

Original Article

Python and MySQL-based Transport Management System

Nayan Sonkalyari¹, Advait Patil², Tiwar Zade³, Kaushal Sharma⁴, Amitkumar Manekar⁵

^{1,2,3,4,5}Information Technology, Shri Sant Gajanan Maharaj College of Engineering, Shegaon, Maharashtra, India.

Received: 13 April 2023

Revised: 21 May 2023

Accepted: 04 June 2023

Published: 15 June 2023

Abstract - Every transport industry, as we all know, has specific paperwork that the company can complete. Transportation entails that their branches have a lot of paperwork. The entire paperwork in the being trained system is completed manually by the personnel. Everything must be done manually, which is exceedingly time-consuming, delicate to handle, and unreliable due to the high likelihood of error. Due to handmade operations, it could also result in significant data loss. The Transport Management System (TMS) design is a comprehensive software result designed to optimize and streamline transportation operations for businesses. The system automates colorful tasks similar to managing orders, dispatching vehicles, managing workers and generating bills. The Transport Management System (TMS) design using Beaker is a web-based software result that enables effective operation of business transportation operations. This design utilizes Flask, a Python web frame, to develop a robust and scalable TMS.

Keywords - Transport management, Operations, Flask, Web-based software, Automates.

1. Introduction

Transportation plays a crucial role in logistics and supply chain management. The manual handling of various transportation processes can be time-consuming, error-prone, and inefficient. To overcome these challenges, a Transport Management System (TMS) can be implemented to automate and optimize transportation operations. This paper presents a TMS developed using Python Flask and MySQL, providing a comprehensive solution for managing transportation activities. Indeed non-technical stoner can use the system effectively and without any difficulty. The system is upgradeable. Hence further improvement to the system is possible. The system maintains speed and delicacy with an eye-catching GUI. Applying Information Technology in the Transport Business can help to ameliorate the working process of association. The transport operation system aims to give window grounded, stoner-friendly and effective system to the staff and the guests. Overall, the objectives of a TMS are to optimize transportation operations, reduce costs, enhance client satisfaction, and ensure compliance with regulations and laws. By achieving these objectives, associations can ameliorate their overall profitability and competitive position. TMS should be designed to meet the association's specific requirements and give a comprehensive result for transportation operations that improve effectiveness, reduces costs, and enhances client satisfaction [1]. Python and MySQL have been used for enforcing similar types of systems [2]. The system is erected with the help of beaker technology as its main element and MySQL database

with it to get a good system in place. Beaker is easy to apply, as well as its featherlight back-end frame. MySQL is the most trusted and dependable- open-source database platform in use moment [3]. MySQL is one of the most extensively used databases for enforcing similar types of systems [4]. Effective transportation operation plays a vital part in enhancing force chain operations. The homemade running of transportation processes can be time-consuming, error-prone, and expensive. To address these challenges, developing a Transport Management System(TMS) exercising Python Flask and MySQL provides a comprehensive result. This paper aims to outline the design, perpetration, and advantages of the Python Flask and MySQL- grounded TMS in the environment of transportation operation.

2. Literature Survey

The paper acknowledges web services' significant role in people's daily lives and emphasizes how technological advancements have shaped the world. With the evolution of networks, the demand for internet connectivity has increased, as it has become essential for communication and various activities. To ensure a smooth experience, high-speed internet and sufficient bandwidth are necessary to prevent poor connections. Furthermore, this paper emphasizes the impact of advancements in computing technology and network infrastructure, which have enabled individuals to stay updated with the latest trends. This, in turn, has led to the rise of web development as a prominent avenue for accessing and gathering information [5].



The paper presents the development of a college database management system using the Python Flask framework. It offers features such as attendance and student grade tracking, accessible by both students and staff. Staff members have permission to update data, and users have separate logins. The system also includes real-time library, transport, and training/placement management. Its purpose is to provide easy access to information for all registered users, ensuring efficient data management in educational institutions [6].

