Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

Internship Report

COMP 408

(For partial fulfillment of 4th Year/2nd Semester in Computer Engineering)

Submitted by:

Name: Pranima Kansakar

Group: Computer Engineering

University Registration Number: 024322-18

Submitted to:

Mrs. Rajani Chulyadyo, PhD

Assistant Professor

Department of Computer Science and Engineering

Submission Date: 24/07/2023

Letter of Transmittal

July 24, 2023

Mrs. Rajani Chulyadyo, PhD

Internship Coordinator,

Department of Computer Science and Engineering,

Kathmandu University

Subject: Submission of Internship Report

Dear Ma'am,

I present to you my internship report for the course COMP 408. The report includes my experiences and observations during my internship period at Cotiviti Nepal Pvt. Ltd. from April 17, 2023, to July 17, 2023. I hope it reflects my learning during the internship program.

Yours sincerely,

Pranima Kansakar

Group: Computer Engineering 4th Year, 2nd Semester

Registration No. 024322-18

Department of Computer Science and Engineering

Kathmandu University

Acknowledgment

This internship period has been an incredibly valuable learning experience for me, as it has provided me with the opportunity to gain practical knowledge beyond the confines of the classroom and textbooks. Moreover, this experience has significantly contributed to the development of my interpersonal skills and self-confidence.

I would like to express my sincere appreciation to the Department of Computer Science and Engineering for granting me the opportunity to acquire hands-on knowledge in the Information and Technology field through their internship program. This experience has greatly enhanced my understanding of my chosen field of study.

I extend my gratitude to Cotiviti Nepal Pvt. Ltd. for offering me an internship opportunity at their esteemed organization. I am thankful to my mentors, Mr. Asim Mahat and Mrs. Tara Manandhar, my supervisor, Mr. Surendra KC, my trainers, Mr. Keshab Maharjan, Mrs. Rina Shrestha, Mr. Avash Tuladhar and Mr. Nitesh Shrestha and the entire Cotiviti Team for giving me the chance to be part of their company and for their support, which has contributed to my progress. I am grateful to my coordinator, Mrs. Rajani Chulyadyo for assisting me with any problems or queries I encountered during my internship.

Abstract

This report details my three-month internship at Cotiviti Nepal Pvt. Ltd., which I completed as a requirement for the COMP 408 course in Kathmandu University's final semester of my Bachelor of Computer Engineering program. It gives a general overview of the tasks I performed, the tools I utilized, and the knowledge and skills I acquired during the internship. It also addresses my reflection on the internship. This report aims to summarize the valuable learning and professional growth I gained during my time at Cotiviti Nepal.

Keywords: *Python, Django Rest Framework, Docker, SQL, PostgreSQL, ReactJS, Oracle, PL/SQL, OOP, US Healthcare, C#, ASP.Net, Postman, Full Stack Development, Web Development, DBMS*

Table of Contents

Chapter 1: Introduction

This report aims to document my internship experience at Cotiviti Nepal Pvt. Ltd., a reputable company known for its goals and objectives in the field of healthcare analytics. Throughout the report, I will discuss the various aspects of my internship, including the training received, the tasks undertaken, the tools and technologies utilized, and the skills acquired during the internship period. Furthermore, I will outline my reflection on the internship.

1.1. Background

The internship program at the Department of Computer Science and Engineering offers us, the final year undergraduate Computer Engineering students, a valuable opportunity to gain practical experience and insight into the IT industry. As part of the COMP 408 course, a 6-credit course, we are required to complete a 3-month internship at an IT company. This internship allows us to explore the world of IT companies, immerse ourselves in their culture, and apply the theoretical knowledge we have acquired throughout our four years of undergraduate studies in a real-world setting. It serves as a bridge between academia and industry, enabling us to utilize our skills and contribute to the IT sector.

