# Industrial Internship Report on

# ”Food Delivery App”

## Prepared by

## Viren Hadawale

*Executive Summary*

This report provides details of the Industrial Internship provided by **upSkill Campus** and **The IoT Academy** in collaboration with Industrial Partner **UniConverge Technologies Pvt Ltd (UCT)**.

This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 4 weeks’ time.

**My Project: Food Delivery App with Real-Time Tracking** I developed a full-stack web application designed to streamline the interaction between Restaurant Owners, Delivery Drivers, and Customers. The system utilizes Python **(**FastAPI**)** for a high-performance backend, React**.**js for a responsive frontend, and PostgreSQL for data management. Key features include role-based access control, real-time GPS tracking using WebSockets, and secure order verification via OTP (SMS).

This internship gave me a very good opportunity to get exposure to Industrial problems—specifically in logistics and real-time data handling—and design/implement a solution for that. It was an overall great experience to have this internship.

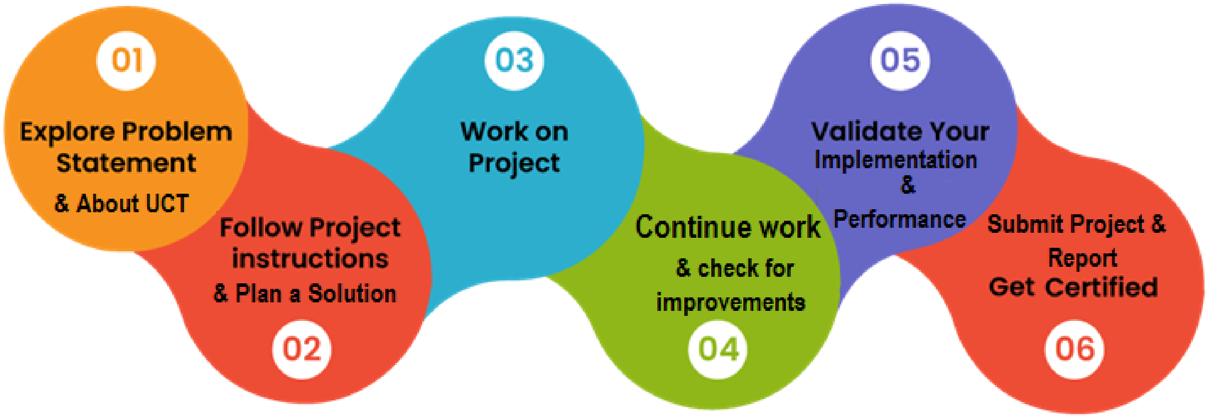
**TABLE OF CONTENTS**

# 1 Preface

Over the course of 4 weeks, I transitioned from understanding the theoretical requirements of a modern delivery system to deploying a functional full-stack application.

This internship was crucial for understanding how software is built in a professional environment.

The internship began with requirement gathering: understanding how to manage three distinct user roles (Owner, Driver, Customer). It progressed into database modeling using SQLAlchemy and API development using FastAPI. The most challenging yet rewarding phase was implementing the "Live Tracking" feature, which required moving beyond standard HTTP requests to persistent WebSocket connections.



I would like to thank the mentors at **UniConverge Technologies** and **upSkill Campus** for providing the platform and guidance. This project bridged the gap between my academic knowledge of Python/JavaScript and industry-standard practices like asynchronous programming and secure authentication (JWT).

# 2 Introduction

## 2.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

uct

Uniconverge Technologies

**IIOT Products**

We offer product ranging from Remote IOs, Wireless IOs, LoRaWAN Sensor Nodes/ Gateways, Signal converter and IoT gateways

**IIOT Solutions**

We offer solutions like OEE, Predictive Maintenance, LoRaWAN based Remote Monitoring, IoT Platform, Business Intelligence…

**OEM Services**

We offer solutions ranging from product design to final production we handle everything for you..

