

Department of Master of Computer Applications

Django Project

On

Online Inventory Management System

26 MAY 2023

Submitted To:

Mr. Satheesh Kumar
Assistant Professor
Dept. of Computer Applications

Presented By:

Name: Amil Dev S
Roll No: 22pmc111

TABLE OF CONTENT

1. ABSTRACT.....	01
2. PROJECT REQUIREMENTS.....	02
3. FEATURES AND HIGHLIGHTS.....	03
4. TECHNICAL ASPECTS.....	04
• ARCHITECTURE OF THE PROJECT.....	05
• DATABASE(CLASS DIAGRAM).....	06
5. CHALLENGES FACED DURING DEVELOPMENT.....	07
6. IMPORTANT SCREENSHOTS.....	08
7. FUTURE ENHANCEMENT.....	09
8. CONCLUSION.....	10
9. REFERENCES.....	11

ABSTRACT

ONLINE INVENTORY MANAGEMENT SYSTEM

The Online Inventory Management System is a web-based application built using the Django framework, designed to streamline and automate the inventory management process for businesses. With the rapid growth of e-commerce and the need for efficient inventory management, businesses require a robust solution to ensure accurate inventory tracking, reduce stockouts, and optimize order fulfillment. The Online Inventory Management System addresses these challenges by offering a user-friendly interface and powerful features.

Key functionalities of the system include real-time inventory tracking, product management, and reporting. The system enables staffs to add, update, and monitor product information, track stock levels, and receive notifications for low inventory. The system leverages the Django framework's capabilities, providing a scalable, secure, and modular architecture. It utilizes Django's models, views, and forms to handle data storage, business logic, and user interaction. Additionally, third-party libraries and APIs are incorporated to enhance functionality and improve integration with external systems.

The Online Inventory Management System offers several benefits to businesses, including improved inventory accuracy, optimized order processing, and enhanced decision-making through reports and analytics. It helps reduce manual effort, streamline operations, and provide a real-time overview of inventory status.

In conclusion, the Online Inventory Management System built with Django provides businesses with an efficient, reliable, and user-friendly solution for effective inventory management. By centralizing inventory data, automating processes, and providing actionable insights, the system enables businesses to optimize their inventory, reduce costs, and improve customer satisfaction.

PROJECT REQUIREMENTS

User Management:

- The system should have two user roles: admin and staff.
- Only the admin user should have the ability to register new staff members.
- Admin users should be able to create and manage staff accounts, including assigning roles and permissions.
- Staff users should be able to log in securely using their credentials.

Product Management:

- Admin and staff users should be able to add and delete products in the system.
- Admin and staff users should have the ability to update product information, such as name, description, quantity, and category.
- The system should support the categorization of products for easier organization and searching.

Inventory Tracking:

- Real-time inventory tracking should be implemented to keep track of the available quantity of each product.
- The system should automatically update inventory levels when purchases are made or products are restocked.
- Admin and staff users should be able to view and search the current inventory levels of each product.
- Provide alert to admin users when inventory levels fall below a predefined threshold.

Order Management:

- Admin and Staff users should be able to issue stock to the customer.
- Admin and staff users should have access to view all orders in the system.
- Invoices and SMS alert should be generated for each order.

Reporting and Analytics:

- Graphical representations, charts, and visualizations should be used to present data effectively.

FEATURES AND HIGHLIGHTS

Login functionality to secure the system:

- Users, both staff and admin, will have their own login credentials to access the system.
- A secure login page will be provided with username and password fields.
- Upon successful authentication, users will be granted access to their respective roles and functionalities within the system.

Add, edit, and delete products in the inventory:

- Admin and staff users will have the ability to add new products to the inventory by providing relevant details such as name, category, reorder level and quantity.
- Users can also edit existing product information to update any changes.
- In case a product is no longer available, users can delete it from the inventory.

Real-time tracking of inventory levels and stock availability:

- The system will provide real-time tracking of inventory levels for each product.
- Users can view the current stock availability and quantity for better inventory management.

Alerts for low stock or out-of-stock items:

- The system will generate alerts to admin or staff users when the stock of a product falls below a predefined threshold.
- This will help users take timely action to restock or reorder items to avoid stockouts.

Export available stocks to a CSV file:

- Users will have the option to export the available stock information to a CSV file.
- This feature allows users to analyse and manipulate the inventory data using external tools such as spreadsheets.

