Industrial Internship Report on

”URL SHORTENER”

Prepared by

[CHILUKURI VENUGOPAL]

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 was on URL shorteners that are valuable tools that enhance the usability and management of web links. They provide benefits such as improved link aesthetics, better tracking and analytics, and easier sharing across platforms.

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 3

2 Introduction 4

2.1 About UniConverge Technologies Pvt Ltd 4

2.2 About upskill Campus 8

2.3 The IoT Academy 8

2.4 Objectives of this Internship program 9

3. Problem Statement.....10

4 Existing and Proposed solution.....11

4.1 Code submission 12

5 Performance Test.....13

6 My learnings.....14

7 Future work scope.....16

1 Preface

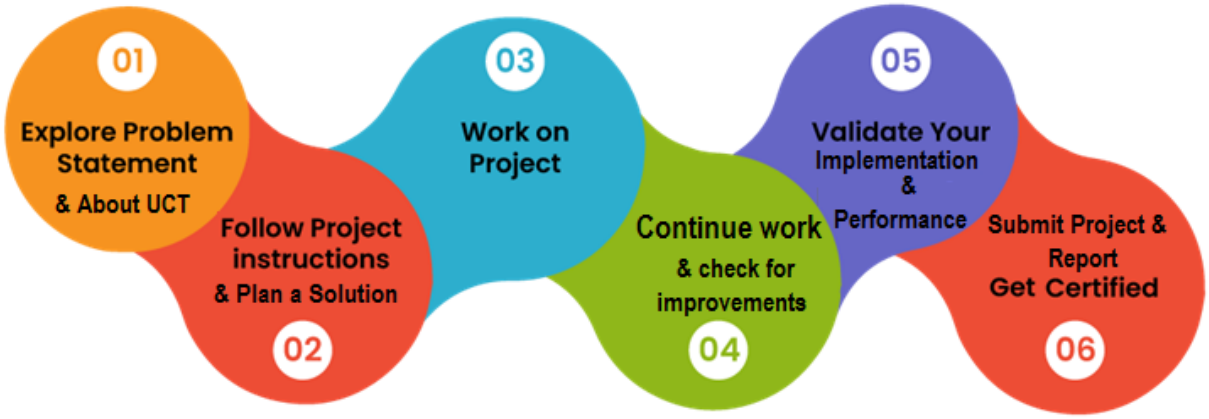
Summary of the whole 6 weeks' work.

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.

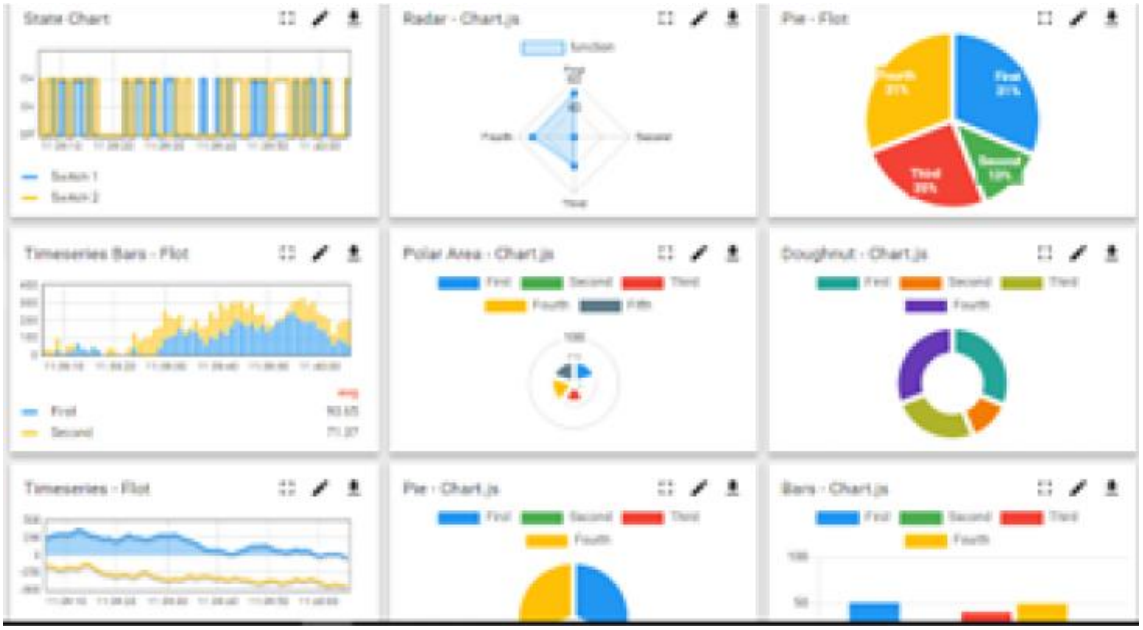
i. UCT IoT Platform (UCT Insight)

