VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI-590018, KARNATAKA

REPORT ON

"Inter/Intra Institutional Internship"

(21INT49)

Submitted in the partial fulfilment of requirement for the Award of degree

Submitted by

ANUSHREE B KUPPAST **USN: 4BD21CS020**

Faculty Mentor

Dr. Naveen Kumar K R

Department of

CS&E BIET,

Davanagere

Department of Computer Science and Engineering

Bapuji Institute of Engineering and Technology

Davanagere-577004

2022-23

VISVESVARAYA TECHNOLOGICAL UNIVERSITY,
BELAGAVI – 590 018

INTERNSHIP REPORT

Submitted in the partial fulfilment of requirement for the Award of degree

Submitted by :

ANUSHREE B KUPPAST

USN : 4BD21CS020

"Inter/Intra Institutional Internship"

(21INT49)

Examiner 1 **Name_____** **Designation _____**		**Examiner 2** **Name_____** **Designation _____**

Bapuji Institute of Engineering and Technology
Davanagere-577004
2022-23

Bapuji Educational Association ®

Bapuji Institute of Engineering and Technology
Davanagere - 577004

Computer Science and Engineering

CERTIFICATE

This is to certificate that the Internship report submitted by **Ms. ANUSHREE B KUPPAST** bearing **USN: 4BD21CS020** a bonafide student of BIET, Davanagere in partial fulfilment for the award of bachelor of engineering in Computer Science and Engineering of VTU Belagavi during the academic year 2022-23. It is certified that all corrections/ suggestions indicated for internal assessment have been incorporated in the report. The report of internship has been approved as it satisfies the academic requirements in respect of internship practice prescribed for the said degree.

INTERNSHIP COORDINATORS

Dr. Naveen Kumar K R	**Prof. Gangadharappa S**	**Dr. Naresh Patel K M**
Associate Professor	**Assistant Professor**	**Associate Professor**
Department of CS&E	**Department of CS&E**	**Department of CS&E**

Head of Department

Dr. Nirmala C R

Professor & Head

Department of CS&E

DECLARATION

I, **ANUSHREE B KUPPAST**,**USN : 4BD21CS020** , student of B.E in Computer Science and Engineering, **Bapuji Institute of Engineering and Technology, Davanagere**, hereby declare that the internship work entitled **"Inter/Intra Institutional Internship"** submitted to the **Visvesvaraya Technological University** during the academic year 2022-2023 is record of an original work done by us under the guidance of **Dr. Naveen Kumar K R**, Department of Computer Science and Engineering, Bapuji Institute of Engineering and Technology Davanagere. This internship work is submitted in partial fulfilment of the requirements for the award of the degree of **Bachelor of Engineering** in **Computer Science and Engineering.** The results embodied in this reporthave not been submitted to any other university or institute for any degree.

ANUSHREE B KUPPAST

USN:4BD21CS020

ACKNOWLEDGEMENT

Salutations to our beloved and highly esteemed institute**, "BAPUJI INSTITUTEOF ENGINEERING AND TECHNOLOGY"** for having well qualified staff and lab furnished with necessary equipment.

We express our sincere thanks to our faculty mentor **Dr. Naveen Kumar K R** for giving us constant encouragement, support and valuable guidance throughout the course of the internship without whose guidance this internship would not have been achieved.

We express our sincere thanks to our Internship Coordinators Prof**. GangadharappaS and Dr. Naresh Patel K M** for giving us constant encouragement, support and valuable guidance throughout the course of internship without whose guidance this internship would not have been achieved.

We express wholehearted gratitude to **Dr. Nirmala C R,** H.O.D of Computer Scienceand Engineering Department. We wish to thank her for making our task easy by providing her valuable help and encouragement.

We also express our wholehearted gratitude to our principal, **Dr. H B Aravind** forhis moral support and encouragement.

We would like to extend our gratitude to all the staff of the Computer **Science and Engineering Department** for their help and support. We have benefited a lot fromthe feedback, suggestions given by them.

We would like to extend our gratitude to all our family members and friends for theiradvice and moral support.

COMPANY CERTIFICATE

GENESIS TRAINING

INTERNSHIP COMPLETION CERTIFICATE

This certificate is issued to Mr./Ms.

ANUSHREE BASAVARAJ KUPPAST

With **USN - 4BD21CS020** Studying in **COMPUTER SCIENCE & ENGINEERING** at Bapuji Institute of Engineering & Technology, Davanagere for the completion of a three-week internship from **October 12,2022 to November 04,2022**

During this period, he/she has completed the internship on "Soft Skills, Fundamentals of Java programming and Fundamentals of Android App Development"

AKSHAY V AGASTHYA
VICE PRESIDENT (L&D)
GENESIS BENGALURU

NAVEEN NAGARAJ
MANAGING DIRECTOR
GENESIS BENGALURU

SUMMARY

Soft Skills:

- I honed my communication skills, both written and verbal, through interactions with colleagues and supervisors.
- I improved my teamwork and collaboration abilities by actively participating in group projects.
- Time management and organization skills were developed as I juggled multiple tasks and deadlines.
- I learned to adapt and remain flexible in a dynamic work environment, which greatly improved my problem-solving skills

Basic Java:

- I gained a foundational understanding of Java, covering concepts like variables, data types, operators, control structures (loops and conditionals), functions, and exception handling.
- I practiced writing efficient and maintainable code by following best practices and coding standards.
- I learned how to use Integrated Development Environments (IDEs) for Java development and became proficient in debugging and troubleshooting code.

Android Development:

- I delved into Android app development, learning how to design user interfaces (UI) using XML layouts and implement functionality with Java.
- I acquired knowledge of the Android Activity lifecycle and how to handle user interactions within apps.
- I became skilled in using Android Studio, the official IDE for Android development, to build, test, and debug Android applications

CONTENTS

LIST OF FIGURES

CHAPTER 1

ABOUT THE COMPANY

Genplus (Formerly known as Genesis Training) is an organization of keen, enthusiastic and driven individuals who are willing to push the boundaries of formal education and to explore and create practical and implementable strategies of learning and retention. Genesis was founded in 2015 by Naveen Nagaraj who is the Alumnus of BMS College of Engineering where Mr. Theertharaj was appointed as a CEO, who is the Alumnus of SJB Institute of Technology and Texas A&M University, USA.

