upSkill CAMPUS

AISSMS
INSTITUTE OF INFORMATION TECHNOLOGY
(IOIT)
ADDING VALUE TO ENGINEERING
An Autonomous Institute Affiliated to Savitribai Phule Pune University
Approved by AICTE, New Delhi and Recognised by Govt. of Maharashtra
Accredited by NAAC with "A+" Grade | NBA - 5 UG Programmes

uct

Industrial Internship Report on

”URL Shortener”

Prepared by

Shivani Santosh Magar

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

Firstly I select the domain of internship is python. Then there are few options to select the project. I selected the 'URL shortner' as a project to develop. In the first week of internship I learned the python for data science in that video is provided. Also gather all the requirements for the project.In week2 various conditional statements, loops syntax, example has been learn. In week 3 various libraries of python is learnt like numpy , pandas, matplotlib etc.In last week the development of project is completed.

Internships offer practical, hands-on experience in a real-world work environment. This experience helps individuals to bridge the gap between theoretical knowledge gained in classrooms and practical skills required in professional settings. Internships provide opportunities to develop and enhance various skills. Internships allow individuals to explore different career paths within their field of interest. By working in various roles and departments, interns can gain insights into different aspects of the industry and discover which career path aligns best with their skills, interests, and goals. Internship experiences add depth and credibility to one's resume. Employers value candidates who have practical work experience, and internships demonstrate a candidate's commitment to their chosen field and their ability to apply theoretical knowledge in real-world scenarios.

The problem statement is URL Shortner. 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.

The USC/UCT offers a very nice platform to students who have to enhance skills. This provides a fresher to a better experience. The program was planned of 4 weeks . It is from 10th February to 10th march. Every week contained a python libraries, loops , conditional statement, etc. This internship gives me a lot of knowledge about python and it was such a nice experience ever.

I would thanks to Ankit sir, Nitin tyagi sir, Jitesh Mathur sir, Sisodia sir who helped me throw out this journey. Thank you for your guidance.

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.

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..

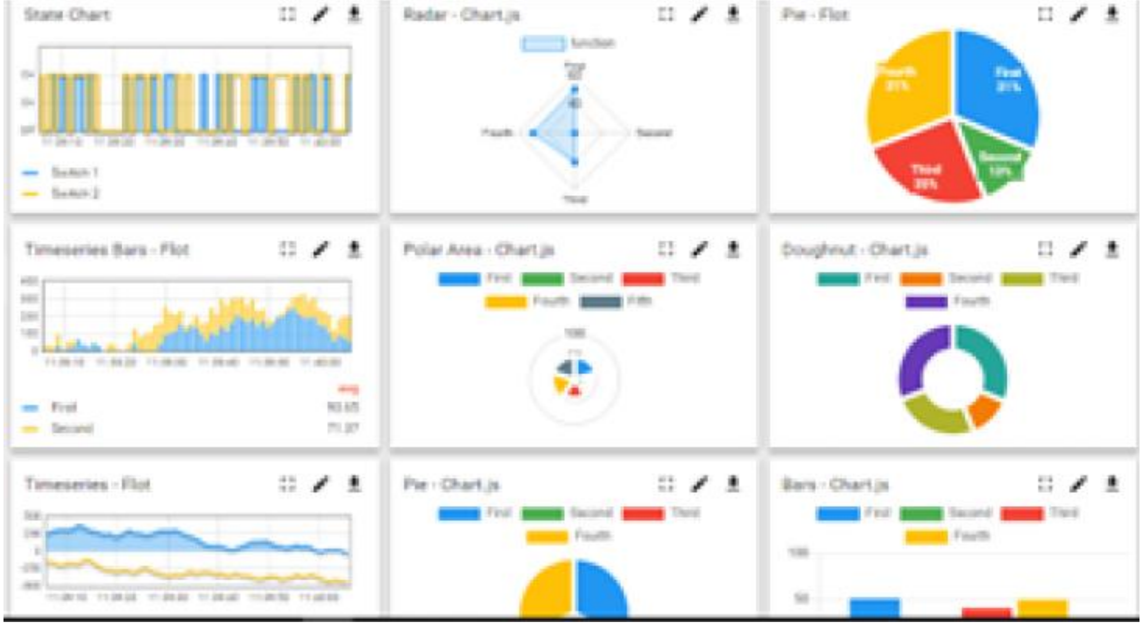
i. UCT IoT Platform (uct Insight)

