Industrial Internship Report on

"Smart Expense Tracker"

Prepared by

[Gautham.B]

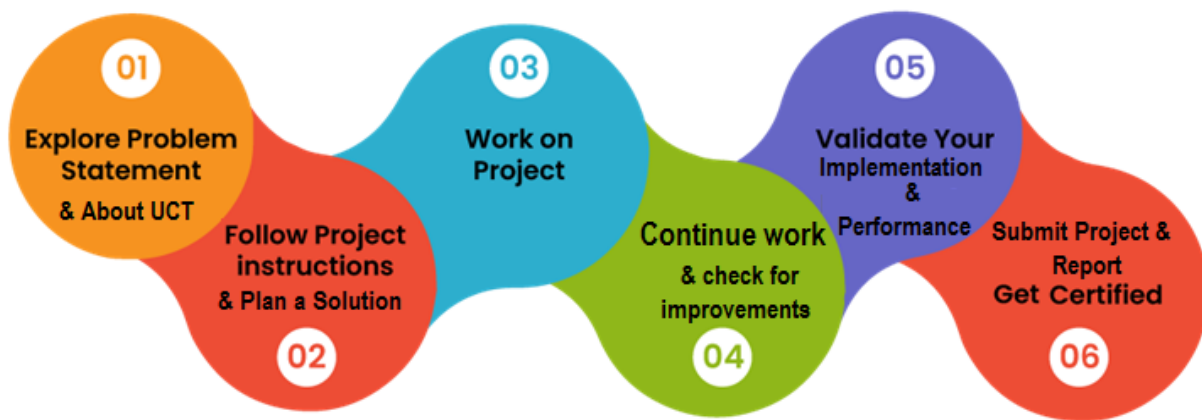| Executive Summary |
| --- |
| This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT). |
| This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks' time. |
| My project was **"Smart Expense Tracker – A Privacy-Focused Personal Finance Management System."** The project involved designing and developing a web-based application to help users track, analyze, and manage their daily expenses. The system works offline, stores data securely using browser LocalStorage, and provides features such as expense categorization, budget tracking, visual analytics, security lock, and report generation. |
| 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 (with names), who have helped you directly or indirectly.

Your message to your juniors and peers.

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 (FACT⌀RY 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.

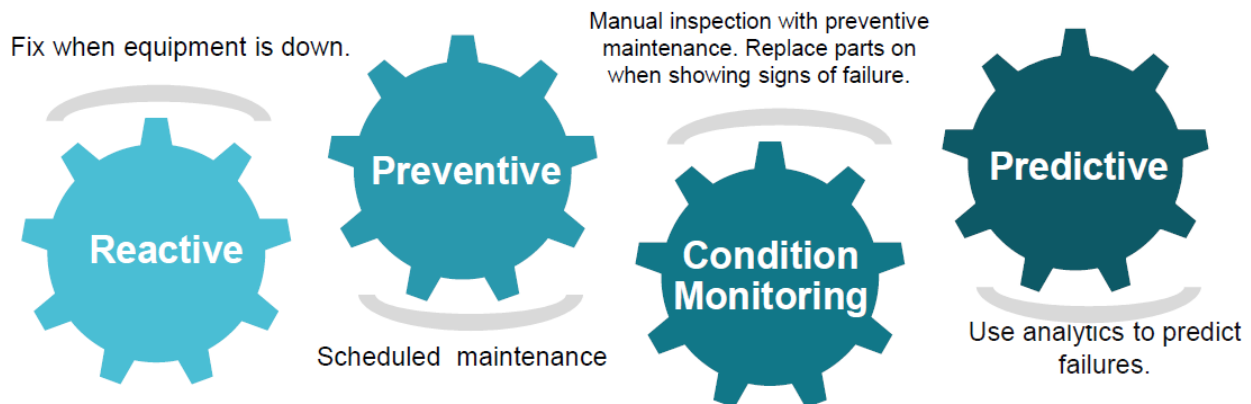| Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator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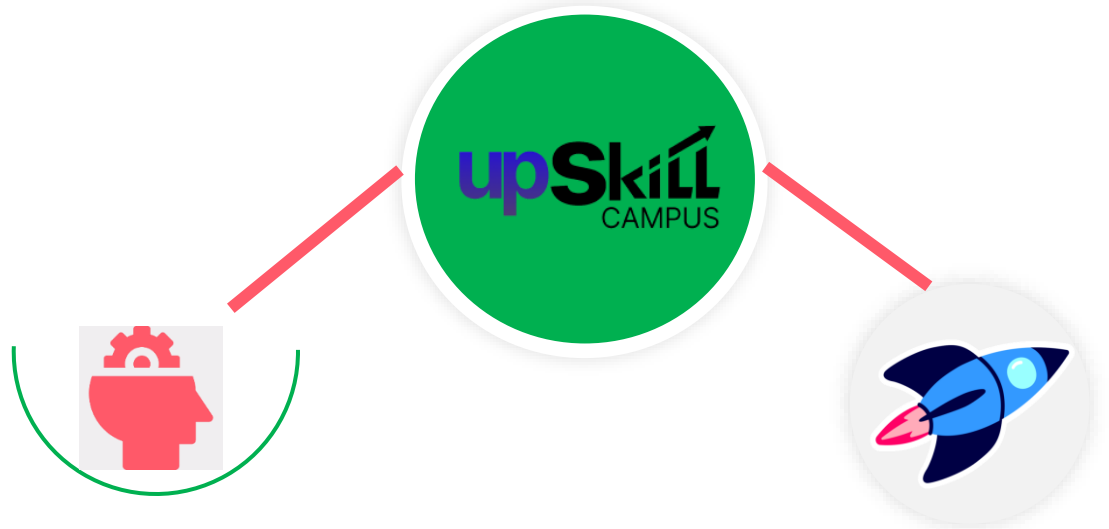
iv. Predictive Maintenance