Genesis is an innovative start-up in the field of Campus Recruitment Training. Their motto is to 'Create Values'. Keeping this in mind, they not only provide training and assess the students but also analyze the results, scientifically identify areas of improvement and provide strategic follow-up programs for the same. They believe that this, one of a kind, training program will better help the students to maximize their areas of strength and work on areas which require improvement.

Vision:

Their vision is to create a platform where learning is no longer a burden but a choice made by that specific learner to transform their dreams into reality.

Mission:

Their mission is to enhance the level of excellence amongst students and pave the way for a better career opportunity.

Profile

Genesis is built on values such as innovation, transparency and co-existence. Knowledge transfer, follow-up and data analysis are their key differentiator and their unique selling point. Their data analysis not only highlights the participant's or student's key strengths but also on their areas of improvement. They believe that this representation of the data enables the learner to better plan their study schedule.

Services Provided

1) Campus Recruitment Training

Refreshing & refining the skill sets, vital for an aspiring apprentice, to crack the initial evaluations of companies visiting their campus for the purpose of recruitment.

2) Technical Workshops

Simulated learning emphasizing on engaging the partakers to involve in collaborative and individual projects that are designed to enhance level of technical familiarity.

3) Placement Drive Management

Collaborative initiatives with the campus in driving company specific programs from Training to Placement process.

4) Corporate Training

Conducting need-based workshops to up-skill the existing workforce and equip them with state-of-the-art dexterity to execute their everyday responsibilities.

5) Intellectual Training and Personality Development

Deliberations on the vastness of the power of knowing and embracing the understanding of the same into daily life to achieve a better self.

6) Staffing Solutions

Delivering qualitative personnel to the job market for varied sectors. Creating varied employment negotiations based on needs of companies associated with.

7) Software Research and Development

Designing client specific software, which are implementable and would enhance the efficiency in delivering its intent. Providing upgrades to existing systems to bring in intensity to performance.

8) Collaborations

Creating platforms to visualize, ideate and bring in a combined effort to turn the same into reality. We appreciate the courage to dream and the will to build it and would like to associate ourselves in such splendid events.

Genesis Methodology

For their training program they adapt to one of a kind, tested and trusted model called **A-A-D-D**. This model helps them to deliver a result driven training program. A-A-D-D is a continuous cycle which consists of 4 key elements:

Assess: Their first step into the program starts by assessing the students with a BASELINE assessment. This assessment is a blend of questions and patterns of all major recruiters.

Analyse: The data from the assessment is crucially analysed to identify the strengths and weaknesses of the students in various topics and subtopics.

Design: They later design the module and plan of action to delivery specifically to the need.

Deliver: This step focuses on executing the standard operating procedure designed in the previous step by our experienced and capable facilitators.

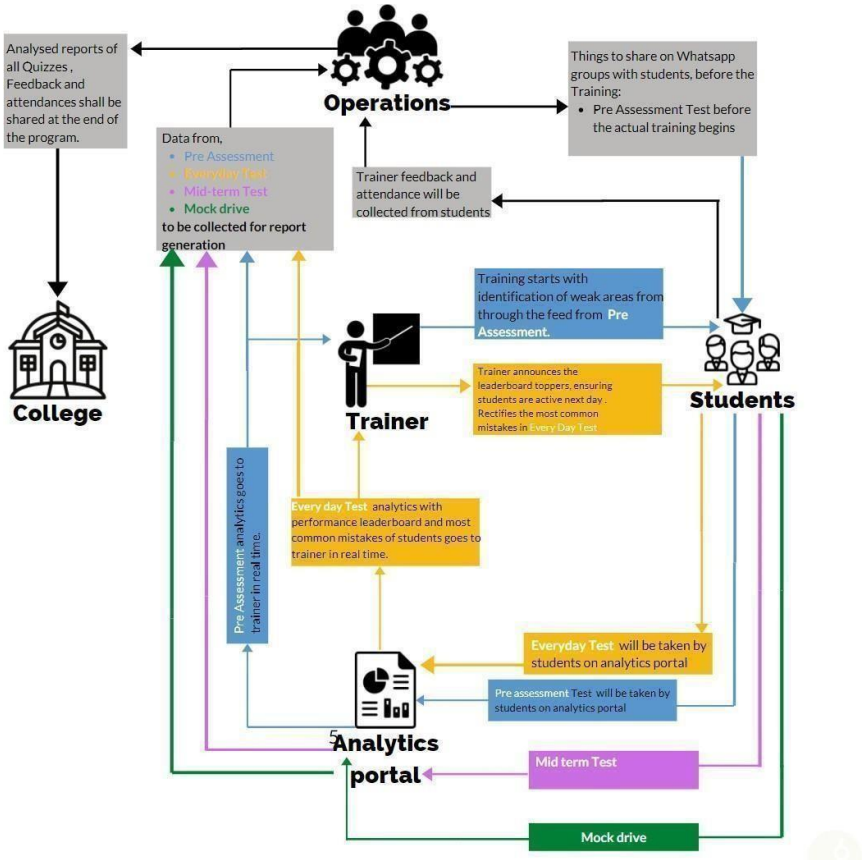

Fig.1.1 Genesis Methodology

Success Story

Founded in 2015, with in a short span of 5 years we have 50+ clients and the clientele include Karnataka's top engineering colleges like RVCE and BMSCE. We could achieve this only because of our dedication to deliver results-oriented training. Genesis Training is an innovative start-up in the field of Campus Recruitment Training. Our motto is to 'Create Values'. Keeping this in mind, we notonly provide training and assess the students, but we also analyze the results,scientifically identify areas of improvement and provide strategic follow-up programs for the same. We believe that this, one of a kind, training program will better help the students to maximize their areas of strength and work on areas whichrequire improvement. We begin where our competitors end. We help transform ordinary to extraordinary!!!

Fig.1.2 Success Map

LEADERSHIP TEAM and Background

Genesis is founded by Naveen Nagaraj who is the Alumnus of BMS College of Engineering. Under his leadership Company had seen a tremendous success in a span of 5 years. To create an impact and accelerate in the EduTech industry, The company was registered as a **Genplus training and consulting services pvt ltd**. With the vision of creating a platform where learning is no longer a burden but a choice made by that specific learner to transform their dreams into reality. With this huge step of reorganization, The board decided to bring in a fresh young talent to **drive the corporate culture into the team, Where Mr. Theertharaj was appointed as a CEO,who is the Alumnus of SJB Institute of Technology and Texas A&M University, USA.**

- **Theertharaj- CEO and BOD**

  Theertharaj, CEO, Genesis comes with 5+ years of industry experience fromworld leading cloud computing organization – Microsoft Corporation working at its Headquarters, Washington, USA. Holds a master‟s degree in Computer Engineering from Texas A&M University, USA. Having a wide experience in C, C++, C#, .Net, Microsoft Azure Technologies implementing highly scalable, reliable software solutions. As a passionate trainer, had trained for 3+ years on various technologies and Campus recruitment oriented technical curriculum.