This study's main goal is to investigate how to combine a complex back-end system with artificial intelligence built on Python with a web-based front-end for a student attendance system that makes use of facial recognition technology. The back-end system incorporates diverse technologies, while the front-end system leverages web technology to ensure accessibility through web browsers from any location, as long as there is a connection to the server. To enable the execution of Python code on the web, a combination of PHP and Flask is employed for integration purposes. The paper presents a detailed explanation of the integration between the front-end and back-end systems, along with performance test results [7].

This research paper centers around creating an indoor climate control system that operates on the client side. The system is made up of user-facing programmes and a web server that makes it easier for the programmes and devices to communicate with one another, aiming to manage the microclimate in a room. The paper also discusses various features and essential aspects involved in developing a web application. Additionally, it highlights the distinctions between the web app and other components of the project [8].

A study in the Philippines tackled the issue of healthcare accessibility by developing a telepresence system that utilized HTML5, Flask, and MySQL technologies. The implemented system included a web-based application and an electronic health record management system, allowing users to remotely create, store, and retrieve patient information [9]. This chapter focuses on building a RESTful Flask application starting from the ground up. The goal is to develop an application that manages a database of Authors and their associated books. The application will include a user authentication system to maintain security, restricting access to some features to only logged-in users [10].

This solution addresses the structural contradiction between talent supply and demand in higher vocational colleges. By utilizing big data technology, it collects real-time job information using Redis and Scrapy, stores it in an HBase database, and analyzes it using the Spark platform. The analysis results are stored in a MySQL database and displayed through charts using the Flask framework and

Echarts library. This system aims to improve talent training quality by aligning job requirements with educational courses and fostering the development of high-quality skilled professionals [11]. This research paper presents a data analysis and visualization platform for lithium-ion batteries. The platform is developed using the Flask framework and Pie charts. The design encompasses both front-end and back-end frameworks, including data preprocessing, visualization, and storage functionalities [12].

This paper presents a front-end development approach using MySQL and HTML5 for forest fire monitoring. The study focuses on the web front-end architecture and system architecture of the forest fire monitoring system. The web front-end architecture outlines the technical implementation of the detection system, while the system architecture describes the overall structure and design concept [21]. In the context of accessing a MySQL database, Flask can be used to design the application's data layer. The data layer is tasked with managing interactions with the database and executing CRUD (Create, Read, Update, Delete) operations. Flask provides various extensions and libraries that facilitate database access and ORM (Object-Relational Mapping) techniques [14].

This research paper centers around creating a lightweight database specifically designed for corrosion analysis in fluorochemical engineering processes. The database is developed using Flask, a Python-based web framework, as the back-end framework, while MySQL is utilized as the foundational database system [15].

This paper presents a stock trading web application that utilizes Flask and MySQL to empower users with knowledge about the stock market and alleviate their fears when entering the market. The application incorporates various machine learning models, web scraping techniques, and sentiment analysis to provide stock recommendations and real-time data [16].

Python is chosen as the preferred programming language for the web application due to its versatility and security features. By utilizing Python, the application is less vulnerable to SQL injection attacks commonly used by hackers. Python's robustness enhances the overall security of the web application [17].

The system's front-end is created using a WeChat applet, which enables users to extract basic information from invoice pictures by scanning the corresponding QR code. To manage the system's business logic and maintain the original voucher images and invoice details, the front-end makes use of a lightweight framework based on Python Flask and incorporates vue.js. This approach allows for the decoupling of the front-end and back-end, thereby enhancing modularity and flexibility [18].

This project centers around creating a network communication platform dedicated to calligraphy culture. The platform utilizes Flask and Python to integrate web technology into promoting and disseminating calligraphy culture. Its main objective is to provide networked and digital services that contribute to advancing calligraphy art and Chinese character culture [19].

3. Materials and Methods

Implementation Details: Back-end Development with Python Flask Python Flask is utilized for developing the back-end of the TMS. It handles routing and request handling and interacts with the MySQL database using appropriate libraries such as Flask-MySQL.

The TMS establishes a connection with the MySQL database using the MySQL Connector/Python library, enabling seamless data retrieval, insertion, and update operations.