Update profile details of staff and admin:

- Staff and admin users will be able to update their profile information, such as name, contact details, or profile picture.
- This feature allows users to maintain up-to-date personal information within the system.

Invoice generation for each issued product:

- When an order is placed or a product is issued, the system will generate an invoice with details such as product name, quantity, customer name, address and phone number.
- The invoice can be printed or saved as a PDF for documentation and record-keeping purposes.

Generate visualized reports to analyze inventory levels:

- The system will generate reports that provide visual insights into inventory levels, such as the current stock of each product.
- Graphs, charts will be used to present the data in a visually appealing and easy-to-understand format.

SMS alerts to notify customers when stock is issued:

- The system can integrate with an SMS service provider to send automated SMS notifications to customers when their ordered products are issued or shipped.
- This feature enhances customer communication and provides them with real-time updates on their orders.

Mini chatbot to guide newly joined staff:

- The system can include a mini chatbot feature to assist newly joined staff members.
- The chatbot can provide basic information about the system, answer frequently asked questions, or guide users through common tasks.
- This feature aims to improve user onboarding and provide immediate assistance for staff members.

TECHNICAL ASPECTS

Django Libraries & Third party APIs:

Admin Page: django-jazzmin:

- Django-jazzmin is a customizable admin interface for Django projects.
- It provides a modern and responsive admin page with additional features and customization options.
- With django-jazzmin, the default Django admin interface can be enhanced and tailored to specific project requirements.

CSS: Bootstrap 5:

- Bootstrap 5 is a popular CSS framework that provides a collection of pre-built responsive components and utilities.
- It simplifies the process of designing and styling web pages, making them visually appealing and mobile-friendly.
- Bootstrap 5 can be used to create a consistent and professional-looking user interface for the online inventory management system.

Forms: django-crispy-forms:

- Django-crispy-forms is a Django package that helps in rendering forms in a clean and elegant manner.
- It provides a way to define form layouts and styles using Python code or templates.
- django-crispy-forms simplifies the process of creating and customizing forms in the online inventory management system.

Login: django-registration-redux:

- Django-registration-redux is a Django package that provides a flexible and customizable user registration and authentication system.
- It simplifies the implementation of user registration, login, and password reset functionalities in the online inventory management system.

ChatBot: Dialogflow:

- Dialogflow is a natural language understanding platform provided by Google.
- It enables the creation of AI-powered chatbots and virtual assistants.
- Dialogflow can be integrated into the online inventory management system to provide a chatbot feature for assisting users and answering common queries.

SMS Alert: Twilio:

- Twilio is a cloud communications platform that allows developers to send and receive SMS messages, make phone calls, and perform other communication tasks programmatically.
- Twilio can be used to integrate SMS alerts into the online inventory management system, enabling the system to send automated notifications to users via SMS.

PDF Generation: reportlab:

- Reportlab is a Python library used for generating PDF documents programmatically.
- It provides a wide range of features for creating dynamic and customized PDFs.
- The reportlab library can be utilized in the online inventory management system to generate invoices, receipts, or other PDF documents.

Import & Export of Stock: django-import-export:

- Django-import-export is a Django package that simplifies the import and export of data in various formats, including CSV, JSON, Excel, etc.
- It provides a user-friendly interface for importing and exporting stock data in the online inventory management system.

Visualization: Chartjs:

- Chart.js is a JavaScript library for creating interactive and visually appealing charts and graphs.
- It offers various chart types, including bar charts, line charts, pie charts, etc.
- Chart.js can be used to visualize inventory data, sales trends, and other relevant information in the online inventory management system, providing a clear representation of data for analysis and decision-making.

ARCHITECTURE OF THE PROJECT

1. Presentation Layer:

1. User Interface (UI) using HTML, CSS, JavaScript
2. Templates for rendering dynamic content
3. Views to handle user requests and responses

2. Application Layer:

1. Django Framework as the web application framework
2. Business logic implementation
3. Models representing inventory items, orders, users, etc.
4. Forms for data validation and user input handling

3. Database:

1. Relational Database Management System (MySQL)
2. Tables representing entities (inventory, orders, users, etc.)
3. Relationships between tables for data integrity

4. Authentication and Authorization:

1. User login, and logout functionality
2. Permissions and roles management
3. User authentication using Django's built-in authentication system

5. Inventory Management Module:

1. CRUD operations for managing inventory items
2. Tracking stock levels, quantities, and availability
3. Search and filtering capabilities