- **Varun Gowda- VP Business development and BOD**

  The travel bug bit this person real hard and he decided to make a career out of it. A motivated and an experienced individual handling the Business expansion and sustenance. He bring with him a bankable experience in business operations and is currently the Vice President of Business development at Genesis

- **Akshay- VP L&D and BOD**

  A corporate trainer with 15 years of experience is an expert in Soft Skills and verbal ability training. Akshay has Bachelor‟s degree in commerce and also holds MA honours in English literature. He comes from an extremely rich corporate training back ground.

- **Naveen- Founder, BOD and Strategic advisor**

  A mechanical engineer who dreamed beyond his might and made it into reality. The founder of Genesis as an idea and company. He is a musician and keen on creating visual delights on screen. He is now the Board of Director of the company and assists in management level decisions

- **Chethan Simha- Technical Content and delivery advisor**

  Techie by passion, Alumnus of **RNSIT** and **University of Illinois, Chicago** who exactly knows what the industry needs. A strong believer of the statement "One who knows how to do can get a job anywhere but the one who knows why to do will be the other persons boss".

Technology Services

The companies‟ services on technology are listed below.

Software Development Life Cycle

SDLC or the Software Development Life Cycle is a process that produces software with the highest quality and lowest cost in the shortest time possible. SDLC provides a well-

structure flow of phases that help an organization to quickly produce high-quality software which is well- tested and ready for production use. The SDLC involves six phases as explained in the introduction. Popular SDLC models include the waterfall model, spiral model, and agile model.

SDLC works by lowering the cost of software development while simultaneously improving quality and shortening production time. SDLC achieves these apparently divergent goals by following a plan that removes the typical pitfalls of software development projects. That plan starts by evaluating existing systems for deficiencies.

Next, it defines the requirements of the new system. It then creates the software through the stages of analysis, planning, design, development, testing, and deployment. By anticipating costly mistakes like failing to ask the end-user or client for feedback, SLDC can eliminate redundant rework and after-the-fact fixes.

It‟s also important to know that there is a strong focus on the testing phase. As the SDLC is a repetitive methodology, you have to ensure code quality at every cycle. Many organizations tend to spend little effort on testing while a stronger focus on testing can save them a lot of rework, time, and money.

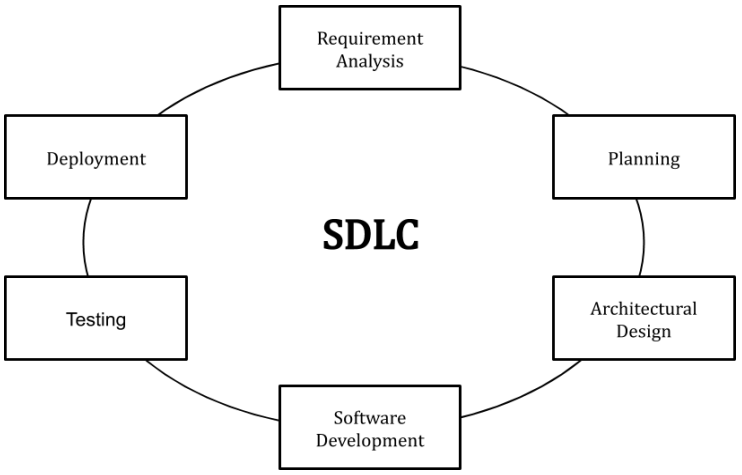

Fig 1.5 SDLC Life cycle

Stages and Best Practices

Following are the best practices and stages of SDLC ensure the process works in a smooth, efficient, and productive way.

1. Identify the Current Problems

“What are the current problems?” This stage of the SDLC means getting input from all stakeholders, including customers, salespeople, industry experts, and programmers. Learn the strengths and weaknesses of the current system with improvement as the goal.

2. Plan

“What do we want?” In this stage of the SDLC, the team determines the cost and resources required for implementing the analysed requirements. It also details the risks involved and provides sub-plans for softening those risks. In other words, the team should determine the feasibility of the project and how they can implement the project successfully with the lowest risk in mind.

3. Design

“How will we get what we want?” This phase of the SDLC starts by turning the software specifications into a design plan called the Design Specification. All stakeholders then review this plan and offer feedback and suggestions. It‟s crucial to have a plan for collecting and incorporating stakeholder input into this document.Failure at this stage will almost certainly result in cost overruns at best and the total collapse of the project at worst.

4. Build

“Let‟s create what we want.” At this stage, the actual development starts. It‟s important that every developer sticks to the agreed blueprint. Also, make sure you have proper guidelines in place about the code style and practices.

For example, define a nomenclature for files or define a variable naming style such as camel Case. This will help your team to produce organized and consistent code that is easier to understand but also to test during the next phase.

5. Code Test

“Did we get what we want?” In this stage, we test for defects and deficiencies. We fix those issues until the product meets the original specifications. In short, we want to verify if the code meets the defined requirements.

6. Software Deployment

“Let‟s start using what we got.” At this stage, the goal is to deploy the software to the production environment so users can start using the product. However, many organizations choose to move the product through different deployment environments such as a testing or staging environment.

This allows any stakeholders to safely play with the product before releasing it to the market. Besides, this allows any final mistakes to be caught before releasing the product.

Scrum

Scrum is a framework that allows development teams flexibility to respond to changing situations. This framework has sufficient control points in place to ensure the team does not stray from the desired outcome, and that issues can be identified and resolved and process adjustments made while the effort is still underway.

The Scrum Lifecycle starts with a prioritized backlog but does not provide any guidance as to how that backlog is developed or prioritized.

The Scrum Lifecycle consists of a series of Sprints, where the end result is a potentially shippable product increment. Inside of these sprints, all of the activities necessary for the development of the product occur on a small subset of the overall product.

