Industrial Internship Report on

Student Record Management System

Prepared by

Megharaj Avin Dandgavhal

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 (Tell about ur Project) 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

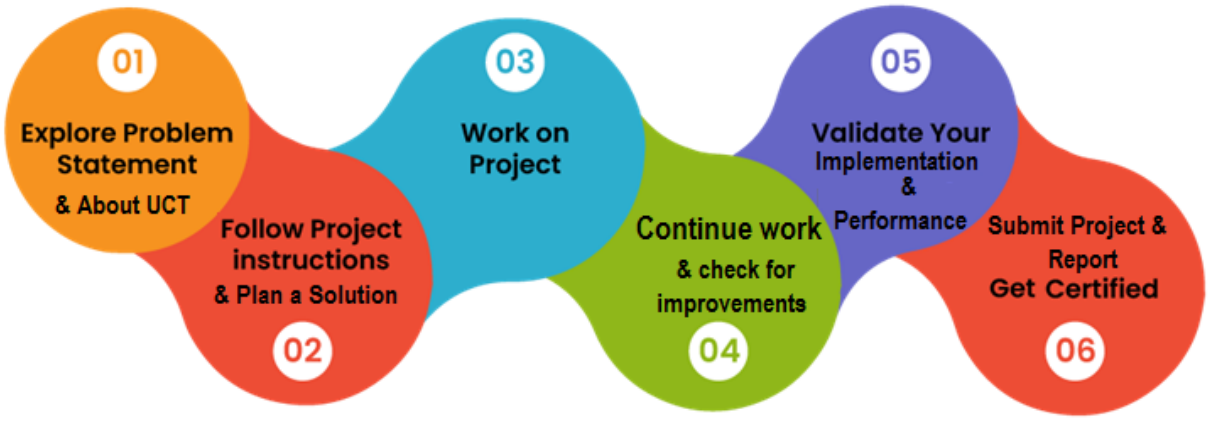
During the 6-week internship program, I worked on building a Java-based Student Record Management System under the guidance of upskill Campus and UniConverge Technologies Pvt. Ltd. The work was structured with milestones, mentorship, and regular feedback, allowing me to progressively complete and document the entire project.

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.

I sincerely thank Mr. Abhishek Sharma, Ms. Aditi Kapoor, and the teams at upskill Campus, The IoT Academy, and UCT for their continuous support. I also thank my peers and mentors who helped me directly or indirectly during this internship.

Make the most of internships. Apply your theoretical knowledge to real-world problems. Don't fear challenges, stay curious, ask questions, and always aim to grow with every experience.

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)

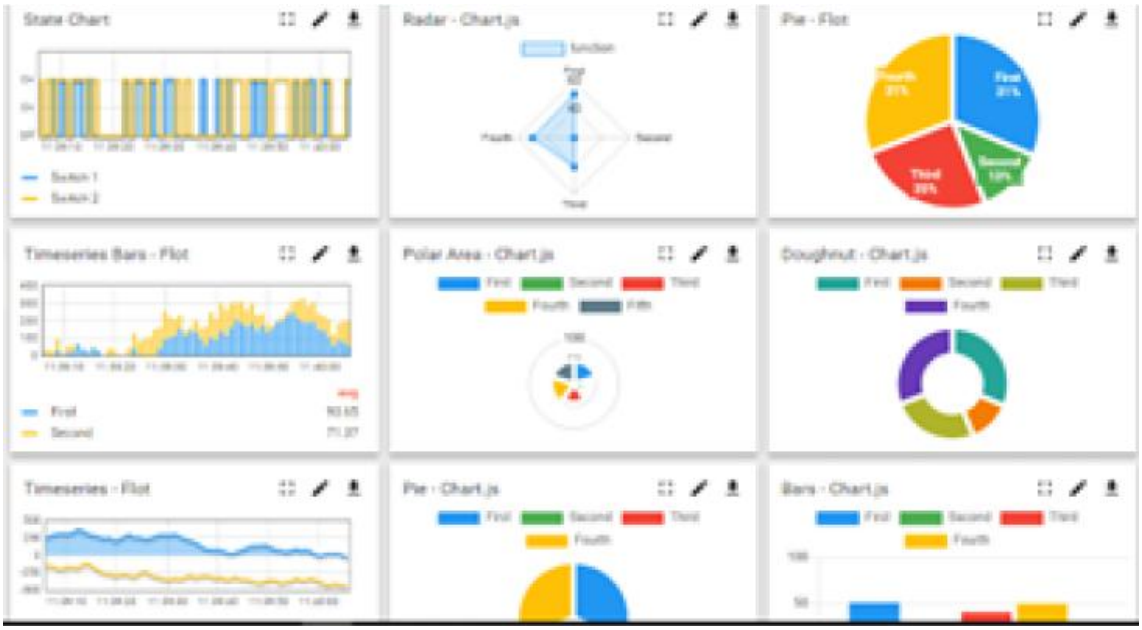
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

