INDUSTRIAL TRAINING REPORT

A. L. D. KANISHA

182061

Department of Computing and Information Systems

Faculty of Applied sciences

Wayamba University of Sri Lanka

November, 2023

INDUSTRIAL TRAINING REPORT

A. L. D. KANISHA

182061

Supervisor: Dr. B. Munasinghe

Program Coordinator: Prof. V.G.T.N Vidanagama

Name and Code of the Course Module: Industrial Training - CMIS 4216

Training Period: 04/07/2023 to 04/01/2024

Mobitel (Private) Limited,
Colombo 08,
Sri Lanka

This dissertation is submitted in partial fulfilment of the requirement of the Degree of Bachelor of Science of the Wayamba University of Sri Lanka

November 2023

DECLARATION

I declare that this dissertation does not incorporate, without acknowledgment, any material previously submitted for a degree or diploma in any university, and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text.

The content of this report is the result of work which has been carried out since the official commencement date of the Industrial training program of the Department. Any editorial work, paid or unpaid, carried out by a third party is acknowledged; and procedures and guidelines of the Department have been followed.

Signature of Candidate: …………………………………… Date:

Name of Candidate: A. L. D. Kanisha

Countersigned by:

Signature of supervisor(s): …………………………… ……….. Date:

Name(s) of Supervisor(s): Dr. B. Munasinghe

Signature and stamp of Training Coordinator: …………………. Date:

Name of Training Coordinator: Prof. V.G.T.N. Vidanagama

APPROVAL FOR SUBMISSION

Internal Supervisor: …………...……………………………………….

: Dr. B. Munasinghe,

Senior Lecturer,

Department of Computing and Information System

External Supervisor: ……….……………………………………..….

: Mr. Chandima Rajaguru,

Senior Tech Lead – Big Data & Data Science,

Information Systems Division,

Mobitel (Private) Limited

Program Coordinator: ……….………………………………………….

: Prof. VGTN Vidanagama,

Senior Lecturer,

Department of Computing and Information System

Assistant Registrar : ……….…………………………………………….

: Mr. Piyathilake

ABSTRACT

During my industrial training as a Trainee AI Software Engineer at Mobitel (Pvt) LTD in my final year of pursuing a BSc. in Computer Science, I undertook a series of transformative tasks that significantly contributed to my practical knowledge and skills. Mobitel (Pvt) LTD is a leading telecommunications company in Sri Lanka, provided me with a unique opportunity to gain invaluable real-world experience in the dynamic field of artificial intelligence. One of the key projects I was involved in was the development of a chatbot. This endeavor allowed me to delve into the realm of natural language processing and AI-driven conversational agents. Building the chatbot involved creating an interactive and intelligent interface, enhancing user experiences, and finding innovative solutions to address customer queries efficiently.

Additionally, I had the opportunity to train the chatbot data, a fundamental step in enhancing its performance. This process entailed fine-tuning the chatbot's responses and behaviors, making it more context-aware and responsive to user input. This hands-on experience provided invaluable insights into the iterative nature of AI model development and refinement.

Furthermore, I contributed to the creation of a Java Spring Boot backend to seamlessly integrate the chatbot with a mobile application frontend. This backend played a pivotal role in ensuring the smooth interaction between the chatbot and users, effectively bridging the gap between the AI system and end-users.

My training experience at Mobitel not only broadened my understanding of AI but also allowed me to apply my skills to real-world projects, reinforcing the importance of AI in modern technology. This abstract provides a concise overview of the tasks I undertook during my training, highlighting their significance and the practical knowledge gained in the process.

ACKNOWLEDGEMENTS

I wish to convey my profound gratitude and deep appreciation to several individuals and organizations who played pivotal roles during my industrial training period.

First and foremost, I am immensely grateful to my internal supervisor, Senior Lecturer Dr. B. Munasinghe of the Department of Computing and Information Systems, for her unwavering support and guidance. Dr. Munasinghe's expertise and mentorship were instrumental in enriching my training experience.

I would also like to extend my sincere thanks to our industrial training coordinator, Prof. VGTN Vidanagama, an Associate Professor and Senior Lecturer in the Department of Computing and Information Systems, for affording me the remarkable opportunity to tackle real-world challenges. His guidance was invaluable in navigating the complexities of the professional world.

My deepest appreciation goes to my external supervisor, Mr. Chandima Rajaguru, Senior Tech Lead – Big Data & Data Science at Mobitel (Pvt) Ltd. His wealth of industrial knowledge and unwavering support greatly enriched my training experience.

I must also express my gratitude to the National Apprentice and Industrial Training Authority (NAITA) for their professional and courteous coordination throughout the training.

Last but certainly not least, I extend my heartfelt thanks to my beloved family members, dearest friends, and dedicated team members whose unwavering support and encouragement were truly invaluable throughout my training journey.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ACRONYMS

AI	Artificial Intelligence
API	Application Programming Interface
DB	Database
DL	Deep Learning
HTTP	Hypertext Transfer Protocol
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
ML	Machine Learning
MVC	Model-View-Controller
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
SDLC	Software Development Lifecycle
SLT	Sri Lanka Telecom
SQL	Structured Query Language
SWOT	Strengths, Weaknesses, Opportunities and Threats

CHAPTER 1
DETAILS OF INDUSTRIAL TRAINING

1.1 Introduction to Industrial Training

The six-month industrial training phase is a crucial segment undertaken by students in their final year of undergraduate studies. It acts as a conduit, linking the theoretical knowledge students amass in classrooms with the practical demands of the industry. The training emphasizes both technical proficiency and essential soft skills while encouraging analytical thinking.

In the realm of software engineering, such industrial training initiatives foster mutual benefits for academic institutions and industries. They equip students with industry-relevant technical skills, foster enhanced collaborations between universities and the corporate sector, and integrate modern practices into academic syllabi. For budding software engineers, this real-world exposure is indispensable, preparing them to navigate and resolve complex professional challenges. Therefore, for undergraduates in the Computing and Information Systems Department of the Faculty of Applied Sciences, this training is not just beneficial but a requisite. As stipulated by the academic framework, students must engage in this immersive six-month training experience during their second semester at Level 04 of their university journey.

1.2 Background of the Organization

Occupying a significant niche in Sri Lanka's telecommunications sector, Mobitel (Pvt) LTD is renowned for its consistent leadership and innovative prowess. This telecommunications giant, as illustrated by its commercial logo in Figure 1.1, stands as one of the foremost service providers in the country, with a legacy marked by relentless innovation and a passion for embracing state-of-the-art technology.

Mobitel's portfolio boasts an impressive array of services, spanning from mobile telephony to broadband, fixed-line offerings, and bespoke enterprise solutions. Rooted in its unwavering dedication to customer delight, the company persistently pursues avenues to augment network reliability, elevate service standards, and roll out avant-garde offerings.

Keeping pace with global tech trends, Mobitel has proactively delved into areas such as artificial intelligence, data analytics, and the Internet of Things (IoT). Such strategic initiatives not only keep Mobitel at the vanguard of industry competition but also underscore its role as a catalyst in Sri Lanka's digital metamorphosis.

