Industrial Internship Report on

”Url Shortener”

Prepared by

Sarthak Padmakar Boralkar

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: " I'm working on a project to build short URLs in Python. It will generate short URLs, store mappings between short and long URLs, and let users download from short URLs to their URLs." the corresponding lengths." It will handle the redirect request centrally 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 5

2.1 About UniConverge Technologies Pvt Ltd 5

2.2 About upskill Campus 9

2.3 Objective 11

2.4 Reference 11

2.5 Glossary 11

3 Problem Statement 12

4 Existing and Proposed solution 13

5 Proposed Design/ Model 16

5.1 High Level Diagram (if applicable) 18

5.2 Low Level Diagram (if applicable) 19

5.3 Interfaces (if applicable) 20

6 Performance Test 20

6.1 Test Plan/ Test Cases 22

6.2 Test Procedure 22

6.3 Performance Outcome 22

7 My learnings 23

8 Future work scope 24

1 Preface

Summary of the whole 6 weeks' work.

Week 1: Introduction to Upskill Development and an overview of Uniconverge Technology

During the first week of the internship, trainees will be introduced to the concept of upskilling, and the importance of continuous learning and technical skill development will be emphasized in a rapidly developing environment. Additionally, interns will receive an overview of Uniconverge Technologies, including its core values, mission, and areas of operation. This introductory phase lays the foundation for training by providing trainees with an understanding of the company's culture and goals.

Week 2: Introductory projects and projects using a strategic plan

During the second week, trainees will explore their specific projects and projects during their training. This includes a detailed introduction to the projects, their objectives, and related technologies. Trainees will receive guidance on project design, including setting up development environments and understanding project requirements. This orientation ensures that trainees have a clear understanding of their responsibilities and are ready to start working effectively in their jobs.

Week 3: Python Training

The third week of the internship focuses on Python training, aimed at enhancing the trainee's programming skills. Trainees will participate in a variety of Python activities and exercises to strengthen their understanding of the language and its basic concepts. To support their learning, trainees have access to extensive Python documentation and learning resources. Additionally, trainees will explore important Python libraries commonly used in software development, their uses and best practices for their use.

Week 4: Program work

During the final week of the internship, interns will transition into administrative work, where they will undertake individual projects independently under the guidance of mentors. This hands-on experience allows the intern to apply the skills and knowledge gained throughout the internship to real-world situations. Support and resources are available for trainees to help with problem solving and overcome any challenges they may encounter during the project. Emphasis will be placed on gaining practical experience and understanding the importance of working in the real world to prepare for future career opportunities.

All of the learning and experience I have gained during internships has been invaluable. Through structured, hands-on training and mentoring, I gained a deeper understanding of various technologies and their practical applications. The opportunity to work on real-world projects sharpened my problem-solving skills and gave me valuable business experience.

I am extremely grateful to everyone who has contributed directly and indirectly to my learning journey. Special thanks to the IoT Academy for providing informative videos that expanded my knowledge base and helped me better understand complex concepts. Additionally, I would like to thank UniConverge Technologies for providing this internship opportunity and providing insight into their organizational culture, values and areas of expertise. Their support and guidance have been invaluable in my professional development.

Overall, this internship has been a transformative experience, and I am grateful for the knowledge, skills and relationships I gained throughout the journey and look forward to applying these lessons in future endeavors and moving on have made progress in my chosen profession. Your message to your juniors and peers.

Message to Junior: - Dear Junior,

Believe in yourself, keep learning, don't hesitate to ask for help, and enjoy the journey.

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.

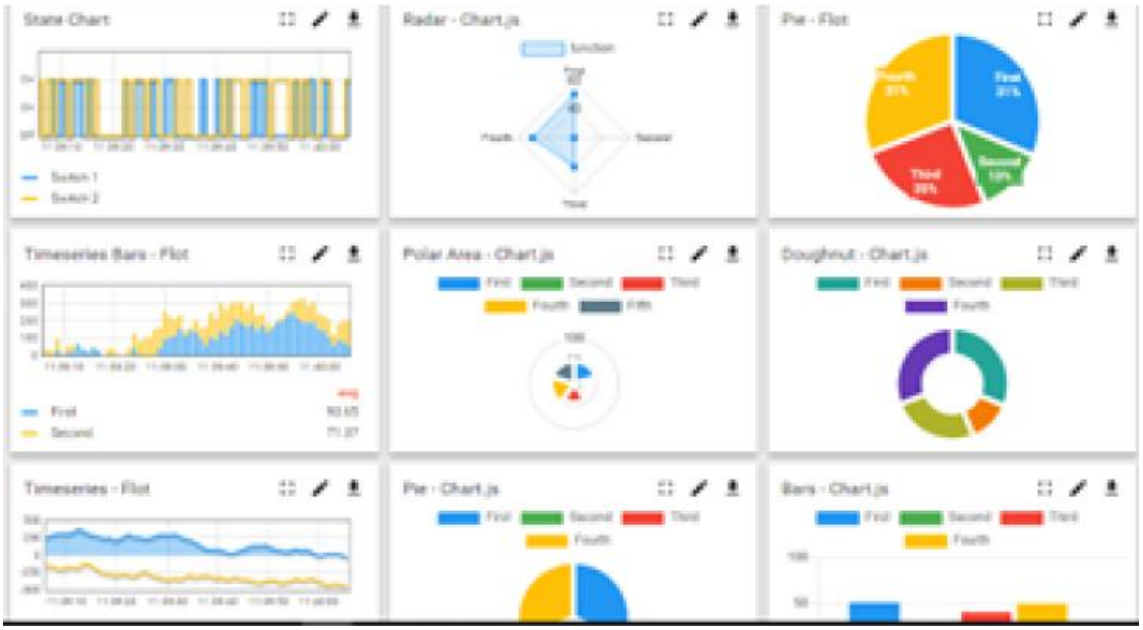
i. UCT IoT Platform (UCT Insight)

