# MULTIPLE TIE-UP BUSINESS MANAGEMENT SYSTEM

## Karthick M[1], Rahul kumar S[2], Rajesh Shanmugam S[3], SivaSubramani T[4], Vimalraj G[5]

[1]Assistant Professor, Department of Computer Science and Engineering, Nandha College of Technology, Perundurai 638 052, Tamilnadu, India.

[2,3,4,5]UG Students – Final Year, Department of Computer Science and Engineering, Nandha College of Technology, Perundurai 638 052, Tamilnadu, India.

## ABSTRACT

Utilising different online solutions for different courier tie-ups can become complicated, confusing, and lead to a lack of uniformity in small to mid scale industries so as in the courier services. The web application is designed to manage multiple courier services through a single platform, eliminating the need for manual pen and paper methods and avoiding the complications of using different online solutions for different courier services. The application was developed using React for the frontend and Django for the backend, with Selenium for web scraping and JWT for authentication. Supabase's PostgreSQL database ensures efficient data management, and the integration of a WhatsApp chat companion makes it easy for users to access courier information quickly and efficiently. This paper describes the development process, the features and functionalities of the web application, and its potential impact on the courier industry.

**Keywords:** Cloud computing, web development, web application, multiple tie-up.

## 1. INTRODUCTION

Introducing the "Multiple Tie-Up Business Management" web application - an innovative solution designed to simplify courier booking and management for small to mid-scale enterprises. Managing multiple courier services can be a challenging task for many businesses. Traditionally, businesses would rely on manual pen and paper methods for recording courier bookings and tracking information. However, as technology advances, such methods become increasingly time-consuming and inefficient. In addition, utilising different online solutions for different courier tie-ups can become complicated, confusing, and lead to a lack of uniformity in courier management. To address these issues, the "Multiple Tie-Up Business Management" web application was created. This application offers businesses a single platform to book and manage multiple courier services. The application simplifies the courier booking process, allowing users to book Akash and Anjani couriers seamlessly. The application was designed using React for the frontend and Django for the backend. Selenium was used for web scraping, while JWT was used for authentication. Supabase's PostgreSQL database ensures efficient data management, and the integration of a WhatsApp chat companion makes it easy for users to access courier information quickly and efficiently. With the "Multiple Tie-Up Business Management" web application, businesses can eliminate the need for manual pen and paper methods and avoid the complications of using different online solutions for different courier services. This results in a streamlined and efficient courier management process that saves businesses valuable time and resources. Overall, the "Multiple Tie-Up Business Management" web application offers a comprehensive solution to the challenges faced by businesses that use multiple courier services. Its intuitive user interface, robust features, and efficient technologies make it an ideal tool for any enterprise looking to streamline its courier management process.

## 2. BACKGROUND

Managing multiple courier services can be a challenging task for many small to mid-scale enterprises. Traditionally, businesses would rely on manual pen and paper methods for recording courier bookings and tracking information, or use different websites for each courier tie-up, which could be time-consuming and lead to a lack of uniformity in courier management. The "Multiple Tie-Up Business Management" web application was developed to address these issues. The process involved confirming which courier company the courier should be booked with and filling in different bills for each of them. The major challenge faced by businesses that use multiple courier services is working with different courier services at once, which can lead to confusion and mistakes in booking couriers in the wrong company. Additionally, managing records and keeping track of courier bills can be complicated and time-consuming. The "Multiple Tie-Up Business Management" web application simplifies courier booking and management for businesses by offering a single platform to book and manage multiple courier services. The application was designed using React for the frontend and Django for the backend. Selenium was used for web scraping, while JWT was used for authentication. Supabase's PostgreSQL database ensures efficient data management, and the integration of a WhatsApp chat companion makes it easy for users to access courier information quickly and efficiently. The benefits of using a single platform to manage multiple courier services include streamlined and efficient courier management,

avoiding the complications of using different online solutions for different courier services. The project also includes a credit system that takes care of the details to be sent to the mail as an excel sheet of all the bookings. The COVID-19 pandemic has impacted the courier industry, and the "Multiple Tie-Up Business Management" web application addresses the challenges faced by businesses during this time by providing a contactless way to manage courier bookings and records. The application can be accessed from any device and from anywhere, making it ideal for remote work and social distancing.