## i. UCT IoT Platform (uct Insight)

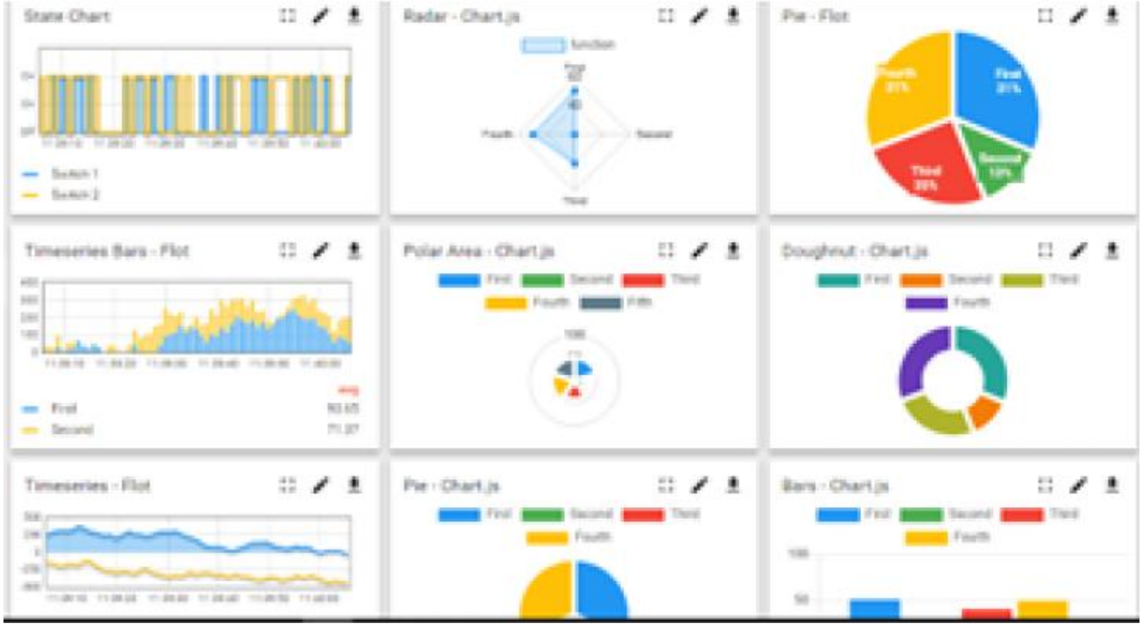
**UCT Insight** is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

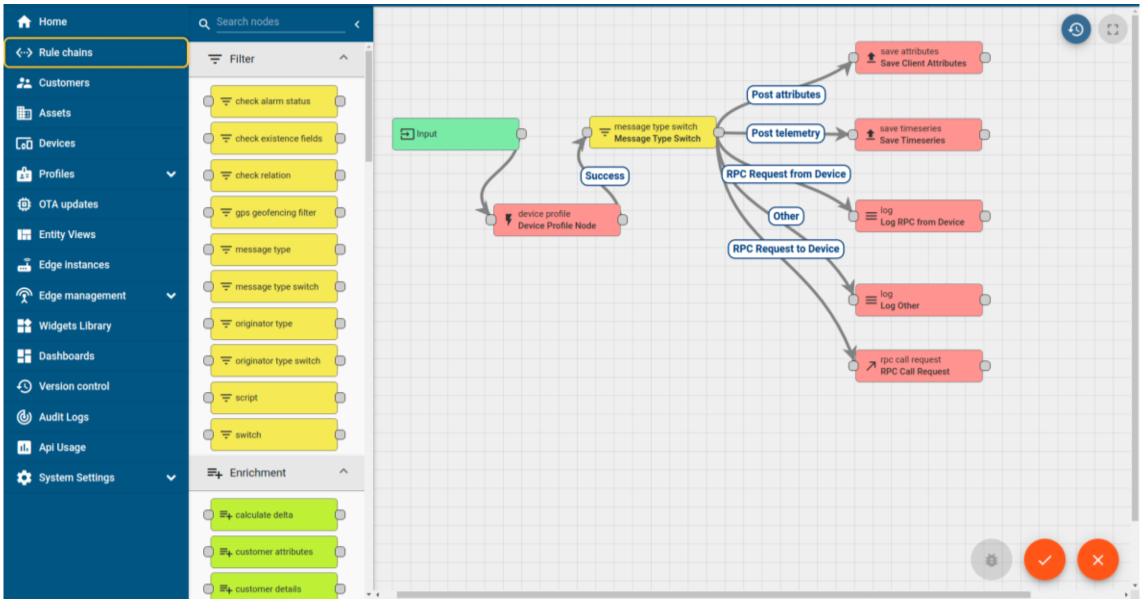
- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA

