

Module 2: Inventory Management

- Product management
- Stock-in and stock-out operations
- Duplicate product prevention

Module 3: Inventory Alerts

- Low-stock detection
- Automatic alert generation

Module 4: Transaction Management

- Recording stock movements
- Maintaining transaction history
- Supporting auditing and reporting

Module 5: Reports & Analytics

- Shows current stock details.
- Tracks inventory usage.
- Helps in inventory planning.

8.Timeline Overview

Week	Activities Planned	Activities Completed
Week 1	Project kickoff, team introduction, understanding inventory management challenges, and defining problem statement and objectives.	Conducted project initiation meeting, finalized project scope and objectives, and studied existing inventory management systems and workflows.
Week 2	Requirement analysis and system design planning, including module identification and technology stack finalization.	Identified core modules (Authentication, Inventory Management, Alerts, Reports), finalized technology stack, and prepared system architecture and database design plan.
Week 3	Database schema design and authentication module planning.	Designed MySQL database schema, created entity relationships, and implemented JWT-based authentication with secure user login and role management.
Week 4	Development of Product and Inventory Management modules with backend APIs.	Implemented CRUD operations for products and inventory using Spring Boot REST APIs and integrated them with the MySQL database.
Week 5	Inventory alert mechanism and business logic implementation.	Developed stock threshold monitoring logic and implemented automated low-stock alert functionality.

Week 6	Reports and analytics module development and frontend integration planning.	Implemented inventory summary and product-wise reports and integrated backend APIs with frontend interfaces using HTML, CSS, and JavaScript.
Week 7	Frontend development, system integration, and testing.	Completed frontend screens, integrated all modules, and performed unit and integration testing for APIs, database operations, and alerts.
Week 8	Final testing, documentation, presentation preparation, and project demonstration.	Completed system documentation, prepared presentation slides, and delivered final demo showcasing end-to-end workflow from login to inventory tracking, alerts, and reports.

9.Key Miles

Milestone	Description	Date Achieved
Project Kickoff	Team formation, problem understanding, requirement analysis, and system architecture planning	26-NOV-25
Prototype / First Draft	Identification of core modules, technology stack finalization, and database schema design	04-DEC-25
Mid-Term Review	Implementation of authentication, product, and inventory management modules	19-DEC-25
Final Submission	Development of inventory alert mechanism and basic reporting features	08-JAN-26
Presentation	Frontend integration, testing, documentation, presentation, and final project demonstration	23-JAN-26

10. Project Execution Details

The Inventra – Intelligent Inventory Management System was developed over an 8-week internship period using a structured and modular approach. The project started with requirement analysis and system design, where key issues such as inaccurate stock handling, lack of transaction tracking, and security concerns were identified. Based on these requirements, a layered architecture consisting of Controller, Service, and Repository layers was designed.

The Authentication module was implemented first using Spring Boot with JWT-based authentication and role-based access control to ensure secure access. Passwords were encrypted using BCrypt for improved security. Next, the Inventory Management module was developed to manage products and perform stock-in and stock-out operations with proper validations to prevent duplicate entries and negative stock values.

The Transaction Management module was integrated to record every inventory operation automatically. Transaction handling was implemented using Spring's @Transactional mechanism to ensure data consistency and rollback in case of failures. The Inventory Alerts module was then added to monitor stock levels and generate alerts whenever inventory fell below the minimum threshold.

Finally, the Reports & Analytics module was implemented by generating reports from existing inventory and transaction data using basic filters. The complete system was tested for functionality and security, followed by documentation and final submission.

11. Snapshots / Screenshots

The screenshot shows a web browser window for the 'Admin Inventory Dashboard' at the URL 'localhost:8081/inventory.html'. The dashboard has a red header bar with 'Logout' and a blue navigation bar with tabs: Category, Supplier, Product, Stock, Alerts, and Transactions. The main content area is titled 'Add Category' and contains fields for 'Category Name' (with value 'I') and 'Description'. Below these is a button labeled 'Add Category'. A table lists four categories:

ID	Name	Description	Action
1	Electronics	Electronic devices and accessories	Delete
2	Office Supplies	General office supplies and stationery	Delete
3	Furniture	Office and workspace furniture	Delete
4	Electronics	Electronic items	Delete

localhost:8081/staff.html

Staff Inventory Dashboard

Logout

Products Stock IN / OUT Alerts Transactions

Transaction History

ID	Type	Product	Qty	User	Date
1	STOCK_OUT	Dell Laptop	100	Unknown	2026-01-21T17:10:01.723762
2	STOCK_IN	Dell Laptop	10	Unknown	2026-01-21T17:26:27.978025
3	STOCK_OUT	Dell Laptop	10	Unknown	2026-01-21T17:26:37.959309

localhost:8081/login.html

Inventory Management System

Forgot Password

krishnamoorthy2004911@gmail.com

Send OTP

[Back to Login](#)

The screenshot shows a web browser window with the URL `localhost:8081/login.html` in the address bar. The title of the page is "Inventory Management System". A central modal dialog box is titled "Sign In". It contains two input fields: the first is labeled "staff" and the second is labeled "..... I". Below these fields is a "Sign In" button. At the bottom of the modal, there are links for "Create new account" and "Forgot Password?". The background of the page is light gray.