Integral to Mobitel's ethos is its zeal for nurturing innovation and continuous learning. This translates into fostering meaningful ties with educational entities and creating avenues for students to immerse in practical, industry-aligned experiences.

Figure 1. 1: Mobitel logo

In essence, Mobitel (Pvt) LTD emerges as a progressive, customer-aligned telecommunication titan that has been instrumental in sculpting Sri Lanka's digital narrative. My tenure at Mobitel was profoundly shaped by its vibrant culture, penchant for innovation, and its authoritative stance in the telecommunications domain.

1.3 Mobitel (Pvt) Limited Head Office, Partnerships & Subsidiaries, and Working Culture

Mobitel (Pvt) Limited, in partnership with SLT as SLT-Mobitel, maintains a robust network of subsidiaries and embodies a dynamic working culture that values technology, diversity, and employee development. My role as a hybrid employee allows to be part of this vibrant culture, fostering both personal and professional growth within the framework of its partnership with SLT.

1.3.1 Mobitel (Pvt) Limited Head Office

The head office of Mobitel (Pvt) Limited is strategically located at No 148/15, Lesley Ranagala Mawatha, Base Line Road, Colombo 08, Sri Lanka. This central headquarters serves as the nerve center of the organization's administrative functions, playing a crucial role in coordinating its nationwide operations.

1.3.2 Mobitel (Pvt) Limited Partnerships & Subsidiaries

1.3.2.1 Partnerships

On January 1, 2021, SLT and Mobitel merged to form SLT-Mobitel. This merger brought together two telecommunications giants, creating a robust entity that combines their expertise and resources to offer enhanced services and innovations to the Sri Lankan market.

Figure 1. 2: SLT-Mobitel logo

This strategic collaboration signifies a major development in the telecommunications landscape, blending the strengths of SLT and Mobitel to drive forward a shared vision of innovation and service excellence. The unified front of SLT-Mobitel, as represented in Figure 1.2, is set to redefine the benchmarks of quality, reliability, and customer-oriented services in the industry.

1.3.2.2 Subsidiaries

Mobitel (Pvt) Limited has a network of subsidiaries, each serving a specific purpose within the organization. These subsidiaries are integral to the company's strategy for providing comprehensive telecommunications services and expanding its reach throughout Sri Lanka. Following the partnership with SLT, known as SLT-Mobitel, these subsidiaries include:

- **eChannelling PLC:** A subsidiary dedicated to providing innovative digital healthcare solutions, including appointment booking and healthcare information services.

- **mCash:** mCash allows to experience a whole new method of carrying out financial transactions by maintaining a Mobile Money Account. This is both convenient and easy to use. This will be able to deposit, withdraw, transfer funds as well as purchase goods and services and even pay utility bills.

- **Sky Network (Pvt) Ltd:** A subsidiary playing a vital role in expanding Mobitel's network infrastructure and coverage.

- **SLT Digital Info Services (Pvt) Ltd:** A subsidiary that specializes in digital information services, ensuring the accessibility of information to customers.

- **SLT Human Capital Solutions (Pvt) Ltd:** A subsidiary dedicated to human resource solutions, emphasizing talent development and management.

- **SLT Property Management (Pvt) Ltd:** A subsidiary that oversees property management, ensuring effective utilization of resources.

- **SLT VisionCom (Pvt) Ltd:** A subsidiary focusing on vision and communication technologies, contributing to Mobitel's cutting-edge services.

- **Sri Lanka Telecom (Services) Ltd:** A subsidiary responsible for providing a wide range of telecommunications services and solutions.

- **Talentfort (Pvt) Ltd:** A subsidiary specializing in talent management and development, essential for the growth and sustainability of Mobitel's workforce.

- **Galle Submarine Cable Depot (Pvt) Ltd (40%, Associate company):** An associate company that plays a role in maintaining and supporting submarine cable infrastructure.

1.3.3 Mobitel (Pvt) Limited Working Culture

Mobitel (Pvt) Limited's working culture is characterized by a dynamic and inclusive environment. It is built on several core principles:

- **Innovation and Technology:** The company places a strong emphasis on adopting cutting-edge technology, particularly in the fields of telecommunications and artificial intelligence. This focus ensures Mobitel stays at the forefront of technological advancements, delivering innovative solutions to its customers.

- **Diversity and Inclusion:** Mobitel values diversity and strives to create an inclusive environment that encourages individuals from various backgrounds to collaborate and contribute their unique perspectives to the organization's success.

- **Employee Well-being:** The well-being of its employees is highly prioritized, recognizing the importance of work-life balance and overall job satisfaction.

- **Hybrid Work Model:** Mobitel's hybrid work model, in which you work as a hybrid employee, provides flexibility by allowing both on-site and remote work. This flexibility empowers employees to effectively manage their professional and personal commitments.

- **Continuous Learning and Growth:** Mobitel's working culture promotes continuous learning and growth through various development opportunities, including training programs, workshops, and mentorship initiatives. This investment in employee development benefits not only the individual but also contributes to the company's overall success.

1.4 SWOT Analysis

This analysis, depicted in Figure 1.3, provides a comprehensive overview of Mobitel’s current position in the telecommunications sector. It offers insights into the company's internal capabilities and areas for improvement, along with an understanding of the external factors that could impact its business strategy and operational success.

Figure 1. 3: SWOT analysis

- Strengths
    1. **Strong Market Presence:** Mobitel has a significant presence in the Sri Lankan telecommunications market, providing a wide range of services and coverage throughout the country.
    2. **Innovative Technology:** The company invests in cutting-edge technology to maintain a competitive edge in the industry.

3. **Diverse Subsidiaries:** Mobitel has a diverse range of subsidiaries and associate companies, offering a broad spectrum of telecommunications services and solutions.
4. **Operational Efficiency:** The organization's commitment to efficiency is evident in its ability to manage and maintain extensive network infrastructure.

- Weaknesses
    1. **Limited Software Market Presence:** Mobitel's software development teams primarily focus on internal solutions, missing opportunities in the external software market.
    2. **Limited Geographic Expansion:** While Mobitel is well-established in Sri Lanka, its geographic expansion beyond the country may be limited.
    3. **Dependence on Telecommunications:** As a telecommunications company, Mobitel's revenue and success are highly dependent on the telecommunications industry's health and competition.

- Opportunities
    1. **Diversification:** Exploring avenues for diversification beyond telecommunications, such as IoT, data services, and content delivery.
    2. **International Expansion:** Expanding services and infrastructure beyond Sri Lanka's borders, tapping into global markets.
    3. **Strategic Partnerships:** Collaborating with global technology giants or other telecom companies for technological advancements and market expansion.

- Threats
    1. **Intense Competition:** The telecommunications sector is highly competitive, and Mobitel faces threats from both established and emerging rivals.

