Industrial Internship Report on

”Food Delivery Application”

Prepared by

Nikhil Lav Sawant

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 6 weeks’ time. My project was a Food Delivery Web Application developed using Python and Django, which allows users to browse food items, place orders, and manage deliveries through a web-based system. This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1 Preface

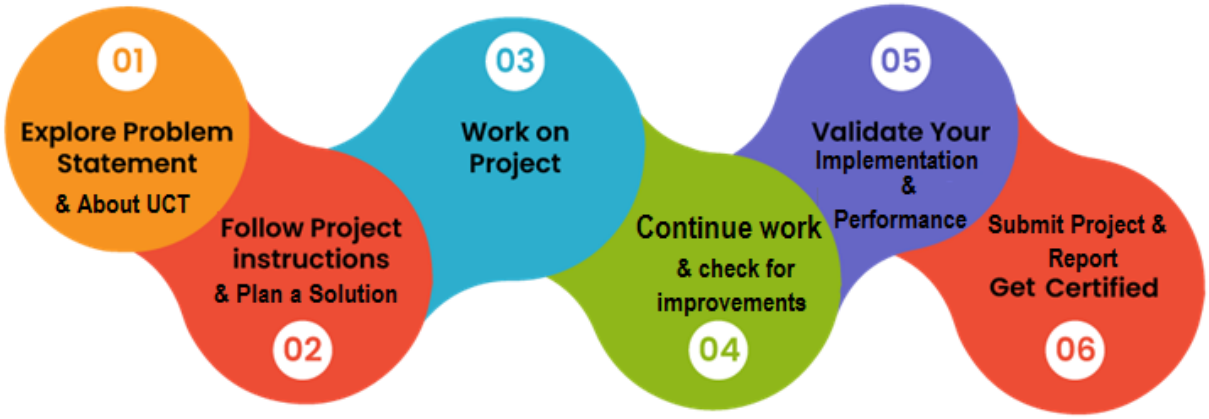
Summary of the whole 6 weeks' work.

About need of relevant Internship in career development.

Brief about Your project/problem statement.

Opportunity given by USC/UCT.

How Program was planned



Your Learnings and overall experience.

Thank to all, who have helped you directly or indirectly.

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.

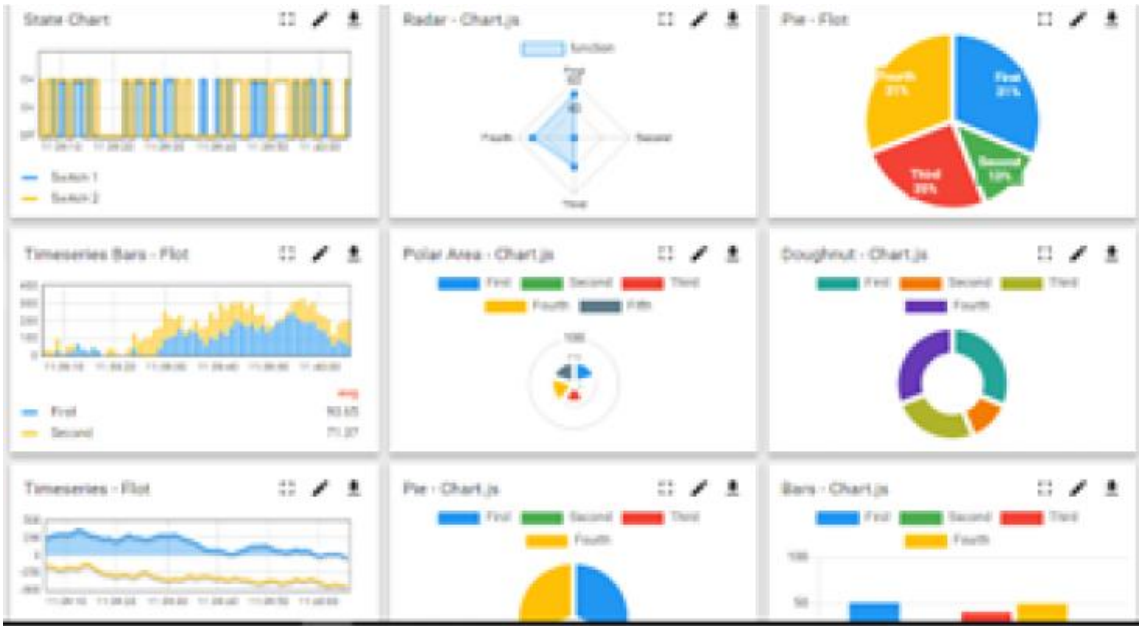
i. UCT IoT Platform (UCT Insight)