## 3. DESIGN AND IMPLEMENTATION

- The project uses a RESTful API design, with different endpoints for different functionalities such as creating a new booking, retrieving booking details, and updating booking information.
- The frontend is designed using React, which provides an intuitive and easy-to-use interface for the user. The use of components in React enables the creation of reusable code, which makes development faster and more efficient.
- Redux is used for state management in the frontend, which helps to manage the different states of the application and enables easy communication between different components.
- Django is used for the backend API, which provides a powerful and flexible framework for building web applications. Django's built-in ORM is used to interact with the database, making it easy to perform database operations and manipulate data.
- The project uses PostgreSQL as the database management system, which is known for its scalability, reliability, and security.
- The web scraping for courier information is done using Selenium, a powerful tool for automating web browsers. This enables the project to gather information about courier services from different websites and display it in a unified manner in the application.
- Authentication is done using JWT, which provides a secure and efficient way to authenticate users and protect the application from unauthorised access.
- The project also includes a WhatsApp chat companion, which enables users to access courier information quickly and efficiently using their mobile devices.

Overall, the project design and implementation have been carefully planned to provide a streamlined and efficient courier management system that is easy to use and flexible enough to handle the needs of small to mid-sized enterprises.
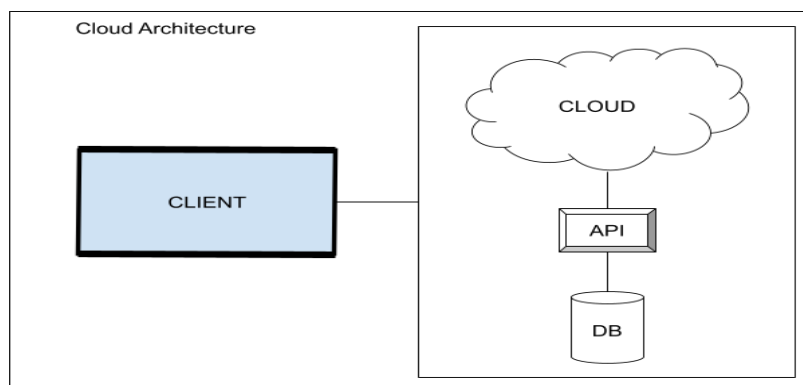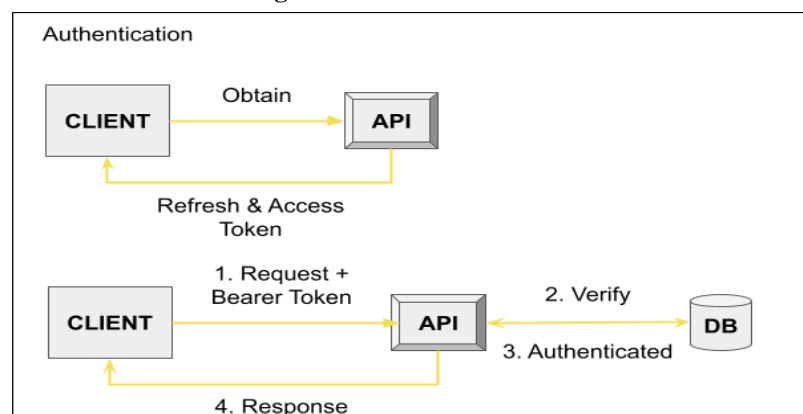


**Figure. 1 Cloud architecture**

**Figure. 2 JWT authentication**

## 4. KEY COMPONENTS

The functionality of "Multiple Tie-Up Business Management" web application includes:

- Simplified courier booking: The application provides businesses with a single platform to book and manage multiple courier services.

- Tile-based booking: Booking a courier is simplified by pre-populated tiles that already contain common information about the courier, such as the company name and courier number.

- Editable records: Users can easily edit and manage their courier booking records through the application's user interface.

- Credit system: The application provides users with a credit system that keeps track of the details of all the bookings made and sends them to the user's email as an excel sheet.

- WhatsApp companion: The application provides users with a WhatsApp companion that allows for easy access to courier information.