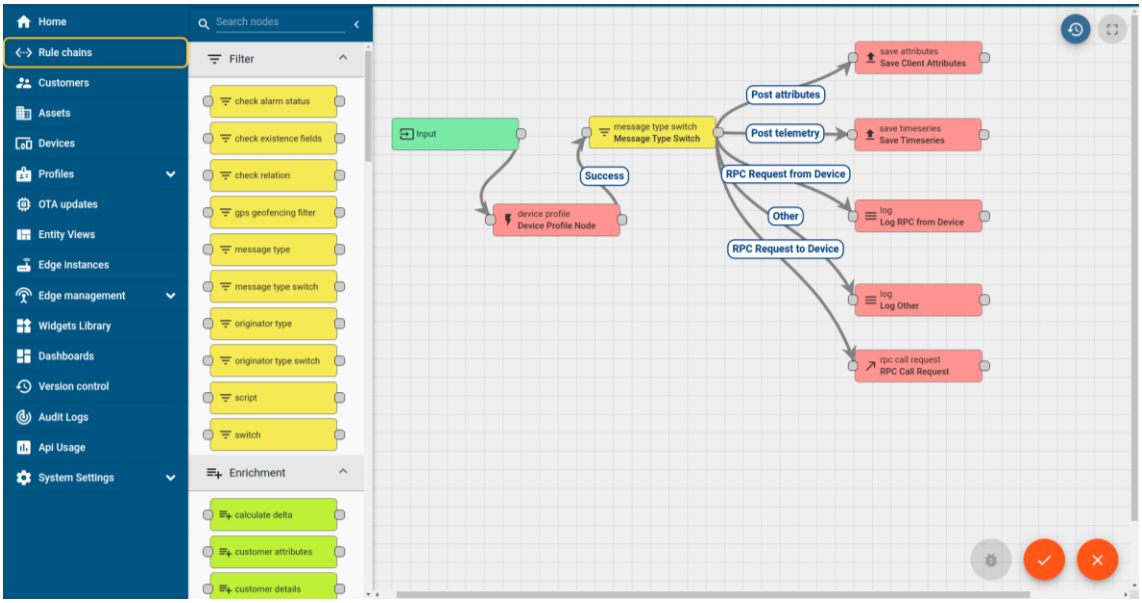
- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA

- 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.

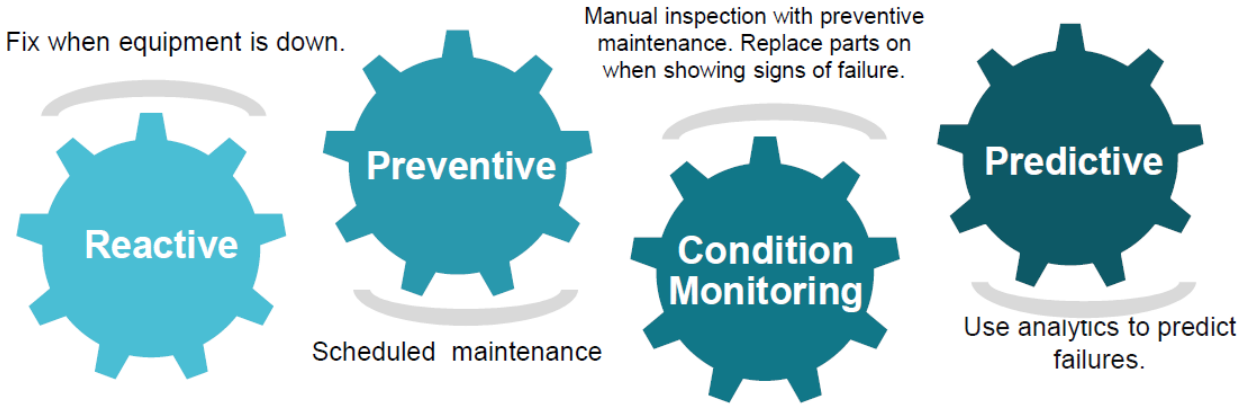
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1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