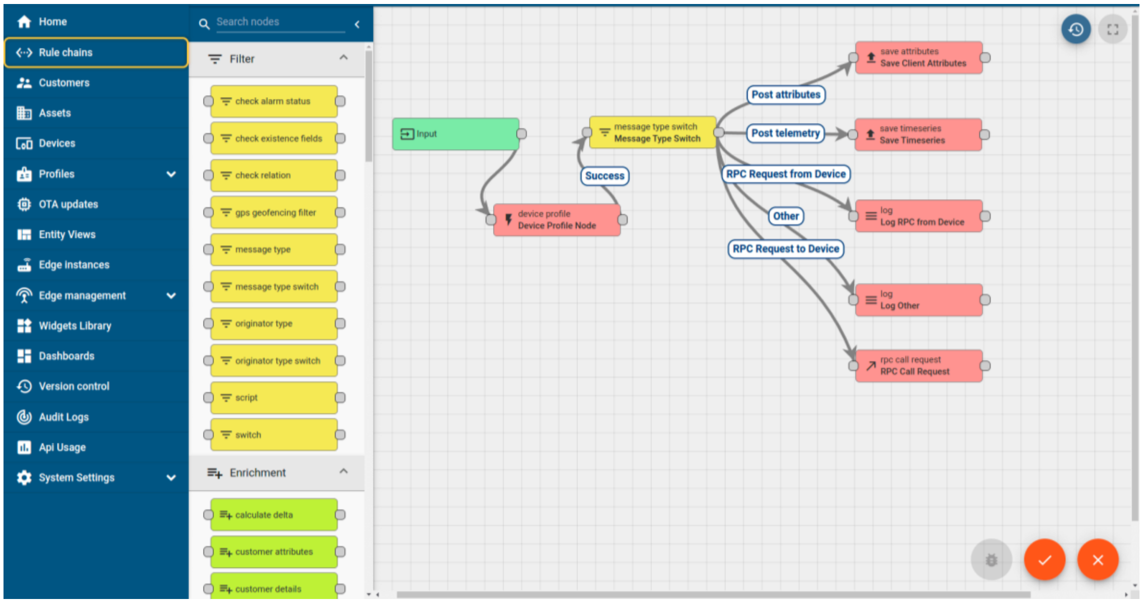
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- 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





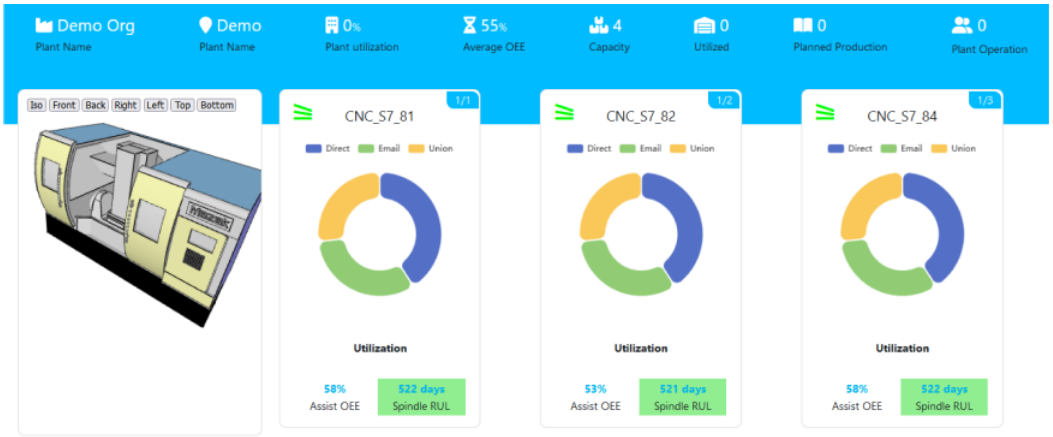
ii. Smart Factory Platform (FACTORY WATCH)

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.