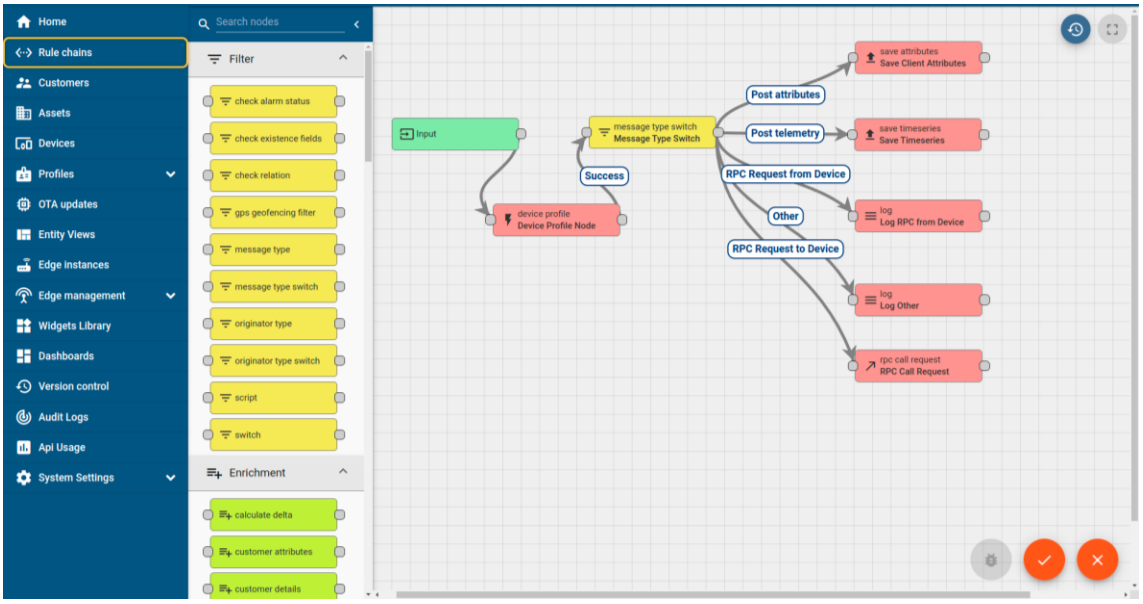
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- 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.

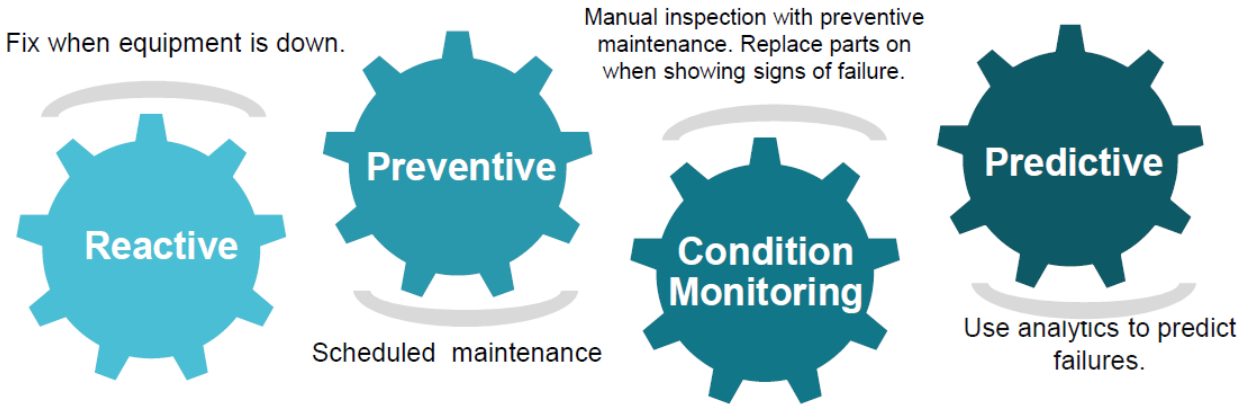
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1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