1.2. Internship Goals and Objectives

The goals and objectives I aim to achieve during my internship program are outlined below:

- To improve my technical, professional and communication skills.
- To gain knowledge and skills to handle real-world projects.
- To learn about software development process in professional companies.
- To become familiar with the corporate work culture and professional work environment.
- To develop professional contacts in the IT industry field in the country.

- To discover and evaluate potential career paths that align with my skills and interests.

1.3. Motivation and Significance

An internship in the IT industry is important for anyone who wants to work in this field. It offers a chance to learn by doing and apply what we have learned in the university to real-life situations. During an internship, we can work with the tools and technologies used in the industry and develop our technical skills. It also gives us a chance to meet and connect with professionals in the field, learn from their experiences, and make valuable contacts for future job opportunities. Internships help us understand how things work in the IT industry, the challenges we may face, and the responsibilities we will have in different roles. They provide a supportive environment where we can learn and grow with guidance from experienced professionals. Overall, internships help us gain practical skills, build a professional network, and make informed career decisions in the IT industry.

Chapter 2: Organization Overview

2.1. Introduction to the Company

Established in 2004, Cotiviti Nepal Pvt. Ltd.[1] is an offshore development center for Cotiviti Inc.[2], with a remarkable identity in the Nepal software industry.

Cotiviti is a renowned analytics and solutions provider that is transforming the healthcare industry by enabling its clients to discover new opportunities and extract value. Their solutions have become a crucial component for healthcare payers, who are striving to improve quality while reducing costs. Cotiviti is committed to delivering higher payment accuracy, managing network performance, risk adjustment, and quality improvement programs. Besides, Cotiviti also supports the retail industry by offering data management and recovery audit services that help enhance business outcomes.

2.2. Company's Vision and Mission

The vision of the company is to enable a high-quality and viable healthcare system.

The mission of the company is to improve the healthcare system through our combination of advanced technology, data analytics, and specialized expertise.

2.3. Company's Market

Cotiviti enables exceptional performance for diverse and evolving organizations, enabling them to make smart decisions and encouraging collaboration—both within organizations and with all of their stakeholders. Some of the entities that benefit from this company are as follows.

1. Payers
2. Government

[1] https://www.cotiviti.com.np/
[2] https://www.cotiviti.com/

3. Healthcare providers
4. Employers, brokers and TPAs
5. Retailers

2.4. Company's Solutions

The solutions that the company provides are listed below.

1. Payment Accuracy
2. Risk Adjustment
3. Quality and Performance
4. Consumer Engagement
5. Cotiviti Retail

Chapter 3: Tasks, Projects, and Activities

During my internship period at Cotiviti Nepal Pvt. Ltd., I engaged in a series of tasks, and activities aimed at enhancing my technical as well as interpersonal skills. The internship started with familiarizing myself with the company practices, company ethics, and policies. And later on, I delved into technical skills development.

Following are the activities I was involved in during my internship.

3.1. Compliance Training

To ensure compliance with company policies, I completed several Compliance Training Courses provided by my supervisor, Mr. Surendra KC. These courses covered topics such as,

1. Cotiviti Acceptable Use Policy Acknowledgement
2. Cotiviti Code of Business Conduct and Ethics
3. CMS Combating Medicare Parts C and D Fraud
4. Waste, and Abuse
5. General Compliance
6. Cotiviti Code of Business Conduct and Ethics
7. Cotiviti Conflict of Interest Disclosure
8. Cotiviti Conflict of Interest
9. Cotiviti HIPAA Privacy, Security, and Breach Notification
10. Cotiviti Information Governance
11. Cotiviti Information Security Awareness

Through these courses, I gained a comprehensive understanding of the company's ethical standards and best practices.

I was also provided with various guidelines during my internship, including the Cotiviti Employee Handbook and the Cotiviti Global Workplace Harassment Employee Policy. These documents outlined the company's expectations and

standards regarding employee conduct and provided valuable information on workplace policies and procedures.

