Industrial Internship Report on

URL Shortener using Python

Prepared by

Simida Parida

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.

The URL Shortener project is a Python-based web application developed using the Flask framework. The primary objective of this project is to create a user-friendly and efficient system for converting long URLs into shorter, more manageable links.

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

I. Preface

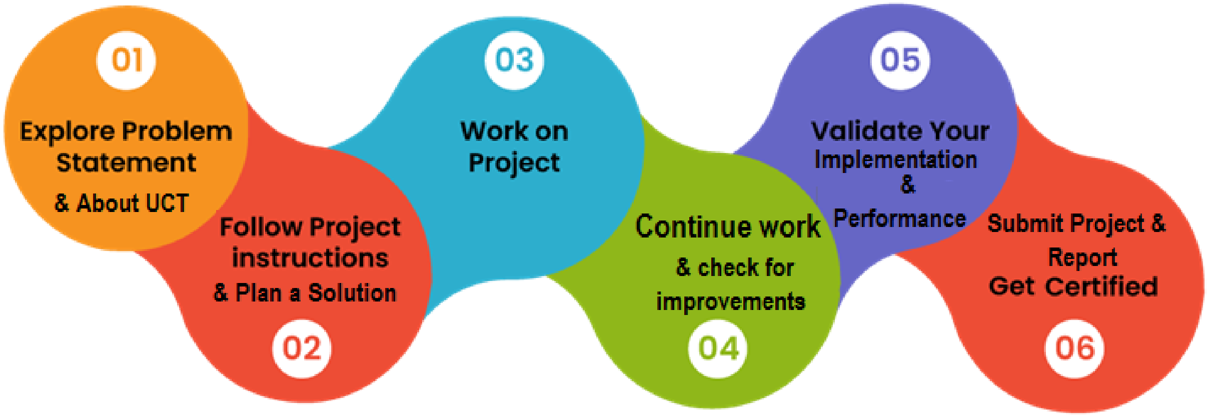
Over the course of six enriching weeks, I had the privilege of undertaking a transformative internship that significantly contributed to my career development. This internship, made possible through the valuable opportunity provided by USC/UCT, proved to be a pivotal experience in enhancing my practical skills and knowledge.

Internships play a crucial role in bridging the gap between academic learning and real-world application. They offer a unique platform for students to integrate theoretical knowledge into practical scenarios, fostering a holistic understanding of industry dynamics. The hands-on experience gained during internships not only sharpens technical skills but also cultivates essential professional attributes such as problem-solving, teamwork, and adaptability. This immersion in a professional environment is instrumental in preparing individuals for the challenges and expectations of their future careers.

The focus of my internship centered around the development of a URL Shortener using Python, a project undertaken to address the need for efficient and user-friendly link management. This web application, built on the Flask framework, aimed to streamline the process of converting long URLs into shorter, more manageable links. The integration of Python for backend logic and Flask for web development allowed for a seamless and responsive user experience.

The opportunity provided by USC/UCT served as a catalyst for practical skill development and exposure to real-world project implementation. The guidance and support received during the internship were instrumental in navigating the complexities of web development and honing my abilities in Python programming and Flask framework utilization.

How Program was planned



The URL Shortener Web Application project undertaken during this internship involved the development of a user-friendly system to convert long URLs into shorter, more manageable links.

Learning Outcomes:

i. **Flask Framework Proficiency:** Gained hands-on experience in utilizing Flask for web application development, understanding routing, templates, and request handling.

ii. **Python Backend Development:** Strengthened skills in Python programming for backend logic, including data manipulation and database interaction.

iii. **Database Management:** Learned to use SQLite for efficient storage and retrieval of short link mappings.

iv. **User Interface Design:** Acquired insights into creating a user-friendly interface, ensuring a positive user experience.

v. **Problem-Solving and Debugging:** Developed effective problem-solving skills and became adept at debugging and troubleshooting.

The six-week internship provided an invaluable opportunity to apply theoretical knowledge in a practical setting. The hands-on experience in developing a URL Shortener using Python and Flask deepened my understanding of web application architecture, database management, and user interface design. The challenges encountered served as learning opportunities, fostering problem-solving skills and adaptability.

This project not only enriched my technical skills but also provided insights into the intricacies of web development, preparing me for future challenges in the dynamic field of software engineering. The URL Shortener project stands as a testament to the effectiveness of internships in bridging the gap between academic learning and real-world application.