Below is a description of the key steps in the Scrum Lifecycle:

1. Establish the Product Backlog.
2. The product owner and development team conduct Sprint Planning. Determine the scope of the Sprint in the first part of Sprint Planning and the plan for delivering that scope in the second half of Sprint Planning.
3. As the Sprint progresses, development team perform the work necessary to deliver the selected product backlog items.
4. On a daily basis, the development team coordinate their work in a Daily Scrum.
5. At the end of the Sprint the development team delivers the Product Backlog Items

selected during Sprint Planning. The development team holds a Sprint Review to show the customer the increment and get feedback. The development team and product owner also reflect on how the Sprint has proceeded so far and adapting their processes accordingly during a retrospective.

6. The Team repeats steps 2–5 until the desired outcome of the product has been met.

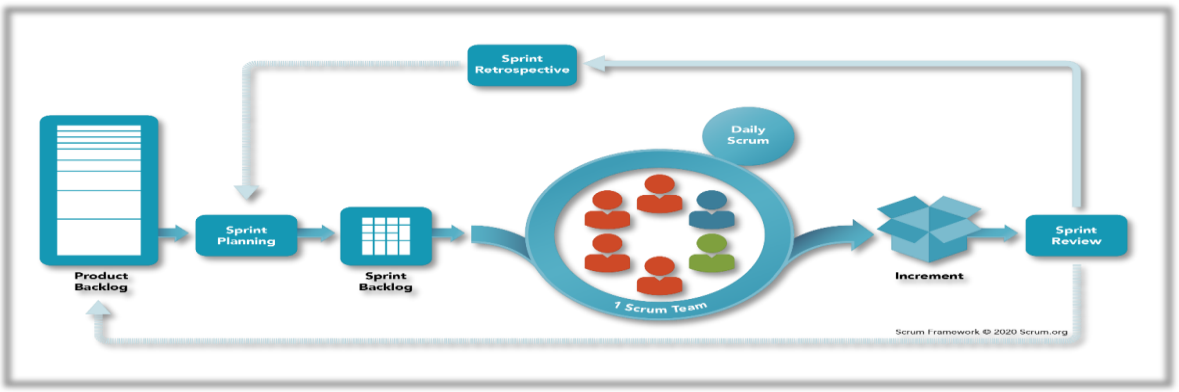

Fig 1.4 Scrum Lifecycle

Azure

Microsoft Azure, formerly known as Windows Azure, is Microsoft's public cloud computing platform. It provides a range of cloud services, including compute, analytics, storage and networking. Users can pick and choose from these services to develop and scale new applications, or run existing applications in the public cloud.

The Azure platform aims to help businesses manage challenges and meet their organizational goals. It offers tools that support all industries -- including e-commerce, finance and a variety of Fortune 500 companies -- and is compatible with open source technologies. This provides users with the flexibility to use their preferred tools and technologies. In addition, Azure offers 4 different forms of cloud computing: infrastructure as a service (IaaS), platform as a service (PaaS), software as a service (SaaS) and serverless.

Microsoft charges for Azure on a pay-as-you-go basis, meaning subscribers receive a bill each month that only charges them for the specific resources they have used.

Working of Azure

Once customers subscribe to Azure, they have access to all the services included in the Azure portal. Subscribers can use these services to create cloud-based resources, such as virtual machines (VM) and databases.

Because Microsoft Azure consists of numerous service offerings, its use cases are

extremely diverse. Running virtual machines or containers in the cloud is one of the most popular uses for Microsoft Azure. These compute resources can host infrastructure components, such as domain name system (DNS) servers; Windows Server services - such as Internet Information Services (IIS); or third-party applications. Microsoft also supports the use of third-party operating systems, such as Linux. Azure is also commonly used as a platform for hosting databases in the cloud.

CHAPTER 2

SOFT SKILLS

Business Communication

Effective business communication involves the exchange of information within and beyond an organization to achieve specific goals. It encompasses verbal, written, and non-verbal forms of communication. Successful business communication is characterized by clarity, precision, relevance, and timeliness, fostering professionalism and adaptability.Its purposes encompass informing, convincing, instructing, and nurturing relationships. Communication channels can be either formal or informal, with potential obstacles such as noise, language diversity, perceptual biases, and insufficient feedback. Proficient business communication encourages collaboration, informed decision-making, and organizational efficiency while nurturing positive relationships with stakeholders, thereby contributing to a company's prosperity and expansion.

Resume Format

Crafting a resume effectively involves a step-by-step approach to showcase your qualifications and abilities. Here's a fundamental structure to help you commence:

• Contact Details

• Summary of Resume

• Professional Work Experience

• Educational Background

• Core Competencies

• Certifications

• Project Highlights

• Accolades and Recognitions

• Professional Affiliations

• References

Presentation Skills

Proficiency in presenting information is crucial for conveying messages to an audience with clarity, confidence, and impact. The process begins with methodically organizing content, employing a well-structured framework. Visual aids, such as slides, should be used judiciously and purposefully. Rehearsing the presentation is essential for ensuring a smooth delivery and managing any anxiety. Engaging the audience can be achieved through storytelling, interactive components, and addressing their interests and needs. Additionally, maintaining eye contact, using gestures for emphasis, and varying vocal tones add to the effectiveness of the presentation. Lastly, readiness for questions and feedback demonstrates expertise and adaptability, ultimately enhancing the capacity to inform, persuade, and connect with the audience.

Resume Writing

Imagine you are transitioning to a new career path after gaining experience in various roles. You need to revamp your resume to emphasize adaptable skills and pertinent experiences. This revised resume underscores your versatility and willingness to tackle fresh challenges, even in different fields.

Self-Awareness

Self-awareness refers to the ability to introspect and comprehend one's own thoughts, emotions, and actions. It encompasses recognizing personal strengths, limitations, values, and beliefs. Self-aware individuals possess an enhanced understanding of theirmotivations and reactions, facilitating improved self-control and decision-making. Additionally, they display empathy and compassion toward others, nurturing stronger interpersonal bonds. Cultivating self-awareness typically involves introspection, mindfulness, and soliciting feedback from peers. It stands as a foundational element for personal development, emotional intelligence, and effective communication in both personal and professional contexts, enabling individuals to align their actions with their values and navigate life's challenges more adeptly.