The TMS includes a user-friendly web interface developed using HTML, CSS, and JavaScript. It provides intuitive features for accessing and managing transportation-related information.

The web application is designed with a focus on simplicity and clarity. The figure below (Fig. 1) illustrates the overall functioning of the system.

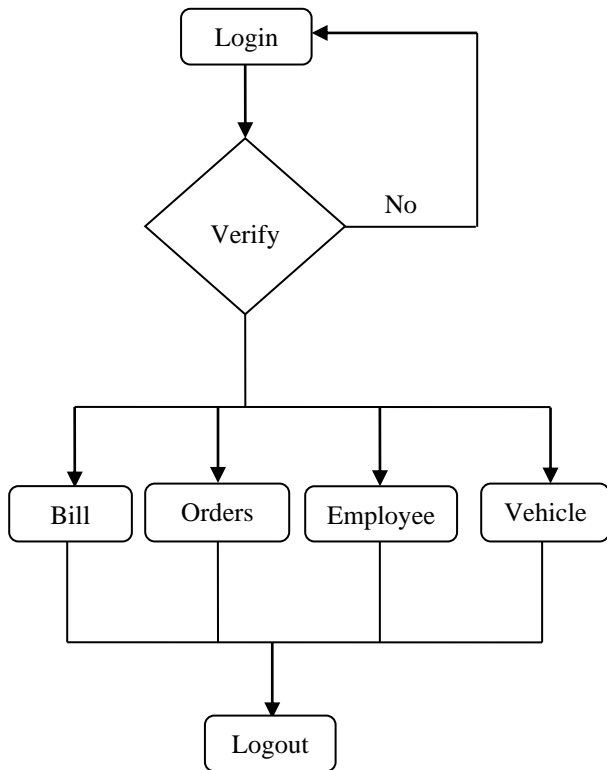


Fig. 1 System overall flow

First, the admin logs into the system and his credentials are verified. After successful login, the admin will see the dashboard containing the Bill, Orders, Employee and Vehicle modules.

In the bill section, we can perform various activities like Generating bills, Saving bills, Print bills and Deleting records, as shown in Fig. 2.

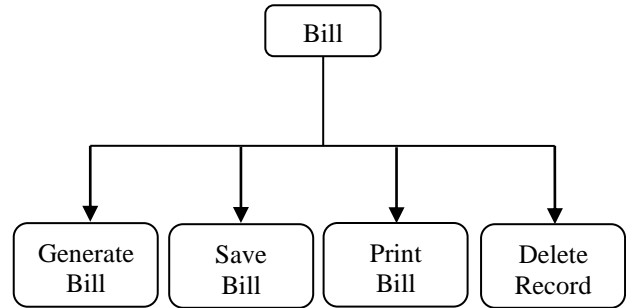


Fig. 2 Bill section

In the Orders section, we can perform various activities like Add Order, Update Status and Cancel Order, as shown in Fig. 3.

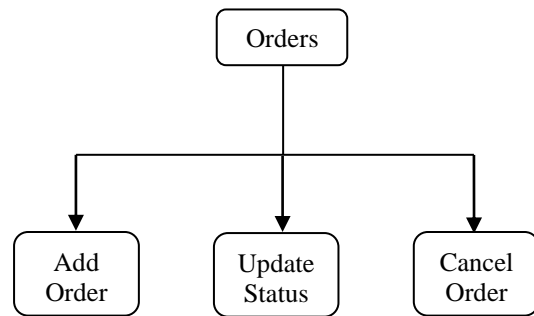


Fig. 3 Orders section

In the employee section, we can perform various activities like Adding, Editing and Deleting employee records, as shown in Fig. 4.