Acknowledgement:

I would like to extend my sincere thanks to **Ankit Kumar** and **Archana Singh** from **upSkill Campus** and **Kaushlendra Singh Sisodia** from **UniConverge Technologies / The IoT Academy** for their unwavering support and guidance throughout the course of this project.

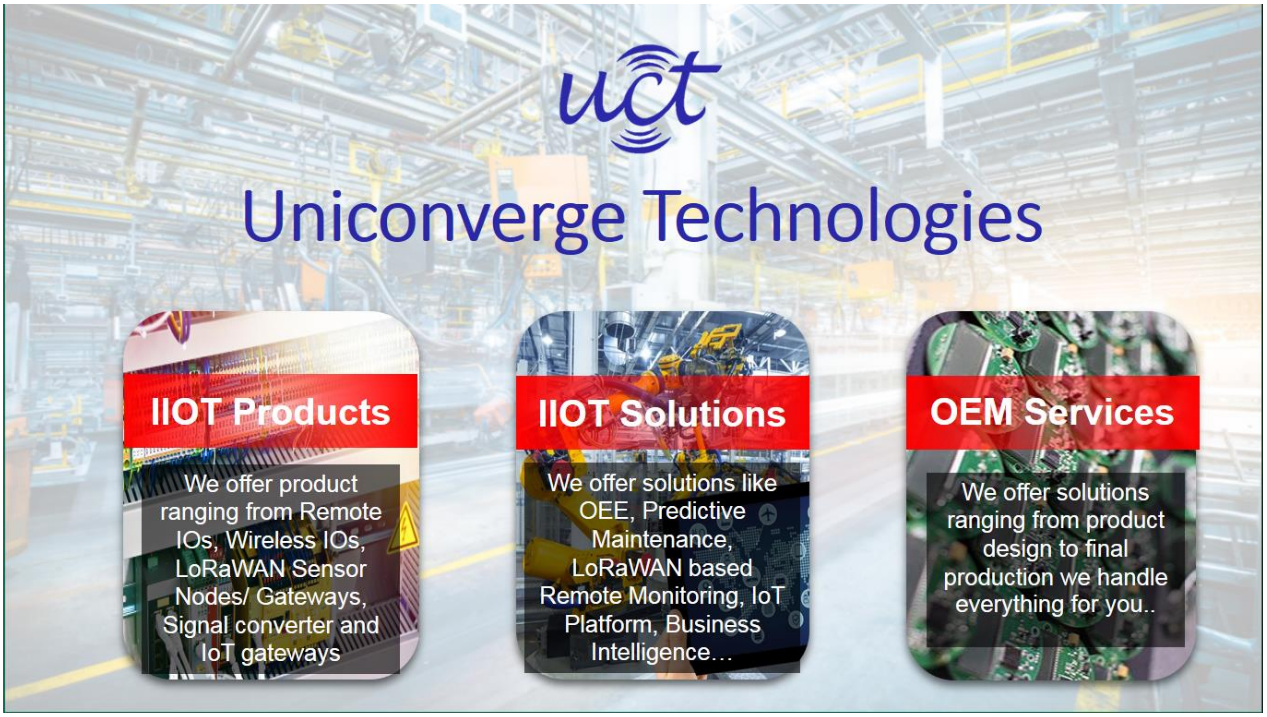
They played a pivotal role in providing insights into industry best practices, offering constructive feedback. Thank you, for your invaluable contributions to this project and for being an exceptional mentor.

II. Introduction

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.