3.2. Python Programming

In parallel with my mentorship and team meetings, I learned Python basics and advanced concepts. I covered various important topics included in the modules[3] provided by my mentor.

I documented my learnings in two Jupyter Notebooks. The first included all the Python basics[4] such as Lists, Dictionaries, File Handling, Exception Handling, Object Oriented Programming and so on. The second included all the Python Advanced Concepts[5] such as Decorators, Iterators, Comprehensions, Mixins, Instropections and so on.

3.3. Relational Database using PostgreSQL

The company has opted to utilize raw SQL queries instead of Object-Relational Mapping (ORM) in the Django framework. This decision stems from the necessity of handling substantial volumes of data, which results in inefficient processing when employing ORM. Consequently, I was tasked with developing a relational database using raw SQL queries within the PostgreSQL database management system. I successfully fulfilled this assignment by establishing a database structure[6] for an online bookstore

3.4. Project Synchron

Continuing my training, I had been entrusted with the development of a project known as 'Project Synchron' as my main project for the internship. This project aims to facilitate the synchronization of daily stand-up meetings across multiple teams, providing a valuable tool for both team members and Scrum Masters

[3]https://github.com/Pranima23/CotivitiInternRepo/blob/main/learnPython/python_topics.md
[4]https://github.com/Pranima23/CotivitiInternRepo/blob/main/learnPython/pythonBasics.ipynb
[5]https://github.com/Pranima23/CotivitiInternRepo/blob/main/learnPython/pythonAdvanced.ipynb
[6]https://github.com/Pranima23/CotivitiInternRepo/blob/main/learnPostgreSQL/learnPostgreSQL.sql

throughout the department. To accomplish this, I utilized Django Rest Framework (DRF) for the backend development, PostgreSQL for efficient database management, and ReactJS for the frontend implementation. These technologies have been chosen to ensure a robust and scalable solution that adheres to practices in software development in Cotiviti Nepal. By leveraging DRF, PostgreSQL, ReactJS, and Docker, I was able to deliver a highly functional and streamlined project that enhances collaboration and coordination within and across teams.

I completed this project in the following steps:

3.4.1 Backend Development with DRF and PostgreSQL

To develop the backend[7] of Project Synchron in a structured and organized manner, I employed various methods and techniques. Here is an overview of the approaches I utilized:

1. **Database Design:** I initiated the backend development by meticulously designing the database schema. This involved identifying the necessary tables and their relationships. I incorporated tables to store information such as organization members, positions, teams, scrum masters, team members, standup cards, and updates. Additionally, I applied careful consideration to add necessary constraints to the tables, ensuring data integrity and adherence to the provided functional requirements.
2. **Django Models:** Building upon the database design, I created Django models to represent the tables in the PostgreSQL database. By utilizing Django's object-relational mapping (ORM), I established a seamless connection between the application's backend and the database. This approach allowed for simplified data manipulation, retrieval, and storage.
3. **Serializers:** Leveraging DRF, I designed API endpoints for various functionalities related to standup cards. I explored different serializers available in DRF, and utilized ModelSerializer to handle the serialization and deserialization of data. I explored and implemented various serializer

[7] https://github.com/Pranima23/project_synchron

relations, including SlugRelatedField, PrimaryKeyRelatedField, and Nested Serializers, to establish appropriate relationships between the models. Additionally, I made use of SerializerMethodField to add extra fields to the model serializers.

4. **Views and Viewsets:** To handle API requests, I employed different views and viewsets offered by DRF. I utilized APIView, ListAPIView, and viewsets like ModelViewSet to implement functionalities such as creating, viewing, and editing standup cards. These views and viewsets provided a structured way to handle HTTP methods, such as GET, POST, PUT, and DELETE, for different endpoints.