SWOT Analysis

SWOT analysis serves as a strategic planning instrument employed by businesses and organizations to assess internal strengths and weaknesses, as well as external

opportunities and threats. The acronym signifies Strengths, Weaknesses, Opportunities, and Threats. Strengths and weaknesses pertain to internal aspects like resources, capabilities, and processes, whereas opportunities and threats are linked to external factors such as market dynamics, competition, and regulatory shifts. By identifying and scrutinizing these four components, organizations can make informed choices, leverage their strengths, address weaknesses, capitalize on opportunities, and mitigate threats. SWOT analysis proves invaluable for strategic planning, helping organizations chart a path towards growth and sustainability.

Public Speaking

Proficiency in public speaking entails effectively delivering a message or presentation to an audience with clarity, confidence, and influence. Accomplishing this involves conveying ideas, information, or viewpoints through well-structured, engaging narratives, employing vocal modulation and body language to convey messages effectively. Adequate preparation, practice, and understanding the audience's expectations are essential components. Public speaking opportunities can arise in diverse settings, spanning formal speeches, business presentations, educational lectures, or informal social gatherings. Mastering this skill empowers individuals to educate, inspire, and wieldinfluence over others, making it a valuable asset in both personal and professional spheres.

Time Management

Time management involves skilfully organizing and prioritizing tasks to maximize available time. It includes setting objectives, crafting schedules, and implementing strategies to minimize procrastination and distractions. Effective time management enhances productivity, reduces stress, and enhances work-life equilibrium. Essential principles comprise establishing clear goals, breaking tasks into manageable steps, and allocating time for both work and relaxation. Time management also encompasses the ability to decline tasks when necessary, delegate responsibilities, and periodically assess and adjust schedules. Mastery of time management empowers individuals to realize their objectives, meet deadlines, and lead more harmonious and fulfilling lives.

Emotional Intelligence

Emotional intelligence (EI) pertains to the capacity to identify, comprehend, manage, and skilfully employ emotions within oneself and in interpersonal relationships. It encompasses self-awareness, recognizing one's own emotions and their impact, and self- regulation, which entails controlling emotional reactions. Additionally, EI comprises empathy, the ability to grasp and respond to the emotions of others, and adept social interactions. Elevated emotional intelligence fosters improved communication, conflict resolution, and leadership skills, rendering it indispensable for personal and professional success. Cultivating EI involves introspection, active listening, and exercises to build empathy, ultimately enhancing emotional awareness and nurturing healthier, more meaningful connections with others.

Casual Conversation

Casual conversation denotes unstructured, relaxed dialogues between individuals, typically revolving around friendly exchanges regarding everyday subjects such as hobbies, interests, experiences, and current events. These interactions aim to cultivate connections, exchange information, and foster a sense of camaraderie. Casual conversations often encompass small talk, serving as a conversational icebreaker or a method to fill conversational lulls. They rely on active listening, empathy, and reciprocity, enabling participants to savour social interactions, strengthen bonds, and gain insights into each other's lives. While less structured than formal discussions, casual conversations play a vital role in building social ties and preserving a sense of community.

Goal Setting

Goal setting encompasses the process of defining specific, attainable objectives that individuals or organizations aspire to achieve within a predefined timeframe. This involves pinpointing clear, measurable targets and outlining the necessary steps to attain them. Effective goal setting bolsters motivation, focus, and productivity by furnishing a sense of purpose and direction. SMART criteria, emphasizing Specific, Measurable,Achievable, Relevant, and Time-bound goals, often serve as a guideline. Regularly reviewing and adapting goals as circumstances evolve proves pivotal for success. Be it in personal or professional spheres, goal setting empowers individuals and teams to pursue growth, improvement, and the realization of their aspirations.

CHAPTER 3

BASICS OF JAVA PROGRAMMING

Introduction to Java

Java is a widely-used, versatile, and platform-independent programming language. Developed by Sun Microsystems (now owned by Oracle) in 1995, Java is celebrated for its portability, security, and versatility. It follows an object-oriented programming paradigm, organizing code into objects with attributes and behaviours, simplifying complex software development. Java applications are compiled into bytecode, which can run on any system with a Java Virtual Machine (JVM). This "Write Once, Run Anywhere" capability makes Java ideal for developing cross-platform applications, from web and mobile apps to large-scale enterprise software.

Data Types and Variables

In Java, data types and variables are fundamental concepts that allow you to work with data in your programs. Here's an overview:

Data Types: Java has two main categories of data types:

1. Primitive Data Types:

These are basic data types that represent single values. There are eight primitive data types in Java:

- byte: 8-bit signed integer.
- short: 16-bit signed integer.
- int: 32-bit signed integer.
- long: 64-bit signed integer.
- float: 32-bit floating-point number.
- double: 64-bit floating-point number.
- char: 16-bit Unicode character.
- boolean : Represents true or false values

2. Reference Data Types:

These data types are used to refer to objects (instances of classes) and do not hold the actual data. Examples include:

- Classes
- Interfaces
- Arrays

Variables in Java:

Variables are used to store data, such as numbers, text, or objects, in Java programs. Before using a variable, you must declare it by specifying its name and data type. 16

The data type determines the kind of data the variable can hold (e.g., int for integers, double for floating-point numbers, String for text).

Properly declaring and initializing variables is a fundamental practice in Java programming, ensuring data is stored and used correctly in your code.

Control Flow Statements

Java provides control flow statements like if, else, while, for, switch, and do-while for decision-making and looping.

Decision-Making

Java offers conditional statements like if, else, and else if for making decisions in your code. These Statements allow you to execute specific blocks of code based on conditions.

Loops

Loops in Java, including while, for, and do-while, enable repetitive execution of code. Loops are used when you want to perform a taskmultiple times, such as iterating through arrays or processing data.

Switch Statement

The switch statement provides an efficient way to select one of many code blocks to be executed based on the value of a variable or expression. It's often used for multiple-choice scenarios.

Control Flow Structures

These control flow structures allow you to control the flow of your program, ensuring that specific parts of your code execute under certain conditions. For example, you can use if to execute code when a conditionis true, or while to repeat code as long as a condition is true

Branching

Control flow statements help your program branch into different paths depending on conditions. For example, you can use if and else to provide

alternate paths of execution.

Iteration

Loops like for and while are used for iteration, where a block of code is executed repeatedly until a specific condition is met.

Termination Conditions

In looping constructs like while and for, you must define termination conditions to avoid infinite loops. An infinite loop can cause your program to hang or crash.