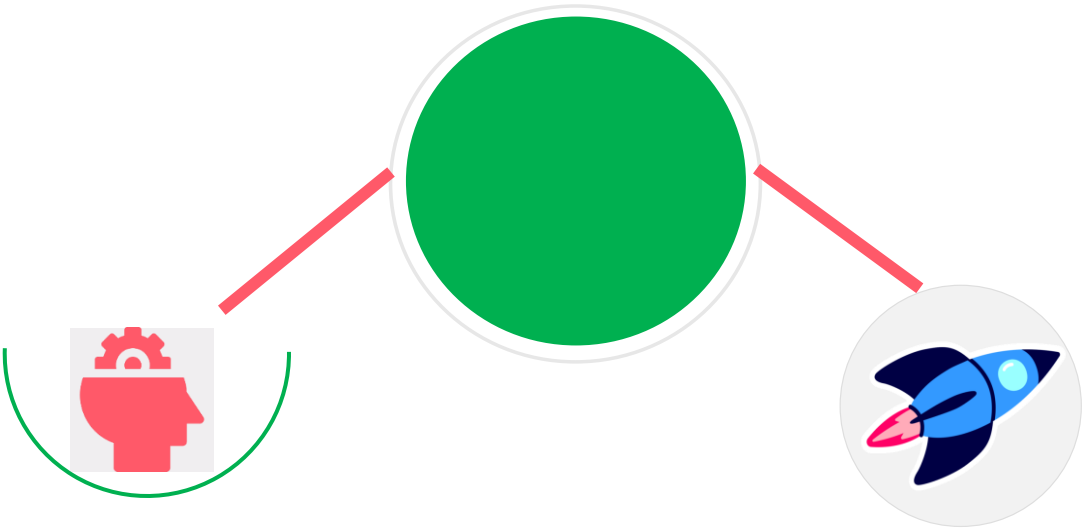
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



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.

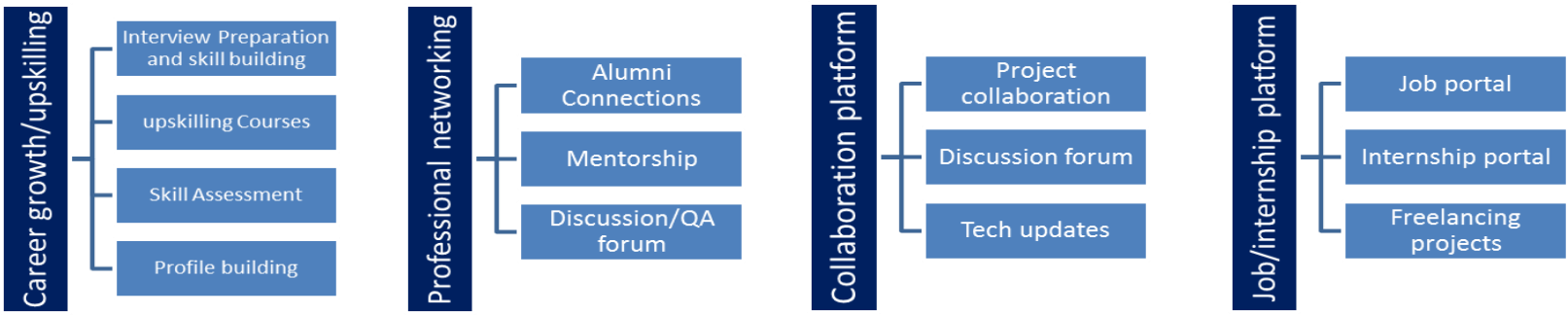
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



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] The Iot Academy

[2] Offline Python Documentation

[3] Community Forum

2.6 Glossary

Terms	Acronym
URL	Uniform Resource Locator
MVT	Model View Template
HTTPS	Hyper Text Transfer Text Protocol Security

3 Problem Statement

The "nUrl Statement" challenge objectives to increase a complicated URL shortener carrier that successfully converts lengthy URLs into shorter, more viable links. This mission addresses the need for a dependable and scalable method to simplify the sharing of web addresses, especially in contexts wherein individual remember is constrained, along with on social media platforms or in messaging applications.

At its center, the task includes designing and implementing a system that takes a long URL as enter and generates a completely unique, shortened alias for it. This alias needs to be concise yet extraordinary, making sure that it accurately represents the unique URL while minimizing its person depend. Additionally, the device needs to cope with a potentially large volume of URL conversions successfully, maintaining rapid reaction instances even beneath heavy load.