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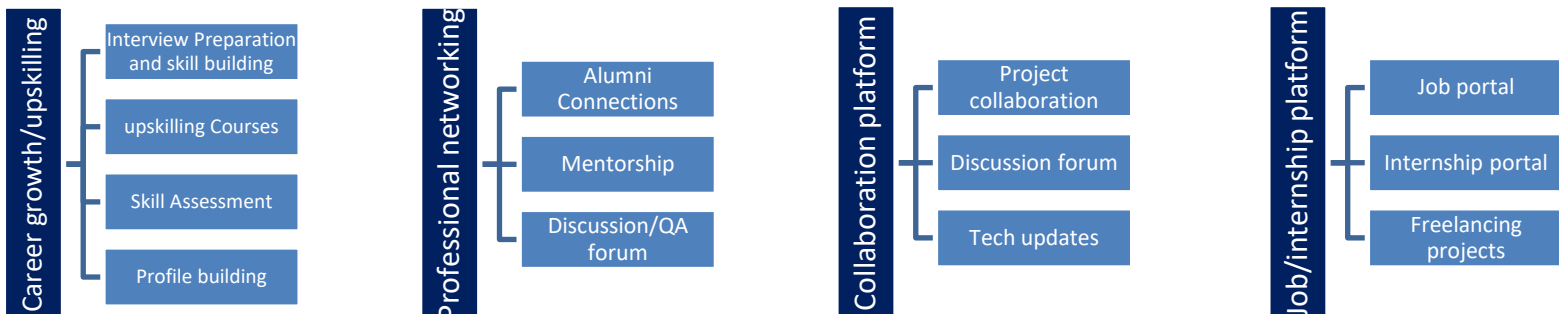
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.

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/

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

☛ 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]   HTML5, CSS3, JavaScript Documentation

[2]   Chart.js Visualization Library

[3]   upskill Campus Learning Resources

2.6 Glossary

Terms	Acronym
UI	User Interface
UX	User Experience
CRUD	Create, Read, Update, Delete
PIN	Personal Identification Number
LOCALSTORAGE	Browser-based data storage

3 Problem Statement

1. In the modern digital lifestyle, individuals perform a large number of financial transactions daily using cash, debit cards, credit cards, UPI applications, and online shopping platforms. Despite earning income, many people struggle to manage their finances efficiently because they do not have a structured system to record and analyze their expenses. This lack of awareness often results in overspending, poor savings habits, unplanned financial decisions, and increased financial stress.

2. Traditional methods such as maintaining handwritten expense notebooks or manually updating spreadsheets are still commonly used. However, these methods are time-consuming, prone to errors, and require continuous discipline. Users often fail to maintain consistency, which leads to incomplete or inaccurate records. Moreover, these manual systems do not provide real-time summaries, visual analytics, or predictive insights that can help users understand their spending behavior.

3. Although many digital expense tracking applications are available, most rely on cloud-based storage and mandatory account creation. This raises serious concerns regarding data privacy, security, and misuse of sensitive financial information. Users are often uncomfortable storing personal expense data on third-party servers. In addition, most existing applications provide only basic reports and lack advanced insights such as emotional spending patterns, future expense prediction, and offline functionality.

4. Another significant issue is the absence of strong security mechanisms. Applications without PIN or lock features increase the risk of unauthorized access to personal financial data.

