IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

DRUG TRACEABILITY IN HEALTHCARE

[1]Arigela Pranay Kumar, [2]B Venkateswarlu, [3]Charan Sai Paritala. [4]Dindar Pranav Sai
[1]Ug Scholar, [2]Assistant Professor, [3]Ug Scholar, [4]Ug Scholar
[1]CSE Data Science,
[1]Geethanjali College of Engineering and Technology, Hyderabad, India

Abstract: The Drug Traceability System is a web-based application developed using the Django framework, designed to enhance the traceability and transparency of pharmaceutical supply chains. In a state like Telangana which is aiming to be a leading pharmacy supply chain to the world it is essential to put quality control and preventing fake medicines business should be a high priority, It also provides an easy way to report fake medicines by the general user .The System comprises modules for users, manufacturers, suppliers, and administrators ,each contributing to the efficient management of drug manufacturing, distribution, and oversight .Presently fake medicine supply is one of a major issues and it can lead to several health hazards this system aims to solve this problem by centralizing the data about manufacturing of medicines and under regular inspection of medicines by the government or by the public we can easily identify fake medicines that are being supplied into the market and if irregularity or side effects of a medicine are observed we can easily identify the manufacturer and supplier of the medicine, Our web Application helps In easily managing Inventory between different stake Holders and also provide users and testers easy way to submit their data and by using Machine learning algorithms identify real and fake medicines and provide a easy way for the user to Check the data regarding their own medicines.

***IndexTerms* – Django, Framework, Stake Holders, SVM, KNN, XGBoost.**

INTRODUCTION

The Drug Traceability System is a cutting-edge web application built on the Django framework, meticulously crafted to revolutionize the pharmaceutical supply chain landscape. Its core mission revolves around bolstering traceability and transparency while tackling the pervasive threat of fake medicines. In regions like Telangana, where ambitions soar high to lead the global pharmacy supply chain, such a system is indispensable.

At its core, the system's architecture is modular, catering to distinct stakeholders such as users, manufacturers, suppliers, and administrators. This modular design fosters seamless collaboration and efficient management across all facets of drug manufacturing, distribution, and oversight. Notably, the system places a paramount emphasis on stringent quality control protocols. Leveraging data management and robust inspection mechanisms, it ensures that only authentic and safe medicines reach consumers.

One of the system's standout features is its user-centric approach to combating fake medicines. By providing a user-friendly interface for reporting suspicious products, it engages the public in a collective effort to identify and eliminate counterfeit drugs from the market. This collaborative approach not only enhances consumer safety but also bolsters trust in the pharmaceutical ecosystem.

Furthermore, the system's comprehensive suite of tools empowers administrators with real-time insights and actionable data, enabling proactive decision-making and swift responses to emerging challenges. By centralizing crucial information about medicine manufacturing, distribution channels, and regulatory compliance, the system fosters a culture of accountability and transparency.

Objective.

- Combating the circulation of fake medicines by implementing robust tracking and verification mechanisms.
- Enhancing traceability throughout the pharmaceutical supply chain to verify the authenticity of medicines and ensure compliance with quality standards.
- Improving quality control measures to uphold the safety and efficacy of medicines during manufacturing and distribution.
- Empowering stakeholders, including users, manufacturers, suppliers, and administrators, through user-friendly interfaces and tools.
- Facilitating regulatory compliance by providing comprehensive data management, audit trails, and reporting functionalities.

Proposed System

The proposed Drug Traceability System, developed using the Django framework, is designed to revolutionize pharmaceutical supply chains in Telangana by prioritizing quality control and combatting the distribution of fake medicines. With modules tailored for users, manufacturers, suppliers, and administrators, the system ensures efficient management of drug manufacturing, distribution, and oversight. By centralizing data and providing easy reporting mechanisms for fake medicines, it aims to swiftly identify irregularities and side effects, enabling quick action to maintain the integrity of the pharmaceutical market and ensure public safety.