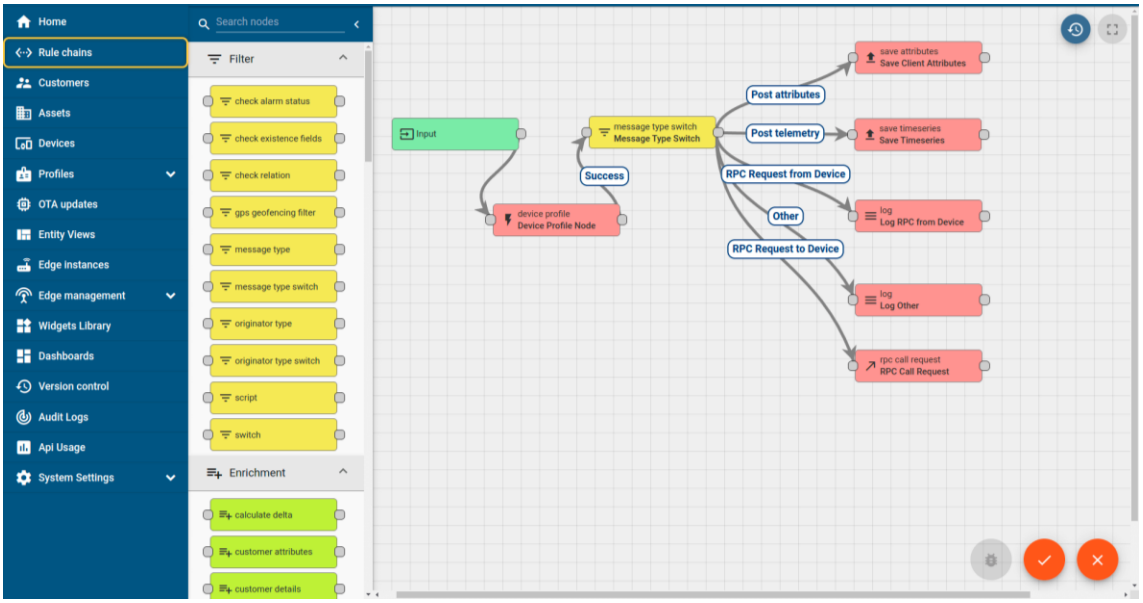
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- 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 (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 m

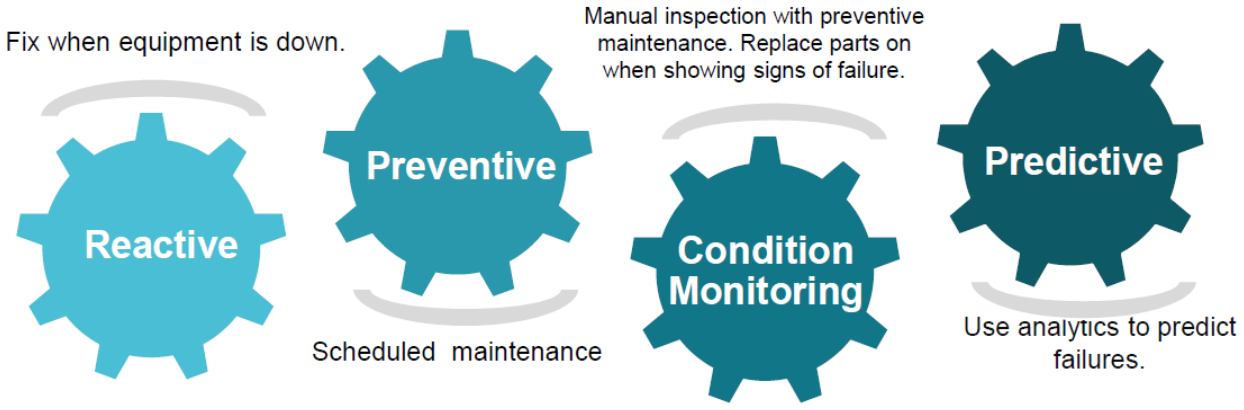
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1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