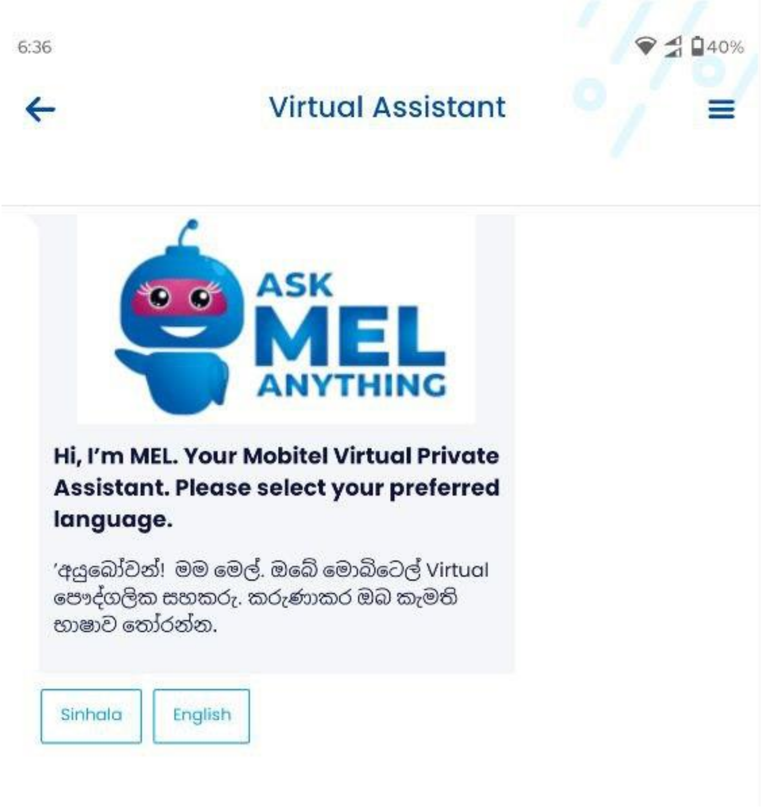
2. **Regulatory Changes:** Frequent regulatory changes in the telecom sector can impact the company's operations and profitability.
3. **Economic Volatility:** Economic downturns or fluctuations can affect consumer spending and investment in telecommunications services.
4. **Technological Disruption:** Rapid technological advancements and disruptive innovations may require substantial investments to stay competitive.

CHAPTER 2

TRAINING EXPERIENCE

2.1 Overview of Approaches, Tools, and Equipment Utilized

This section delves into the diverse methodologies, techniques, and tools incorporated throughout the project's lifecycle. The efficacy of a project often rests on the judicious use of these assets, and this segment is designed to elucidate the core processes and choices that steered the project's direction. By elucidating these facets, the intention is to grant readers an unambiguous perspective on the tactical approaches and tools chosen to achieve the project objectives. Furthermore, this overview aspires to equip readers with practical insights and lessons that might prove invaluable for subsequent project initiatives.



Hi, I'm MEL. Your Mobitel Virtual Private Assistant. Please select your preferred language.

'අයුබෝවන්! මම මෙල්. ඔබේ මොබිටෙල් Virtual පෞද්ගලික සහකරු. කරුණාකර ඔබ කැමති භාෂාව තෝරන්න.

Figure 2. 1: Mobitel Virtual Assistant mobile application

Figure 2.1 showcases a screenshot of the Mobitel Virtual Assistant mobile application. This image represents a significant aspect of the project, illustrating the integration of the backend work that was a focus during the internship. While the development of the user interface was handled by other team members, the Figure 2.1 reflects the successful collaboration of various project components, including those on which I worked directly.

2.1.1 The Interplay of Artificial Intelligence & Software Development

During my training, the harmonious integration of AI and software development emerged as the cornerstone for devising innovative solutions tailored to the organization's unique demands. Below are the principal arenas where AI intertwined with software development during my training:

- **Natural Language Processing (NLP)**: NLP emerged as a foundational element, crucial for crafting chatbots and enhancing user communication. This entailed designing algorithms and models for tasks such as textual evaluation, sentiment discernment, and automated chatbot replies. Tools like NLTK and spaCy were indispensable for proficiently processing and scrutinizing text-based information.

- **Chatbot Creation**: The art of chatbot construction seamlessly married AI principles with software design techniques. Leveraging platforms like Dialogflow and Rasa was essential for forging conversational interfaces adept at understanding user intents and providing fitting replies.

- **AI-centric Development Environments**: Tools such as Jupyter Notebook and PyCharm were integral to the AI development journey—spanning from initial coding stages to the final deployment of AI solutions. These platforms furnished a dynamic and user-friendly backdrop for AI engineering.

- **Integration of AI into Applications**: Integrating AI capabilities into functional applications was a pivotal task that connected sophisticated AI models with actionable, user-facing tools. Flask, a micro web framework, was invaluable in this process. With its lightweight and modular design, Flask enabled the swift deployment of AI models, making them available as user-friendly APIs. On the other hand, Spring Boot, renowned for its convention-over-configuration approach, amplified this integration process. It provided a robust platform to craft and deploy microservices, ensuring that AI-driven features meshed seamlessly into broader application ecosystems. The synergy between Flask and Spring Boot meant that AI models were not only developed and honed to perfection but also effortlessly integrated into applications, thereby enriching their utility and efficiency. The union of these technologies, enriched with AI enhancements, paved the way for the creation of intelligent, responsive, and cutting-edge applications.

- **Ethical Dimensions of AI**: Incorporating ethical considerations in AI ventures was a non-negotiable aspect. This ensured that our AI models met the benchmarks of ethical integrity, equity, and data confidentiality. Techniques to minimize biases and safeguard data anonymity were vital to uphold ethical AI standards.

The marriage of AI with software development provided a robust strategy to tackle pragmatic challenges, optimize network functions, and uplift user engagement in the telecom sector. The convergence of AI-based strategies and techniques, married with the right tools, laid the foundation for the triumphant completion of the training project. This integration was not only a reservoir of rich learning but also an impetus for the growth of AI-infused solutions within the telecom realm.

2.1.2 Frameworks and Programming Languages

Understanding the tools of the trade is essential for any software project. In this section, we delve into the frameworks and programming languages that formed the backbone of our AI-driven solutions during the training.

2.1.2.1 RASA

RASA [1] (logo depicted in Figure 2.2) is an open-source framework designed primarily for creating conversational AI, such as chatbots and virtual assistants. Its flexibility and built-in natural language processing capabilities, powered by spaCy and TensorFlow, make it a go-to choice for crafting customized and advanced chatbot solutions. The adaptability of RASA allows for the creation of bots that can engage in multi-turn conversations, making user interactions more dynamic and intuitive.