This system's robust features not only streamline processes within the supply chain but also empower users to actively participate in reporting and verifying medicines, fostering a transparent and accountable pharmaceutical ecosystem. Through proactive measures such as regular inspections and data analytics, the system not only prevents the proliferation of fake medicines but also facilitates rapid response to any potential health hazards, thus positioning Telangana as a leader in ensuring pharmaceutical traceability and quality assurance on a global scale.

Machine Learning Model

This step involved identifying all the key factors that are essential in approval step, Choose the right model and extract the model from the ipynb file, we tested different Machine Learning models like KNN, SVM, XGBoost, Lightgbm. we ended up using XGboost as it had an accuracy of around 92 percent after training and evaluating the model we used pickle package to extract the Machine Learning model into the disk and later this extracted file will be used to run prediction in the Flask application, The training dataset used was from Kaggle but for testing data we generated a new dataset using Excel random function, the accuracy of various ML models are as following SVM 88 percent, KNN 90 percent and Lightgbm 92 percent, The reason for choosing XGboost over Lightgbm is because it is more standard Algorithm in the industry and also Lightgbm only performs well for small datasets, it tends to suffer when huge amount of data is available.

System Architecture

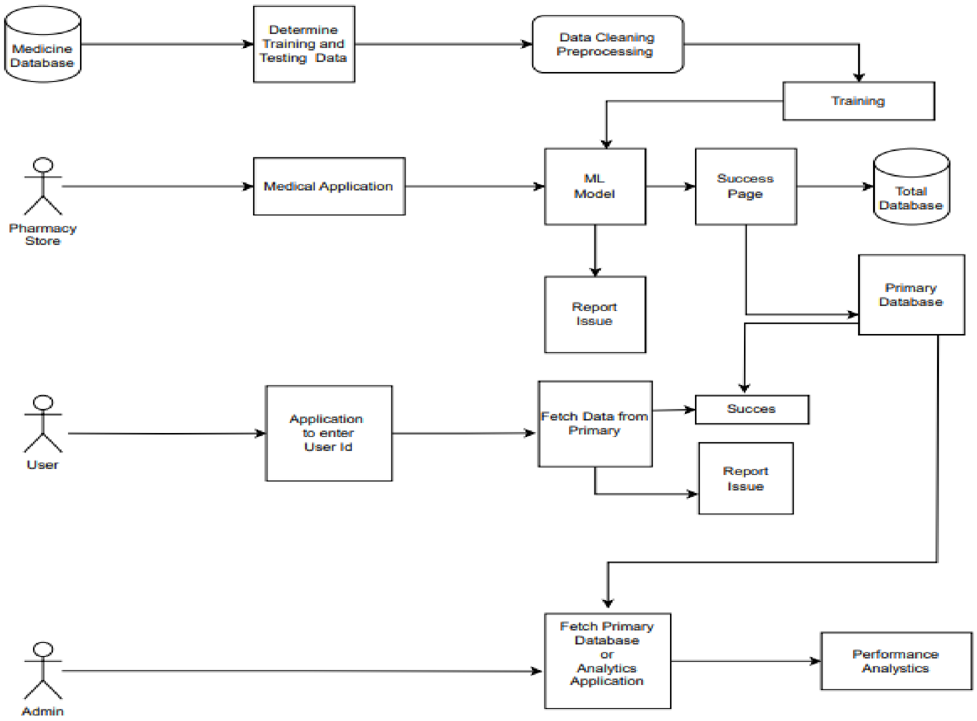


Fig 1 System Architecture of proposed System

System architecture encompasses the fundamental structure and organization of a software system, detailing the arrangement of its components, modules, and their interactions. It defines the layers of the system, such as the presentation layer responsible for user interfaces, the application layer managing business logic, and the data layer handling data storage and retrieval. This architectural design facilitates efficient communication between different parts of the system, ensures scalability, modularity, and robustness, and provides a framework for developers to implement and maintain the system effectively.