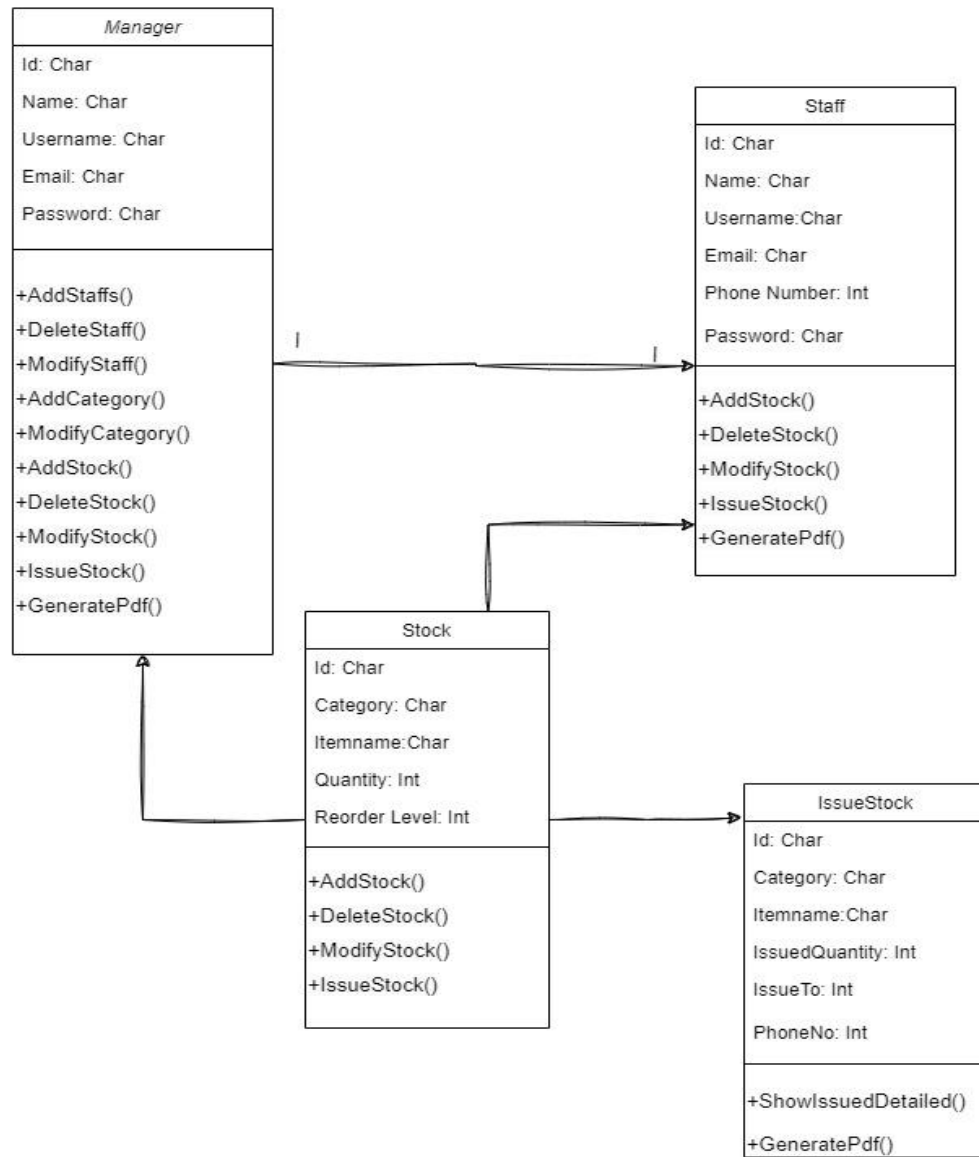
6. Order Management Module:

1. Placing and processing customer orders

7. Reporting Module:

1. Generating reports on inventory levels, sales, etc.
2. Data visualization and analytics
3. Exporting reports in various formats (e.g., PDF, CSV)

DATABASE (CLASS DIAGRAM)