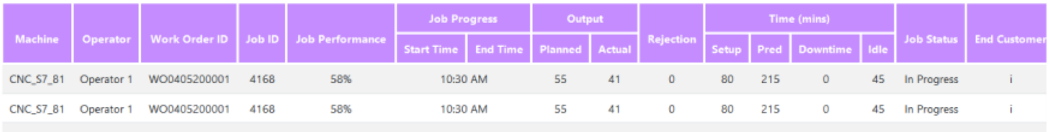


Demo Org — Plant Name
Demo — Plant Name
0% — Plant utilization
55% — Average OEE
4 — Capacity
0 — Utilized
0 — Planned Production
0 — Plant Operation

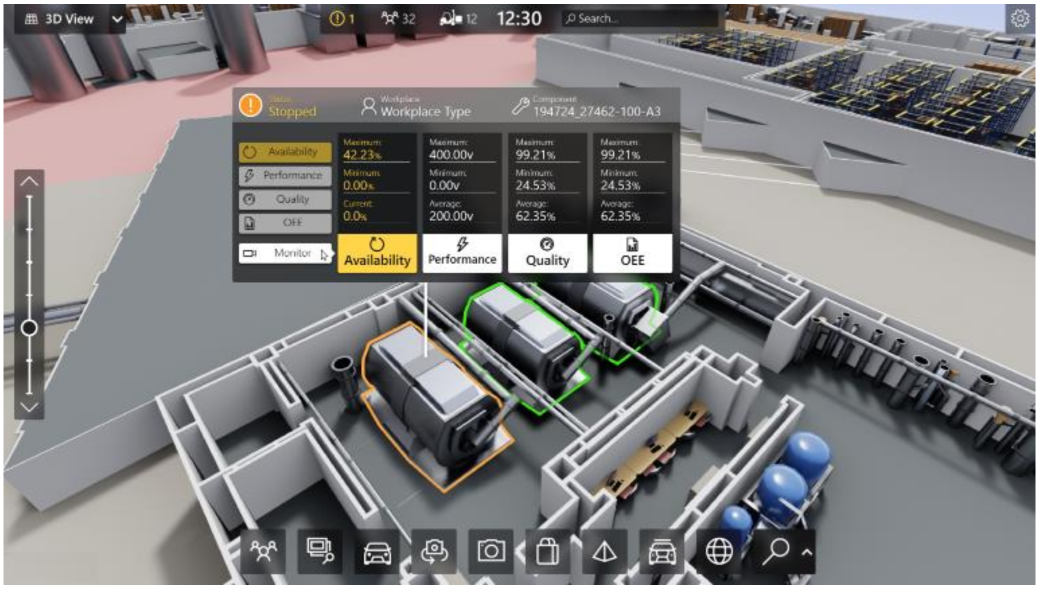
CNC_S7_81 (1/1) — Utilization — 58% Assist OEE — 522 days Spindle RUL

CNC_S7_82 (1/2) — Utilization — 53% Assist OEE — 521 days Spindle RUL

CNC_S7_84 (1/3) — Utilization — 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

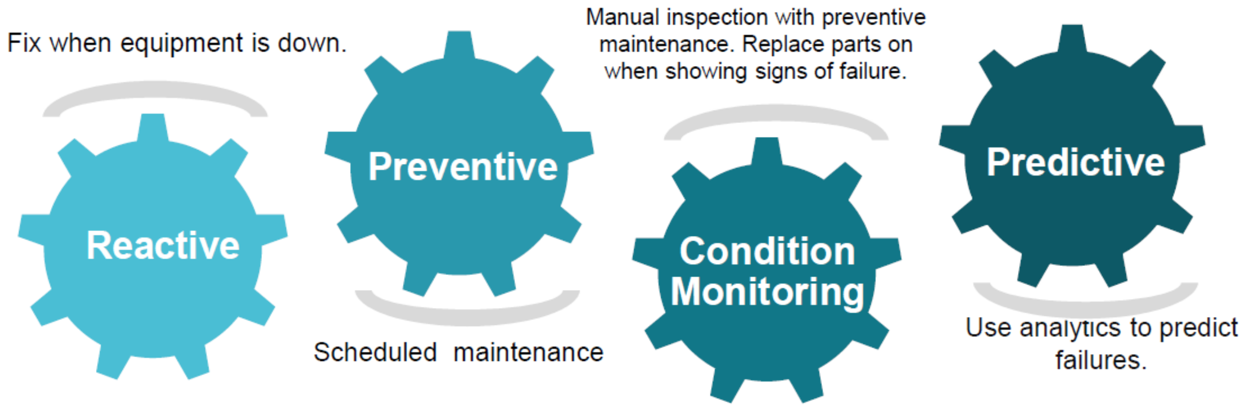
Availability	Performance	Quality	OEE
Maximum: 42.23%	Maximum: 400.00v	Maximum: 99.21%	Maximum: 99.21%
Minimum: 0.00%	Minimum: 0.00v	Minimum: 24.53%	Minimum: 24.53%
Current: 0.0%	Average: 200.00v	Average: 62.35%	Average: 62.35%