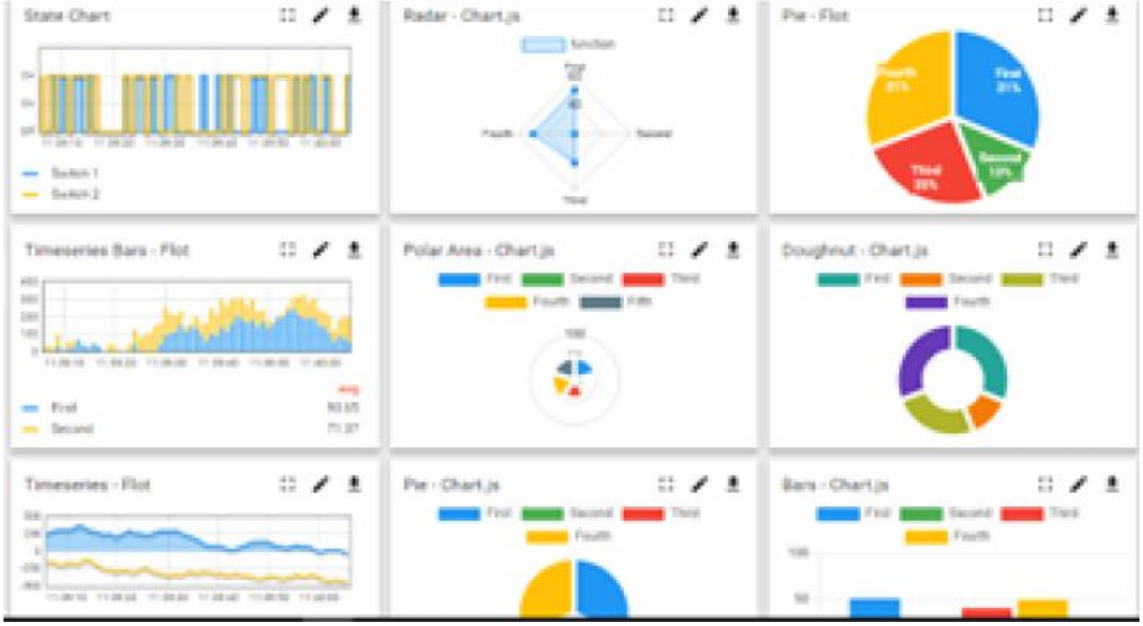
- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA

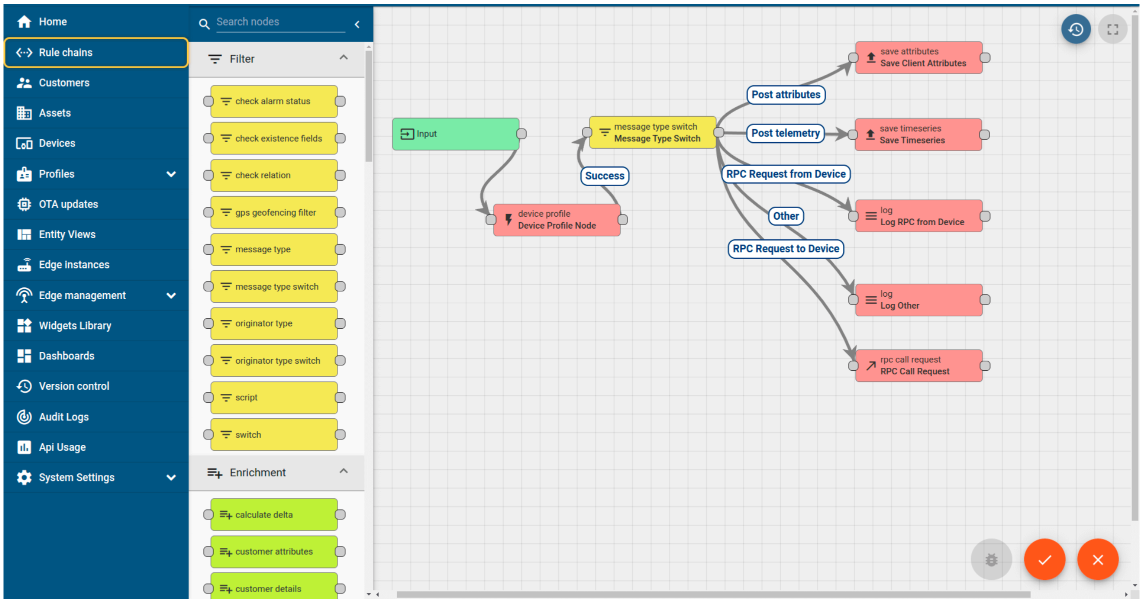
- It supports both cloud and on-premises deployments.

