**Industrial Internship Report on**

**”Python code of URL shortener”**

**Prepared by**

**SUMIT SINGH**

|  |
| --- |
| *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 a Python project for a URL shortener it is a useful and straightforward code that converts long URLs into shorter ones. URL shorteners are commonly used to create more manageable and shareable links, especially for platforms with character limitations like social media.  Here's a basic overview of how I create a URL shortener project in Python:  Dependencies and Tools:  - Python: I need Python installed on my system. Python 3 is used as it's the latest stable version.  Setup:  - I create a new directory for my project and set up a virtual environment to manage dependencies.  Shortening Algorithm:  - I create a function to generate the shortened URL. I use various methods, like hashing algorithms (e.g., MD5, SHA-256) or base conversion (e.g., base64 encoding), to generate unique shortcodes for each URL.  Shortened URL Handling:  - Define a route to handle the redirection from a shortened URL to the original URL. When a user accesses the shortened URL, the server should look up the original URL in the database and redirect the user accordingly.  Error Handling and Validation:  - Implement error handling to deal with invalid input or potential issues when processing URLs.  Testing:  - Test my URL shortener thoroughly to ensure it works as expected. Consider edge cases and handle them appropriately.  Deployment:  - Once my URL shortener is ready, I deploy its code.  This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solutions for them. It was an overall great experience to have this internship. |

**TABLE OF CONTENTS**

[1 Preface 3](#_Toc139702806)

[2 Introduction 4](#_Toc139702807)

[2.1 About UniConverge Technologies Pvt Ltd 4](#_Toc139702808)

[2.2 About upskill Campus 8](#_Toc139702809)

[2.3 Objective 9](#_Toc139702810)

[2.4 Reference 9](#_Toc139702811)

[2.5 Glossary 10](#_Toc139702812)

[3 Problem Statement 11](#_Toc139702813)

[4 Existing and Proposed solution 12](#_Toc139702814)

[5 Proposed Design/ Model 13](#_Toc139702815)

[5.1 High Level Diagram (if applicable) 13](#_Toc139702816)

[5.2 Low Level Diagram (if applicable) 13](#_Toc139702817)

[5.3 Interfaces (if applicable) 13](#_Toc139702818)

[6 Performance Test 14](#_Toc139702819)

[6.1 Test Plan/ Test Cases 14](#_Toc139702820)

[6.2 Test Procedure 14](#_Toc139702821)

[6.3 Performance Outcome 14](#_Toc139702822)

[7 My learnings 15](#_Toc139702823)

[8 Future work scope 16](#_Toc139702824)

# Preface

I have learned many things about python in this internship like Pandas ,Numpy, functions like IF ,If-Else and many more terms like series, tuple, etc.

I also learn to run codes in python and done a project of URLS shorteners .

Many times its complicated for everyone to copy and use large URLs and its sometimes gets irritating if the URLs are very lengthy so to eliminate this problem I worked on this project.

Opportunity given by USC/UCT was uncommendable as its provide me an opportunity to learn a new skill and gave me the chance to explore my technical skills.



My experience was really good with lots of information and learnings throughout this project.

Thanks to all USC/UCT team members to be so supportive and motivative, who have helped me to learn this new skill directly or indirectly.

I want my juniors to work on such new skills which are required by the industry so they can keep themselves up to date.

# 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.



1. UCT IoT Platform **(****)**

**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

 

1. **Smart Factory Platform (****)**

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.

 

1.  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.

1. 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.



## 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

<https://www.upskillcampus.com/>

upSkill Campus aiming to upskill 1 million learners in next 5 year



## 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.

## 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.

# Problem Statement

In the assigned problem statement of long URLs that sometimes irritate peoples and can be a hectic task to copy such long urls and search the link.

Lengthy URLs can present various difficulties in real-life scenarios. Some of the common challenges and issues associated with lengthy URLs include:

1. User Experience: Long URLs can be challenging to read, remember, and share. They can look cluttered and confusing, leading to potential errors in manual entry or when shared verbally.

2. Social Media: Many social media platforms have character limits for posts. Long URLs can take up a significant portion of this limit, leaving less space for meaningful content or additional hashtags.

3. Text Messaging: In SMS or other text-based communication, long URLs can be split into multiple messages, making it difficult for recipients to click on the link directly.

4. QR Codes: Long URLs can generate complex and dense QR codes, which may become less scannable and harder for QR code readers to interpret.

5. Link Sharing: Some platforms or applications may automatically truncate long URLs, potentially breaking the link and rendering it non-functional.

6. SEO and Analytics: Long URLs with multiple query parameters and dynamic content may negatively impact search engine optimization (SEO) and tracking analytics.