Use Case Diagram

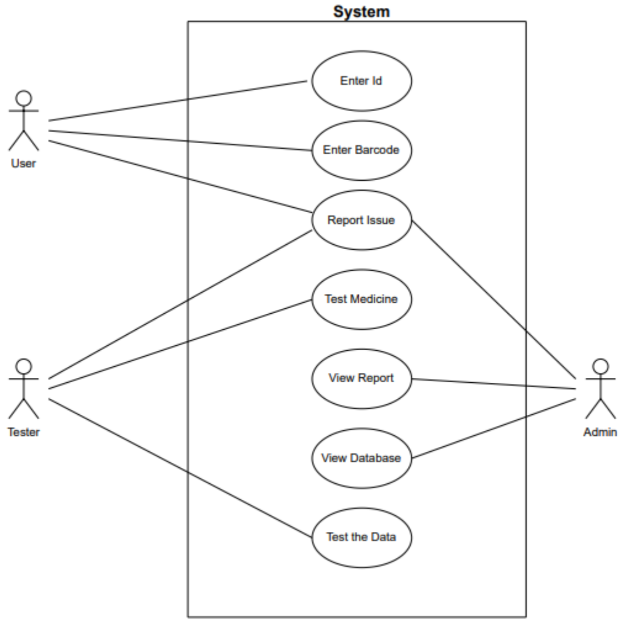


Fig.2 Use Case Diagram

A collection of use cases, actors, and their relationships are shown in use case diagrams. They represent a system's use case perspective. A use case represents a certain system functionality. The relationships between the functionalities and their internal/external controllers are therefore described using a use case diagram.

Implementation

Integrating Machine Learning Model in Backend

To bring machine learning capabilities to your Django application, start by ensuring you have a trained and optimized XGBoost model. This model, honed on a relevant dataset and tuned through hyperparameter optimization, is the foundation of your application's predictive abilities. Once satisfied with its performance, export the model to a format compatible with Django, such as using Pickle. This creates a serialized file containing the model's architecture and learned weights, essentially capturing its knowledge in a way that can be loaded and utilized within your Django application.

The next step involves creating a Django application specifically for this project. This application, generated using the Django-admin startapp command, will house the code responsible for interacting with the XGBoost model. Within this application, you'll create a view function to act as the intermediary between the user interface and the machine learning model.

To enable this interaction, the view needs to load the pre-trained XGBoost model you exported earlier. Using the Pickle library, the view deserializes the model from the stored file. This process essentially unpacks the model's configuration and knowledge from the file and loads it into memory, making the model ready to receive data and generate predictions within your Django application.

By integrating the XGBoost model in this way, you empower your Django front-end to interact with the model and benefit from its predictions. This can unlock various functionalities within your application, such as product recommendations, risk assessments, or any other prediction task suited to your XGBoost model's design.

Barcode Reader

The process involves several steps seamlessly integrated. Firstly, ensure you have a Django application set up. Utilize a JavaScript barcode scanning library like QuaggaJS to enable barcode scanning directly from the front end. Upon scanning, the captured barcode data is sent to a Django view using AJAX or a form submission. In the Django view, the received barcode data is processed, and a query is made to the database to fetch corresponding information. This involves defining models and querying the database using Django's ORM. Once the database query retrieves the relevant data, it is returned as a response from the Django view back to the front end. Finally, the front end displays the fetched data to the user, completing the process seamlessly integrated into the Django application.