One key thing of the nUrl Statement assignment is making sure the uniqueness of each shortened URL alias. This involves imposing robust algorithms or techniques to generate aliases which might be not going to collide with present ones. Duplicate aliases ought to cause confusion or unintentional redirection, undermining the usability and reliability of the service.

Additionally, the project includes features such as URL optimization and analytics tracking. Users want to be able to rename their shortened URLs, allowing for personalized and memorable links. Additionally, the system can track metrics such as the number of times each shortened URL is accessed, providing valuable insights into user engagement and usage patterns.

Security is another important consideration in the nUrl Statement function. Systems must be protected against potential abuse, such as the creation of shortened URLs for malicious purposes or unauthorized access to sensitive information through internal authentication mechanisms is robust, data authentication and security best practices are essential to protect service integrity and reliability.

Overall, the nUrl Statement project presents an exciting opportunity to develop a powerful and versatile URL shortening solution that meets the needs of today’s Internet users with new technologies and robust security protocols and features a user-friendly combination, the project aims to facilitate shared web addresses in various reliable online contexts -Easy-to-use service delivery

4 Existing and Proposed solution

Existing solutions offered by others include various URL shortening services commonly found on the internet, such as:

1. Bitley:

- Bitly is one of the most popular URL shortening services.

- Provides basic link shortening functionality and analysis features.

- Users can create custom short links and track click-through rates, users' geographic location, and referral sources.

- Bitly offers free and premium subscription plans with additional features.

- Website: [Bitly](https://bitly.com/)

2. TinyURL:

- TinyURL is another widely used URL shortening service.

- This allows users to shorten long URLs.

- TinyURL offers basic link management features but lacks advanced analytics capabilities.

- Users can create custom aliases for their shortened URLs.

- TinyURL is free to use and does not require registration.

- Website: [TinyURL](https://tinyurl.com/).

3. Renamed:

- Rebrandly is a URL shortening service that focuses on branding and optimization.

- Offers extensive customization options, allowing users to create branded short links and custom domains.

- Rebrandly offers analytics features to track link performance and user engagement.

- The service offers both free and premium plans with additional features such as link retargeting and link management.

- Website: [Rebranded](https://www.rebrandly.com/).

4. Ov.Lee (Hootsuite):

- Ow.ly is part of the Hootsuite social media management platform.

- Provides integrated URL shortening services and social media planning and analytics.

- Ow.ly adds basic link management features to analytics and tracking, including click-through rates and social media engagement metrics.

- The role is primarily based on businesses and marketers who manage social media campaigns

- Website: [Ow.ly](https://hootsuite.com/products/owly).

This service provides basic functionality for shortening long URLs, and allows users to quickly share links. However, restrictions are often imposed e.g.

1. Limited Customization: Many existing solutions offer limited customization options for shortened URLs. Users cannot customize or customize a title to reflect their logo or content.

2. Relying on Third Party Services: Users of existing URL shorteners often rely on third party services to manage their links. This raises concerns about data privacy, reliability, and possible service outages.

3. Analytics Restrictions: While some services offer basic analytics features, they may impose restrictions or require a premium subscription to get a general view of link performance and user engagement.

4. Scalability and Performance: Depending on the service provider's infrastructure and resources, existing URL shorteners may face limitations in scalability and performance, resulting in slow response times or performance disruptions over time in widespread use.

5. Security Risks: There may be security risks associated with using short URLs, including the possibility of hacking or redirecting users to websites on a terrible scale.

Our proposed solution for the nUrl Statement project aims to address these limitations and provide a comprehensive, user-friendly, and secure URL shortening service. Key features and value additions of our proposed solution include:

1. Advanced Customization: Our solution will offer extensive customization options, allowing users to personalize the shortened URL alias, customize it with their brand name or keywords, and even choose from various domain options for the shortened URL.

2. Robust Analytics: We will implement robust analytics tracking features, providing users with detailed insights into link performance, including click-through rates, geographic distribution of users, and referral sources. These analytics will help users understand the effectiveness of their shared links and optimize their marketing or promotional strategies accordingly.

3. Scalability and Performance: Our solution will be built on a scalable and high-performance infrastructure, ensuring fast response times and reliable service even under heavy load. We will leverage cloud technologies and optimization techniques to efficiently handle a large volume of URL conversions and user traffic.

4. Enhanced Security: Security will be a top priority in our solution. We will implement stringent authentication mechanisms, data encryption, and proactive monitoring to protect against potential security threats such as phishing attacks, unauthorized access, and data breaches.