5. **Customization and Filtering:** I took advantage of the flexibility offered by DRF to customize the default behavior of serializers and views. I implemented writable nested serializers by overriding the default behavior of ModelSerializer, allowing for the creation and updating of related models within the standup card serializers. Additionally, I made use of filter querysets in all views to enable filtering of standup cards based on specific criteria. I implemented filters based on the logged-in user's team, restricting access to only their team's standup cards. This measure ensured that teams could not view or edit standup cards belonging to other teams. Furthermore, I incorporated filters in ListAPIView for date, month, sprint ID, and other relevant criteria to enhance the filtering capabilities of the API. These customizations enhanced the overall usability and security of the application by restricting access to appropriate teams and enabling advanced filtering options for standup cards.

6. **Groups and Permissions**: I leveraged the built-in groups feature provided by Django to categorize users into distinct roles, specifically as scrum masters and scrum members. By assigning users to the appropriate groups, I could effectively differentiate their access levels and permissions within the application. For instance, scrum masters were granted read and write permissions, enabling them to create, view, and update standup cards. On the other hand, scrum members were assigned read-only permissions,

limiting their access to viewing standup cards without the ability to modify or delete them.

7. **JWT Authentication:** I further enhanced the authentication and security aspects of Project Synchron by implementing JWT (JSON Web Token) authentication.

By utilizing these structured methods and techniques, I was able to develop the backend of Project Synchron efficiently. The process not only enhanced my understanding of database design, Django models, authentication, permissions and DRF but also strengthened my problem-solving capabilities in the context of developing RESTful APIs.

3.4.2. Frontend Development with ReactJS

During the frontend[8] development of Project Synchron, I followed a structured approach to effectively utilize ReactJS and implement the desired functionalities. The steps involved in the frontend development process are as follows:

1. **Understanding ReactJS Concepts:** I began by familiarizing myself with key concepts in ReactJS, including components, state management, props, and hooks. This understanding laid the foundation for building efficient and reusable components.
2. **Building the Basic Structure:** To structure my React components effectively, I initially developed the frontend using HTML and CSS. This approach allowed me to create a well-organized layout for the frontend pages.
3. **Creating Functional Components:** I proceeded to develop functional components that were essential for the frontend page, specifically the detailed view of standup cards for today and yesterday. Each component served a specific purpose and encapsulated a particular functionality.
4. **Event Handling and Form Handling:** To enable editing functionality for the scrum master, I implemented event handling and form handling. I

[8] https://github.com/Pranima23/project_synchron_frontend

passed event handler methods as props to the respective components and utilized the useState hook to manage the component's state effectively.

5. **Implementing API Calls with useEffect:** To retrieve standup card data from the backend, I utilized the useEffect hook. I made API calls within the useEffect hook to fetch the necessary data and update the component's state accordingly.
6. **Testing and Bug Fixing:** Throughout the development process, I rigorously tested the frontend components and functionality to identify and address any issues or bugs that arose. This iterative testing and bug fixing phase ensured a smooth and seamless user experience.

By following these structured steps, I was able to successfully develop the frontend of Project Synchron using ReactJS. The result was a user-friendly and interactive interface that provided the desired functionalities, including the detailed view of standup cards and the ability to edit them for the scrum master.

3.5. US Healthcare Training

During my internship at Cotiviti Nepal, I had the opportunity to attend a comprehensive four-day session on US healthcare conducted by Mrs. Rina Shrestha. Through this session, I learned:

1. The reasons behind the complexity of the US healthcare system.
2. How patients access medical facilities and treatment in the US healthcare system.
3. The major players involved in providing medical and financial services in the US healthcare market.
4. Different types of insurance plans, including HMO and PPO.
5. The concept of private health insurance, Medicare, and Medicaid.
6. The process of professional claims and facility claims, along with billing forms like CMS-1500 and UB-04.
7. Procedure codes, diagnosis codes, place of services, and drug codes (NDC).
8. The difference between branded and generic drugs.

9. Mail order indicators.
10. The concept of formulary versus non-formulary drugs.

These learnings provided me with a comprehensive understanding of the US healthcare system, insurance coverage, claims processing, and important codes and indicators used in the industry.