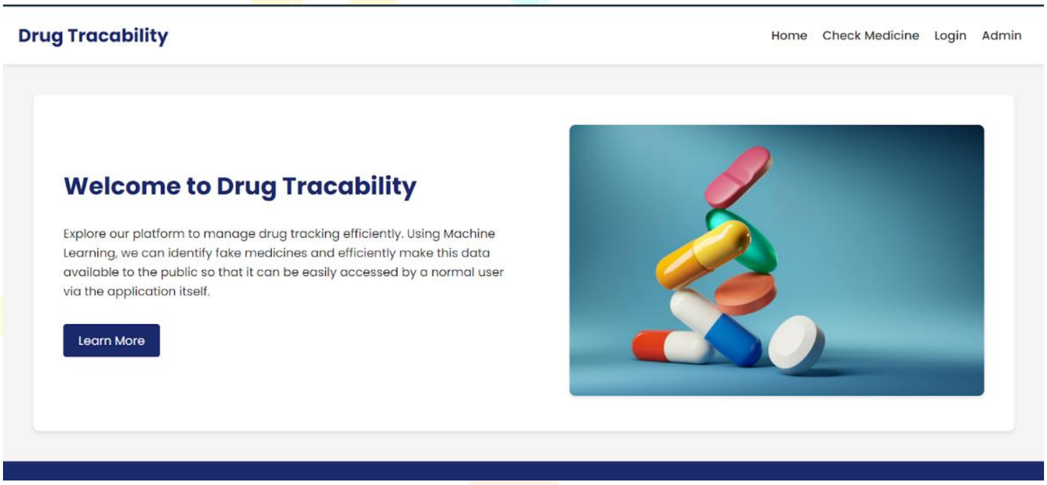
Dashboard

In crafting an interactive user dashboard within a Django front-end application, we embark on a comprehensive integration journey. Firstly, we ensure Django's installation and create a Django project, employing its built-in functionalities for simplicity and efficiency. Leveraging Django's templating engine, we design the dashboard's interface using HTML, CSS, and possibly JavaScript frameworks like Bootstrap or Vue.js for enhanced interactivity. Next, we define Django models representing the data stored in the database, ensuring they reflect the structure and relationships accurately. To enable live data fetching, we implement Django views responsible for querying the database and retrieving the latest data. These views utilize Django's ORM to interact seamlessly with the database. Subsequently, we connect these views to corresponding URLs, enabling navigation and interaction within the dashboard. Additionally, we incorporate AJAX or WebSocket’s to facilitate real-time updates, ensuring the dashboard reflects the most recent data dynamically. Finally, we integrate user authentication and authorization mechanisms to personalize the dashboard's content based on user roles and permissions, thereby enhancing security and usability.

Submitting Report

In implementing a Submit Report function within a Django application, we orchestrate a series of steps to seamlessly collect user data, merge it with a PDF template, and provide a means for users to download the generated PDFs. Initially, we craft a Django form, defining fields that correspond to the data we wish to collect from users. This HTML form, rendered within a Django template, allows users to input relevant information conveniently. Upon submission, the form data is sent to a Django view for processing. Within this view, we leverage Python's PDF manipulation libraries such as PyPDF2 to dynamically generate PDF documents. We utilize a pre-designed PDF template, incorporating placeholders where user-submitted data will be inserted. Subsequently, we extract the form data submitted by the user and populate the appropriate fields within the PDF template. The merged PDF document is then generated programmatically. To provide users with easy access to the generated PDFs, we implement functionality within Django view to serve the PDF file for download. This typically involves saving the generated PDF to a temporary location on the server and returning a response with the PDF file's content type, triggering the browser to prompt the user to download the file. Additionally, we may include a link or button in the Django template interface, enabling users to initiate the PDF download effortlessly. Through this meticulous integration of Django's form handling, PDF manipulation, and file serving capabilities, we establish a streamlined "Submit Report" function, facilitating the seamless collection, merging, and distribution of user-generated data in PDF format.