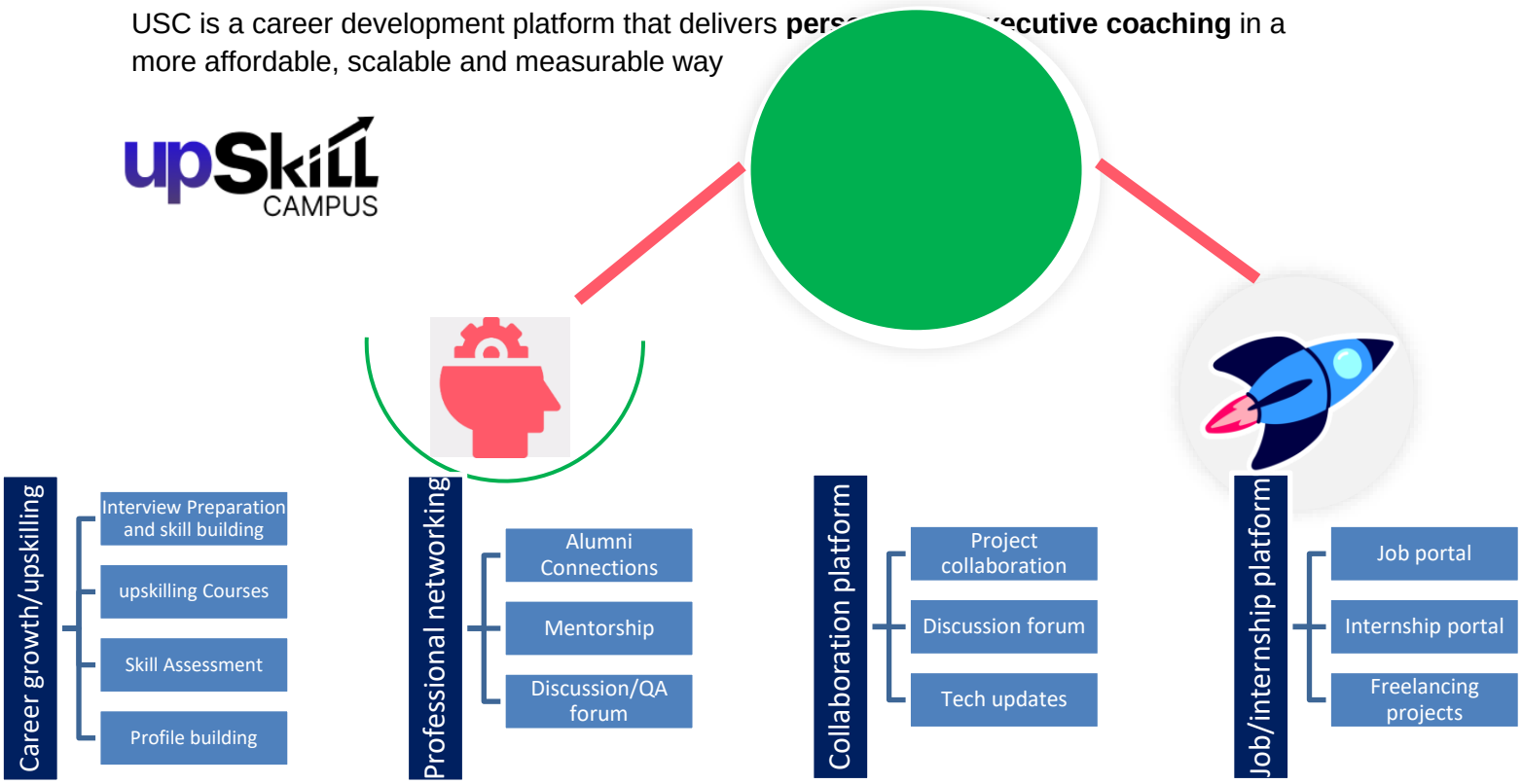
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



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 **pers** **ecutive coaching** in a more affordable, scalable and measurable way



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.

3 Problem Statement

- **URL Shortener**:

Description: The URL shortener is a Python project that converts long URLs into shorter, more manageable links. It takes a long URL as input, generates a unique shortened URL, and redirects users to the original URL when the shortened link is accessed.

Scope: The scope of this project involves designing a user interface to input long URLs and display the shortened links, implementing a database to store the mapping between original and shortened URLs, and developing functions to generate unique shortened URLs and handle redirection.

4 Existing and Proposed solution

1. **Bitly**

- **Overview**: Bitly is one of the most popular URL shorteners, offering features such as branded links, custom short URLs, and comprehensive analytics.
- **Limitations**:
    - **Cost**: While it offers a free version, many of the advanced features are locked behind a paid plan.
    - **Privacy Concerns**: The service tracks user data and link performance, which might be a concern for privacy-focused users.