3.6. Intermediate to Advanced Database Training

During the database sessions led by Mr. Keshab Maharjan, I learned and practiced various aspects of SQL and PL/SQL using Oracle Live SQL. Here is a summary of the key topics covered:

1. Basic SQL queries: I revised and practiced basic SQL queries.
2. PL/SQL statements and variables: I gained an understanding of PL/SQL statements, variables, and their usage.
3. Procedures and packages: I learned how to create and utilize procedures and packages in PL/SQL.
4. Triggers: I explored row-level and statement-level triggers and completed assignments involving trigger implementation.
5. Advanced SQL concepts: I learned about inserting multiple rows, NULL values, and used keywords such as IN, BETWEEN, GROUP BY, ORDER BY, HAVING, DISTINCT, COUNT, EXISTS, NOT EXISTS, JOIN, and TO_DATE.
6. Metadata tables and dynamic queries: I discovered metadata tables in Oracle and also practiced dynamic queries.
7. Exception handling: I learned techniques for handling exceptions in PL/SQL and successfully completed assignments related to debugging through exception handling.
8. Data analysis techniques: In the final sessions, I learned about data analysis using dummy data and tables, and was introduced to advanced database concepts.

These database sessions provided me with a comprehensive understanding of SQL and PL/SQL, and equipped me with practical skills for working with databases effectively.

3.7. OOP with C# Training

During the OOP (Object-Oriented Programming) sessions using C# in my internship, conducted by Mr. Avash Tuladhar and Mr. Nitesh Shrestha, I gained valuable knowledge and accomplished various tasks. Here is a summary of what I learned and achieved during the sessions:

1. Basic C# Programming: I learned about C# features, syntax, variables, operators, and logical expressions. I practiced writing programs to perform actions like depositing, withdrawing, and checking balance.
2. OOP Concepts: I delved into the building blocks of OOP, such as classes, objects, methods, and attributes. I grasped important concepts like inheritance, encapsulation, abstraction, and polymorphism.
3. Collections: I learned about different types of collections in C#, including arrays, lists, dictionaries, stacks, and queues. I explored their usage and benefits in programming.
4. Interfaces and Exception Handling: I gained knowledge about interfaces by building a file storing system with specific methods. I also learned about exception handling techniques to handle errors and exceptions in my programs.
5. Generics and Wallet Project: I practiced using generics by developing a wallet class capable of holding objects of any value type. I implemented methods to add and remove items from the wallet.
6. Student Information System: I was assigned a project to design a Student Information System using Blazor and MVC in the .NET framework. I successfully created models, controllers, and APIs for various classes, and implemented frontend functionalities using Blazor.
7. Project Presentation: I presented my code for the Student Information System project, and received positive feedback on its comprehensiveness.

I also had the opportunity to view and learn from the code demos and presentations of my colleagues on diverse topics such as arrays, delegates, enums, threading, and more.

Throughout the OOP sessions, I not only gained a strong understanding of C# programming and OOP principles but also applied my knowledge to practical projects. The hands-on assignments and project work allowed me to develop skills in designing classes, implementing functionalities, and troubleshooting issues. Overall, the sessions provided me with a solid foundation in C# and OOP, equipping me with valuable skills for software development.

3.8. Docker

During my internship, I had the opportunity to learn about Docker, a popular containerization platform. I gained a solid understanding of Docker concepts and techniques. I explored different aspects of Docker, such as container creation, management, and deployment. I learned how to work with Docker images and containers, and became familiar with essential Docker commands for pulling, running, and stopping containers. Additionally, I learned about the importance of Docker compose for using multiple services, Docker volumes for persisting data, and Dockerfiles for building custom Docker images. Overall, my experience with Docker during my internship provided me with valuable insights into containerization and its significance in modern software development and deployment.