Inventory Management System

Sign In

staff

..... I

Sign In

[Create new account](#)

[Forgot Password?](#)

The screenshot shows a web browser window with the URL `localhost:8081/login.html` in the address bar. The title of the page is "Inventory Management System". A central modal dialog box is titled "Verify OTP". It contains one input field with the value "851922" and a "Verify OTP" button below it. The background of the page is light gray.

Inventory Management System

Verify OTP

851922

Verify OTP

12. Key Learnings & Skills Acquired

Through the development of the **Inventra – Intelligent Inventory Management System**, the project team gained valuable technical and professional skills. This internship provided hands-on exposure to real-world software development practices and strengthened both individual and collaborative abilities.

Firstly, the team gained a strong understanding of **backend application development** using **Java and Spring Boot**. We learned to design and implement RESTful APIs, handle business logic in the service layer, and manage data access using repository patterns. This helped us understand how enterprise-level applications are structured and maintained.

Secondly, the project enhanced our knowledge of **database design and management** using **MySQL**. We learned to design normalized database schemas, define relationships between entities, and perform CRUD operations while ensuring data integrity and consistency. Handling inventory transactions improved our understanding of real-time data updates.

We also acquired practical experience in implementing **secure authentication and authorization mechanisms**. By using **JWT-based authentication, BCrypt password encryption, and role-based access control**, we understood how to protect applications from unauthorized access and manage user roles securely.

Additionally, the project improved our **frontend development skills** through the use of **HTML, CSS, and JavaScript**. We learned how to integrate frontend components with backend APIs and display dynamic data effectively in a user-friendly interface.

From a professional perspective, the internship strengthened our **teamwork, communication, and problem-solving skills**. Regular discussions, task division, testing, and documentation helped us learn collaborative development practices. Overall, this project significantly improved our confidence in building secure, scalable, and real-world software applications.

13. Challenges Faced

Challenge 1: Handling Concurrent Inventory Updates

Problem:

Multiple users performing stock-in and stock-out operations at the same time could lead to inconsistent inventory values and data conflicts.

Solution:

Transactional management was implemented using Spring Boot's `@Transactional` mechanism to ensure atomic database operations and prevent race conditions during concurrent updates.

Challenge 2: Performance and Response Time

Problem:

As the number of products and transactions increased, system response time and performance became a concern.

Solution:

Optimized database queries, indexed frequently accessed columns, and structured REST APIs efficiently to improve performance and reduce response time.

Challenge 3: Error Handling and Data Validation

Problem:

Invalid inputs such as negative stock values or incomplete product details could affect system reliability.

Solution:

Implemented input validation at both frontend and backend levels, along with proper exception handling to ensure system stability and meaningful error messages.

Challenge 4: Testing and Debugging Across Modules

Problem:

Identifying and resolving issues across multiple integrated modules was challenging during system testing.

Solution:

Performed systematic unit and integration testing for each module and used detailed logging to trace errors and ensure smooth module interaction.

14. Testimonials from team

Working on the **Inventra – Intelligent Inventory Management System** provided valuable hands-on experience in designing and developing a real-world inventory application. The project helped us understand the complete software development lifecycle, from requirement analysis to implementation and testing.

The team gained practical exposure to **Spring Boot, REST APIs, database integration, and secure authentication mechanisms**. Collaborative problem-solving and regular discussions improved our technical knowledge as well as teamwork and communication skills.

Guidance from the mentor and continuous feedback played a crucial role in successfully implementing key features such as inventory alerts and reporting. Overall, this project enhanced our confidence in building secure, scalable, and industry-relevant software solutions.

15.Conclusion

The **Inventra – Intelligent Inventory Management System** successfully demonstrates the design and implementation of a secure, efficient, and scalable inventory management solution. The project effectively addresses common challenges such as inaccurate stock tracking, lack of real-time visibility, and manual inventory handling by automating inventory operations through a centralized system.

Throughout the development process, the project provided practical experience in backend development, database design, secure authentication, and system integration. The modular architecture ensures maintainability and supports future enhancements, making the system adaptable to evolving business requirements.

Overall, this project enhanced our understanding of real-world software development practices and strengthened our technical, analytical, and teamwork skills, making it a valuable learning experience and a strong foundation for future industry-level applications.

16.Acknowledgement

We sincerely express our gratitude to **Infosys Springboard** for providing us with the opportunity to participate in the Virtual Internship program and gain valuable practical experience. We extend our heartfelt thanks to our mentor for their continuous guidance, encouragement, and constructive feedback throughout the project.

We also thank our team members for their cooperation, dedication, and teamwork, which played a vital role in the successful completion of this project. Finally, we are grateful to everyone who supported us directly or indirectly and contributed to making this project a successful learning experience