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





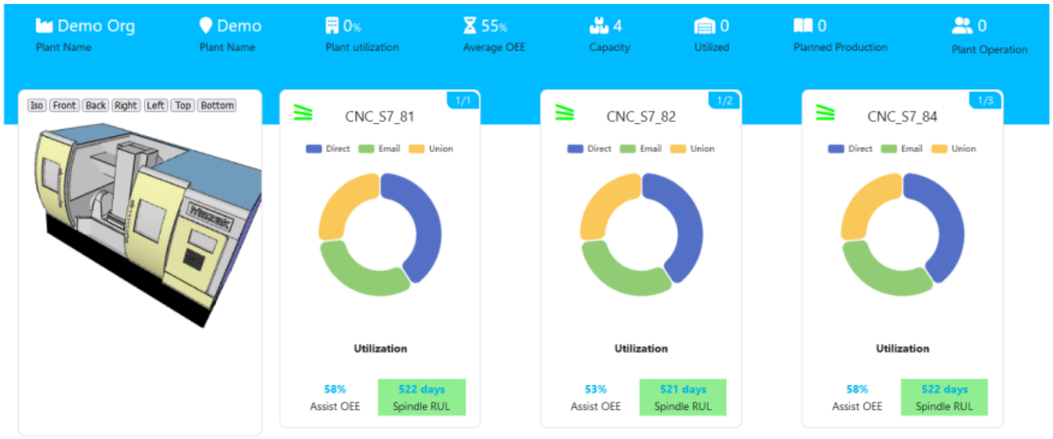
ii. Smart Factory Platform (FACTORY WATCH)

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

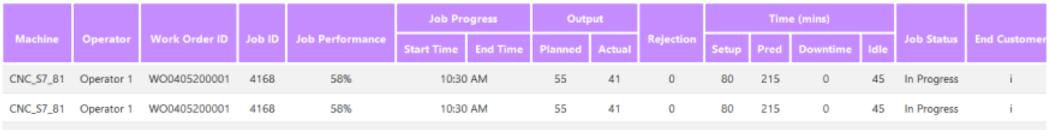
0
Planned Production

0
Plant Operation

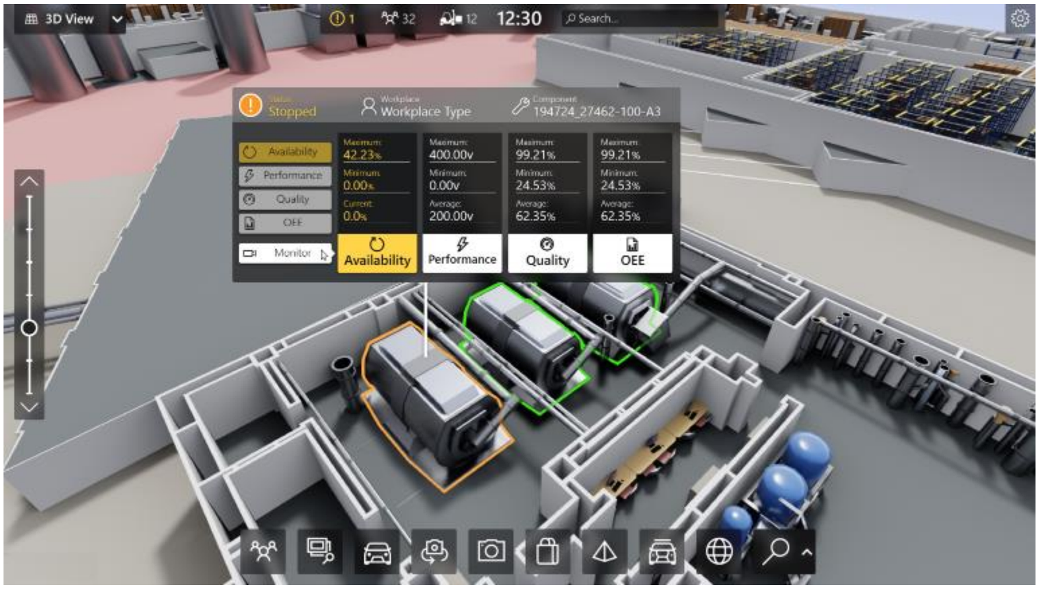
CNC_S7_81 (1/1): 58% Assist OEE; 522 days Spindle RUL