iii. LoRaWAN based Solution

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

iv. Predictive Maintenance

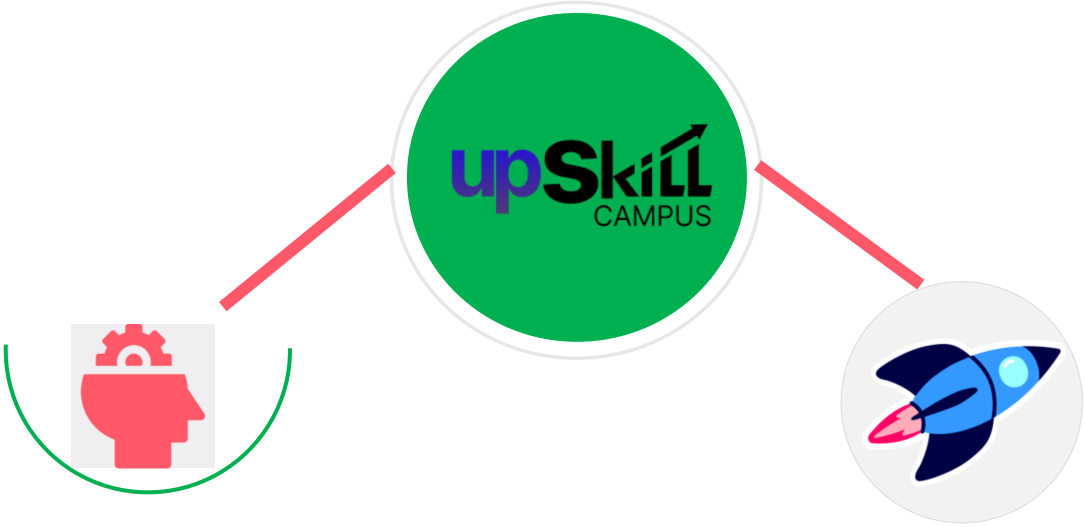
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.


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.

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.

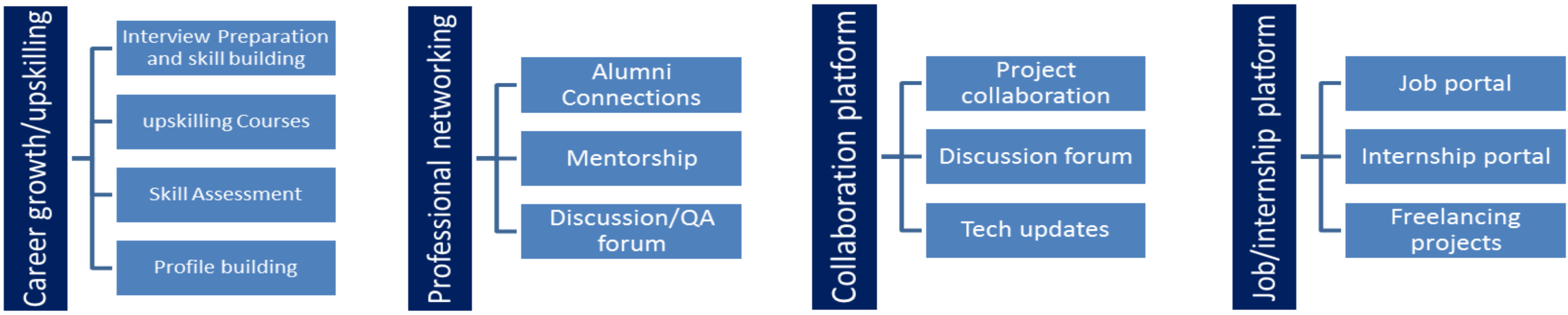
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



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] Edunet Foundation

2.6 Glossary

Terms	Acronym
UCT	Uniconverge Technologies
USC	Upskill Campus
IoT	Internet Of Things

3 Problem Statement

URL Shortener:

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.