- Automated web scraping: The application uses Selenium for web scraping, which automatically retrieves the relevant courier information from the official courier websites.

- JWT-based authentication: The application uses JWT for authentication, ensuring secure access to the user's courier booking records.

- Database management: The application uses Django's ORM to interact with the Supabase's PostgresQL database for efficient data management.

- Intuitive user interface: The application's user interface is designed to be intuitive and easy to use, providing businesses with a streamlined and efficient courier management process.

Overall, the "Multiple Tie-Up Business Management" web application provides businesses with a comprehensive solution to the challenges of managing multiple courier services. Its robust features, efficient technologies, and intuitive user interface make it an ideal tool for any enterprise looking to streamline its courier management process.
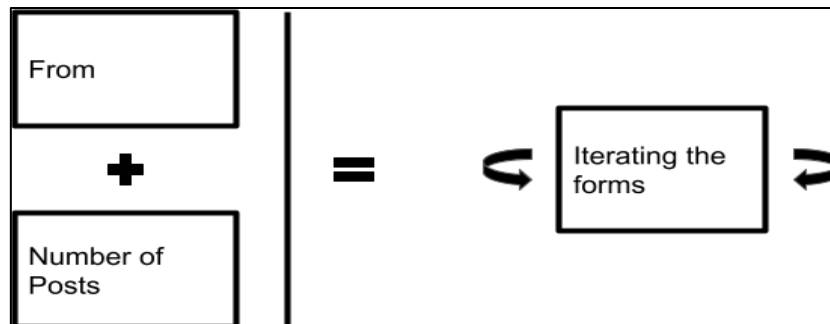
**Figure. 3 Credit information flow**
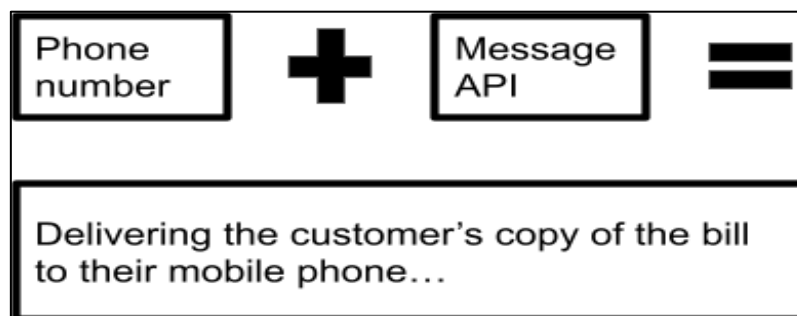
**Figure. 4 Cash information flow**

## 5. CONCLUSION

The "Multiple Tie-Up Management" web application is a solution designed to simplify courier booking and management for small to mid-scale enterprises. The project was born out of the need to streamline the tedious process of booking couriers for multiple courier services, which are outdated as well. Traditionally, businesses had to use

different books or websites for each courier service, making the process time-consuming and inefficient. The "Multiple Tie-Up Management" project aims to provide a one-stop solution accessible from any device and location. This project was implemented using React for the frontend and Django for the backend API. Th application offers a range of features, including component-based UI, state management using react-redux-toolkit, routing using react router dom, and styling using styled components. Authentication was implemented using JWT, while Supabase's PostgresQL database was used for efficient data management. The web application provides a range of functionalities, including booking and managing multiple courier services, record management, credit system, and a WhatsApp companion for easy access to courier information. The editable records feature was implemented using React JSX, while the web scraper was implemented using Selenium. The Django ORM was used to work with the Supabase's PostgresQL database.

## 6. REFERENCES

[1] Django Software Foundation. (n.d.). Django documentation. Retrieved March 13, 2023, from https://docs.djangoproject.com/en/3.2/

[2] React. (n.d.). React – A JavaScript library for building user interfaces. Retrieved March 13, 2023, from https://reactjs.org/

[3] RFC 7519: JSON Web Token (JWT) - This is the official specification for JSON Web Tokens, published by the Internet Engineering Task Force (IETF). https://jwt.io

[4] Redux Toolkit with React - The official documentation for using Redux Toolkit with React https://redux-toolkit.js.org/tutorials/intermediate-tutorial