Overall, our proposed solution for the nUrl Statement project aims to provide users with a comprehensive and reliable URL shortening service that meets their needs for customization, analytics, scalability, performance, and security. By addressing the limitations of existing solutions and incorporating value-added features, we aim to deliver a superior user experience and empower users to effectively manage and share their links online

4.1 Code submission (Github link): SarthJain37/Url-Shortener (github.com)

4.2 Report submission (Github link): Url-Shortener_Sarthak_USC_UCT.pdf

5 Proposed Design/ Model

The design flow of our proposed URL shortening solution:

1. User-Interface (UI) Design:

- The user interface will provide users with an easy and convenient way to interact with the URL shortener service.

- Users will have options to insert long URLs that they want to shorten, and to change the alias if they want.

- Additionally, the UI will add features for users to view analytics data, manage their short links, and access statistical settings.

2. External Options:

- Develop the backend infrastructure using a microservices approach, allowing for modularity, scalability and flexibility.

- Key components of the backend system include:

- URL Shortening Service: Responsible for creating unique short aliases for long URLs and checking mappings between short and long URLs.

- Analytics Service: Handles tracking and analytics functionality, recording data such as click- through rates, geographic location of users, and referral sources

- Authentication Service: Manages user authentication and authorization, ensuring secure access to the service.

- Database: Stores information about URL shortenings, analytics data, and user accounts.

3. URL Shortening Function:

- If a user submits a long URL to be shortened, the short URL function will generate a unique short name for it.

- The service will ensure that the generated alias is unique to avoid crashes and ensure that each shortened URL is unique.

- The shortened URL, along with its corresponding long URL, will be stored in the database for future reference.

4. Customization Options:

- Users will have the option to customize the alias of their shortened URL, allowing for personalization and branding.

- Customization features may include choosing a custom alias, selecting a preferred domain, and specifying additional parameters for the shortened URL.

5. Analytics Tracking:

- The Analytics Service will track various metrics related to shortened URLs, such as click-through rates, geographic location of users, and referral sources.

- Analytics data will be recorded in the database and presented to users through the UI in an intuitive and visually appealing manner, such as charts and graphs.

6. Security Measures:

- Security will be a top priority in the design of our solution.

- We will implement robust authentication mechanisms to ensure secure access to the service, including features such as multi-factor authentication and token-based authentication.

- Additionally, we will employ encryption techniques to protect sensitive data, such as user credentials and analytics information, both in transit and at rest.

7. Scalability and Performance:

- The solution will be designed to be highly scalable and performant, capable of handling a large volume of URL conversions and user traffic.

- We will leverage cloud technologies and horizontal scaling techniques to dynamically allocate resources based on demand and ensure optimal performance.

By following this design flow, our proposed URL shortener solution will provide users with a comprehensive, user-friendly, and secure platform for shortening URLs, managing links, and tracking analytics data effectively.