5. The core problem addressed by this project is the absence of a secure, offline-capable, privacy-focused, and intelligent expense tracking system. The Smart Expense Tracker is designed to overcome these challenges by providing local data storage, meaningful analytics, visual insights, and enhanced security

4 Existing and Proposed solution

4.1.1 Existing Solutions

Existing expense tracking solutions include mobile applications, cloud-based financial platforms, and spreadsheet-based systems. Mobile apps provide basic expense logging and categorization but usually require internet connectivity and account registration. Cloud-based platforms store user data on remote servers, which introduces privacy and security risks.

Spreadsheet-based solutions offer flexibility but require manual calculations and technical knowledge. They lack automation, visualization, and predictive analytics, making them unsuitable for long-term use by non-technical users.

4.1.2 Limitations of Existing Systems

The major limitations of existing systems include dependency on internet connectivity, lack of data privacy, limited analytical features, and poor personalization. Most systems do not provide emotional or behavioral spending insights. Security features such as PIN-based locking are often missing, increasing the risk of unauthorized access.

4.1.3 Proposed Solution

The Smart Expense Tracker is proposed as an offline-first, browser-based application that stores all data locally using LocalStorage. This ensures complete privacy and eliminates the need for internet connectivity. The system provides advanced features such as budget tracking, remaining balance calculation, expense prediction, mood-based spending analysis, and interactive visualizations using charts and graphs.

4.2 Code submission (Github link)

/https://github.com/gauthamb2711/upskillcampus

5 Proposed Design/ Model

The Smart Expense Tracker follows a client-side modular architecture where all processing, storage, and visualization are handled within the browser. This design ensures high performance, data privacy, and ease of maintenance.

The system begins with user input through a simple and intuitive interface. Expense details are validated before being stored in LocalStorage. The analytics module processes stored data to generate summaries, predictions, and behavioral insights. Visualization components display the results using charts, heatmaps, and graphs.

The design also includes a security module that manages PIN-based access control, ensuring that user data remains protected. The modular structure allows easy extension and future upgrades.
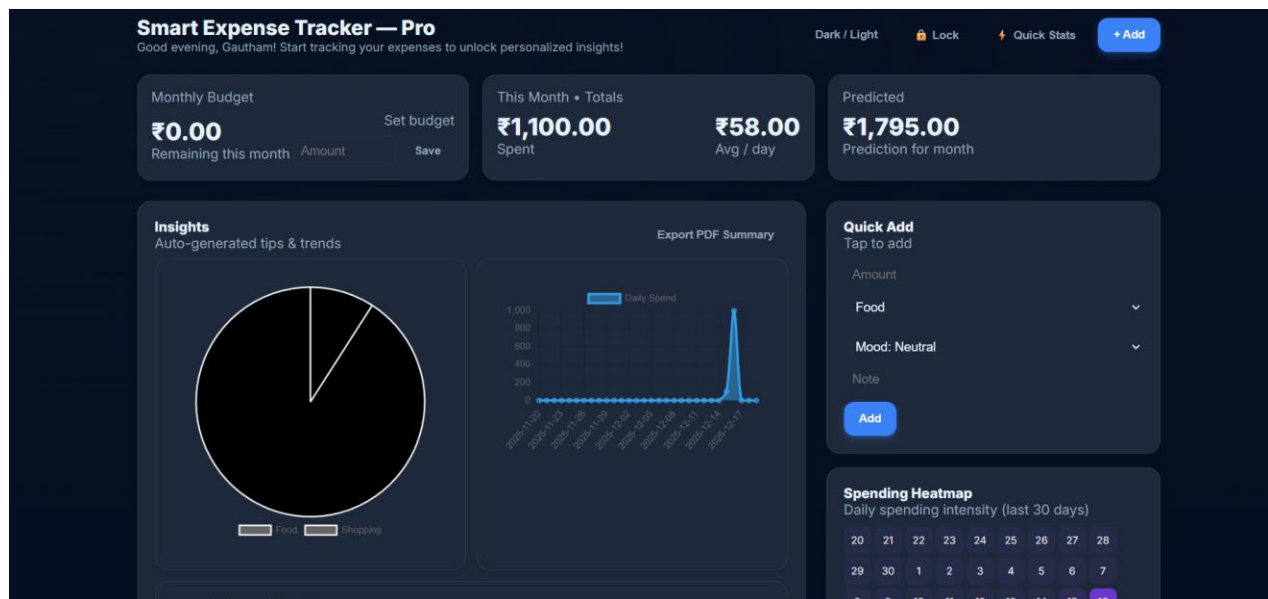
5.1 High Level Diagram (if applicable)

Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM

6 Performance Test

- Performance testing is a critical phase in application development, as it ensures that the system functions efficiently under real-world conditions. For the Smart Expense Tracker, performance testing focused on evaluating responsiveness, accuracy, memory usage, and overall user experience. Since the application is designed as an offline-first, browser-based system, it was essential to verify that all operations perform smoothly without dependency on internet connectivity.