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

User Interface: A simple web interface or API endpoint where users can submit long URLs and retrieve shortened ones.

Shortening Algorithm: Utilize libraries like hashlib in Python to generate hash values for long URLs, which are then truncated to create short identifiers.

I have proposed Improved Shortening Algorithm: Utilize a combination of hashing and encoding techniques for shorter and more unique short URLs.

Distributed Architecture: Break down the application into microservices using tools like Flask Blueprints or deploy serverless functions on platforms like AWS Lambda.

Caching Mechanism: Implement caching using libraries like redis-py to cache frequently accessed mappings.

Rate Limiting and Authentication: Use middleware like Flask-Limiter for rate limiting and integrate authentication mechanisms like JWT (JSON Web Tokens) using libraries such as PyJWT.

Analytics and Monitoring: Integrate analytics services like Google Analytics and monitoring tools like Prometheus and Grafana for tracking usage statistics and system health.

4.1 Code submission (Github link)

https://github.com/shivanismagar/upskillcampus2

4.2 **Report submission (Github link) :**

https://github.com/shivanismagar/upskillcampus2

5 Proposed Design/ Model

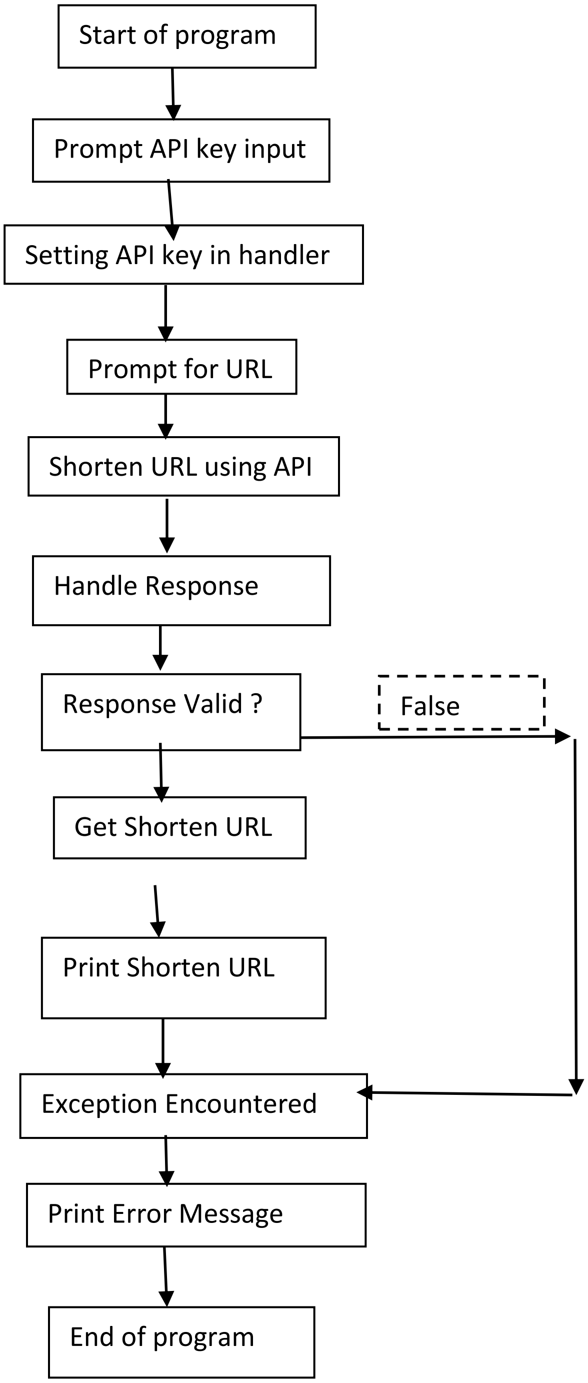
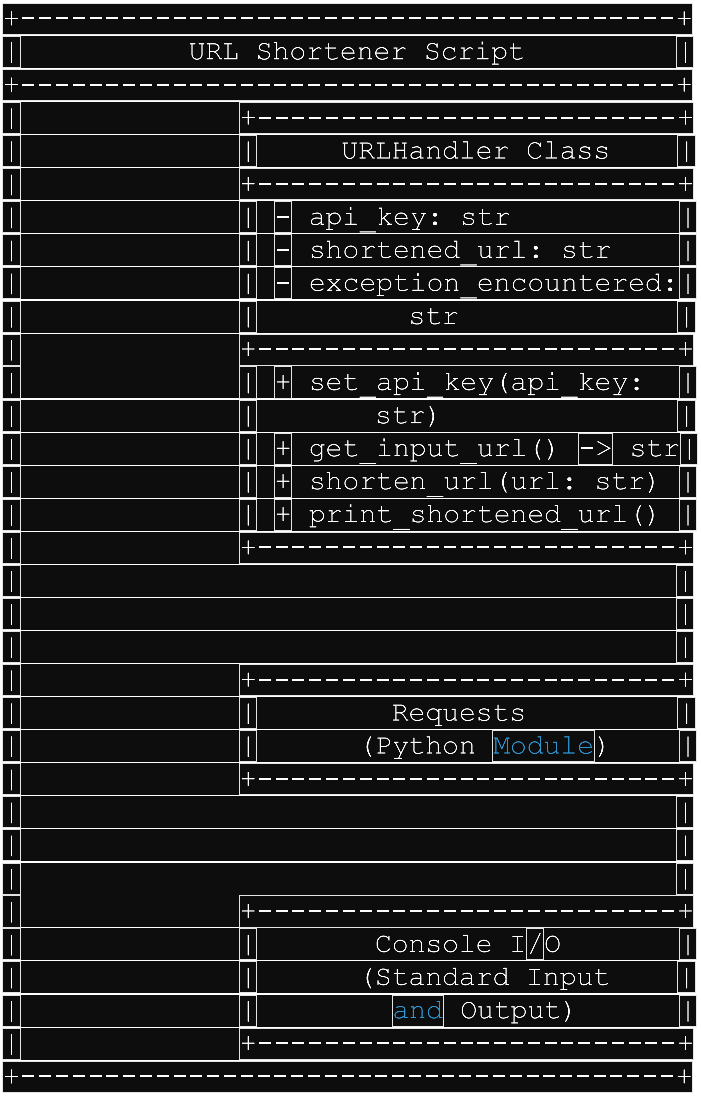


Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM

5.1 Interfaces (if applicable)



```
+---------------------------------------+
|          URL Shortener Script         |
+---------------------------------------+
|             +-------------------------+
|             |     URLHandler Class    |
|             +-------------------------+
|             | - api_key: str          |
|             | - shortened_url: str    |
|             | - exception_encountered:|
|             |         str             |
|             +-------------------------+
|             | + set_api_key(api_key:  |
|             |       str)              |
|             | + get_input_url() -> str|
|             | + shorten_url(url: str) |
|             | + print_shortened_url() |
|             +-------------------------+
|                                       |
|                                       |
|                                       |
|             +-------------------------+
|             |        Requests         |
|             |     (Python Module)     |
|             +-------------------------+
|                                       |
|                                       |
|                                       |
|             +-------------------------+
|             |       Console I/O       |
|             |     (Standard Input     |
|             |       and Output)       |
|             +-------------------------+
+---------------------------------------+
```

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.

Define Performance Metrics: Determine which performance metrics you want to measure. This could include response time, throughput (requests per second), error rate, and resource usage (CPU, memory).

Select Testing Tools: Choose appropriate tools for conducting performance testing. For HTTP-based testing, popular tools include Apache JMeter, Gatling, Locust, or even Python's requests module combined with libraries like concurrent.futures.

Create Test Scenarios: Develop test scenarios that mimic realistic usage patterns. For example, simulate simultaneous requests from multiple users, varying URL lengths, and different API key inputs.

Execute Tests: Run the performance tests with different configurations and load levels. Gradually increase the load to observe how the system behaves under stress.

Analyze Results: Collect and analyze performance data, including response times, throughput, and error rates. Identify any bottlenecks or performance issues.

Optimize and Retest: Based on the analysis, optimize the script or underlying infrastructure to improve performance. Then, rerun the tests to validate the improvements.