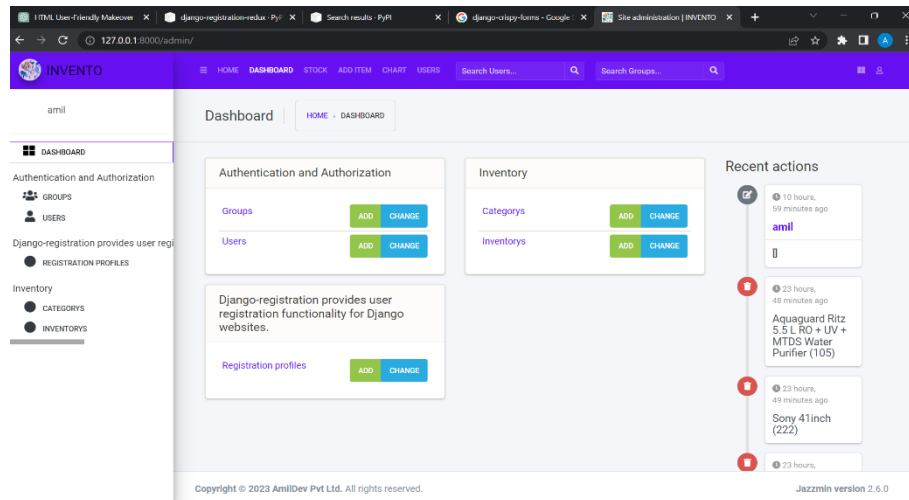
CHALLENGES FACED DURING DEVELOPMENT

- Ensuring accurate and consistent data across different modules and database tables.
- Handling data validation and ensuring data integrity during various operations.
- Integrating the inventory system with Django libraries and third-party APIs.
- Creating a user-friendly interface that is intuitive and easy to navigate.
- Balancing the need for a visually appealing design with functional requirements.
- Implementing secure user authentication and authorization mechanisms.

IMPORTANT SCREENSHOTS

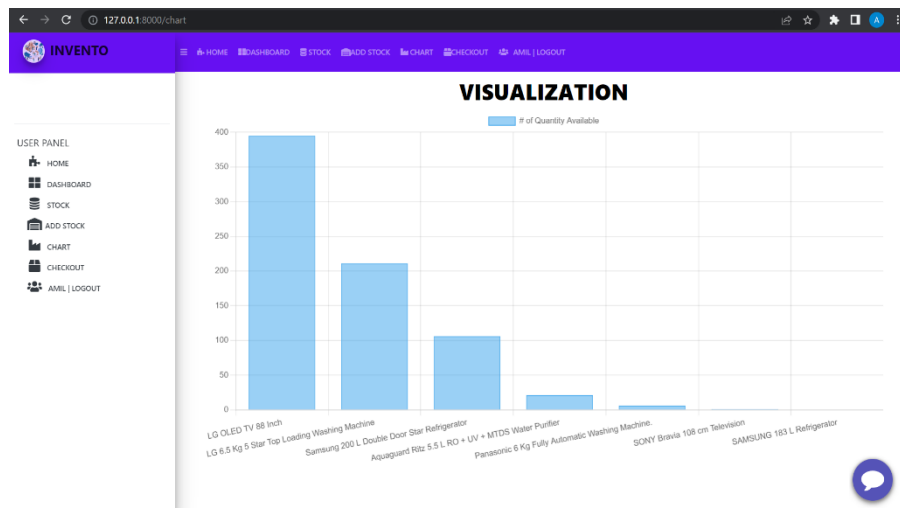
- Admin Panel

Customised Django admin panel with Django-jazzmin library, which helps to make the user interface more interactive. It provides more facilities such as sidebar, logo, logo image, filter, show recent actions and so on.



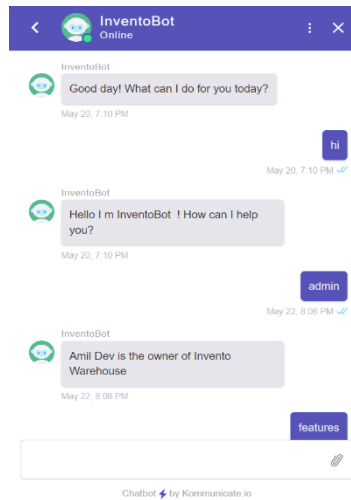
- Visualization

This page provides the graphical visualization of the stock according to their quantity. Here I used Chart.js library to build the visualization and the chart used is Bar Graph.