- It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine





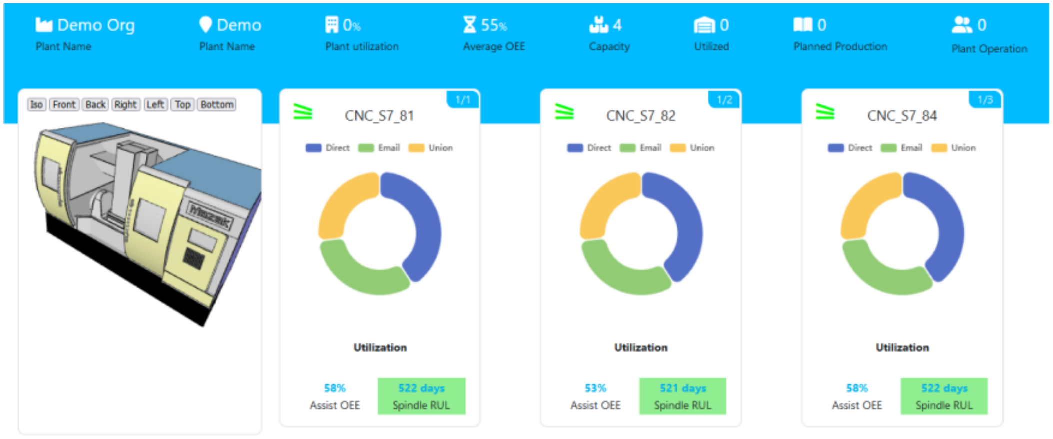
## ii. Smart Factory Platform ( FACTORY WATCH )

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.