Chapter 4: Tools and Techniques

Following are the tools I have used during my internship period.

4.1. Project Management Tools

4.1.1. Jira

Jira[9] is a software application developed by the Australian software company Atlassian that allows teams to track issues, manage projects, and automate workflows.

4.2. Development Tools

4.2.1. Visual Studio Code

Visual Studio Code[10] is a source-code editor made by Microsoft used for writing code and development. It has great support for development operations like debugging, task running, and version control. It has features like syntax highlighting, intelligent code completion, code snippets and embedded Git.

4.2.2. Postman

Postman[11] is an application used for API testing. It is an application platform for building and using APIs.

4.2.3. Docker

Docker[12] is an open source platform that enables developers to build, deploy, run, update and manage containers—standardized, executable components that combine application source code with the operating system (OS) libraries and dependencies required to run that code in any environment.

[9] https://www.atlassian.com/software/jira
[10] https://code.visualstudio.com/
[11] https://www.postman.com/
[12] https://www.docker.com/

4.2.4. Visual Studio

Visual Studio[13] is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs including websites, web apps, web services and mobile apps.

4.2.5 Jupyter Notebook

Jupyter Notebook[14] is an open-source web application that allows users to create and share documents containing live code, visualizations, and explanatory text. It provides an interactive computing environment that supports various programming languages, including Python, R, and Julia. Jupyter Notebook is widely used in data science, machine learning, and scientific research due to its ability to combine code execution, data visualization, and narrative text in a single document.

4.2.6 Oracle Live SQL

Oracle Live SQL[15] is a web-based SQL editor with an Oracle database built-in. It's maintained by Oracle and allows you to write and run SQL statements easily without having to set up your own Oracle database.

4.3. Version Control Tools

4.3.1. Git

Git[16] is a version control system (VCS) used for tracking changes in the project directory and managing source code. It is a command line tool, and is free and open source. It can be used to track long-term change history of every file. It takes snapshots of a project and stores it as unique versions.

[13] https://visualstudio.microsoft.com/
[14] https://jupyter.org/
[15] https://livesql.oracle.com/
[16] https://git-scm.com/

4.3.2. GitHub

GitHub[17] is a Git repository hosting service. It is a provider for Internet hosting for software development and version control using Git. It provides a Web-based graphical interface.

4.4. Languages, Libraries and Frameworks

4.4.1. Python

Python[18] is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

4.4.2. JavaScript

JavaScript is a light-weight, high-level programming language. It is often referred to as the programming language of the Web. It runs on the client side of the web and can be used to create and control dynamic website content, i.e. anything that refreshes, moves or changes on your screen without requiring to reload a web page manually.

4.4.3. React

React[19] (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces based on components. It is

[17] https://github.com/
[18] https://www.python.org/
[19] https://react.dev/

maintained by Meta (formerly Facebook) and a community of individual developers and companies. React can be used to develop single-page, mobile, or server-rendered applications with frameworks like Next.js. Because React is only concerned with the user interface and rendering components to the DOM, React applications often rely on libraries for routing and other client-side functionality.

4.4.4. Django Rest Framework

Django REST framework[20] is a powerful and flexible toolkit for building Web APIs. It is a python-based framework which is highly scalable.

4.4.5. PL/SQL

PL/SQL[21] is a combination of SQL along with the procedural features of programming languages. It was developed by Oracle Corporation in the early 90's to enhance the capabilities of SQL. PL/SQL is one of three key programming languages embedded in the Oracle Database, along with SQL itself and Java.

4.4.6 C#

C#[22] (pronounced C sharp) is a general-purpose high-level programming language supporting multiple paradigms. C# encompasses static typing, strong typing, lexically scoped, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines.

4.4.7 ASP.NET Framework

.NET[23] (pronounced as dot net) is the free, open-source, cross-platform framework for building modern apps and powerful cloud services. ASP.NET is a framework for building web apps and services with .NET and C#.