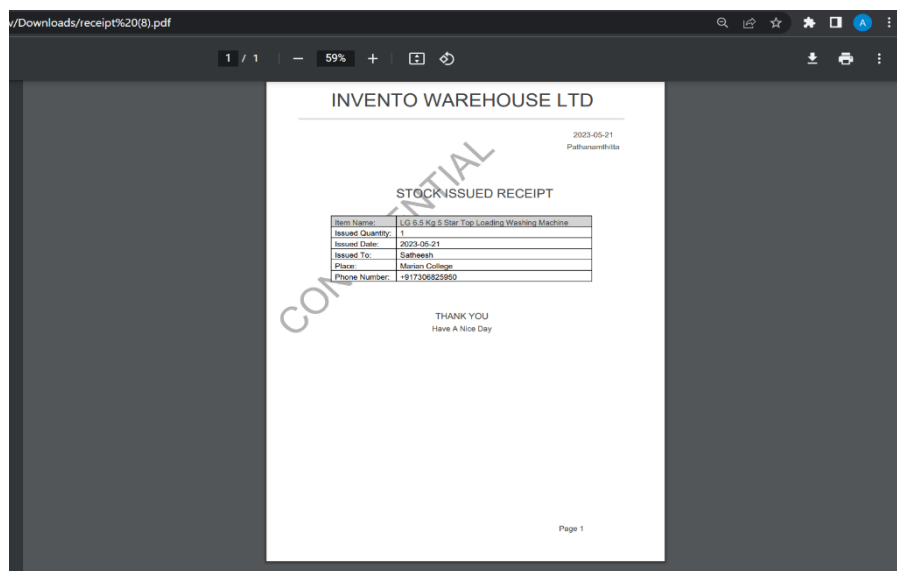
- Mini ChatBot

Mini Chatbot is one of the interesting feature in the Online Inventory Management System. The main purpose of the chatbot is to provide guide for the newly joined staff to know the features and procedures of the website. Dialogflow ,owned by Google is used to build the chatbot with the help of Kommunicate.io



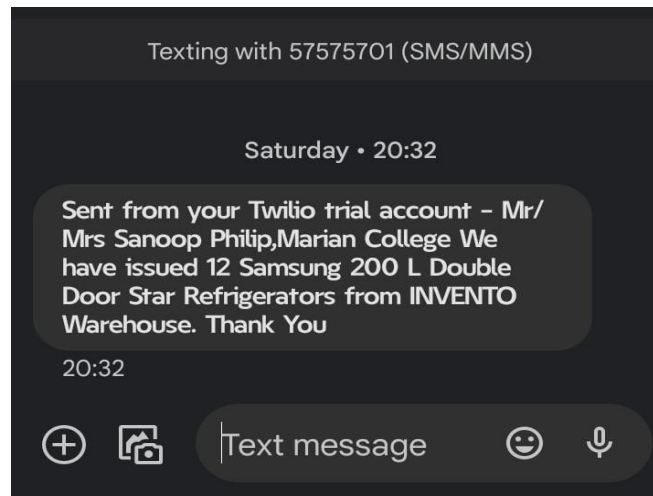
- Stock Issue Receipt

When the stock is issued by the staff, the system will automatically provide a stock issued receipt, which is in the form of PDF. It includes the selected product, quantity, issued date, customer name and details .



- Stock Issued SMS

If the stock issued successfully, an sms notification will send to the customer's phone number. It includes the customer details and the name of issued product with the number of quantity issued. SMS Notification is integrated with the help of Twilio, which is a third party API to provide notifications and calls etc.



FUTURE ENHANCEMENT

- Develop a mobile application to provide on-the-go access to the inventory system.
- Enable barcode scanning using mobile device cameras for quick product identification.
- Enhance reporting capabilities to provide more detailed insights into inventory levels, sales trends, and profitability.
- Extend the system to include warehouse management features such as location tracking, bin management, and stock transfers between warehouses.
- Predict future stock requirements based on historical data and market trends.

CONCLUSION

In conclusion, the development of an Online Inventory Management System offers significant benefits to businesses in effectively managing their inventory, streamlining operations, and improving overall efficiency .

By implementing features such as real-time inventory tracking, the system provides accurate and up-to-date inventory information. The user-friendly interface ensures easy navigation and efficient management of stock . The project has leveraged the power of Django, a robust web framework, along with third-party libraries to achieve a scalable and secure solutions . It contributes to improved productivity and cost savings.

By continually evolving and incorporating future enhancements, the system can adapt to the changing business landscape and ensure sustained growth and success.

REFERENCES

- <https://youtu.be/iINE3iaN1Oo>
- <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django>
- <https://djangopackages.org/>
- <https://pypi.org/>
- <https://chat.openai.com/>
- <https://dialogflow.cloud.google.com/>
- <https://www.twilio.com/en-us>
- <https://bard.google.com/>