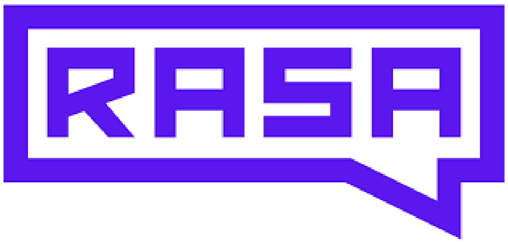


Figure 2. 2: RASA logo

2.1.2.2 Python

Python [5] is a versatile, high-level programming language revered for its simplicity and readability. Given its extensive library support and community-driven modules, Python is a favored choice for AI, data science, and web development projects. During the training, Python was predominantly used for scripting AI algorithms, particularly in areas like natural language processing and machine learning.

2.1.2.3 Flask

Flask [2], represented by the logo in Figure 2.3, is a lightweight and micro web framework written in Python. It offers the tools and libraries required to build web applications seamlessly. With its minimalistic and modular architecture, Flask is particularly adept for creating RESTful web services. In the context of our project, Flask facilitated the deployment of AI models as APIs, enabling easy integration with front-end applications or other systems.

Figure 2. 3: Flask logo

2.1.2.4 Java

Java is an object-oriented programming language that boasts platform independence due to its “write once, run anywhere” (WORA) capability. Its robustness, security features, and extensive library support make it a preferred language for a plethora of applications. During the training, Java played a crucial role in backend development, particularly in interfacing with databases and handling business logic.

2.1.2.5 Spring Boot

Spring Boot [4], as depicted in Figure 2.4, is an open-source Java-based framework used to create stand-alone, production-grade Spring-based applications. It simplifies the process of building production-ready applications by eliminating much of the boilerplate code and providing the necessary tools out of the box. In the project, Spring Boot was instrumental in developing scalable microservices, allowing for efficient and seamless integration of AI functionalities into broader system architectures.

Figure 2. 4: Spring Boot logo

2.1.3 Libraries

Libraries, in the context of application development, serve as pre-written code modules that can be integrated to augment an application's functionality and usability. They represent a valuable resource for developers, offering time-saving, tried-and-tested solutions to common challenges. Throughout the training period, a diverse array of libraries was harnessed to enrich the application's codebase. These libraries encompassed both open-source and proprietary options, thoughtfully selected based on their appropriateness, performance, and alignment with the application's requirements.

2.1.3.1 NLTK

NLTK, or the Natural Language Toolkit, is a Python library that specializes in natural language processing and text analysis. It is a crucial tool in text mining, linguistic analysis, and building applications that involve human language understanding. Key aspects of NLTK include:

- **Text Analysis:** NLTK provides a wide range of tools for processing and analyzing text data. It supports tasks such as tokenization, stemming, lemmatization, and part-of-speech tagging, which are essential for understanding the structure of text.

- **NLP Algorithms:** The library includes various algorithms and methods for NLP, including named entity recognition, sentiment analysis, and machine learning-based classification. These are used in a wide range of applications, from chatbots to information retrieval.

- **Corpora and Resources:** NLTK offers access to numerous text corpora, lexicons, and language datasets, making it a valuable resource for linguistic research and model training.

- **Research and Education:** NLTK is widely used in research and education, providing a platform for experimenting with NLP techniques and teaching the fundamentals of natural language processing.

- **Community and Documentation:** NLTK boasts an active community and comprehensive documentation, which make it a valuable resource for developers, researchers, and educators in the NLP field.

In the context of our project, NLTK played a pivotal role in enhancing the natural language understanding of our applications. It empowered us to develop features like sentiment analysis, chatbot interactions, and language processing, contributing to more intuitive and context-aware user experiences.

2.1.3.2 Request

The Requests library is a vital tool for Python developers, simplifying the process of making HTTP requests to interact with web services and APIs. Key aspects of the Requests library include:

- **HTTP Requests:** Requests allows developers to send HTTP requests, such as GET, POST, PUT, and DELETE, to web services, servers, and APIs.

- **Parameters and Headers:** It supports the inclusion of parameters, query strings, and custom headers in requests, making it versatile for various data exchange scenarios.

- **Response Handling:** The library enables the handling of HTTP responses, which may include data in formats like JSON, XML, or HTML. Developers can extract and manipulate the data retrieved from these responses.

- **Session Management:** Requests offers the capability to manage sessions, handling cookies and authentication tokens, which is important for maintaining state across multiple requests.

- **Error Handling:** It provides robust error handling, allowing developers to respond to different HTTP status codes and network-related issues.

- **HTTPS Support:** Requests inherently supports secure HTTPS connections, ensuring data integrity and privacy when interacting with secure web services.

- **Community and Documentation:** With an active user community and extensive documentation, Requests is well-supported, making it a reliable choice for HTTP communication in Python projects.

In our project, the Requests library was a key component for communicating with external APIs, fetching data from web services, and integrating third-party resources. Its simplicity and versatility made it a valuable tool for managing HTTP requests, facilitating data retrieval, and enabling seamless interactions with external systems.

2.1.3.3 SpaCy

spaCy [9] is a Python library specifically designed for natural language processing (NLP) and text analysis. It is widely used in various NLP applications and offers the following key features:

- **Efficient Processing:** spaCy is known for its speed and efficiency in text processing. It’s designed to be fast and can handle large volumes of text efficiently.

- **Linguistic Analysis:** The library provides various linguistic annotations, such as tokenization, part-of-speech tagging, named entity recognition, and dependency parsing, which are essential for understanding the structure and meaning of text.

- **Pre-trained Models:** spaCy offers pre-trained models for different languages, allowing developers to perform tasks like text classification and entity recognition without building models from scratch.

- **Rule-based Matching:** It includes a rule-based matching engine that enables the extraction of specific patterns or entities from text, making it useful for information extraction and custom NLP tasks.

- **Customization:** Developers can train their own models or fine-tune existing ones to suit specific tasks, making spaCy highly adaptable for a wide range of applications.

- **Community and Documentation:** spaCy has an active community and comprehensive documentation, which are valuable resources for developers working with NLP and text analysis.

In our project, spaCy played a significant role in enhancing natural language understanding, text analysis, and entity recognition. It allowed us to process and analyze textual data efficiently, enabling features like named entity recognition and sentiment analysis in our applications.

2.1.4 Database Technology

For the project's database requirements, MySQL [10], as illustrated in Figure 2.5, was the chosen platform. Renowned as a leading relational database management system, MySQL stands out for its proficiency in managing structured datasets. Its alignment with SQL standards simplifies data handling and queries. A key strength of MySQL is its scalability; it's adept at catering to both compact and expansive datasets, making it apt for diverse project scopes. Moreover, MySQL boasts rigorous security features, including stringent user access guidelines, advanced data encryption, and solid authentication methods, which collectively ensure data integrity and security. Another significant advantage of MySQL is its active user community, which consistently provides rich documentation and resources, aiding in effective troubleshooting and system enhancement. In the context of the project, MySQL was integral to data storage and handling, reinforcing the project's stability and performance benchmarks.