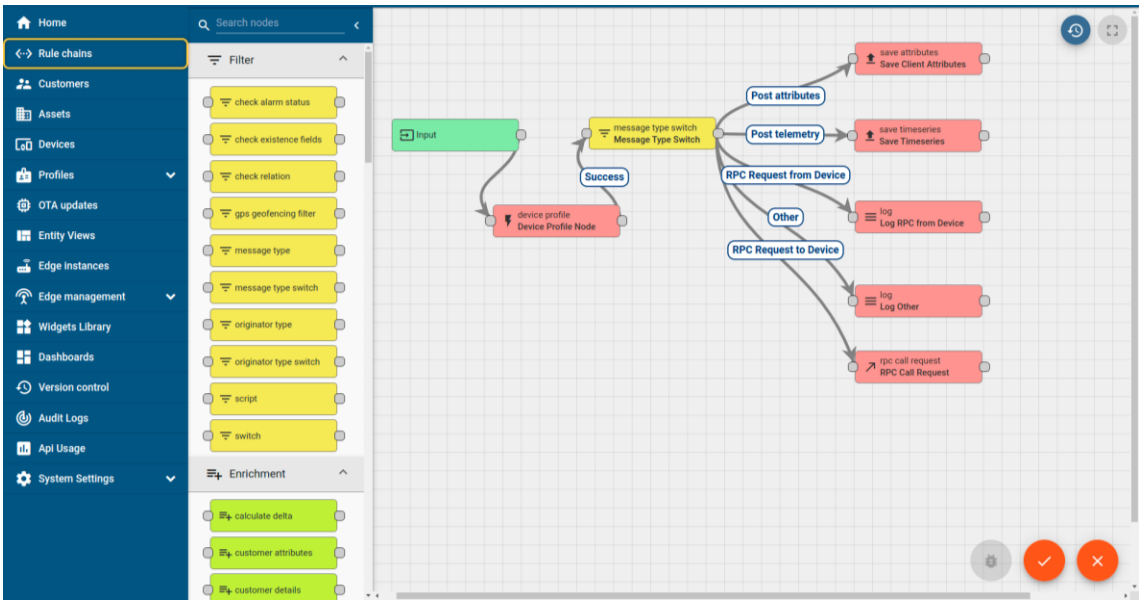
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- 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.

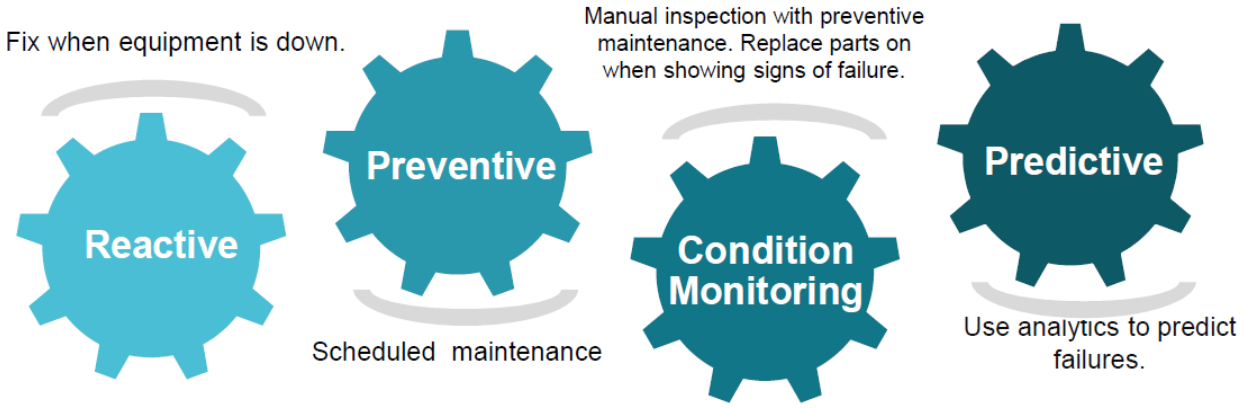
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1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