[20] https://www.django-rest-framework.org/
[21] https://www.oracle.com/database/technologies/appdev/plsql.html
[22] https://learn.microsoft.com/en-us/dotnet/csharp/
[23] https://dotnet.microsoft.com/en-us/

4.5. Database

4.5.1. PostgreSQL

PostgreSQL[24] is a powerful, open-source object-relational database system with over 35 years of active development that has earned it a strong reputation for reliability, feature robustness, and performance.

[24] https://www.postgresql.org/

Chapter 5: Reflection on The Internship

My internship experience at Cotiviti Nepal Pvt. Ltd. has been incredibly valuable and fulfilling. Throughout the internship, I had the opportunity to enhance my technical skills, gain practical experience, and learn about industry best practices. Here are my reflections on the internship:

Technical Skills Development: The internship provided me with hands-on experience in using various tools, technologies, and programming languages such as Python, JavaScript, ReactJS, Django, PostgreSQL, and C#. I learned how to develop RESTful APIs, design databases, implement frontend interfaces, and work with different frameworks and libraries. Throughout the internship, I gained a solid understanding of different aspects of software development, including backend and frontend development, database management, and OOP concepts using C#. I also had the opportunity to attend training sessions on database management using SQL and PL/SQL. This practical exposure enhanced my technical skills and improved my ability to develop efficient and scalable software solutions.

Collaboration and Professionalism: Working in a professional environment like Cotiviti Nepal allowed me to collaborate with experienced professionals and colleagues. I gained insights into effective teamwork, communication, and time management. I also learned about the importance of adhering to company policies, maintaining ethical standards, and delivering quality work within deadlines. These experiences enhanced my professionalism and prepared me for future roles in the IT industry.

Practical Application of Theoretical Knowledge: The internship bridged the gap between theoretical knowledge gained during my undergraduate studies and its practical application in real-world scenarios. Through tasks, projects, and training sessions, I was able to apply the concepts and principles learned in the classroom to develop functional software solutions. This practical experience

deepened my understanding of programming concepts and their real-world implications.

Exposure to Industry Practices: During the internship, I had the opportunity to work with industry-standard tools, follow best practices in software development, and gain insights into the IT industry's work culture. I learned about version control, project management tools, and collaborative development practices. This exposure to industry practices equipped me with the necessary skills and knowledge to adapt to the professional software development environment.

Personal and Professional Growth: The internship provided a platform for personal and professional growth. I developed problem-solving skills, critical thinking abilities, and the capacity to handle real-world challenges. I gained confidence in my technical capabilities and learned to approach complex tasks systematically. Moreover, the exposure to diverse projects, training sessions, and interactions with professionals helped me explore potential career paths and make informed decisions about my future in the IT industry.

Overall, my internship at Cotiviti Nepal Pvt. Ltd. was a rewarding experience that allowed me to acquire new skills, apply my knowledge in practical settings, and grow both personally and professionally. The hands-on training, challenging tasks, and exposure to industry practices have significantly contributed to my development as a software engineer. I am grateful for the opportunity and believe that the skills and experiences gained during the internship will serve as a strong foundation for my future career endeavors in the IT industry.

Chapter 6: Conclusion

The internship at Cotiviti Nepal Pvt. Ltd. was a transformative experience that enriched my technical skills, practical experience, and understanding of the IT industry. Through training sessions, hands-on projects, and exposure to various tools and technologies, I gained valuable experience in software development practices and project management. The guidance and mentorship provided by experienced professionals at Cotiviti Nepal were instrumental in my learning journey. Interacting with colleagues and learning from their projects broadened my perspective and enriched my understanding of different domains within the IT industry.

The internship also provided insights into the professional work environment, emphasized communication, collaboration, and adhering to company policies. Exploring career paths and exposing myself to different domains further broadened my knowledge. Overall, the internship provided a strong foundation for my personal and professional growth in the IT industry.