Nested Control Flow

Java allows nesting of control flow statements. For example, you can place an if statement within a while loop to create complex decision-making structures.

Functions

In Java, functions are referred to as "methods." Methods are blocks of code within classes that perform specific tasks or actions. They are essential for structuring and organizing code. A method in Java typically has a name, a return type (which can be void if the method doesn't return a value), and parameters that may or may not be passed when calling the method. Methods encapsulate functionality, promoting code reuse and modularity. They are invoked using the method name, followed by parentheses containing any required arguments. Java's object-oriented nature also allows methods to be associated with objects, enabling the use of instance methods that operate on specific instances of classes.

Object Oriented Programming Concepts

Object-Oriented Programming (OOP) is a fundamental programming paradigm in Java that revolves around the concept of "objects." Here are the key principles of OOP in Java:

Classes and Objects: Java programs are built using classes, which are blueprints for creating objects. Objects are instances of classes and represent real-world entities, each with attributes (data) and methods (functions).

Encapsulation: Encapsulation is the concept of bundling data (attributes) and methods (functions) that operate on that data into a single unit called a class. Access to the data is controlled through methods, allowing for data security and abstraction.

Inheritance: Inheritance is a mechanism that allows one class (subclass or derived class) to inherit the properties and behaviours of another class (superclass or base class). It promotes code reuse and establishes a hierarchical relationship between classes.

Polymorphism: Polymorphism allows objects of different classes to be treated as objects of a common superclass. It enables dynamic method dispatch, where the appropriate method is called at runtime based on the actual object type, facilitating flexibility and extensibility.

Abstraction: Abstraction is the process of simplifying complex systems by modelling classes and objects at the appropriate level of detail while hiding unnecessary implementation details. It helps in managing program complexity and promoting a high-level view of the system.

Arrays

In Java, arrays are data structures that store collections of values of the same data type under a single variable name. They provide an efficient way to work with groups of data elements. Arrays are declared by specifying the data type of their elements followed by square brackets, and they can be initialized with specific values or a predefined size. Accessing elements in an array is done by their index, with the first element at index 0. Java arrays are fixed in size once created, meaning you need to know the number of elements you'll store in advance. To work with dynamic collections, Java offers other datastructures like Array Lists.

Strings

In Java, strings are sequences of characters that represent text. They are a fundamental data type and are treated as objects in the Java language. Strings in Java are immutable, meaning their values cannot be changed once they are created. You can perform various operations on strings, such as concatenation, substring extraction, and searching forspecific characters or substrings. Java provides a rich set of methods through the String class for manipulating strings, making it versatile for text processing tasks. Additionally, Java allows for easy comparison of strings using the „equals()“ method. String literals in Java are typically enclosed in double quotes, making them straightforward to work within code.

CHAPTER

BASICS OF ANDROID DEVELOPMENT

Introduction to Android

Android is an open-source mobile operating system developed by Google. It powers billions of smartphones, tablets, and other devices globally. Android offers a customizable, user-friendly interface, robust app ecosystem through Google Play, and extensive developer tools. It's based on the Linux kernel and supports a wide range of applications, from social media and games to productivity and utilities. Android Studio, the official Integrated Development Environment (IDE), simplifies app creation. Android's versatility, regular updates, and strong community support make it a dominant force in the mobile industry, enabling developers to create innovative and diverse applications for a broad audience.

Layouts

In Android app development, layouts are a crucial part of the user interface (UI) design. They define the structure and arrangement of UI elements, such as buttons, text fields, images, and more, within an app's user interface. Layouts help ensure that your app's UI is visually appealing, consistent, and responsive on various screen sizes and orientations. Here are some common layout types used in Android:

Linear Layout, Relative Layout, Constraint Layout, Frame Layout, Grid Layout, Table Layout, Constraint Set, Scroll View and other Custom ones.

UI Components

UI components, or widgets, are the building blocks of the user interface in Android app development. They allow you to create interactive and visually appealing interfaces for your Android applications. Here are some common UI components in Android:

Text View, Edit Text, Button, Image view, Radio Button, Check Box, Toggle Button, Switch, Spinner, Progress Bar, Seek bar and many more.

CHAPTER 5

TASK PERFORMED

Soft Skills Developed

Communication: Internships offer chances to interact with colleagues, superiors, and clients, improving verbal and written communication skills. Clear and effective communication is vital in any workplace.

Teamwork: Collaborating with diverse teams on projects helps build teamwork and cooperation skills. Learning to contribute your strengths and respect others' ideas fosters a harmonious work environment.

Problem-Solving: Facing real-world challenges during internships hones problem- solving abilities. You learn to analyse situations, generate solutions, and adapt to changing circumstances.

Time Management: Balancing tasks, meeting deadlines, and juggling responsibilities enhance time management and organizational skills, valuable for productivity.

Adaptability: Internships expose you to new environments and tasks, teachingadaptability and the ability to thrive in varied settings.

Initiative: Demonstrating a proactive attitude, taking on additional responsibilities, and seeking opportunities to contribute positively reflect well on your initiative.

Networking: Building professional relationships with colleagues, mentors, and industry contacts can help you throughout your career.

Leadership: Assuming leadership roles in projects or teams during internships cultivates leadership skills, including decision-making and motivating others.

Conflict Resolution: Exposure to workplace conflicts provides opportunities to develop conflict resolution skills, crucial for maintaining a harmonious work atmosphere.

Critical Thinking: Internships encourage critical thinking by requiring you to analyse situations, gather information, and make informed decisions.

Customer Service: If applicable, internships in customer-facing roles enhance customer service and interpersonal skills.

Self-Confidence: Successfully completing tasks and achieving goals during internships boosts self-confidence and self-esteem.

Cultural Awareness: Working with diverse teams or in international settings fosters cultural awareness and sensitivity.

Professionalism: Internships teach professionalism, including punctuality, dressing appropriately, and conducting oneself in a business-like manner.

Emotional Intelligence: Interactions with colleagues and clients during internships improve emotional intelligence, enhancing empathy and interpersonal relationships

Java Programming-What I Learnt

Learning Java programming involves mastering its syntax, concepts, and best practices. Key topics include variables, data types, control structures (if, loops), functions (methods), classes, objects, inheritance, polymorphism, and exception handling. Understanding the Java Standard Library is crucial for practical development. You'll also learn about memory management, garbage collection, and multithreading for concurrent programming. Additionally, frameworks like JavaFX for desktop apps and Android for mobile apps expand your skills. Continuous practice, debugging, and problem-solving areessential for proficiency. Collaborating with a developer community and staying updated with Java's evolving features and libraries contribute to becoming a proficient Java programmer.