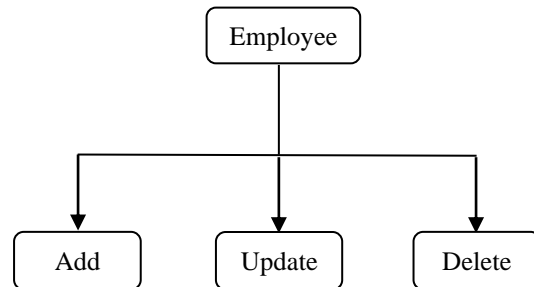


Fig. 4 Employee section

In the vehicle section, we can perform various activities like Adding, Edit and Delete vehicle records as shown in Fig. 5.

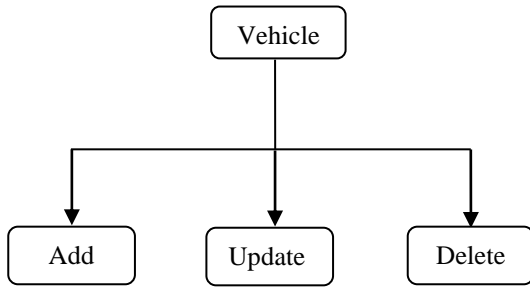


Fig. 5 Vehicle section

4. Results and Discussion

The transport management system is being shown, which is used for managing and creating bills for the customer, adding order details, and keeping records of employees and vehicles. The login screen appears when the “Admin” enters the system. “Admin” needs to enter his email address and password to access the system. The login screen is shown in Fig. 6.

After logging in to the system, you will see the chat symbol in the bottom right of the homepage, as shown in Fig. 7. Admin can click on it and start a conversation with the employees (drivers). This is important because employees may have queries so they can write their queries here, and the admin can give his response and solve their queries.

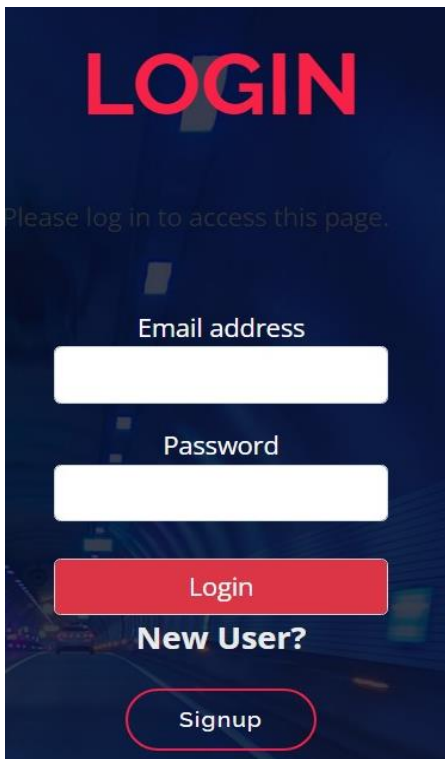


Fig. 6 Login screen

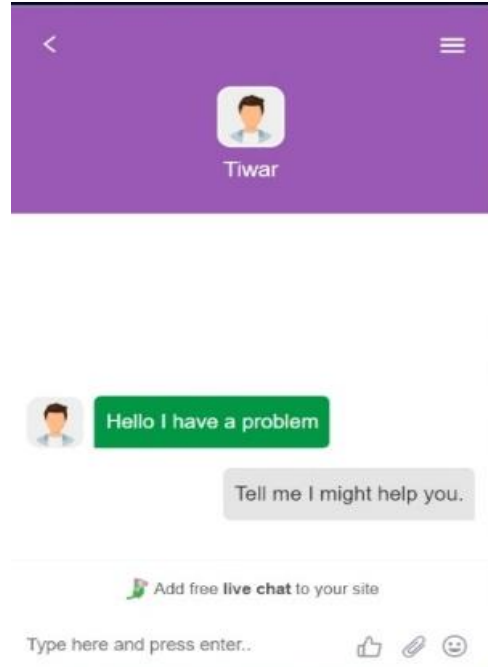


Fig. 7 Home screen

After clicking on Bill, you will see the following page. You can add bill details in the fields and save them to the database, as shown in Fig. 8.

Fig. 8 Bill Form

After clicking “Bill Records”, you will see the following page. You can print the bill by clicking on the Bill button and delete the record by clicking the delete button. If you want to get the complete record in the ‘csv’ file, you can click on “Import”, as shown in Fig. 9.