5.1 High Level Diagram (if applicable)

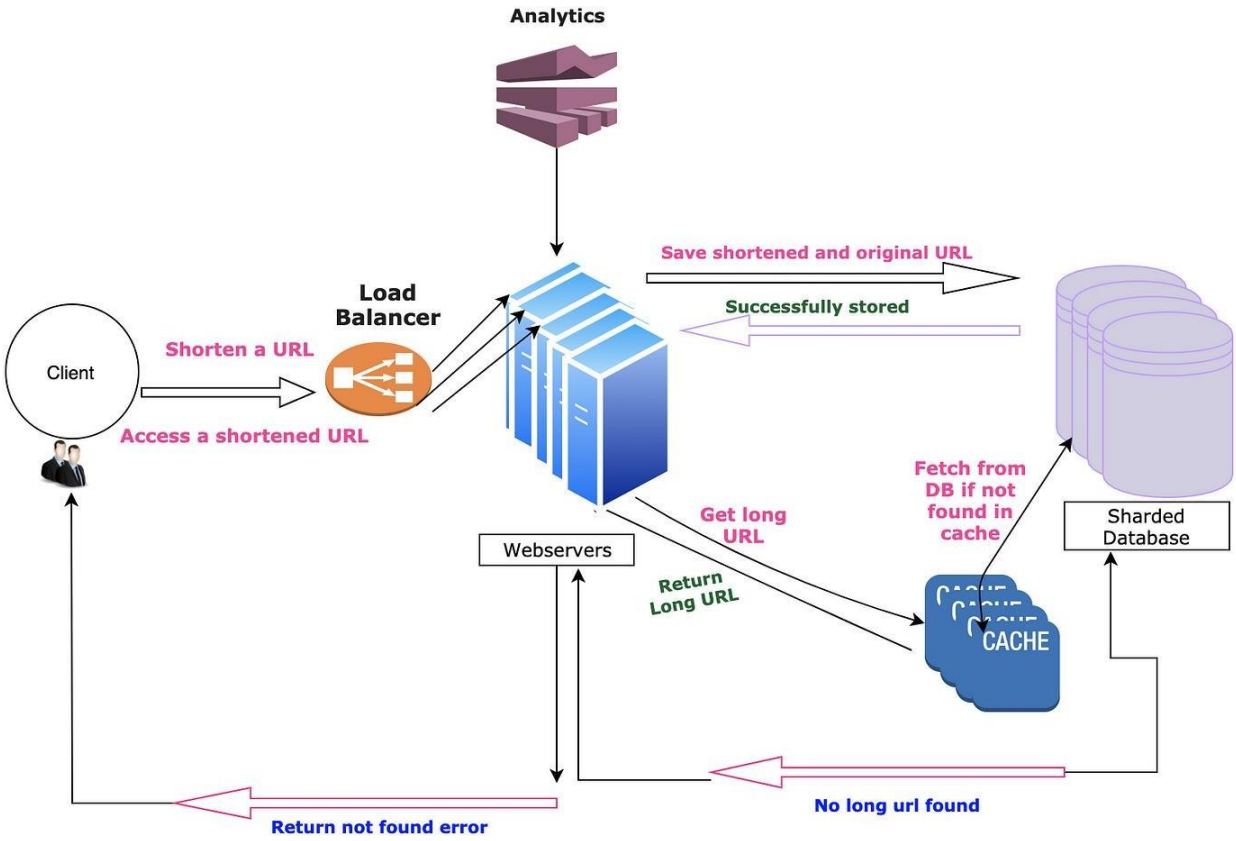
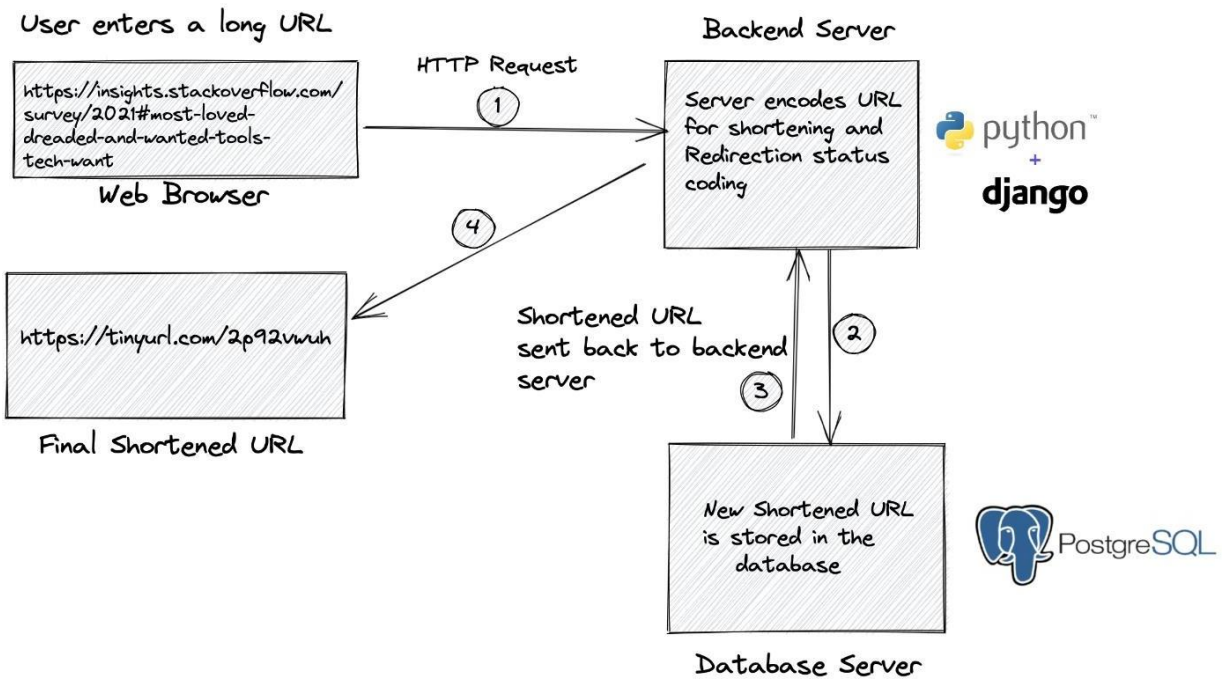
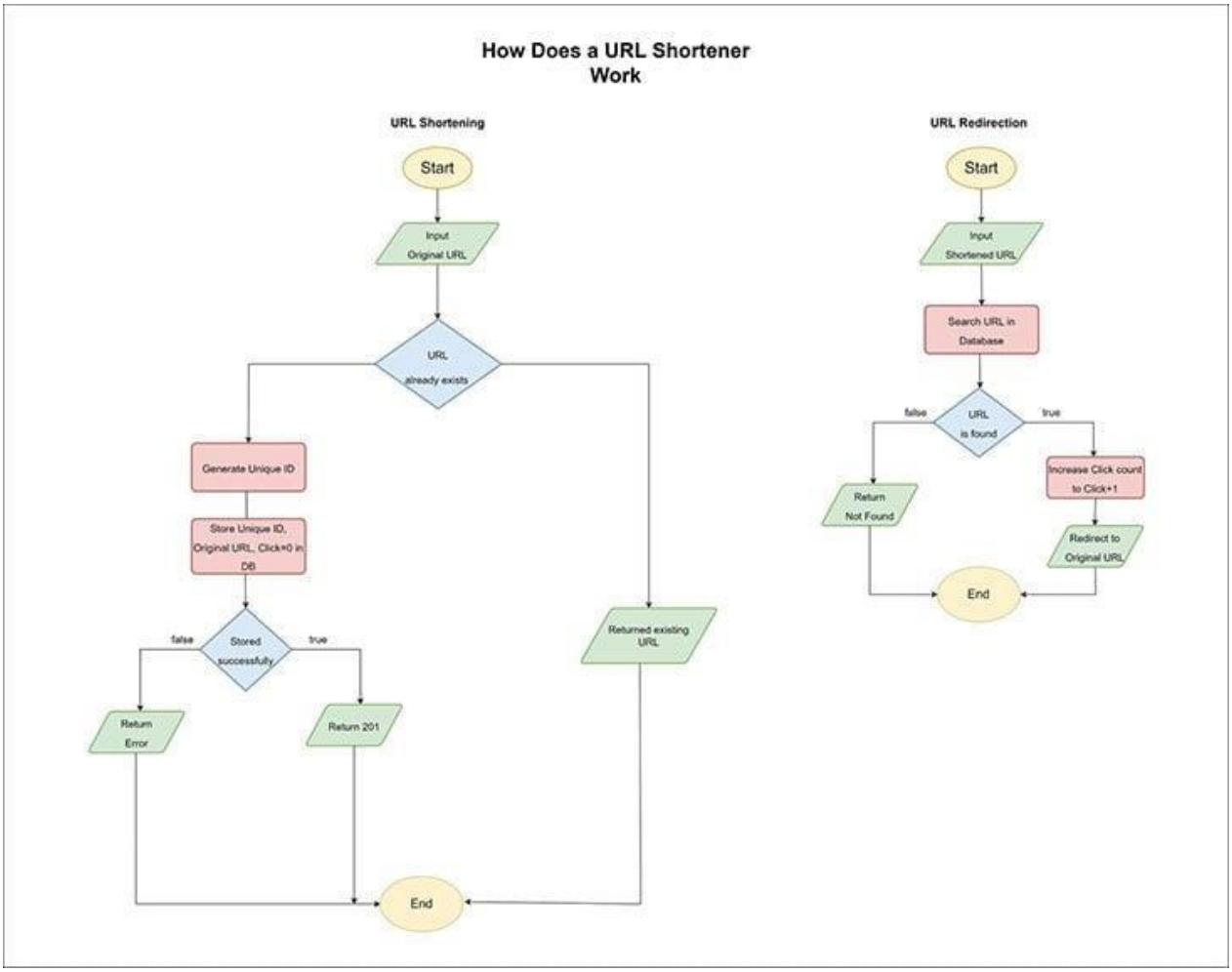


Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM

5.2 Low Level Diagram (if applicable)



5.3 Interfaces (if applicable)



6 Performance Test

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

1. Memory and Performance:

- Constraint: The application must be memory-efficient and efficient, especially when processing multiple URL mappings and user requests simultaneously.

- Structural approach: Django ORM (Object-Relational Mapping) was used to interact with the database, which helps in efficient data retrieval and usage. Appropriate indexing and database optimization techniques were used to improve performance.

- Test results: The application was tested using simulated load and stress tests to evaluate its performance in high traffic conditions. The goal was to ensure that the application remained viable and stable even during peak usage.

2. Accuracy and durability:

- Constraints: URL shortening must map exactly to URL shortening to their original counterparts and maintain data integrity over time.

- Design approach: The application uses unique identifiers for shortened URLs, ensuring that each mapping is unique and consistent. Database connections are used to maintain data integrity and persistence.

- Test results: Performed extensive unit testing and integration testing to verify the accuracy of the URL map and ensure that the data remains consistent and consistent across applications.

3. Power management:

- Limitations: Although not directly related to web applications, power consumption can indirectly affect server infrastructure and operating costs.

- Design Methodology: The design of the application focuses on resource efficiency to reduce server load and indirectly optimize energy consumption. Cloud hosting services with energy efficient infrastructure were considered for implementation.

- Recommendation: Implementing caching techniques, optimizing database queries, and using serverless or containerized deployment options can further reduce capacity consumption and operational costs.