Here are the things I have learnt in Java:

Introduction to Java:

Java is a high-level, object-oriented programming language known for its portability and wide range of applications. It's used in web development, mobile app development, and more.

Variables and Data Types:

I've learned how to declare variables to store data and the different data types available in Java, including integers, floating-point numbers, characters, and strings.

Operators:

I've explored various operators, such as arithmetic, comparison, and logical operators, to perform calculations and make decisions in your programs.

Control Structures:

Conditional Statements: I've learned how to use if, else if, and else statements to make decisions based on conditions.

Loops: I've studied for, while, and do-while loops to execute code repeatedly, making the programs more efficient.

Object-Oriented Programming (OOP):

Classes and Objects: I've delved into OOP by creating classes to define blueprints for objects and then creating instances (objects) from those classes.

Methods: I've learned how to define methods (functions) within classes to encapsulate behaviour.

Inheritance and Polymorphism: I might have touched upon concepts like inheritance (creating subclasses from super classes) and polymorphism (the ability of objects to take on different forms).

Functions and Methods: I've gained the ability to write reusable code by defining your own functions and methods, allowing you to break down complex tasks into manageable pieces.

Exception Handling: I've explored error handling techniques like try, catch, throw, and finally to gracefully manage exceptions and errors in your programs.

Standard Library: I've been introduced to Java's extensive standard library, which provides pre-built classes and packages for various tasks, from file input/output to networking.

IDE and Compilation: You've likely used an Integrated Development Environment(IDE) to write and manage your Java code, and you've learned how to compile your code to produce executable Java applications.

Basic Input and Output: You've worked with input and output streams to interact with users and external data sources, making your programs interactive and dynamic.

Debugging: You've acquired skills to identify and fix errors (bugs) in your code, an essential part of the development process.

Many examples were done based on the above learnt concepts.

Android Development

Learning Android development involves mastering the fundamentals of Java or Kotlin programming, understanding XML layouts, and grasping key Android components like activities, fragments, and services. You must become proficient in using Android Studio, the official IDE, for designing interfaces and writing code. Knowledge of the Android API, including UI elements, permissions, and intents, is crucial. Learning how to manage data with SQLite, RESTful APIs, or Firebase is essential. Understanding app lifecycle, handling UI responsiveness, and debugging skills are vital. Continuous learning, staying updated with Android versions, and practicing app development are key to becoming a proficient Android developer.

After learning about the Basics of Android, An App called “lognpg” app was created. The App looks as follows:

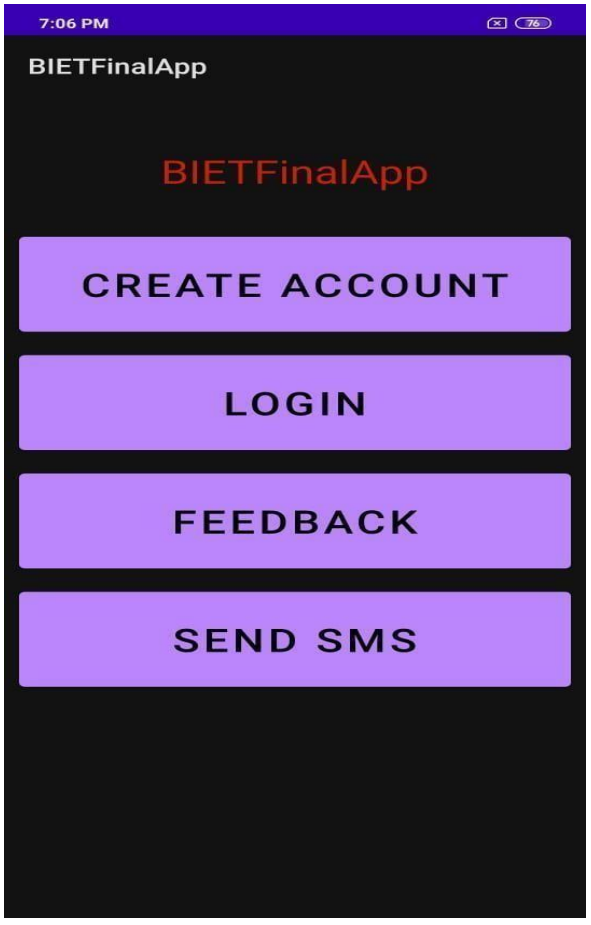

Fig. 4.1 Home Page

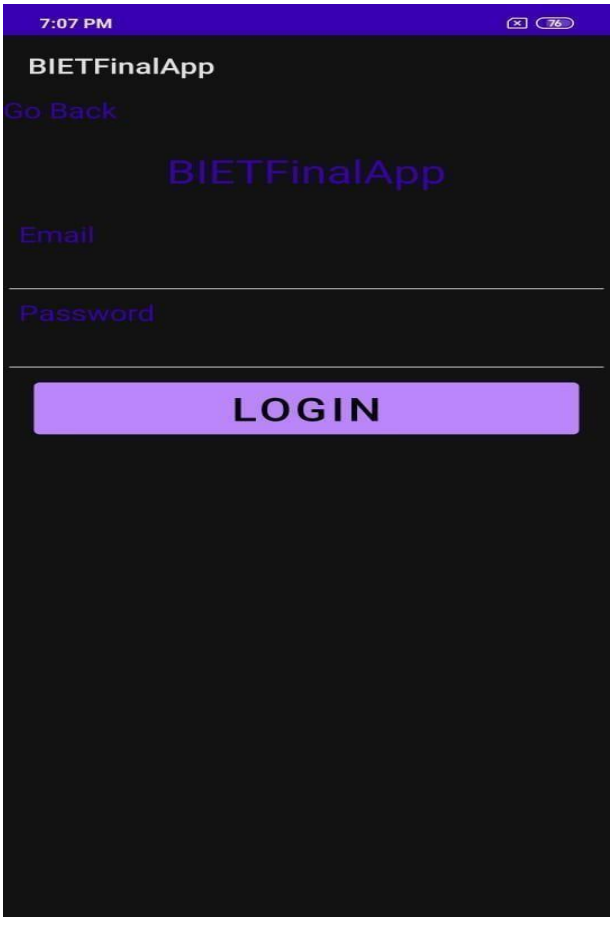

Fig. 4.2 Login Page

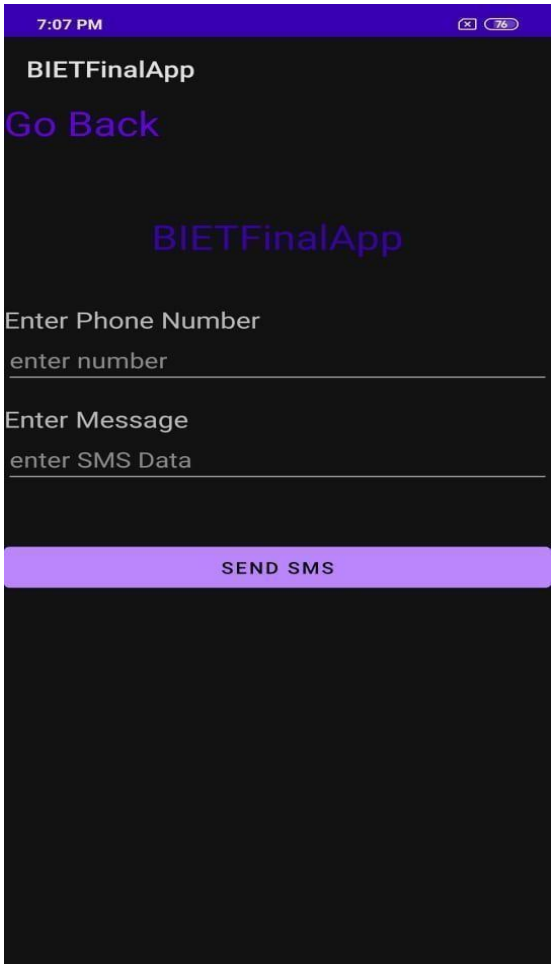

Fig. 4.3 SMS Sender

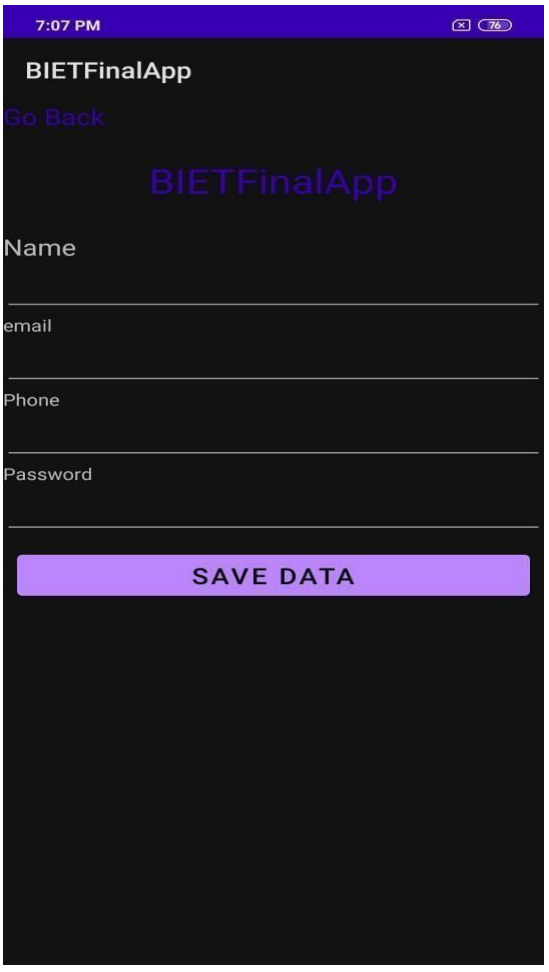

Fig. 4.4 Registration Page

REFLECTION NOTES

Enhanced Communication Skills: The internship experience significantly improved my communication skills. I learned the importance of clear and concise communication, whether it was in written reports, emails, or in-person discussions with colleagues and supervisors.

Teamwork and Collaboration: Working on various projects with diverse teams taught me the value of teamwork and collaboration. I realized that different perspectives can leadto innovative solutions, and effective collaboration is crucial for project success.

Time Management and Prioritization: Balancing multiple tasks and deadlines during the internship helped me develop strong time management and prioritization skills.

Adaptability: The dynamic nature of the work environment required me to adapt quickly to changing circumstances and priorities. This experience improved my adaptability and problem-solving abilities.

Java Fundamentals: My exposure to Basic Java was a great starting point for my programming journey. I grasped key concepts like variables, data types, and control structures, which laid a solid foundation for more advanced programming.

Debugging and Troubleshooting: I gained confidence in debugging and troubleshooting Java code, which is an essential skill for a developer.

Android App Development: Exploring Android development was a fascinating experience. Designing user interfaces, understanding the Activity lifecycle, and working with Android Studio allowed me to create functional Android applications.

Technical Growth: Throughout the internship, I witnessed substantial growth in my technical abilities. I transitioned from a novice to a more confident and capable developer, and I now have a better understanding of how software development projects unfold.

Overall Impact: This internship not only helped me acquire technical skills but also shaped my professional demeanour. It instilled in me the importance of continuouslearning, adaptability, and effective communication, which are essential qualities in any career.

CONCLUSION

My internship experience, which encompassed soft skills, Basic Java, and Android development, was a transformative journey. It highlighted the significance of effective communication, emphasizing the importance of clear and concise expression in written and verbal interactions. Working within diverse teams on various projects enhanced my teamwork and collaboration skills, teaching me the value of diverse perspectives and innovative problem-solving. In the realm of technical skills, I acquired a foundational understanding of Basic Java, which laid the groundwork for my programming knowledge. Additionally, delving into Android app development equipped me with the ability to design user interfaces and develop functional applications. This hands-on experienceallowed me to confront real-world challenges, further solidifying my problem-solving abilities. Overall, this internship not only fostered my technical growth but also sharpenedmy soft skills, making me a more adaptable, effective, and well-rounded professional**.**

REFERENCES

https://ieeexplore.ieee.org/document/9465935

https://ieeexplore.ieee.org/document/7003608

https://ieeexplore.ieee.org/document/7503306

https://developer.android.com/studio

https://stackoverflow.com/questions/tagged/android

https://docs.oracle.com/en/java/