2. **TinyURL**

- **Overview**: TinyURL provides a straightforward and easy-to-use service for shortening URLs without requiring an account. It also offers browser extensions for quick access.
- **Limitations**:
    - **Lack of Analytics**: TinyURL lacks advanced analytics and tracking features that are available in other URL shorteners.
    - **Limited Customization**: Customizing short links and branding options are limited compared to other services.

3. **Ow.ly (by Hootsuite)**

- **Overview**: Ow.ly integrates with Hootsuite's social media management platform, providing URL shortening along with social media analytics.
- **Limitations**:
    - **Integration Dependency**: Full functionality requires a Hootsuite subscription, which might not be suitable for users only needing URL shortening.
    - **Complexity**: The platform can be complex for users who only need basic link shortening and are not interested in social media management tools.

MY SOLUTION :

Overview of the URL Shortener Solution

1. **Requirements**
    - Programming Language: Python
    - Web Framework: Flask or Django (Flask is simpler and more lightweight for this project)
    - Database: SQLite or a more robust option like PostgreSQL if needed
    - Hosting: Optional, for deployment (e.g., Heroku, AWS, etc.)
2. **Key Features**
    - Shorten URL: Convert a long URL into a short, unique URL.
    - Redirect: Redirect users from the short URL to the original long URL.
    - Analytics: Track clicks on the short URLs (optional but useful).
    - Custom Aliases: Allow users to specify custom short URL aliases (optional).

4.1 Code submission :

GITHUB LINK: https://github.com/Venu145pro/upskillcampus

Report submission (Github link) :

GITHUB LINK:

5 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 My learnings

You should provide summary of your overall learning and how it would help you in your career growth.

- **1. Understanding of URL Shortening Logic:**
- **Hashing Algorithms**: You learned how to implement hashing algorithms (like Base62 encoding) to convert long URLs into shorter, unique keys.
- **Unique URL Generation**: Ensuring that the shortened URL is unique, even when users submit the same URL multiple times, may have required using techniques like random string generation or a counter-based system.
- **2. Web Framework Knowledge (Flask/Django):**
- **Routing**: You gained an understanding of how to handle URL routing in Python-based frameworks like Flask or Django.
- **HTTP Methods**: Understanding the difference between GET and POST requests, especially when users submit long URLs for shortening.
- **Database Interaction**: You might have learned how to store original and shortened URLs in a database, and how to retrieve them efficiently.
- **3. Database Schema Design:**
- You learned how to design a schema to store long URLs and their corresponding short URLs, considering scalability and efficiency.
- Handling collisions or duplicate URLs, perhaps through the use of indices or constraints.
- **4. Handling Redirects:**
- You likely learned how to handle HTTP redirections, where the shortened URL redirects users to the original long URL.
- Managing different HTTP status codes, such as 301 (Moved Permanently) or 302 (Found), in redirect operations.
- **5. Front-End Interaction:**
- **Form Handling**: You may have developed the front-end interface to allow users to submit URLs, learn basic HTML form processing, and validate user input.

- **User Feedback**: Ensuring that users get appropriate feedback (success or error messages) after URL shortening.
- **6. Error Handling:**
- Implementing error handling, like managing invalid URLs, expired URLs, or handling edge cases such as when the URL shortening service is down.
- **7. Scalability Considerations:**
- You might have considered how to scale your URL shortening service to handle a large number of requests, such as using caching mechanisms or load balancing for heavy traffic.
- **8. Security Aspects:**
- Addressing security concerns like preventing malicious URLs, spam, or phishing attempts.
- Implementing input validation and output sanitization to avoid issues like SQL injection or cross-site scripting (XSS).

7 Future work scope

You can put some ideas that you could not work due to time limitation but can be taken in future.

Here are some future work ideas for the URL Shortener project that could not be implemented due to time limitations but can be considered for future development:

1. Custom Short URLs:

- While the current project generates random short links, future work could focus on allowing users to create custom URLs (e.g., example.com/custom-name) that are easy to remember.

2. User Authentication and Account Management:

- Implementing a full-fledged user authentication system (using OAuth, JWT, or session-based authentication) was not included due to time constraints. In the future, users could create accounts, log in, and manage their shortened URLs from a personalized dashboard.

3. Analytics Dashboard:

- Although the basic functionality is implemented, a detailed analytics dashboard for users was not possible within the given time. The dashboard could provide metrics such as click-through rates, geographic data of visitors, browser usage, and device types.

4. Link Expiration and Deletion:

- Future work can include adding features where users can set expiration dates for their shortened links, automatically deactivating them after a specified time, or deleting them after reaching a certain number of clicks.

5. QR Code Integration:

- Integrating QR code generation for each shortened URL was another feature not completed due to time limitations. This feature would allow users to easily share URLs in offline settings by scanning QR codes.