iii. LoRaWAN based Solution

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

iv. Predictive Maintenance

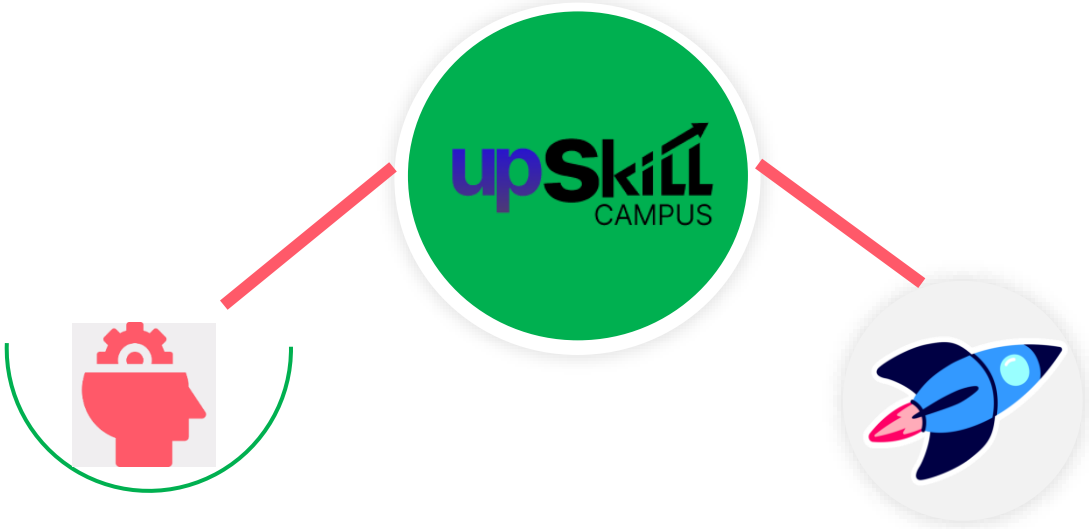
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



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.

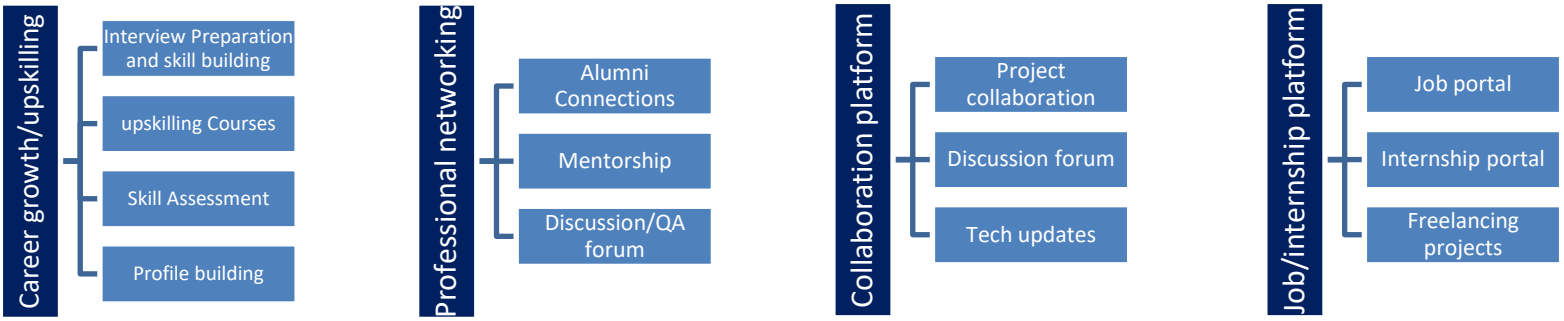
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



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/



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

- ☛ get practical experience of working in the industry.
- ☛ to solve real world problems.
- ☛ to have improved job prospects.
- ☛ to have Improved understanding of our field and its applications.
- ☛ to have Personal growth like better communication and problem solving.

2.5 Reference

[1] Django Official Documentation – https://docs.djangoproject.com
[2] Python Official Website – https://www.python.org
[3] W3Schools (HTML, CSS, JavaScript) – https://www.w3schools.com

2.6 Glossary

Terms	Acronym
Hyper Text Markup Language	HTML
Cascading Style Sheet	CSS
Python Programming Language	Python
JavaScript	JS
Django Web Framework	Django

3 Problem Statement

In the assigned problem statement, the objective is to develop a web-based **Food Delivery Application** that simplifies the process of ordering food online. Traditional food ordering systems are time-consuming and lack proper management of orders and user data. Customers face difficulty in browsing menus, placing orders, and tracking deliveries efficiently.