Figure 2. 5: MySQL logo

2.1.5 Version Controlling Tools

Version control systems are fundamental tools in software development, providing developers with the capability to manage, monitor, and collaborate on source code changes. They enable teams to work together on projects, maintain a historical record of alterations, and revert to previous versions when necessary. These systems are considered a best practice in the industry, contributing to the stability, security, and reliability of software projects. This section outlines the version controlling tools employed during the training period.

2.1.5.1 Git

Git, as shown in Figure 2.6, a widely recognized version control system, was utilized in the application's development process. Git empowers developers to effectively track changes to the codebase, enabling seamless collaboration on projects. This tool allows multiple developers to work on the same codebase from different locations while ensuring that code is securely backed up in the event of system failures. Developers can create project branches, make modifications in their respective branches, and subsequently merge those changes into the primary codebase. This approach facilitates collaborative development while preserving distinct code versions. Git also offers a range of features for tracking changes and resolving conflicts, making it a robust and adaptable tool for software development.

Figure 2. 6: Git

2.1.5.2 Bitbucket

In addition to Git, BitBucket was used as a complementary platform. BitBucket enhances collaboration by providing a centralized location for code repositories and project management. It seamlessly integrates with Git, offering features for version control, code review, and team collaboration. BitBucket's capabilities, such as pull requests and issue tracking, streamline the development process and ensure the codebase's reliability and integrity throughout the training period. Following Figure 2.7 shows the logo of Bitbucket.

Figure 2. 7: Bitbucket

2.1.7 Hosting Method

It's worth noting that all projects were deployed on Mobitel's private cloud server. This private cloud infrastructure served as the hosting environment for the applications developed during the training period. By leveraging Mobitel's private cloud server, the projects benefited from a secure and dedicated hosting solution, ensuring the confidentiality, accessibility, and performance of the applications. This approach allowed for efficient deployment, management, and scaling of the projects, meeting the specific hosting requirements of the training experience.

Private Cloud: The projects were hosted on a private cloud server, which operated on the Ubuntu Linux system. This private cloud environment ensured the applications' security, scalability, and flexibility. In addition, the deployment and management of the applications on this platform were facilitated using Xshell [6] and Xftp [7]. These tools, in conjunction with the private cloud server, supported efficient and reliable hosting, contributing to the success of the projects during the training period. The following figure, Figure 2.8, showcases the logos of Xshell 7 and XFTP 7 respectively.

Figure 2. 8: XSHELL & XFTP

2.1.8 Development Tools

Development tools, encompassing integrated development environments (IDEs) and software extensions, provide an array of functionalities that empower developers to create, test, and deploy top-notch software. These tools include code highlighting, debugging capabilities, and version control, among others. Their primary role is to expedite, simplify, and refine the software development process, freeing up developers to concentrate on code creation and innovative problem-solving. As the demand for software development continues to surge, the need for potent and efficient development tools to streamline the development workflow is increasingly pronounced. This section elaborates on the tools employed to facilitate the development process.

2.1.8.1 Visual Studio Code

Visual Studio Code (VS Code), developed by Microsoft, is a freely available, open-source code editor renowned for its versatility in software development. This sophisticated tool features an intuitive and user-friendly interface that simplifies the tasks of coding, debugging, and testing. With capabilities such as syntax highlighting, code completion, and seamless integration with version control systems like Git, it equips developers with a comprehensive set of functionalities. What further enhances its utility is an extensive repository of extensions that enable users to customize the editor according to their specific needs. VS Code has garnered popularity in the development community due to its cross-platform compatibility, swift performance, and support for a wide array of programming languages. Its user-friendliness and the wealth of available add-ons make it an ideal choice for developers at all levels of expertise. Figure 2.9 showcases the logo of Visual Studio Code.

Figure 2. 9: Visual Studio Code

2.1.8.2 IntelliJ IDEA

IntelliJ IDEA (logo shows in Figure 2.10) is an IDE designed to enhance the software development experience. Developed by JetBrains, it provides developers with a comprehensive set of tools to create, test, and deploy software solutions. Some of its key features include code analysis, intelligent code completion, and a wide range of plugins that streamline the development process. IntelliJ IDEA supports various programming languages and frameworks, making it versatile and adaptable to different projects. Its user-friendly interface, rich set of functionalities, and consistent updates have made it a preferred choice for developers aiming to boost productivity and code quality throughout the software development lifecycle.

Figure 2. 10: IntelliJ IDEA

2.1.8.3 Eclipse IDE

Eclipse IDE, a widely used integrated development environment, is a versatile tool that caters to software developers' needs for code creation, testing, and deployment. Eclipse offers a range of features such as code analysis, intelligent code completion, and a library of plugins that expedite development tasks. It supports multiple programming languages and frameworks, rendering it adaptable to a variety of project requirements. Eclipse's user-friendly interface, extensive feature set, and regular updates make it a favored choice among developers seeking to enhance productivity and code quality throughout the software development lifecycle. The following figure, Figure 2.11, showcases the logo of Eclipse IDE.

Figure 2. 11: Eclipse IDE

2.2 Details of Operations, Processes and Procedures Learned

During the industrial training period, I gained valuable insights into a wide array of operations, processes, and procedures that are fundamental in the field of AI software engineering.

2.2.1 Software Development Lifecycle

The Software Development Lifecycle (SDLC) is a structured process that encompasses various phases, including planning, analysis, design, implementation, testing, and maintenance of software solutions. Each of these steps is illustrated in Figure 2.12. It consists of a series of actions and procedures aimed at ensuring software is consistently, reliably, and efficiently developed. Best practices and industry standards were employed in the creation of the web application to ensure that the software was developed in alignment with the SDLC. These practices and procedures included responsibilities such as requirement gathering, analysis, design, coding, testing, and deployment to ensure that the software met high-quality standards and user expectations.

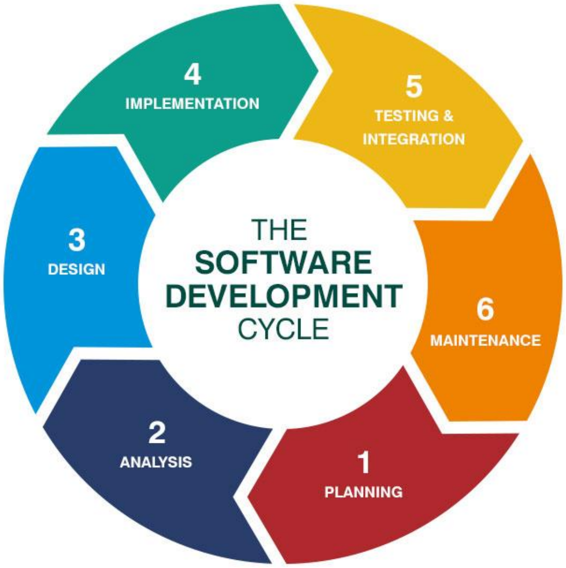


Figure 2. 12: Software Development Life Cycle

In this specific scenario, the software engineers primarily engaged in the fourth phase of the SDLC, which is the implementation phase. The preceding stages, including requirement gathering, analysis, and design, were conducted by other developers and designers within the organization.