Monitor and Continuous Testing: Implement monitoring to continuously track the performance of the URL shortener in production. Regularly conduct performance tests to ensure the system can handle increasing loads and maintain acceptable response times.

6.1 Test Plan/ Test Cases:

1. Valid API Key Test:

Input: Valid API key.

Expected Output: The script should successfully set the API key without any errors.

2. Invalid API Key Test:

Input: Invalid or expired API key.

Expected Output: The script should handle the error appropriately and provide a descriptive error message.

3. Valid URL Test:

Input: Valid URL to be shortened.

Expected Output: The script should successfully shorten the URL and provide the shortened URL as output.

4. Invalid URL Test:

Input: Invalid URL (e.g., missing protocol).

Expected Output: The script should handle the error and provide a descriptive error message.

5. Empty URL Test:

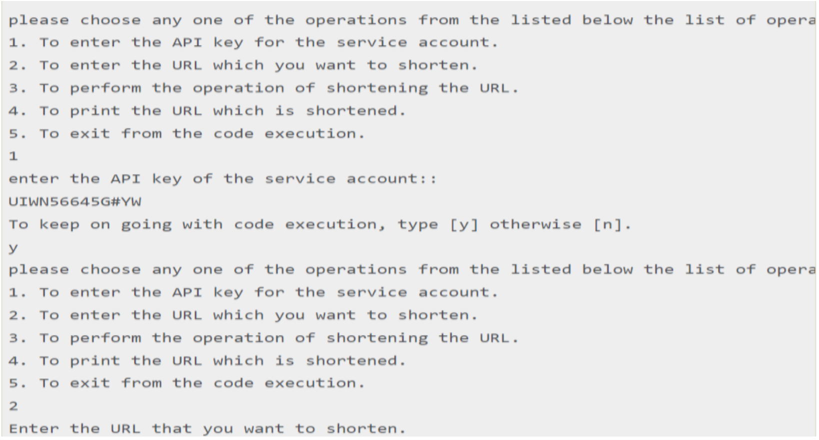
Input: Empty URL.

Expected Output: The script should handle the error and prompt the user to input a valid URL.

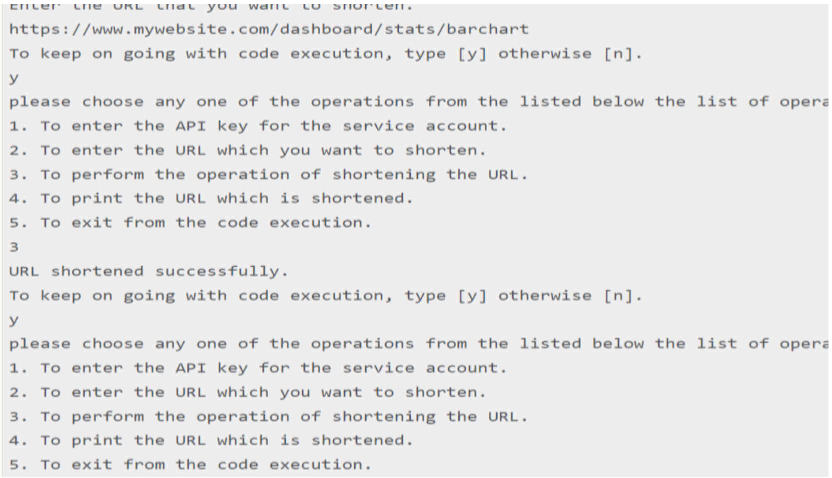
6.2 Test Procedure

- Preparation
- Test environment setup
- Test case execution
- Test case validation
- Performance Testing
- Error handling and recovery
- Documentation and reporting