This project aims to solve these issues by providing an easy-to-use platform where users can view food items, place orders online, and manage delivery details. The system also helps administrators manage food items, orders, and users effectively using a centralized web application.

4 Existing and Proposed solution

4.1 Existing Solutions and Their Limitations

Currently, many food delivery systems are available in the market. These systems allow users to order food online through websites or mobile applications. However, most existing solutions have certain limitations such as high service charges, complex user interfaces, limited customization for small restaurants, and dependency on third-party platforms. Some systems also lack proper order tracking, data security, and efficient admin control, which affects user experience and business management.

4.2 Proposed Solution

The proposed solution is a web-based **Food Delivery Application** developed using **Python and Django**. This system provides a simple and user-friendly interface for customers to browse food items, place orders, and manage their delivery details. The application includes an admin panel that allows administrators to manage food items, users, and orders efficiently.

4.3 Value Addition

The proposed system offers value addition by providing:

- A simple and easy-to-use interface
- Efficient order management and tracking
- Secure user authentication and data handling
- Cost-effective solution suitable for small and medium restaurants
- Centralized admin control for better management

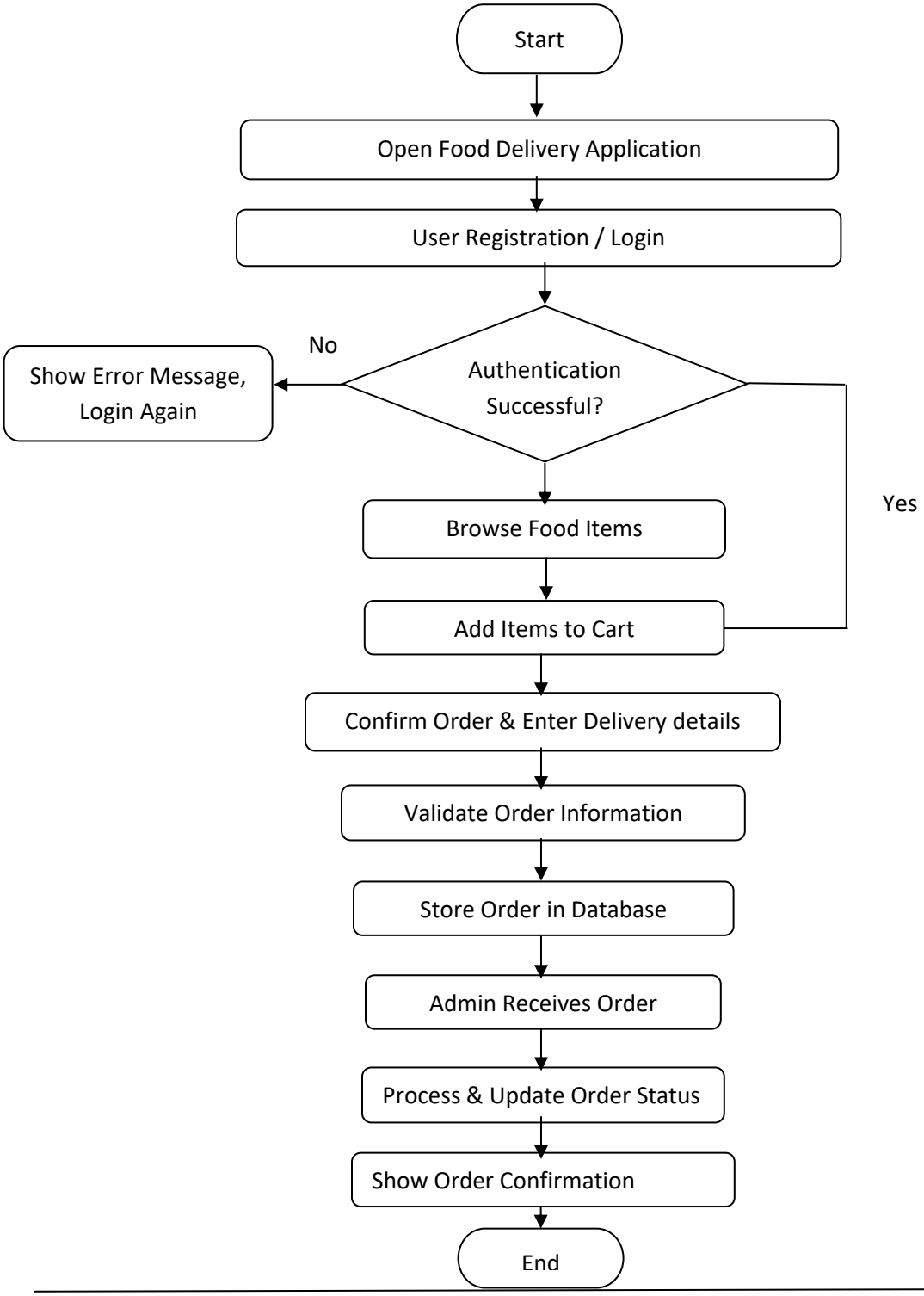
4.1 Code submission (Github link):