Demo Org
Plant Name

Demo
Plant Name

0%
Plant utilization

55%
Average OEE

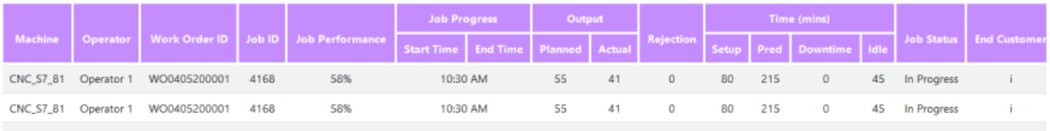
4
Capacity

0
Utilized

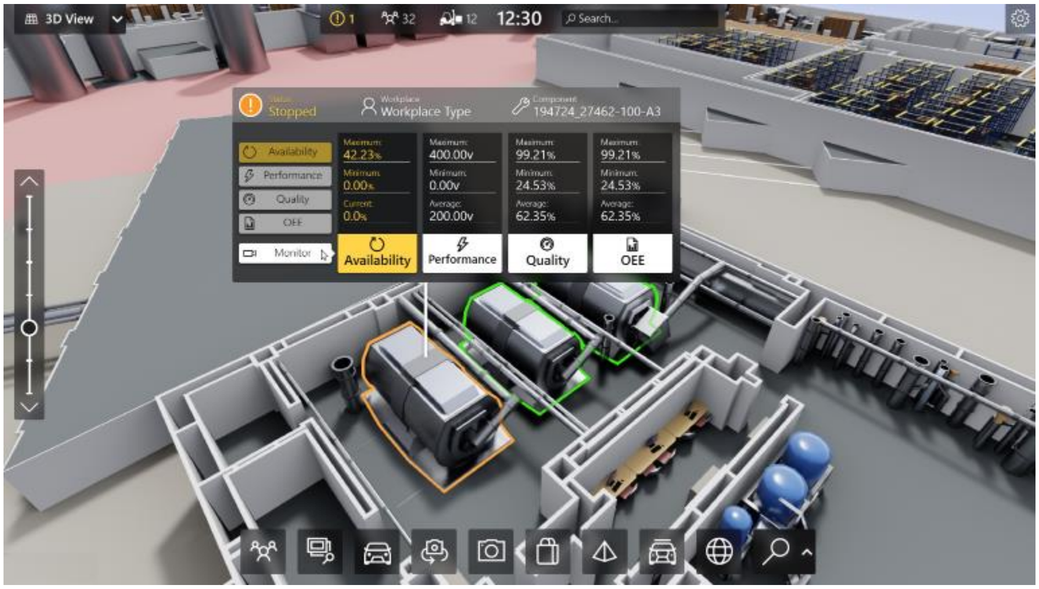
0
Planned Production

0
Plant Operation

| Machine | Assist OEE | Spindle RUL |
|---|---|---|
| CNC_S7_81 | 58% | 522 days |
| CNC_S7_82 | 53% | 521 days |
| CNC_S7_84 | 58% | 522 days |



| Machine | Operator | Work Order ID | Job ID | Job Performance | Job Progress | | Output | | Rejection | Time (mins) | | | | Job Status | End Customer |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Start Time | End Time | Planned | Actual | | Setup | Pred | Downtime | Idle | | |
| CNC_S7_81 | Operator 1 | WO0405200001 | 4168 | 58% | 10:30 AM | | 55 | 41 | 0 | 80 | 215 | 0 | 45 | In Progress | i |
| CNC_S7_81 | Operator 1 | WO0405200001 | 4168 | 58% | 10:30 AM | | 55 | 41 | 0 | 80 | 215 | 0 | 45 | In Progress | i |



Status: Stopped
Workplace: Workplace Type
Component: 194724_27462-100-A3

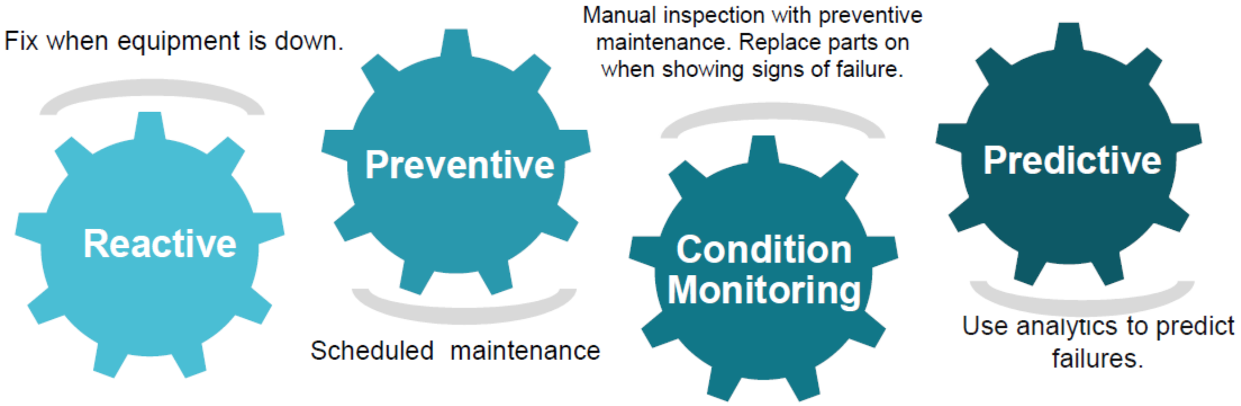
| Availability | Performance | Quality | OEE |
|---|---|---|---|
| Maximum: 42.23% | Maximum: 400.00v | Maximum: 99.21% | Maximum: 99.21% |
| Minimum: 0.00% | Minimum: 0.00v | Minimum: 24.53% | Minimum: 24.53% |
| Current: 0.0% | Average: 200.00v | Average: 62.35% | Average: 62.35% |