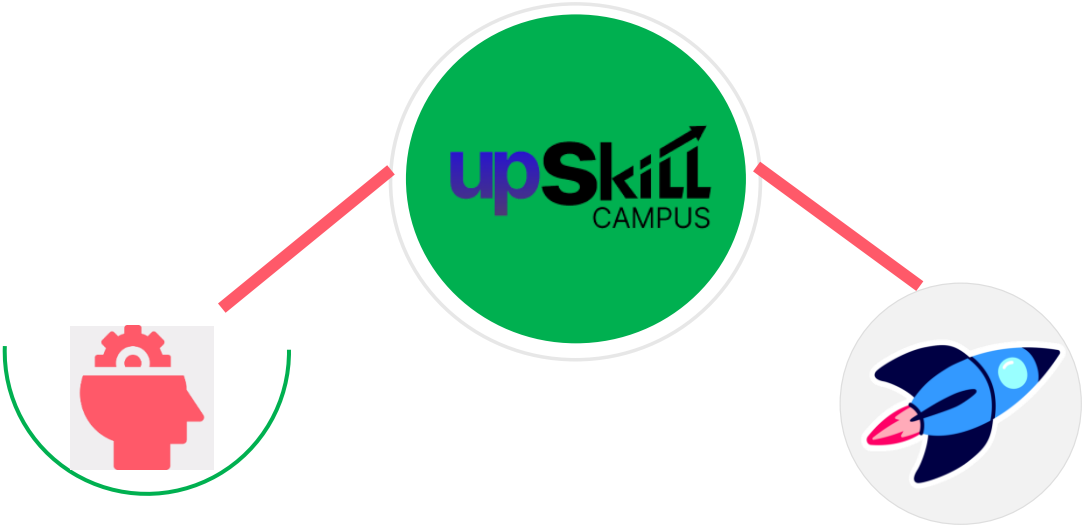
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



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.

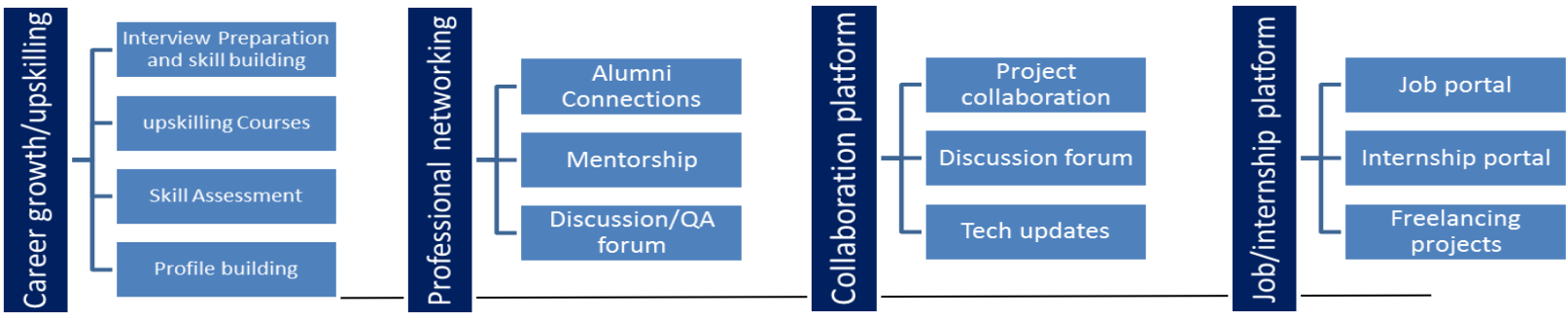
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



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]

[2]

[3]

2.6 Glossary

Terms	Acronym

3 Problem Statement

In the assigned problem statement

The problem was to create a simple, interactive, and reliable system for managing student records using Java. The system should support core operations like adding a new student, displaying a student's details, deleting a record, and showing all records.

4 Existing and Proposed solution

Provide summary of existing solutions provided by others, what are their limitations?

What is your proposed solution?

What value addition are you planning?

4.1 Code submission (Github link): https://github.com/arjun/upskillcampus/blob/main/StudentRecordManagementSystem.java

4.2 Report submission (Github link): https://github.com/arjun/upskillcampus/blob/main/StudentRecordManagementSystem_Arjun_USC_UCT.pdf

5 Proposed Design/ Model

Given more details about design flow of your solution. This is applicable for all domains. DS/ML Students can cover it after they have their algorithm implementation. There is always a start, intermediate stages and then final outcome.