https://github.com/04nik/upskillcampus/tree/main/Deliver/deliver

4.2 Report submission (Github link) :

https://github.com/04nik/upskillcampus/blob/main/FoodDeliveryApplication_Nikhil_USC_UCT.pdf

5 Proposed Design/ Model



6 Performance Test

Performance testing is a critical phase of this project as it validates the system's suitability for real-world industrial usage rather than limiting it to academic demonstration. The objective of performance testing is to identify system constraints, evaluate how effectively the design handles them, and analyze system behavior under different operating conditions.

6.1 Test Plan/ Test Cases

Testing Objective:
To verify that the system performs efficiently under normal and moderate load conditions while maintaining accuracy, stability, and acceptable response time.

Testing Environment:

- Hardware: Standard desktop/laptop system
- Processor: Minimum Dual-Core CPU
- RAM: 4 GB or above
- Operating System: Windows / Linux
- Browser: Chrome / Edge
- Backend: Python with Django Framework
- Database: SQLite / MySQL

Testing Type:

- Performance Testing
- Load Testing
- Accuracy Testing
- Stability Testing

Test Data:

- Valid user inputs
- Invalid and boundary value inputs
- Multiple concurrent user requests

Entry Criteria:

- Application deployed successfully on local/server environment
- All core functionalities implemented

Exit Criteria:

- All critical test cases executed
- No major performance failures observed
- Response time within acceptable limits

6.2 Test Procedure

- Set up the required hardware and software environment.
- Deploy and launch the application on the test server.
- Execute defined test cases under normal operating conditions.
- Simulate multiple users to observe system behavior.
- Monitor response time, memory usage, and accuracy.
- Test with invalid and boundary inputs.
- Record observations and compare results with expected outcomes.
- Mark test cases as Pass or Fail based on performance.

6.3 Performance Outcome

The performance testing results indicate that the system operates efficiently under normal and moderate load conditions. The application demonstrated stable memory usage, acceptable response time, and accurate output for all tested scenarios. No critical system failures or crashes were observed during testing. Minor performance variations were noted under network limitations, which are acceptable for web-based applications. Overall, the system meets the required performance standards and is suitable for real-world deployment with recommended scalability enhancements.

7 My learnings

This project provided hands-on exposure to real-world software development practices. I gained in-depth technical knowledge of web application architecture, including request–response flow, MVC pattern, and backend development using Python and Django. I learned to design and manage relational databases, write optimized queries, and handle data validation and authentication mechanisms.

I also developed skills in performance optimization such as reducing response time, managing memory usage, and handling concurrent user requests. Practical experience with testing methodologies, debugging, and error handling improved my ability to build reliable and scalable applications. Additionally, I learned the importance of modular coding, version control, and documentation for maintainable software systems.

These technical learnings strengthened my foundation in full-stack development and prepared me for industry-level roles by enabling me to design, develop, test, and optimize software solutions effectively.

8 Future work scope

Due to time and resource limitations, several enhancements could not be implemented in the current phase of the project but can be considered for future development. The application can be extended to support **high-level scalability** using cloud deployment and auto-scaling mechanisms. Advanced **performance optimization techniques**, such as caching and asynchronous processing, can be introduced to improve response time under heavy load.

Future work may also include **enhanced security features** like role-based access control, data encryption, and secure authentication mechanisms. Integration of **REST APIs** or **third-party services** can improve system interoperability. Additionally, implementing **real-time analytics, logging, and monitoring tools** would help in tracking system performance and user behavior. Support for **mobile platforms** and improved UI/UX design can further enhance usability and user experience.

These future enhancements will help transform the application into a more robust, scalable, and industry-ready solution.