### iii. LoRaWAN based Solution

UCT  is one of the early adopters of LoRAWAN technology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

### iv. Predictive Maintenance

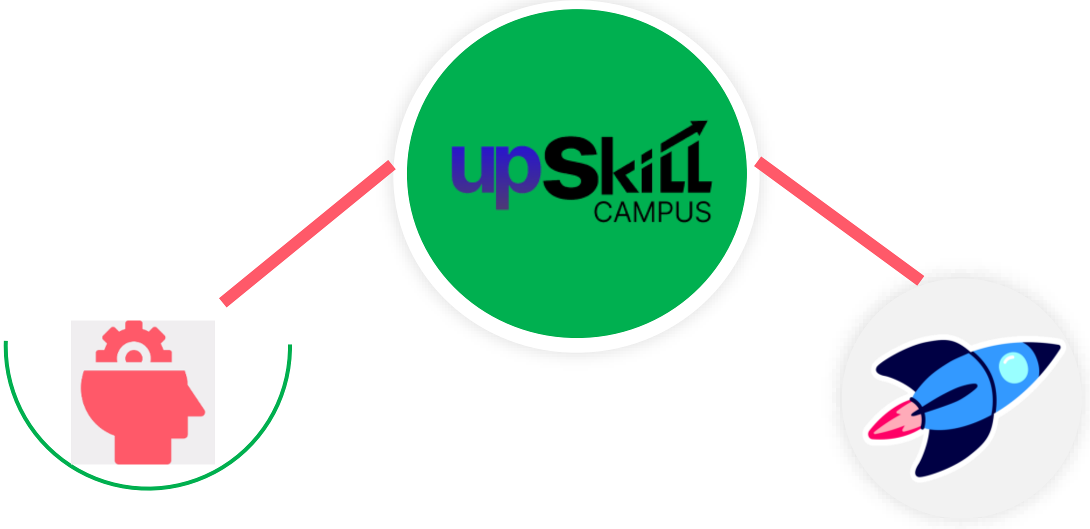
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.


Fix when equipment is down.
Reactive
Preventive
Scheduled  maintenance
Manual inspection with preventive maintenance. Replace parts on when showing signs of failure.
Condition Monitoring
Predictive
Use analytics to predict failures.

## 2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

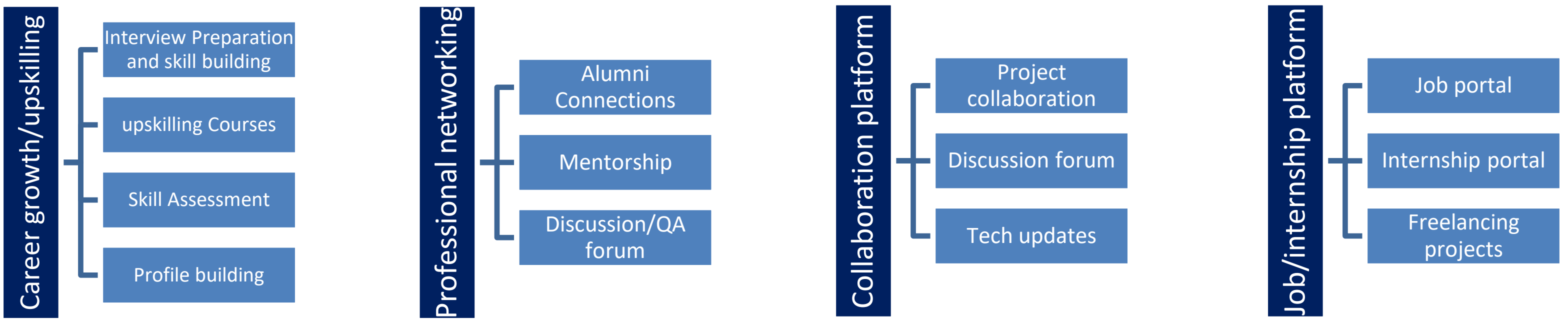
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



Career growth/upskilling
- Interview Preparation and skill building
- upskilling Courses
- Skill Assessment
- Profile building

Professional networking
- Alumni Connections
- Mentorship
- Discussion/QA forum

Collaboration platform
- Project collaboration
- Discussion forum
- Tech updates

Job/internship platform
- Job portal
- Internship portal
- Freelancing projects

## 2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

## 2.4 Objectives of this Internship program

The objective for this internship program was to

- To get practical experience working with full-stack web technologies (React & FastAPI).
- To solve real-world logistics problems involving real-time data.
- To improve understanding of database management and API security.
- To enhance personal growth in debugging and system architecture design.