CNC_S7_82 (1/2): 53% Assist OEE; 521 days Spindle RUL

CNC_S7_84 (1/3): 58% Assist OEE; 522 days Spindle RUL



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

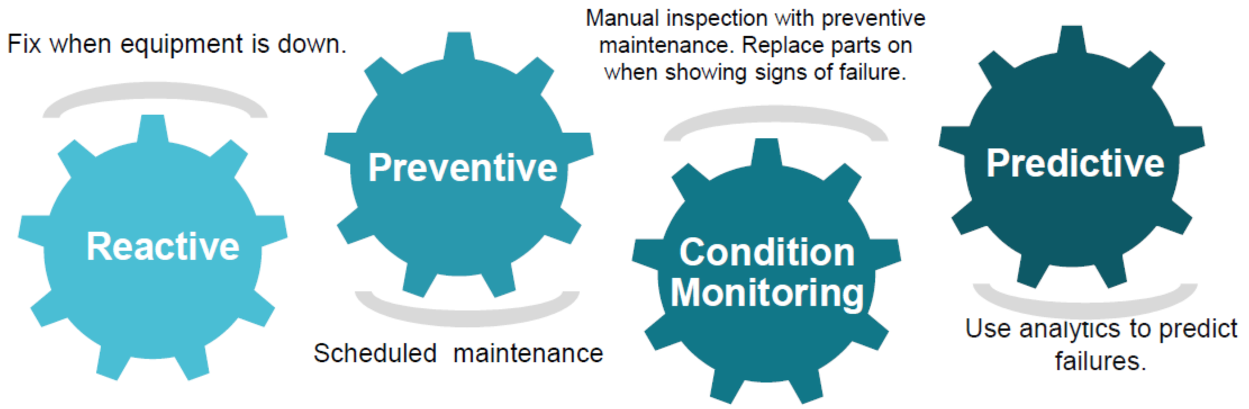
Availability	Performance	Quality	OEE
Maximum: 42.23%	Maximum: 400.00v	Maximum: 99.21%	Maximum: 99.21%
Minimum: 0.00%	Minimum: 0.00v	Minimum: 24.53%	Minimum: 24.53%
Current: 0.0%	Average: 200.00v	Average: 62.35%	Average: 62.35%

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

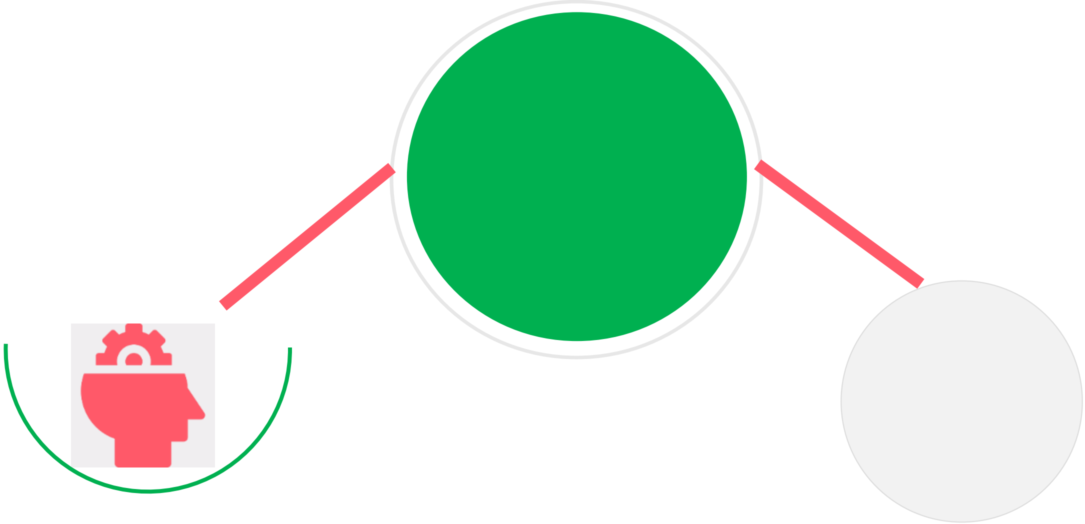
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.


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.

II.1 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.