Bill Records

ID	TRUCK NUMBER	TRUCK TYPE	FROM	TO	CONSIGNOR	CONSIGNEE	TRANSPORTER	DISTANCE	RATE	GOODS	QUANTITY	DRIVER	DRIVER'S MOBILE NO.	TOTAL AMOUNT
7	RJ09GG946	8-Tyre	Akola	Nagpur	Manali Traders	Shakti Enterprises	Sharma Transport	270	75	Soyabean	30	ABC	2147483647	20250
8	MP09HC941	10-Tyre	Amravati	Mulapur	Pall Industries	Prabhakar Traders	Sharma Transport	200	75	Cement	30	BBB	2147483647	15000

Export

Fig. 9 Bill records

After selecting “Vehicle,” you will see the following page. You can add vehicle details and edit and delete the record, as shown in Fig. 10.

Add Vehicle

Vehicle Number Model

Vehicle Year Truck Type

Mileage Color

Booked

Yes No

Add Record

View Vehicle Records

ID	Vehicle Number	Model	Vehicle Year	Truck Type	Mileage	Color	Booked
7	RJ09GG946	Ashok Leyland	2017	10-Tyre	5	Red	No
8	MP09HC941	Mahindra	2019	12-Tyre	3	Blue	No

Fig. 10 Vehicle page

After selecting “Order”, you will see the following page. You can add order details and also edit and delete the record, as shown in Fig. 11.

Order

Customer Order date

Goods From

To Due date

Order status

Completed

Add Record

Records

ID	Customer	Order date	Goods	From	To	Due date	Order status
7	Pall Industries	2023-05-30	Soyabean	Akola	Nagpur	2023-06-02	Pending
8	Sharma Traders	2023-05-31	Cotton	Pune	Shigaon	2023-06-03	Completed

Fig. 11 Order page

After selecting “Employee”, you will see the following page. You can add employee details and also edit and delete the record, as shown in Fig. 12.

Add Employee

Employee Name Age

Mobile Number Gender

Salary Address

Add Record

View Employee Records

ID	Employee Name	Age	Mobile Number	Gender	Salary	Address
9	transport	250	2147483647	Male	3344	3eddededc
10	hbjj	424	2147483647	Female	112537	hrtij bfgelgfn
11	jjjt	50	388838	Male	27522	gfbfnhm n

Fig. 12 Employee page

You can log out of the system after clicking the “Logout” button in the navigation bar.

5. Conclusion

Transport Management System project is a comprehensive software system designed to help transport companies streamline their operations and improve their overall efficiency. The system includes several key features, such as vehicle management, order management, driver management, and customer invoicing. The system is designed to be user-friendly and intuitive, with a modern and visually appealing interface allowing users to access the information they need quickly. It is also highly scalable, allowing transport companies to easily add or remove vehicles and drivers as their business needs change. In addition to these core features, the TMS also includes a range of other features that are designed to improve the efficiency and effectiveness of the transport management process. This includes features such as customer invoicing, which allows companies to generate invoices for their customers automatically, and reporting and analytics, which provides companies with valuable insights into their operations. Overall, the Transport Management System project represents a significant improvement over traditional paper-based systems, which can be slow, error-prone, and difficult to manage. Through the implementation of this system, transportation companies can streamline numerous routine tasks, mitigating the potential for errors and enhancing operational efficiency. This can help them to save time and money while also improving the quality of their service and enhancing their reputation among customers.