5.1 Interfaces

Update with Block Diagrams, Data flow, protocols, FLOW Charts, State Machines, Memory Buffer Management.

```
import java.util.*;

class Student {
    private String id;
    private String name;
    private int age;
    private double grade;

    public Student(String id, String name, int age, double grade) {
        this.id = id;
        this.name = name;
        this.age = age;
        this.grade = grade;
    }

    public void display() {
        System.out.println("Student ID   : " + id);
        System.out.println("Name         : " + name);
        System.out.println("Age          : " + age);
        System.out.println("Grade        : " + grade);
        System.out.println("-----------------------------");
    }

    public String getId() {
        return id;
    }
}

public class StudentRecordManagementSystem {
    private static Scanner sc = new Scanner(System.in);
```

```
    private static Map<String, Student> studentMap = new HashMap<>();

    public static void main(String[] args) {
        int choice;
        do {
            System.out.println("\n*** Student Record Management System ***");
            System.out.println("1. Add Student");
            System.out.println("2. Display Student");
            System.out.println("3. Delete Student");
            System.out.println("4. Display All Students");
            System.out.println("5. Exit");
            System.out.print("Enter your choice: ");
            choice = sc.nextInt();
            sc.nextLine(); // Clear buffer

            switch (choice) {
                case 1:
                    addStudent();
                    break;
                case 2:
                    displayStudent();
                    break;
                case 3:
                    deleteStudent();
                    break;
                case 4:
                    displayAllStudents();
                    break;
                case 5:
                    System.out.println("Exiting system. Goodbye!");
                    break;
                default:
                    System.out.println("Invalid choice! Try again.");
            }
        } while (choice != 5);
    }

    private static void addStudent() {
        System.out.print("Enter Student ID: ");
        String id = sc.nextLine();
        if (studentMap.containsKey(id)) {
            System.out.println("Student already exists!");
```

```
            return;
        }

        System.out.print("Enter Name: ");
        String name = sc.nextLine();
        System.out.print("Enter Age: ");
        int age = sc.nextInt();
        System.out.print("Enter Grade: ");
        double grade = sc.nextDouble();
        sc.nextLine(); // Clear buffer

        Student student = new Student(id, name, age, grade);
        studentMap.put(id, student);
        System.out.println("Student added successfully.");
    }

    private static void displayStudent() {
        System.out.print("Enter Student ID to display: ");
        String id = sc.nextLine();
        Student student = studentMap.get(id);
        if (student != null) {
            student.display();
        } else {
            System.out.println("Student not found!");
        }
    }

    private static void deleteStudent() {
        System.out.print("Enter Student ID to delete: ");
        String id = sc.nextLine();
        if (studentMap.remove(id) != null) {
            System.out.println("Student deleted successfully.");
        } else {
            System.out.println("Student not found.");
        }
    }

    private static void displayAllStudents() {
        if (studentMap.isEmpty()) {
            System.out.println("No student records found.");
        } else {
            for (Student student : studentMap.values()) {
```

```
                student.display();
            }
        }
    }
}
```

Output:

```
C:\Users\megha\.jdks\openjdk-23\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.4\lib\idea_rt.jar=61981:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.4\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath E:\StudentRecordManagementSystem.java\out\production\StudentRecordManagementSystem.java StudentRecordManagementSystem

*** Student Record Management System ***
1. Add Student
2. Display Student
3. Delete Student
4. Display All Students
5. Exit
Enter your choice: 1
Enter Student ID: 101
Enter Name: Megharaj
Enter Age: 21
Enter Grade: 8.5
Student added successfully.

*** Student Record Management System ***
1. Add Student
2. Display Student
3. Delete Student
4. Display All Students
5. Exit
Enter your choice: 2
Enter Student ID to display: 101
Student ID    : 101
Name          : Megharaj
Age           : 21
Grade         : 8.5
----------------------------

*** Student Record Management System ***
1. Add Student
2. Display Student
3. Delete Student
4. Display All Students
5. Exit
Enter your choice:
```

6 Performance Test

This is very important part and defines why this work is meant of Real industries, instead of being just academic project.

Here we need to first find the constraints.

How those constraints were taken care in your design?

What were test results around those constraints?

Constraints can be e.g. memory, MIPS (speed, operations per second), accuracy, durability, power consumption etc.

In case you could not test them, but still you should mention how identified constraints can impact your design, and what are recommendations to handle them.

6.1 Test Plan/ Test Cases

6.2 Test Procedure

6.3 Performance Outcome

7 My learnings

Through this internship, I enhanced my skills in Java programming, especially in writing modular and reusable code. I learned how to implement real-world logic, handle exceptions, and work with data structures. It also improved my understanding of project planning and documentation.

8 Future work scope

In the future, the system can be enhanced by adding a graphical user interface (GUI), integration with databases like MySQL for persistent storage, and login authentication for data security.