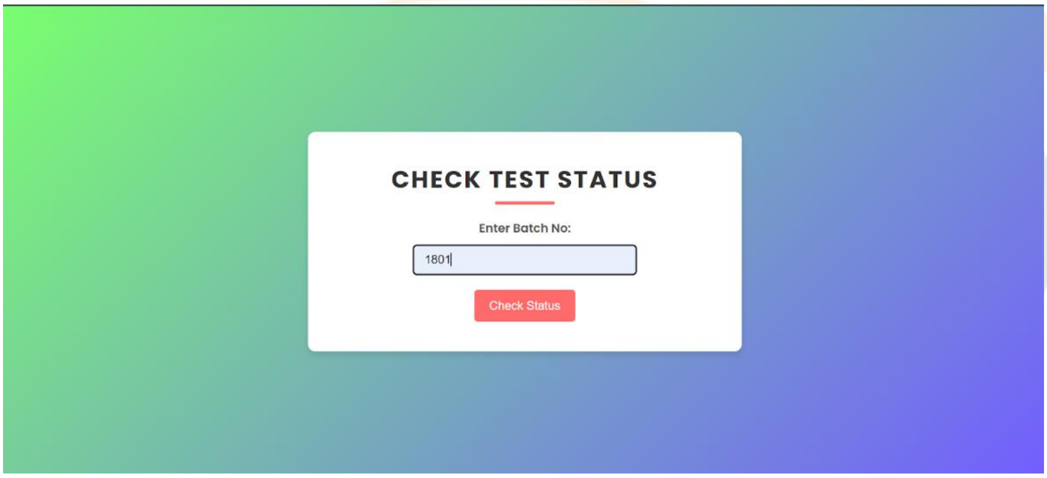
Outputs



Welcome to Drug Tracability

Explore our platform to manage drug tracking efficiently. Using Machine Learning, we can identify fake medicines and efficiently make this data available to the public so that it can be easily accessed by a normal user via the application itself.

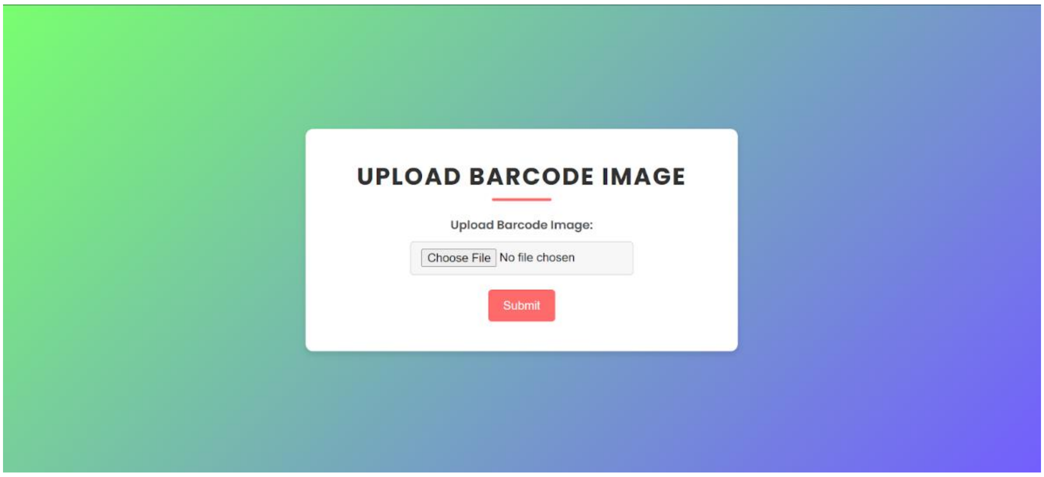
Fig.3 Home Page



CHECK TEST STATUS

Enter Batch No: 1801

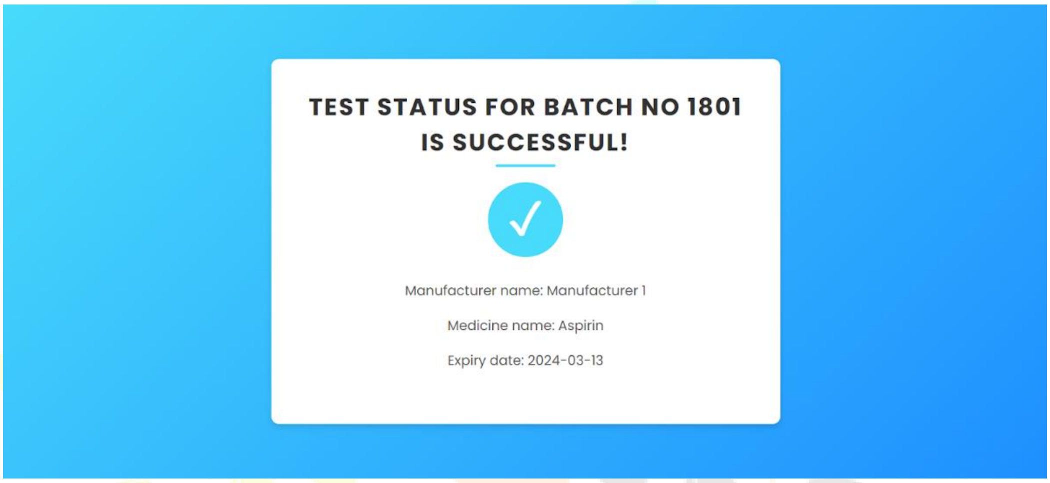
Fig.4 Check Test Screen



UPLOAD BARCODE IMAGE

Upload Barcode Image:

Fig.5 Upload Barcode Page



TEST STATUS FOR BATCH NO 1801 IS SUCCESSFUL!

Manufacturer name: Manufacturer 1

Medicine name: Aspirin

Expiry date: 2024-03-13

Fig.6 Success Page

Conclusion

The Drug Traceability in Healthcare, developed using Django, is a pivotal solution in Telangana's quest to lead the global pharmacy supply chain. It prioritizes quality control and the prevention of fake medicines, offering a user-friendly platform for reporting such incidents. With modules catering to users, manufacturers, suppliers, and administrators, the system streamlines drug manufacturing, distribution, and oversight. By centralizing data and enabling regular inspections, it becomes easier to detect and eliminate fake medicines from the market, ensuring public health safety, This system's significance is amplified by the pressing issue of fake medicine supply, which poses serious health risks. Through robust data management and swift identification of irregularities, it empowers authorities and the public to take decisive action against counterfeit medicines. By fostering transparency and accountability across the pharmaceutical supply chain, this system not only safeguards consumers but also strengthens Telangana's reputation as a leader in pharmacy traceability and integrity.

Acknowledgment

We are greatly indebted to the authorities of Geethanjali College of Engineering and Technology, Cheeryal, Medchal District, for providing us the necessary facilities to successfully carry out this major project work titled "Drug Traceability in Health Care", Firstly, we thank and express our solicit gratitude to our HOD, Dr L. Kiran Kumar Reddy, Associate Professor, CSE(DS) department, Geethanjali College of Engineering and Technology, for his invaluable help and support which us a lot in successfully completing our main project, Secondly, we express our gratitude to Mr. B. Venkateswarlu, Assistant Professor CSE(DS), internal guide, Geethanjali College of Engineering and Technology, for his suggestion and encouragement which helped us in the successful completion of our main project.

We would like to express our sincere gratitude to our Principal Dr. S. Uday Kumar for providing the necessary infrastructure to complete our project, we convey our gratitude to our Chairman, Mr. G. Ravinder Reddy, for his invaluable support and encouragement for propelling our innovation forward. Finally, we would like to express our heartfelt thanks to our parents who were very supportive both financially and mentally and for their encouragement to achieve our goals.

Finally, we would like to express our heartfelt thanks to our parents who were very supportive both financially and mentally and for their encouragement to achieve our set goals.

REFERENCES

[1] Counterfeit Medicine Detection using Deep Learning PROF. KIRTI MOTWANI1, RACHAEL DSOUZA2 , RHEA DSOUZA3 , JANICE JOSE4 1 Professor, Xavier Institute of Engineering 2, 3, 4 Student, Xavier Institute of Engineering

[2] A Machine Learning Technique to Detect Counterfeit Medicine Based on X-Ray Fluorescence Analyser, Muna Alsallal, Mhd Saeed Sharif, Baydaa Al-Ghzawi, Sabah Mohammed Mlkat al Mutoki.

[3] "Pharmaceutical Anti-Counterfeiting: Combating the Real Danger from Fake Drugs" by Mark Davison.

[4] "Pharmaceutical Traceability and Counterfeit Detection" by Himanshu Das and A.K. Jana.

[5] "Drug Traceability in Healthcare: A Comprehensive Guide" by James R. Edwards.

[6] "Pharmaceutical Supply Chain: Drug Quality and Security Act (DQSA) Overview" by John Stewart.

[7] "Healthcare Supply Chain Management: Basic Concepts and Principles" by Philip Kaminsky and Brian Jacobs