## 2.5 Reference

[1] **FastAPI Documentation:** https://fastapi.tiangolo.com/

[2] **React.js Documentation:** https://react.dev/

[3] **MDN Web Docs (WebSockets**): https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API

## 2.6 Glossary

| Terms | Acronym |
|---|---|
| **API** | Application Programming Interface (The bridge between Frontend and Backend) |
| **JWT** | JSON Web Token (Used for secure user authentication) |
| **WebSocket** | A protocol for full-duplex communication channels (Used for live tracking) |
| **ORM** | Object-Relational Mapping (SQLAlchemy - interacting with DB using Python) |
| **OTP** | One-Time Password (Used for secure delivery verification) |
| **DBMS** | Database Management System (PostgreSQL) |

# 3 Problem Statement

In the unorganized food delivery sector, small restaurant owners face significant challenges:

1. **Lack of Visibility:** Owners cannot track where their drivers are once they leave the restaurant.
2. **Customer Anxiety:** Customers constantly call the restaurant to ask, "Where is my food?" because they lack live updates.
3. **Security Gaps:** Orders are often handed to the wrong person or marked delivered without actual handover.
4. **Data Fragmentation:** Sales data, menu management, and order history are often manual or scattered across different tools.

The project aim was to build a **Unified Food Delivery System** that centralizes these operations, provides transparency via live tracking, and secures the handover process.

# 4 Existing and Proposed solution

**Existing Solutions:** Currently, large aggregators (like Zomato/Swiggy) exist but charge high commissions (20-30%). Small businesses often resort to taking orders via phone calls or WhatsApp, which is unscalable and lacks tracking.

**Proposed Solution:** I developed a proprietary "Food Delivery App" that allows restaurant owners to manage their own fleet and orders without high commissions.

**Value Addition:**

1. **Hybrid Communication:** Uses WebSockets for live map updates (free, fast) and SMS (Twilio) for critical alerts (OTP, Confirmation).
2. **Role-Based Dashboards:** Specific interfaces for Owners (Analytics), Drivers (Navigation), and Customers (Ordering).
3. **Secure Handover:** A delivery is only marked "Complete" when the driver enters the correct 4-digit OTP provided by the customer.

## 4.1 Code submission

- GitHub Repository: github.com/Viren021/upskillcampus

## 4.2 Report submission

- Report Link: github.com/Viren021/upskillcampus/FoodDeliveryApp_Viren_USC_UCT.pdf

# 5 Proposed Design/ Model

The system follows a **Client-Server Architecture**. The frontend (Client) is built with React.js, which communicates with the backend (Server) via REST API for standard data and WebSockets for real-time streams.