References

- [1] Stanley E. Griffis, and Thomas J. Goldsby, “Transportation Management Systems: An Exploration of Progress and Future Prospects,” *Journal of Transportation Management*, vol. 18, no. 1, pp. 18-32, 2007. [CrossRef] [Google Scholar] [Publisher Link]
- [2] Swapnil Shah et al., “Python and MySQL based Smart Digital Retail Management System,” *6th International Conference for Convergence in Technology (I2CT)*, pp. 1-6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

- [3] Kodrat Iman Satoto et al., "Optimizing MySQL Database System on Information Systems Research, Publications and Community Service," *3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, pp. 1-5, 2016. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [4] Jerzy Letkowski, "Doing database design with MySQL," *Journal of Technology Research*, vol. 6, pp. 1-15, 2014. [[Google Scholar](#)] [[Publisher Link](#)]
- [5] Nuruldelmia Idris, Cik Feresa Mohd Foozy, and Palaniappan Shamala, "A Generic Review of Web Technology: Django and Flask," *International Journal of Advanced Computing Science and Engineering*, vol. 2, no. 1, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [6] Mandeep Singh et al., "Implementation of Database Using Python Flask Framework," *International Journal of Engineering and Computer Science*, vol. 8, no. 12, pp. 24894-24899, 2019. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [7] Claudia Audia Trianti, Budhi Kristianto, and Hendry, "Integration of Flask and Python on the Face Recognition Based Attendance System," *2nd International Conference on Innovative and Creative Information Technology (ICITech)*, pp. 164-168, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [8] Ildar Nigmatzyanov et al., "The Client-side System for Management of the Smart Home Climate Control System," *International Seminar on Electron Devices Design and Production (SED)*, pp. 1-6, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [9] Nicole Jillian B. Day et al., "Design of a Web-based and Electronic Health Record Management System for Medical Teleconsultation," *IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, pp. 1-5, 2018. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [10] Kunal Relan, *CRUD Application with Flask (Part 1), Building REST APIs with Flask*, Apress, Berkeley, pp. 59-96, 2019. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [11] Taizhi Lv, Jun Zhang, and Chenyong He, "Research on Posts Analysis Based on Data Process Automation," *2nd International Symposium on Computer Engineering and Intelligent Communications (ISCEIC)*, pp. 156-159, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [12] Zuyi Liang et al., "Data Analysis and Visualization Platform Design for Batteries Using Flask-Based Python Web Service," *34th International Electric Vehicles Symposium and Exhibition*, vol. 12, no. 4, p. 187, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [13] D. J. Samatha Naidu, and R. Lokesh, "Missing Child Identification System using Deep Learning with VGG-FACE Recognition Technique," *SSRG International Journal of Computer Science and Engineering*, vol. 9, no. 9, pp. 1-11, 2022. [[CrossRef](#)] [[Publisher Link](#)]
- [14] Xi Liya, and Pen Yan, "Design and Application of a Development Framework Based-on Shared Farm Digital Control Platform," *E3S Web Conference, International Conference on Agricultural Science and Technology and Food Engineering*, vol. 189, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [15] Xiaoran Wei et al., "A New Lightweight Online Database for Corrosion Rate Analysis of Fluorochemical Engineering Processes," *7th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS)*, pp. 332-337, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [16] Arjun Vaish, "Machine Learning for Asset Wealth Management," *Nanyang Technological University*, 2022. [[Publisher Link](#)]
- [17] M. Shyam Manikanta et al., "Web based E-commerce System Integrated with Chatbot," *International Journal of Research Publication and Reviews*, vol. 3, no 4, pp. 1655-1659, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] Tao Wang, and Yiyu Chen, "Intelligent Form Filling System of Original Invoice Voucher Based on WeChat Applet," *Asia Conference on Computers and Communications*, pp. 120-123, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [19] Wenbin Yang, and Wenbin Yang, "Development and Application of Calligraphy Culture Communication Platform Based on Web Technology," *International Conference on Educational Innovation and Multimedia Technology*, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [20] Sweta Tripathi et al., "I-Voting – Raspberry Pi Based Anywhere Voting System," *SSRG International Journal of Electronics and Communication Engineering*, vol. 7, no. 4, pp. 21-23, 2020. [[CrossRef](#)] [[Publisher Link](#)]
- [21] Liqing Tao et al., "Research on Web Front-End Development Based on Integrated Forest Fire Monitoring System of Communication and Remote Control," *International Conference on Computer Application and Information Security*, vol. 12609, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]