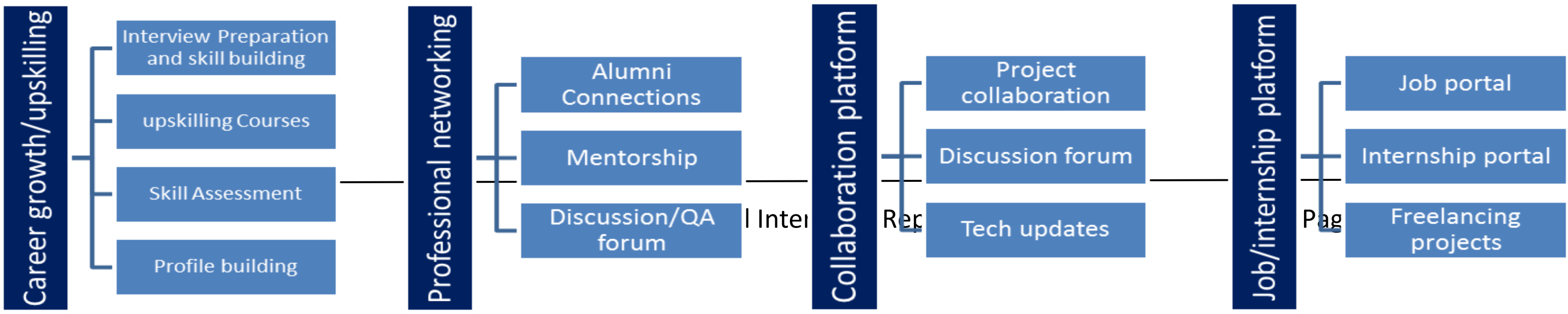
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



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

l Inter

Collaboration platform

- Project collaboration
- Discussion forum
- Tech updates

Rep

Job/internship platform

- Job portal
- Internship portal
- Freelancing projects

Pag

II.2 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.

II.3 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.

II.4 References

[1] “How to create a URL Shortener using Python and Flask”.

Available: https://www.freecodecamp.org/news/python-tutorial-how-to-create-a-url-shortener-using-flask/

[2] “Introduction to Flask and Python Flask Documentation.”

Available: https://medium.com/@knowledgelibary/python-flask-documentation-760b718d4246

[3] “SQLAlchemy Documentation.”

Available: https://docs.sqlalchemy.org/en/20/

III. Problem Statements

1. **Usability:** Many users find it inconvenient to share long URLs, especially on platforms with character limits. Create a URL Shortener using Python to provide a simple and efficient way to generate short links for easy sharing.

2. **Customization:** Users often want personalized and memorable short links for branding purposes. Design a URL Shortener that allows users to customize short links, making them more relevant and distinctive.

3. **Integration:** Organizations may need an in-house URL Shortener that integrates seamlessly with existing systems. Develop a Python solution that is easily integrable, catering to organizations requiring a customized URL shortening solution.

4. **Open Source for Developers**: Developers seek customizable and open-source solutions for URL shortening. Create an open-source Python-based URL Shortener to enable developers to contribute, customize, and extend functionality according to their specific needs.

IV. Existing and Proposed solution

1. Existing Solutions for URL Shortening:

a) Bitly: A popular URL shortening service providing analytics, custom short links, and link management features.
b) TinyURL: Offers basic URL shortening with a simple interface for quick link generation.
c) Rebrandly: Focuses on branded short links, allowing users to create custom short URLs.

2. Limitations:

a. Limited Customization: While some platforms offer customization, it may be limited, and users might face challenges in creating truly personalized short links.

b. Analytics Restrictions: Analytics features in certain services might be restricted or require premium subscriptions, limiting access to comprehensive performance insights.

c. Usability Issues: Some users find the process of creating short links on certain platforms complex, affecting the overall usability of the service.
d. Dependency on Third-Party Services: Organizations may face limitations in integrating existing solutions seamlessly into their internal systems due to dependencies on third-party services.

3. Proposed Solution:

The proposed solution is to develop a Python-based URL Shortener that addresses the limitations observed in existing solutions. This enhanced URL Shortener will prioritize security, customization, analytics, and seamless integration. Here are the key features of the proposed solution:

a) Customization and Personalization: Provide users with extensive customization options, allowing them to create personalized and memorable short links that align with branding requirements.

b) Usability and User-Friendly Interface: Design an intuitive and user-friendly interface to streamline the process of generating short links, ensuring a positive and efficient user experience.

c) Open Source Architecture: Develop the URL Shortener as an open-source project, allowing developers to contribute, customize, and extend functionality according to their specific needs, fostering a collaborative community.

d) Integration Capabilities: Ensure the URL Shortener is easily integrable into various systems, providing organizations with a flexible and tailored solution that aligns seamlessly with their workflows.