4. Security:

- Constraints: The application must properly handle user data, prevent unauthorized access, and protect against common network security threats.

- Design approach: Implemented built-in Django security features such as CSRF protection, user authentication, and input validation. HTTPS was introduced to encrypt data in transit. Best practices such as parameterized queries were followed to prevent SQL injection attacks.

- Test results: Performed security testing including penetration testing and vulnerability assessment to identify and mitigate potential security risks Regularly applied security updates and patches to the application and underlying infrastructure.

6.1 Test Plan/ Test Cases

- The test cases were designed to cover various aspects of the application, including short URL processing, redirection, database processing, input validation, error handling, and performance under load.

- To divide the test cases into unit tests, integration tests, system tests, and performance tests.

6.2 Test Procedure

- Unit tests were run using the Django testing framework to validate individual components such as models, views, forms, and utility functions.

- Performed integration tests to check the interaction between modules and application components.

- Conducted configuration tests to evaluate end-to-end functionality including user interface, database operations and error handling.

- Simulation of performance tests using tools such as JMeter or Locust to measure response time, throughput, and server stability under load.

6.3 Performance Outcome

- Performance tests gave good results, indicating that the application can handle a significant number of concurrent users and maintain an acceptable response time.

- Further improve application performance through optimizations such as caching, database indexing, and improved query design.

- Used monitoring tools to monitor key performance metrics and identify areas for further improvement.

7 My learnings

1. Technical Skills Development: Through this internship, you gained practical experience in Python programming, Django programming, database administration (using Django ORM), and front-end development. This greatly improved your technical skills, especially in web development and back-end planning.

2. Problem Solving and Design: You learned how to solve complex problems like URL shortening by focusing on scalable solutions to effective problems. This includes considerations of uniqueness, security, performance, and user experience.

3. Project Management: Managing a project within a defined timeline (6 weeks) has taught you project management skills, including work prioritization, resource allocation and deadlines will be used in addition.

4. Collaboration and Communication: Working with industry partners like UniConverge Technologies Pvt Ltd and educational institutions like upskill Campus and The IoT Academy, I learned the importance of effective communication, teamwork and collaboration work to achieve project objectives.

5. Testing and Best Practices: Experienced in conducting test designs, test cases, and performance tests to ensure the functionality, reliability, and performance of a short URL application. Includes testing for memory efficiency, accuracy, stability, power consumption and security.

6. Continuous learning: Participants emphasized the importance of continuing education, staying abreast of industry developments, and making use of resources such as flyers, community forums and online forums to emphasize knowledge.

7. Real-world applications: Applying theoretical knowledge to real-world projects gained practical insights into technical problems, technical solutions and business applications, contributing to your overall business performance.

Together, these courses contribute to your development as a competent and experienced professional in software development and IT solutions.

8 Future work scope

1. Feature Expansion: The future scope of the "Url Shortener" project will include the addition of advanced features to enhance the user experience and capabilities. These features can allow users to create custom URL renames, set expiration dates for short URLs, generate QR codes for easy sharing, use password protection for sensitive connections Expand the list of features to meet broader project user needs and preferences, creating a more versatile and user-friendly URL shortening service.

2. Analytics and Reporting: Improving analytical tools is another important aspect of the future of the profession. Enhanced analytics and reporting capabilities will provide users with deeper insights into link performance, user engagement metrics, geographic segmentation usage, and referral sources This data-driven approach this will help users make informed decisions about their link-sharing strategies, optimize marketing efforts and track the effectiveness of their campaigns.

3. Integration: Opportunities for integration with third-party services may be sought to increase project efficiency and improve efficiency. Integration with social media platforms, content management systems, and analytics tools can streamline the sharing process, enable link tracking, and provide a seamless user experience with integrated delivery, "Url Shortener" projects can become an integral part of users' digital ecosystem, facilitating efficient link management and data analysis.

4. Mobile App Development: An important future of the industry is to develop a dedicated mobile app. The mobile app can provide all the features of the web platform, including short URLs, link management, analytics tracking, and push notifications. A mobile app will allow users to access services on the go, share links from their mobile devices, and receive real-time updates on link performance and users.

5. Security measures: Strengthening security measures is essential to maintain reliability and protect user data. The application can use advanced authentication methods such as multifactor authentication (MFA) and token-based authentication to enhance account security. To obtain critical information, data encryption techniques must be used both during travel and at rest. Proper security threat management, regular security audits, and adherence to industry standards will ensure that the "Url Shortener" service remains secure and reliable.

6. Scalability and Performance: Optimizing scalability, response time, and overall performance is essential to dealing with increasing user traffic and ensuring a seamless user experience. Implementing cloud technologies, implementing archiving techniques, and optimizing database queries can improve system scalability and response time. Ongoing performance monitoring, load testing, and optimization efforts will help maintain optimal performance even during peak usage.