7. Branding and Aesthetics: Lengthy URLs can detract from the overall visual appeal of a website or promotional material, making it less aesthetically pleasing.

8. Link Decay: If a URL is too long and not properly formatted, it can lead to issues with link decay, where the URL becomes invalid due to line breaks or spaces inserted during copy-pasting.

9. Security Concerns: Long URLs can obscure the destination, making it challenging for users to identify potential phishing or malicious links.

10. QR Code and Barcode Generation: In printed materials like brochures or business cards, long URLs may not fit well within the limited space available for QR codes or barcodes.

# Existing and Proposed solution

To overcome the difficulties stated above, many web services, applications, and marketing campaigns use URL shorteners to create concise and more manageable URLs. URL shorteners can create short and user-friendly links that redirect to the original, lengthy URLs, thereby mitigating the issues associated with long URLs.

However, it's essential to be cautious when using URL shorteners, especially for critical links or in sensitive contexts, as they can obscure the actual destination and potentially lead to security risks. Always ensure that you trust the source of the shortened link before clicking on it.

## Code submission (Github link): https://github.com/sumit412singh/Upskill-Sumitsingh-Python-project-url.git

# 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.

# My learnings

My learning from Python and URL shorteners can be a valuable skill set that allows me to build practical and useful application of real time url shortner.

1. Learn Python Basics:

Start by learning the fundamentals of Python programming. Understand variables, data types, control structures (if-else, loops), functions, and object-oriented programming concepts. There are numerous online resources, tutorials, and courses available for Python beginners.

2. Web Development Basics:

Familiarize yourself with web development basics. These are essential for creating user interfaces for your URL shortener application.

3. Choose a Framework:

Decide on the web framework you want to use for building your URL shortener. Popular choices include Flask and Django. Flask is lightweight and easy to get started with, while Django offers a more comprehensive set of features.

4. Setting Up the Project:

Create a new project directory, set up the virtual environment, and install the necessary dependencies (Flask/Django) using `pip`.

5. Design the Database:

Decided how i'll store the mapping between short URLs and long URLs.

6. URL Shortening Logic:

Implement the URL shortening logic in your Python code. This involves generating a short code for a given long URL and storing the mapping in the database.

# Future work scope

The future scope of URL shorteners remains promising, with potential opportunities for innovation and expansion. While the core functionality of converting long URLs to shorter ones remains constant, the following areas offer exciting possibilities for URL shorteners:

1. Improved Analytics and Tracking: Enhancing the analytics and tracking capabilities of URL shorteners can provide valuable insights to businesses and marketers. Advanced tracking features may include click-through rates, geographical location of users, device types, and conversion metrics.

2. Link Customization and Branding: Customizable short URLs that incorporate branding elements, such as company names or keywords, can make links more recognizable and trust-worthy. This feature can be particularly useful for businesses and marketing campaigns.

3. Security Enhancements: Strengthening security measures to prevent abuse, spam, and malicious links is critical for building trust and maintaining user safety. Implementing anti-phishing mechanisms and link expiration policies can be essential improvements.

4. API Integration: Providing robust and well-documented APIs will enable developers to integrate URL shortening services seamlessly into their applications, websites, and services.

5. QR Codes and Deep Linking: Integrating with QR code generation and supporting deep linking can improve the user experience and expand the usability of URL shorteners across various platforms and marketing channels.

6. Domain Whitelisting: Allowing users to whitelist specific domains for shortening URLs can prevent misuse and unauthorized URL shortening.

7. Real-time Link Management: Offering users the ability to edit, update, or delete shortened URLs in real-time can add flexibility and convenience to URL shortener services.

8. Link Bundling and Sharing: Enabling users to bundle multiple links into one shortened URL can facilitate easier sharing and tracking of multiple resources.

9. Integration with Social Media Platforms: Seamless integration with popular social media platforms can streamline the sharing of shortened URLs on sites like Twitter, Facebook, and Instagram.

10. Smart URL Shortening: Implementing intelligent algorithms to generate meaningful short codes based on the content of the long URL can improve URL readability and user experience.

11. Personalization and User Profiles: Introducing user accounts and profiles can allow users to manage and track their shortened URLs, view link performance, and access additional features.

12. Global Reach and Multi-language Support: Expanding the availability of URL shorteners to support various languages and handle internationalized domain names (IDNs) can cater to a more diverse user base.

13. Mobile App Integration: Developing dedicated mobile apps for URL shortening can provide additional convenience and accessibility to users.

As technology evolves and user needs change, the future of URL shorteners will likely see continuous innovation and integration with emerging technologies. However, it is essential to consider potential challenges, such as link decay, security vulnerabilities, and ethical use of data, while exploring the future possibilities of URL shortening services.