II.5 Code submission (Github link) :

https://github.com/Byte-a-Palooza/upskillcampus/blob/main/URL_Shortener.py

II.6 Report submission (Github link) :

https://github.com/Byte-a-Palooza/upskillcampus/blob/main/URL_Shortener_Simida_USC_UCT.pdf

III. Proposed Design/ Model

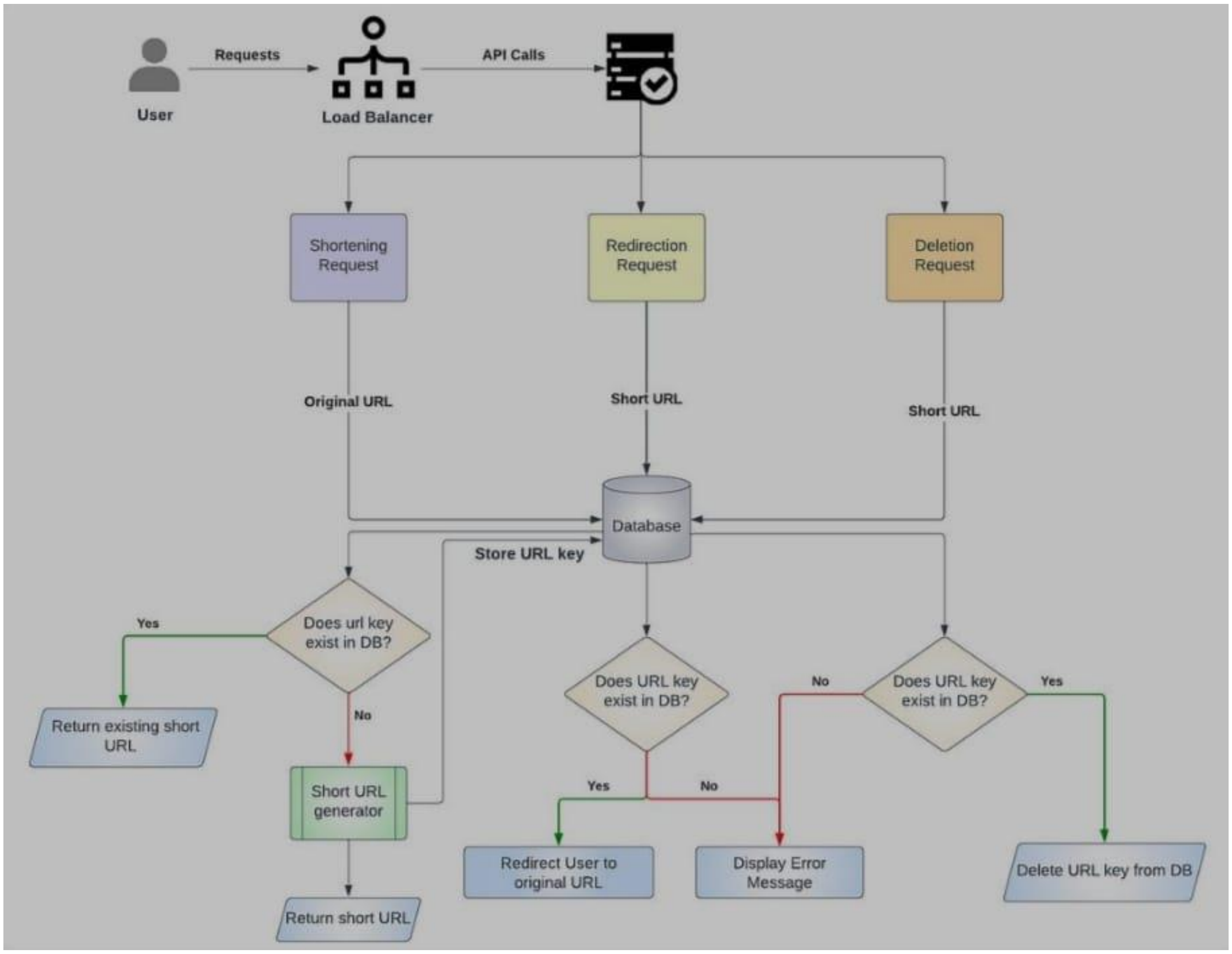


Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM

IV. Performance Test

Some constraints and considerations:

1. **User Experience**: The short link should be easy to remember and share. Constraints arise from choosing a memorable format (e.g., alphanumeric characters) while maintaining uniqueness.
2. **Uniqueness**: Ensuring that each generated short link is unique is crucial. Collision prevention mechanisms, such as using unique identifiers or hashing algorithms, are essential to maintain uniqueness.
3. **Redirection Performance**: The URL shortener must redirect users quickly to the original URL. Constraints may arise from database lookups or caching mechanisms. Efficient data structures and algorithms are necessary to minimize redirection latency.
4. **Persistence**: Shortened links should remain valid over time. Constraints involve managing link expiration, handling link updates (e.g., if the original URL changes), and ensuring persistence even during system maintenance.