During the coding stage, responsibilities included translating the application's design into functional features. This entailed writing, debugging, and testing the code, as well as integrating various components. This phase allowed for the creation of a robust and dependable codebase while ensuring the application met user requirements and expectations.

The testing stage involved conducting a variety of tests to validate the application's reliability and resilience. These tests included unit tests, integration tests, and manual acceptance tests, along with identifying and rectifying any identified defects or errors. This phase aided in identifying and resolving faults, ensuring the application met the project's quality standards.

The deployment phase involved packaging the application and transferring it to the production environment. However, tasks such as infrastructure configuration, database setup, and code and content delivery to servers were handled by the DevOps team. This phase ensured the application's availability and accessibility to end-users.

In summary, the SDLC played a pivotal role in the application's development by enabling a structured approach to planning, designing, building, testing, and deploying the software. SDLC best practices were diligently followed, leveraging various tools, techniques, and processes to ensure the application's high quality and alignment with user needs and expectations.

2.2.2 Version Control

In the context of our training experience, version control emerged as a crucial element in the software development process. It played a pivotal role in tracking, documenting, and managing code changes efficiently, fostering collaborative development. The significant aspects of version control in our scenario included the implementation of Git, a widely adopted version control system. Git was central to our development workflow, facilitating the following:

- Git enabled the simultaneous work of multiple developers on the same codebase, ensuring that changes were tracked and code backups were maintained for resilience.

- It allowed for parallel development efforts and seamless code synchronization among team members.

- The branching feature in Git was employed to create distinct branches for specific development tasks, allowing developers to work on features or fixes independently until they were ready for integration.

- Git provided tools for tracking and resolving code conflicts that might arise during the merging of changes from different branches, ensuring smooth integration.

- Every code change was systematically documented through commits in Git, resulting in a comprehensive commit history. This history served as a record of code modifications, enabling the tracing of the codebase's evolution and the identification of the purpose of each change.

Version control practices enhanced the integrity and reliability of the codebase by systematically managing, reviewing, and documenting code changes. It ensured that the software development process was well-organized and collaborative, contributing to the stability and consistency of the project.

2.2.3 Agile Methodologies

The training project focused on the adaptability and continuous improvement aspects of Agile methodologies. These methodologies allowed the team to respond to evolving project requirements and priorities. Furthermore, the culture of continuous improvement was fostered through regular retrospectives. These sessions provided an opportunity to reflect on the development process, identify areas for enhancement, and implement changes to optimize workflow and product quality. The steps in the Agile development process in software development, are illustrated in Figure 2.13.

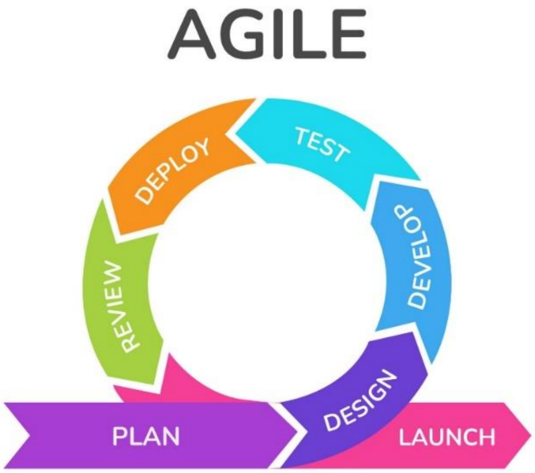


Figure 2. 13: Agile development process

2.2.4 Artificial Intelligence Techniques

The training experience placed significant emphasis on the implementation of various artificial intelligence techniques. These techniques included NLP, machine learning, and chatbot development. NLP techniques were extensively utilized in creating chatbots and enhancing customer interactions. This involved the development of algorithms and models for text analysis, sentiment analysis, and chatbot responses. Tools such as NLTK and spaCy were leveraged for processing and analyzing textual data. Machine learning algorithms played a crucial role in tasks like predictive analytics, recommendation systems, and anomaly detection. Tools such as scikit-learn and TensorFlow were instrumental in developing and fine-tuning machine learning models. The development of chatbots was a fusion of AI and software engineering techniques, using frameworks like Dialog flow and Rasa to build conversational agents capable of understanding user queries and providing relevant responses. Data analysis and visualization tools were used to extract insights from datasets, with Python libraries such as Pandas and Matplotlib aiding in data preprocessing and visualization. Development environments like Jupyter Notebook and PyCharm provided a flexible workspace for coding, testing, and deploying AI solutions. Ethical considerations were integrated into AI projects, ensuring that solutions adhered to ethical guidelines, fairness, and data privacy. This involved techniques for bias mitigation and data anonymization.

2.2.5 Cloud Hosting

The training experience delved into cloud hosting, where projects were hosted on a private cloud server. This hosting environment operated on the Ubuntu Linux system and was supported by tools like Xshell and Xftp. The private cloud infrastructure played a vital role in ensuring the security, scalability, and flexibility of the applications developed during the training period. It facilitated efficient deployment, management, and scaling of the projects, effectively meeting the specific hosting requirements of the training experience.

2.2.6 Ethical Considerations

Ethical considerations were a significant aspect of the training experience, particularly in the context of AI projects. These ethical considerations encompassed fairness, privacy, and compliance with ethical guidelines. Bias mitigation was a crucial strategy, with techniques employed to ensure that AI algorithms delivered fair and equitable outcomes for all users. Data privacy was safeguarded through the anonymization of sensitive information, protecting user identities and confidential data. These ethical considerations were integral to the development of AI-driven solutions, guaranteeing that the solutions adhered to ethical standards, respected user privacy, and delivered fair and unbiased results.

2.3 Details of New Learning

Developing a software application is a complex and dynamic endeavor that demands a diverse range of skills, technologies, and methodologies. It represents an ongoing journey of learning and growth, wherein developers encounter fresh challenges, prospects, and insights that contribute to their advancement and enhancement. Throughout the development of this web application, we encountered numerous novel concepts that expanded our understanding and competencies.

2.3.1 Technical Skills

The training experience provided exposure to an array of technical skills and technologies that proved valuable in the development of the application. This encompassed programming languages, frameworks, libraries, tools, and platforms that were previously unfamiliar. Proficiency in these technical skills and technologies was acquired, enabling the effective utilization of these resources in the construction of the application throughout the industrial training period.

2.3.1.1 Programming Languages

It's noteworthy that during this training experience, the primary new programming language introduced and learned was Python. The acquired proficiency in Python played a pivotal role in the development of the web application. This language provided a valuable addition to the skill set, enhancing the capability to address the various demands of the project. I believe that the knowledge and skills that I acquired during this project will be valuable in my future career.

2.3.1.2 Tools and Technologies

Within the scope of the training, several new tools and technologies were introduced, encompassing:

- **RASA:** This technology opened up opportunities for advanced chatbot development and conversation management.

- **Spring Boot:** The understanding and application of Spring Boot provided valuable insights into developing robust and scalable web applications.