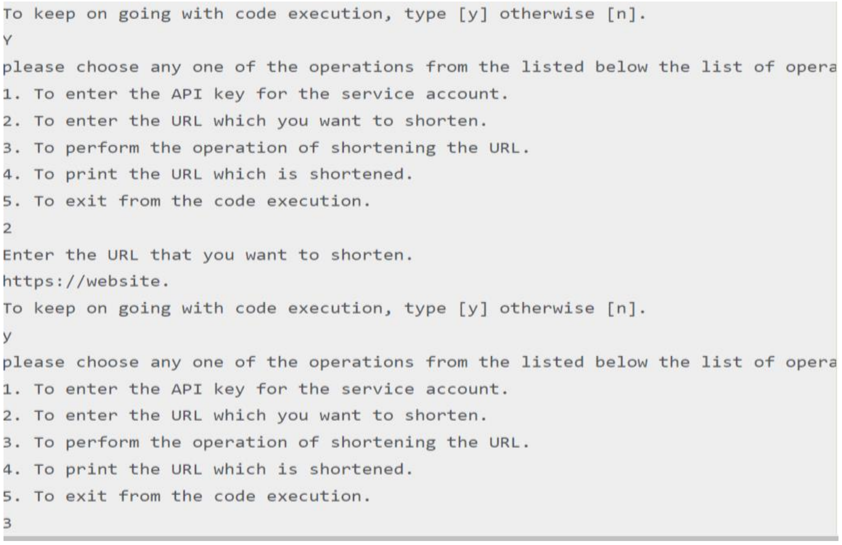
6.3 Performance Outcome



```
please choose any one of the operations from the listed below the list of opera
1. To enter the API key for the service account.
2. To enter the URL which you want to shorten.
3. To perform the operation of shortening the URL.
4. To print the URL which is shortened.
5. To exit from the code execution.
1
enter the API key of the service account::
UIWN56645G#YW
To keep on going with code execution, type [y] otherwise [n].
y
please choose any one of the operations from the listed below the list of opera
1. To enter the API key for the service account.
2. To enter the URL which you want to shorten.
3. To perform the operation of shortening the URL.
4. To print the URL which is shortened.
5. To exit from the code execution.
2
Enter the URL that you want to shorten.
```



```
Enter the URL that you want to shorten.
https://www.mywebsite.com/dashboard/stats/barchart
To keep on going with code execution, type [y] otherwise [n].
y
please choose any one of the operations from the listed below the list of opera
1. To enter the API key for the service account.
2. To enter the URL which you want to shorten.
3. To perform the operation of shortening the URL.
4. To print the URL which is shortened.
5. To exit from the code execution.
3
URL shortened successfully.
To keep on going with code execution, type [y] otherwise [n].
y
please choose any one of the operations from the listed below the list of opera
1. To enter the API key for the service account.
2. To enter the URL which you want to shorten.
3. To perform the operation of shortening the URL.
4. To print the URL which is shortened.
5. To exit from the code execution.
```



```
To keep on going with code execution, type [y] otherwise [n].
Y
please choose any one of the operations from the listed below the list of opera
1. To enter the API key for the service account.
2. To enter the URL which you want to shorten.
3. To perform the operation of shortening the URL.
4. To print the URL which is shortened.
5. To exit from the code execution.
2
Enter the URL that you want to shorten.
https://website.
To keep on going with code execution, type [y] otherwise [n].
y
please choose any one of the operations from the listed below the list of opera
1. To enter the API key for the service account.
2. To enter the URL which you want to shorten.
3. To perform the operation of shortening the URL.
4. To print the URL which is shortened.
5. To exit from the code execution.
3
```

7 My learnings

Through writing code for the project, I became familiar with Python's syntax and language features, such as variable assignment, control structures (if statements, loops), functions, and classes. Working with URLs involves string manipulation, including concatenation, slicing, and formatting. Integrating an external API into your project provided you with hands-on experience in making HTTP requests, parsing JSON responses, and handling API errors and exceptions. Storing and managing data, such as mappings between short and long URLs, likely involved using Python data structures like dictionaries or lists. Dealing with potential errors and exceptions, such as network errors or invalid API responses, taught you about Python's exception handling mechanisms and how to handle errors gracefully in your code.

8 Future work scope:

- Analytics and Reporting: Implement analytics capabilities to track usage statistics, such as the number of clicks, geographic locations of users, and referral sources. Providing insights through reports and dashboards can help users understand the impact of their shortened URLs.
- Link Management Features: Enhance the link management capabilities by allowing users to categorize, tag, or organize their shortened URLs into folders. This can help users better manage and navigate their links.
- Integration with external services : Integrate the URL shortener with other external services, such as social media platforms, email marketing tools, or content management systems. This integration can streamline workflows and provide additional functionality for users.
- Mobile Application : Develop a mobile application for iOS and Android platforms, providing users with a convenient way to create and manage shortened URLs on the go.