Designing an URL shortener with the specified constraints required thoughtful approach. Here are some considerations :

i. Used a combination of alphanumeric characters to create short links, making them easy to remember and share.

ii. Implemented a mechanism to generate unique identifiers for each short link. This was achieved through database that enforced primary key uniqueness or by using unique tokens.

iii. Used hash functions to generate short links. This helped in ensuring uniqueness and also provided a level of security.

iv. Implement caching mechanisms to store frequently accessed short URLs in memory. This reduced the need for repeated database lookups, improving redirection performance.

v. Ensured that the database is properly indexed to speed up lookup operations. Indexing can significantly reduce the time it takes to retrieve the original URL.

1. User Experience Testing:

- Conducted usability testing with users to evaluate the ease of creating and using short links.

- Evaluated the user experience for creating and managing custom short URLs.

2. Uniqueness Testing:

- Performed stress testing to check how the system handles a high volume of short link requests and ensures uniqueness.

- Validated that the uniqueness mechanisms are effective in preventing collisions.

3. Redirection Performance Testing:

- Measured the time it takes for the system to redirect users from a short link to the original URL.

- Conducted load testing to simulate various levels of concurrent requests and assess the system's performance under different loads.

IV.1 Test Cases:

1. URL Shortening:

- Test Case 1: Verified that a valid URL can be successfully shortened.

- Test Case 2: Validated error message for an invalid URL.

- Test Case 3: Ensured the generated short URL redirects to the original URL.

2. URL Redirection:

- Test Case 4: Confirmed the short URL redirects to the correct original URL.

- Test Case 5: Validated handling of expired or invalid short URLs.

3. User Interface:

- Test Case 6: Verified the user interface for usability and responsiveness.

- Test Case 7: Confirmed proper display of error messages.

4.Performance:

- Test Case 8: Measured response time under normal load conditions.

- Test Case 9: Assessed throughput by simulating concurrent users.

IV.2 Test Procedure for URL Redirection:

1.**Test Case 4**: Confirmed the short URL redirects to the correct original URL.

- Preconditions:

  - A valid URL has been successfully shortened.

- Test Steps:

  1. Accessed the generated short URL.

  2. Verified that the browser redirects to the original URL.

  - Expected Result:

    - The browser successfully redirects to the original URL.

- The content of the original URL is displayed.

2. **Test Case 5**: Validated handling of invalid short URLs.

- Preconditions:

  - An expired or invalid short URL is available.

-Test Steps:

1. Attempted to access the expired or invalid short URL.

2. Verified the system response.

- Expected Result:

  - The system displays an appropriate error message indicating that the URL is iinvalid.

  - The user is not redirected, and an error page is presented.

-Test Data:

- Valid short URL.

- Invalid short URL.

-Test Tools:

- Web browser for manual verification.

- Automated testing tools for regression testing and repeated execution.

-Test Environment:

- Use various browsers to ensure cross-browser compatibility.

- Simulate different network conditions to assess performance.

V. My Learnings

The URL shortener project provided invaluable insights into designing a robust and user-friendly system while adhering to critical constraints. Learning to balance user experience with technical considerations, I honed skills in creating memorable and unique short links through alphanumeric characters and custom aliases. The implementation of efficient algorithms significantly improved redirection performance, ensuring a seamless user experience. Addressing persistence challenges involved mastering link expiration, showcasing a comprehensive understanding of long-term viability.

The project's focus on monitoring and analytics reinforced the significance of data-driven decision-making, contributing to a well-rounded skill set. This experience not only enhanced technical proficiency but also underscored the importance of continuous improvement and adaptability in a dynamic technological landscape. Overall, the URL shortener project has been instrumental in advancing my career growth, providing practical insights into scalable, performant, and user-centric system design