- **Flask:** Proficiency in Flask empowered the development of web services and applications with elegance and efficiency.

- **XShell and Xftp:** These tools became instrumental in connecting to servers, enabling secure and efficient data transfer and remote access.

- **Ubuntu Linux:** Familiarity with Ubuntu Linux enriched the understanding of Linux-based systems and their deployment for hosting applications.

The acquisition of expertise in these tools and technologies significantly contributed to the versatility and capability to address the diverse requirements of the training projects.

2.3.1.3 Design Patterns

One of the design patterns applied during the application's development was the MVC (Model-View-Controller) pattern. The MVC pattern is a software architecture concept that partitions an application into three core components: the model, responsible for handling data and business logic; the view, managing the user interface; and the controller, which acts as a mediator between the model and the view. This architectural pattern encourages the separation of concerns, enhances modularity, and supports testability, making it a widely adopted choice in web application development.

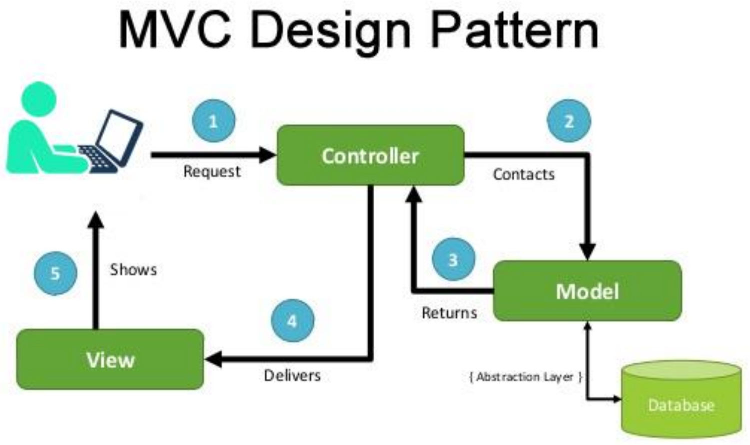


Figure 2. 14: MVC design pattern

Above figure, Figure 2.14 showcases an architecture of MVC design pattern. The application development process involved the application of the MVC pattern to structure and organize the codebase. This included the creation of models, views, and controllers, along with the establishment of connections and interactions between them. The utilization of the MVC pattern was instrumental in constructing a flexible and scalable architectural foundation capable of accommodating future modifications and expansion.

2.3.2 Soft Skills

A range of soft skills and personal attributes emerged as crucial in the application's development. These encompassed skills in problem solving, communication, cooperation, and adaptability. The training involved the acquisition of the knowledge and techniques necessary for the effective utilization of these attributes.

2.3.2.1 Problem Solving

The training encompassed the mastery of problem-solving skills, specifically in the context of identifying and resolving issues and hurdles encountered during the application's development. This encompassed the process of debugging and troubleshooting the program, employing logical reasoning, analysis, and critical thinking in conjunction with diverse tools and methodologies.

2.3.2.2 Time Management

A spectrum of time management skills and techniques were acquired and applied during the development of the web application. These competencies facilitated the achievement of project completion within the established time frame. The process included the establishment of explicit objectives and priorities, enabling a focused approach to essential tasks and the efficient allocation of time and resources. Techniques for dividing projects into smaller, more manageable segments were also learned, enhancing the ability to plan and execute work systematically. Additionally, instruction was received on creating timetables and routines to optimize productivity and efficiency, as well as managing workloads and responsibilities effectively.

2.3.2.3 Communication

Communication skills were honed in the course of developing the application. These abilities proved essential for conveying ideas effectively and collaborating with the senior tech lead, the primary point of contact for this internal project. The training underscored the significance of clear and constructive communication, enabling efficient interaction with the senior tech lead, aligning project objectives, and ensuring its successful development.

CHAPTER 3

OVERCOMMING OBSTACLES

Every learning journey presents its own set of challenges, especially when transitioning from academia to an industrial setting. My training was no exception. This chapter delves into specific obstacles I encountered during the training period and elucidates how I tackled and overcame them.

3.1 Familiarizing with New Frameworks

Obstacle: My academic curriculum did not cover Flask, Spring Boot, or RASA, making them entirely new terrains for me. Their complex architectures and functionalities posed initial hurdles, as I had to navigate through unfamiliar documentation and tools.

Overcome: To bridge this knowledge gap, I adopted a multi-pronged approach. Starting with the official documentation provided foundational understanding. I complemented this with comprehensive Udemy courses [3], which offered structured learning pathways and hands-on practice sessions. To further reinforce concepts and resolve specific queries, I explored relevant YouTube tutorials and insightful articles penned by industry experts. This blended learning strategy ensured a well-rounded grasp of the frameworks and empowered me to apply them effectively in real-world projects.

3.2 Navigating the Linux Environment

Obstacle: My prior exposure was predominantly to Windows-based platforms. The project's reliance on a private cloud anchored in Ubuntu Linux presented a steep learning curve, especially concerning server management and project deployment.

Overcome: Recognizing the importance of mastering this environment, I immersed myself in dedicated Linux tutorials [8]. By starting with the basics of the Linux command line and gradually venturing into advanced server management tasks, I progressively built my confidence and skill set. Networking with colleagues who had expertise in the Linux domain provided invaluable hands-on tips and solutions to specific challenges. Their mentorship, coupled with rigorous practice, enabled me to proficiently manage the server environment, ensuring seamless project deployments.

3.3 Scalability Concerns

Obstacle: As user engagement grew and datasets expanded, the project faced potential bottlenecks, risking system performance and response times.

Overcome: Embracing a microservices architecture, enabled by Spring Boot, allowed components to scale independently. Database performance was optimized using indexing and query optimization techniques. Redis was employed as a caching mechanism to reduce redundant database queries, ensuring swift data retrieval even under load.

3.4 Integration Challenges

Obstacle: The amalgamation of AI models with legacy systems and applications introduced unforeseen complexities. The challenge was ensuring uninterrupted and coherent communication across diverse system components.

Overcome: A middleware layer was introduced to bridge the gap between AI components and traditional systems. Custom API endpoints were developed using Flask, ensuring that data flow and model predictions were routed efficiently. Additionally, consistent feedback sessions with integration teams ensured swift resolution of issues.

CHAPTER 4

FUTURE OPPORTUNITIES AND IMPROVEMENT

4.1 Technological Skill Augmentation

In the rapidly evolving technological landscape, the opportunity to gain new technological skills cannot be understated. An internship provides a unique platform to delve into the intricacies of new tools, technologies, and programming languages. By mastering these, interns can increase their efficiency, executing tasks with greater precision and speed. This hands-on experience fosters flexibility, allowing them to adapt to diverse projects and challenges seamlessly. In the long run, this diversified technical skill set not only bolsters their position in the job market but also secures them against potential job redundancies due to technological advancements. As they continue to adapt and grow, they become indispensable assets to organizations, offering solutions that cater to both present needs and future challenges.