## 5.1 High Level Diagram

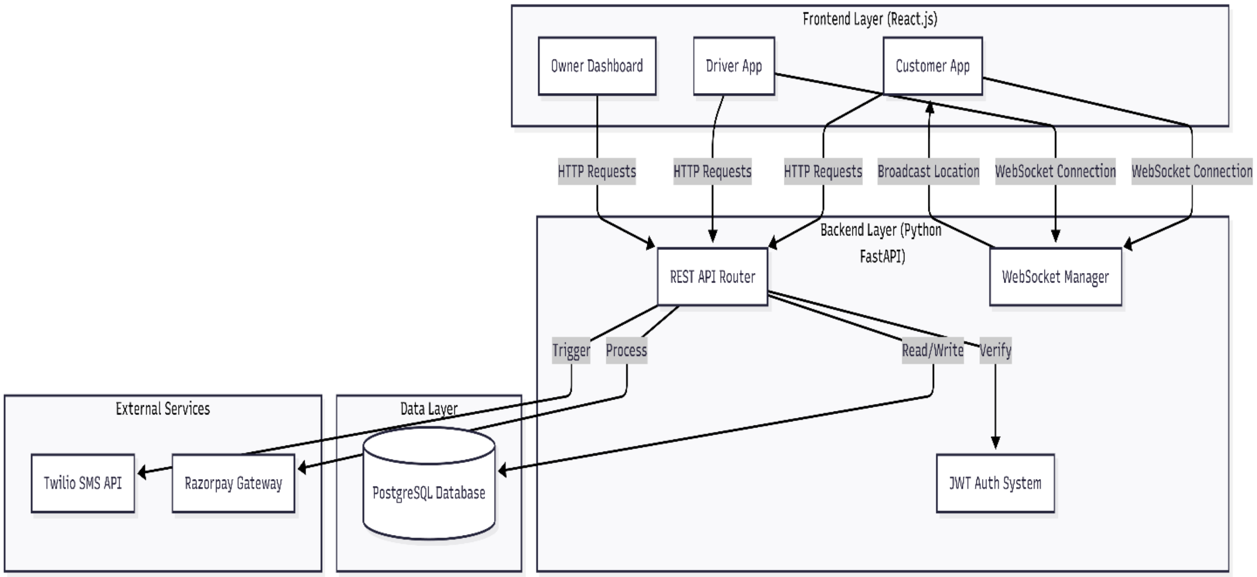


**Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM**

## 5.2 Low Level Diagram

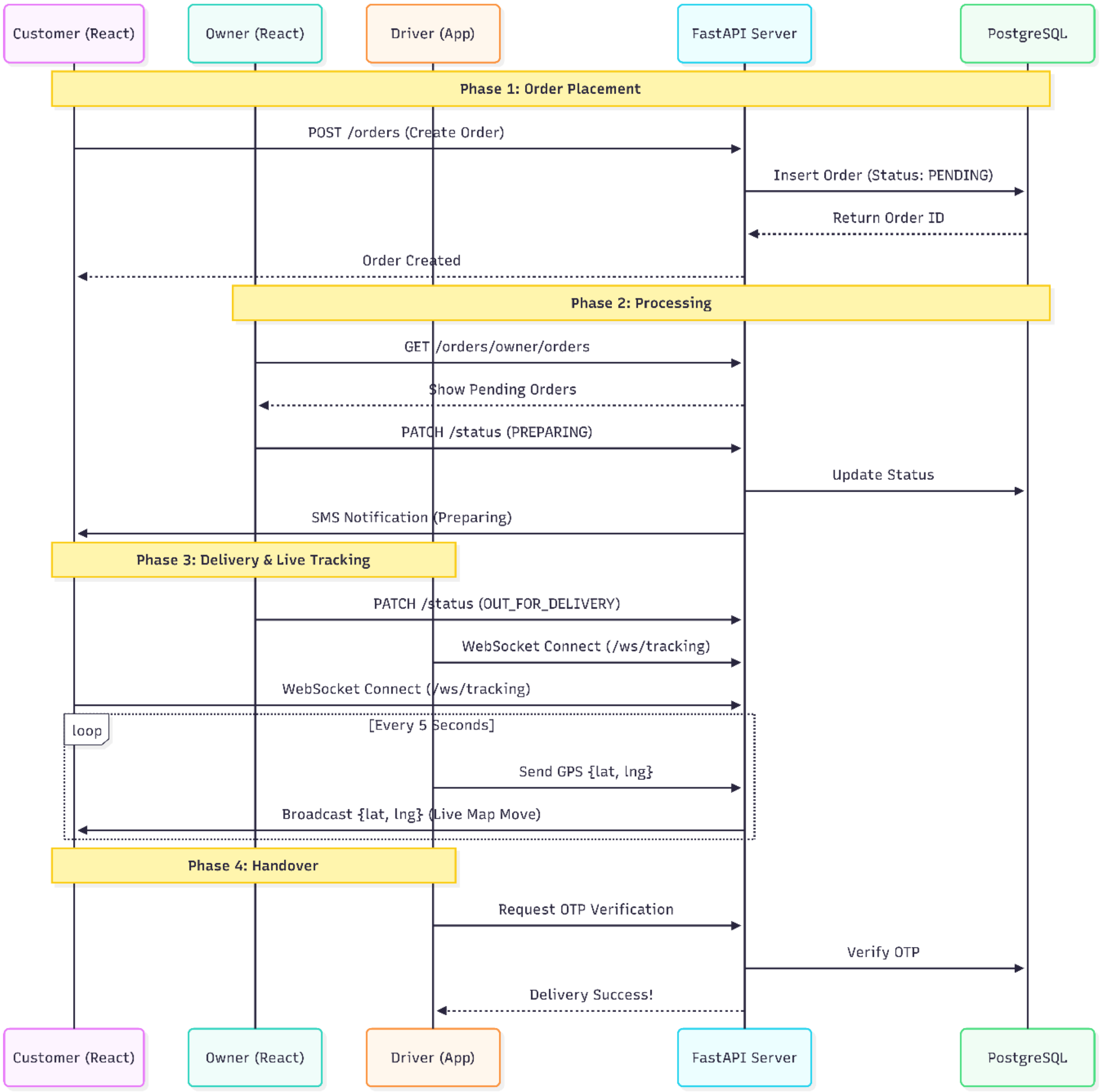


**Figure 2: LOW LEVEL DIAGRAM OF THE SYSTEM**

## 5.3 Interfaces

- **REST API:** Endpoints for POST /auth/login, GET /orders, POST /orders/{id}/driver-location.
- **Database:** PostgreSQL tables for Users, Restaurants, Orders, and OrderItems.
- **WebSocket:** Endpoint /ws/tracking for real-time coordinate exchange.

# 6 Performance Test

The application was tested to ensure it can handle real-world constraints such as network latency and concurrent data updates.

## 6.1 Test Plan/ Test Cases

1. **Authentication:** Verify that Drivers cannot access Owner dashboards (Role-Based Access Control).
2. **Concurrency:** Test if 2 users can order simultaneously without DB locking.
3. **Tracking Latency:** Measure the delay between Driver update and Customer map movement.
4. **OTP Security:** Ensure order cannot be closed with an incorrect OTP.

## 6.2 Test Procedure

- **Unit Testing:** Used Python's pytest to check API endpoints.
- **Manual Testing:** Opened the "Driver App" on a mobile phone and "Customer App" on a laptop to verify live movement synchronization.
- **Load Simulation:** Simulating multiple WebSocket connections to check server stability.

## 6.3 Performance Outcome

- **Latency:** WebSocket updates were received in under **200ms**, providing a smooth "live" feel.
- **Security:** The JWT middleware successfully blocked unauthorized access to the Admin panel.
- **Database:** SQLAlchemy queries were optimized using selectinload to prevent "N+1" query performance issues.

# 7 My learnings

This internship was pivotal in transforming my theoretical knowledge into practical skills:

1. **Full Stack Integration:** I learned how to connect a React Frontend to a Python Backend, handling CORS issues and data serialization.

2. **Asynchronous Programming:** I mastered async/await in Python to handle multiple database and WebSocket requests without blocking the server.

3. **Real-Time Architecture:** I understood the difference between HTTP Polling (slow) and WebSockets (fast) and when to use each.

4. **Debugging:** I gained significant experience in reading server logs, debugging HTTP 401/500 errors, and fixing frontend state management issues.

## 8 Future work scope

Due to the 4-week time constraint, some features can be expanded in the future:

1. **Machine Learning Recommendations:** Implementing the TF-IDF algorithm (which is currently in the code) to provide personalized food suggestions based on order history.
2. **Payment Gateway Integration:** Fully activating the Razorpay integration for live real-money transactions.
3. **Mobile Native App:** Converting the React web frontend into React Native for a better mobile experience for drivers.