- The application was tested under various usage scenarios, including frequent expense entry, large volumes of stored data, and continuous interaction with charts and analytics. Particular attention was given to the performance of LocalStorage operations, as all expense data is stored and retrieved locally. The system demonstrated fast data retrieval and minimal delay during expense addition, deletion, and modification.

- Chart rendering performance was also tested, as the application includes pie charts, line graphs, heatmaps, and dependency tree visualizations. The visualization components loaded efficiently without noticeable lag, even when handling extended datasets. Budget calculations, remaining balance updates, and expense predictions were computed accurately in real time.

- Security-related performance was tested by enabling and disabling the PIN lock feature. The locking and unlocking mechanism responded instantly without affecting stored data. Overall, the Smart Expense Tracker exhibited stable performance, efficient memory usage, and consistent responsiveness, making it suitable for daily use across different devices and browsers.

6.1 Test Plan/ Test Cases

- A structured test plan was designed to validate the functionality, performance, and reliability of the Smart Expense Tracker. The test plan covered all major modules, ensuring that each feature worked as intended under normal and edge-case scenarios. The objective was to identify defects,ensure data integrity, and validate system behavior.

- Test cases were created for expense CRUD operations (Create, Read, Update, Delete), ensuring that expenses could be added, edited, and removed correctly. Budget-related test cases verified correct calculation of total expenses, remaining balance, and overspending alerts. Prediction test cases validated the accuracy of estimated monthly expenses based on average daily spending.

- Additional test cases were designed to verify visualization accuracy, ensuring that charts and graphs reflected real-time data changes. Security-related test cases tested PIN setup, incorrect PIN entry, and lock/unlock behavior. PDF export functionality was tested to confirm that reports were generated correctly with accurate summaries.

- Edge cases such as empty input fields, extremely large expense values, and invalid data formats were also tested. The test plan ensured comprehensive coverage of system behavior, resulting in a reliable and robust application.

6.2 Test Procedure

- The testing process followed a systematic and manual approach due to the client-side nature of the application. Each module was tested individually before performing integrated testing to verify end-to-end functionality. Testing was conducted across different browsers to ensure cross-browser compatibility.

- The procedure began by initializing the application with empty LocalStorage to test first-time usage behavior. Expenses were then added incrementally to evaluate real-time updates and calculations. Data persistence was verified by refreshing the browser and reopening the application.

- Visualization modules were tested by adding expenses across different categories, dates, and moods to ensure correct graphical representation. Security testing involved enabling the PIN lock, attempting unauthorized access, and verifying successful authentication. PDF export functionality was tested by generating monthly reports and verifying content accuracy.

- All observed results were compared against expected outcomes. Any inconsistencies were resolved through debugging and refinement. This systematic testing approach ensured high reliability and correctness

6.3 Performance Outcome

1. The performance testing results confirmed that the Smart Expense Tracker meets all functional and non-functional requirements. The application demonstrated fast response times, accurate calculations, and smooth visual transitions. Memory usage remained stable even with extended usage.

2. The offline-first design proved highly effective, allowing uninterrupted operation without internet connectivity. No data loss or corruption was observed during repeated operations. The security mechanism did not introduce any noticeable performance overhead.

7 My learnings

This internship project provided valuable technical and professional learning experiences. I gained hands-on experience in developing a complete web application from concept to deployment. Working on the Smart Expense Tracker strengthened my understanding of frontend development using HTML, CSS, and JavaScript.

I learned how to implement client-side data persistence using LocalStorage and how to manage structured data efficiently. The project enhanced my ability to design user-friendly interfaces and integrate interactive data visualizations. Implementing security features such as PIN-based access control improved my awareness of data protection practices.

Beyond technical skills, this project improved my problem-solving ability, analytical thinking, and attention to detail. I learned how to break down complex requirements into manageable modules and how to test and debug applications systematically. The internship also enhanced my communication skills through documentation and mentor interactions.

Overall, this experience increased my confidence in software development and prepared me for future industry roles.

8   Future work scope

The Smart Expense Tracker has significant potential for future enhancement and scalability. One major improvement would be the integration of optional cloud backup and synchronization, allowing users to access their data across multiple devices while maintaining privacy controls.

Developing a mobile application version would improve accessibility and convenience. AI-based financial recommendations could be implemented to provide personalized savings suggestions and spending alerts. Multi-user support could enable family or group expense tracking.

Additional features such as bank statement integration, SMS transaction parsing, advanced analytics dashboards, and multilingual support could further enhance usability. These future enhancements would transform the Smart Expense Tracker into a comprehensive and intelligent financial management platform.