4.2 Professional Skill Cultivation

While technical skills form the backbone of many professions, soft skills or professional skills often emerge as the silent pillars of a successful career. The beauty of internships lies in their ability to mold these very skills. Engaging with teams, clients, and projects during the internship, interns invariably hone their communication skills. They learn the art of articulating their thoughts clearly, ensuring collaborative success. Simultaneously, real-world challenges sharpen their analytical and problem-solving prowess. They begin to approach challenges with a more critical eye, dissecting problems to find efficient solutions. The teamwork involved introduces them to the nuances of group dynamics and often kindles leadership qualities that they weren't previously aware of. Over time, they also learn the invaluable skill of adaptability. As they navigate a plethora of tasks, tools, and environments, they cultivate the ability to thrive amidst change.

4.3 Networking Foundations

One of the more subtle yet profound aspects of an internship is the foundation it lays for professional networking. The industry thrives on connections, and early exposure to this network can catalyze numerous opportunities in the future. By building relationships with experienced professionals, interns can glean insights, receive mentorship, and stay updated with the latest industry trends. This not only aids in personal growth but also opens doors to job referrals and potential collaborations in the future.

4.4 Personal & Professional Growth Assessment

An internship serves as a mirror, reflecting both the strengths and areas demanding more attention of the intern. By recognizing these facets, they can channel their efforts towards amplifying their strengths while simultaneously addressing and mitigating their weaknesses. This focused approach augments their overall productivity and can pave the way for significant advancements in their career. Moreover, it fosters a sense of holistic development, enriching both their personal and professional lives.

4.5 Career Path Exploration

The diversity of roles and projects encountered during an internship is akin to a sampler of the vast career opportunities available. By experiencing varied roles, interns get a clearer picture of which domains resonate with their passion and skill set. This clarity can be instrumental in guiding their future decisions, ensuring that they embark on a fulfilling career path.

4.6 Practical Expertise Acquisition

Theoretical knowledge, while essential, finds its true value when applied practically. Internships offer this golden chance. By working on tangible projects, interns get to bridge the gap between theory and practice. This not only solidifies their understanding but also prepares them for the real-world challenges they will face in their subsequent professional endeavors.

4.7 Strengths Identification

A significant byproduct of internships is the self-awareness they foster. By navigating through various tasks and receiving feedback, interns become acutely aware of their strengths. This self-awareness can amplify their confidence, guiding them towards roles and projects where they can play to their strengths, ensuring both personal satisfaction and professional excellence.

4.8 Weaknesses Awareness

Conversely, internships also shine a light on areas of improvement. Acknowledging these areas is the first step towards personal growth. By actively seeking feedback and being introspective, interns can chart a path to mitigate these weaknesses, transforming them into strengths over time.

Collectively, internships are more than mere professional stints. They are transformative journeys that shape, challenge, and prepare young professionals for the dynamic world that lies ahead. Through technical skill acquisition, soft skill development, networking, and introspective learning, interns are better equipped to navigate their careers and personal growth trajectories.

CHAPTER 5

CONCLUSION

The journey of an internship is a transformative experience, filled with opportunities, challenges, growth, and profound realizations. From mastering new technologies to understanding the subtleties of professional collaboration, internships provide a comprehensive taste of what lies ahead in the professional realm.

The Power of Hands-on Experience

The most striking takeaway from this internship phase is the unparalleled value of hands-on experience. While academic foundations provide the necessary theoretical grounding, it's the practical application during internships that solidifies this knowledge. Navigating real-world challenges, working on tangible projects, and interacting with seasoned professionals offer insights that are impossible to gain within the confines of a classroom.

Skill Augmentation and Holistic Growth

The skill set an intern cultivates during this period extends beyond the mere technical. The professional landscape demands a blend of technological proficiency and soft skills, and internships are the crucibles where this amalgamation occurs. Learning to communicate effectively, solving problems critically, leading teams, and adapting to ever-changing scenarios are as vital as understanding a new programming language or software tool. These skills not only prepare interns for the complexities of the professional world but also contribute to their personal growth, shaping them into well-rounded individuals.

Navigating Career Paths

Internships also provide a unique vantage point, allowing young professionals to gauge their interests and aptitudes. Through varied projects and roles, they gain clarity on what resonates with their passions, facilitating informed decisions about their future career trajectories. Such clarity is invaluable, ensuring that they embark on paths that are both professionally rewarding and personally fulfilling.

Networking and Building Foundations

Beyond the skills and experiences, the relationships cultivated during internships can have lasting impacts. From mentors who provide invaluable guidance to peers who might become future collaborators, the professional network laid down during this phase can yield dividends throughout one's career.

Reflection and Forward Momentum

In retrospection, while internships come with their set of challenges - be it grappling with unfamiliar tools or navigating workplace dynamics they invariably steer growth. Every obstacle faced and overcome during the internship is a stepping stone, preparing young professionals for larger challenges and grander victories in their careers.

In conclusion, the internship phase is not just a transient period but a significant chapter in a professional's journey. It lays down the foundational blocks, equips with the necessary tools, and instills the confidence to face the dynamic world of professionalism. As the pages of this chapter close, they leave behind invaluable lessons, memories, and a sense of anticipation for the exciting journey ahead.

REFERENCES

[1] RASA, "What is Rasa?" RASA Documentation, [Online]. Available: https://rasa.com/docs/rasa/. [Accessed Oct. 28, 2023].

[2] Flask Documentation, "Welcome to Flask," Flask, [Online]. Available: https://flask.palletsprojects.com/en/2.1.x/. [Accessed Oct. 28, 2023].

[3] Udemy, Inc., "[New] Spring Boot 3, Spring 6 & Hibernate for Beginners," Udemy, [Online]. Available: https://www.udemy.com/course/spring-hibernate-tutorial/. [Accessed Oct. 28, 2023].

[4] Spring Boot, "Spring Boot," [Online]. Available: https://spring.io/projects/spring-boot. [Accessed Oct. 28, 2023].

[5] Python Software Foundation, "Welcome to Python.org," [Online]. Available: https://www.python.org/. [Accessed Oct. 28, 2023].

[6] NetSarang Computer, Inc., "XShell: Secure Terminal Emulator Software," [Online]. Available: https://www.netsarang.com/en/xshell/. [Accessed Oct. 28, 2023].

[7] ExaVault, "Xftp: Secure File Transfer Protocol Tool Documentation," [Online]. Available: https://www.exavault.com/docs/xftp. [Accessed Oct. 28, 2023].

[8] DigitalOcean, "A Comprehensive Guide to Linux Commands," [Online]. Available: https://www.digitalocean.com/community/tutorials/linux-commands. [Accessed Oct. 28, 2023].

[9] Real Python, "Natural Language Processing with spaCy and Python," [Online]. Available: https://realpython.com/natural-language-processing-spacy-python/. [Accessed Oct. 28, 2023].

[10] Oracle Corporation, "MySQL Documentation," [Online]. Available: https://dev.mysql.com/doc/. [Accessed Oct. 28, 2023].