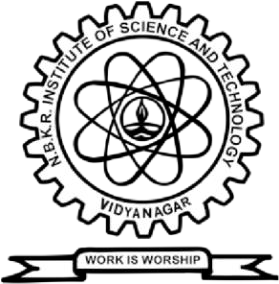
# FINAL SEMESTER INTERNSHIP REPORT



**IV B. Tech I Semester (2024-25)**

*A report submitted in partial fulfillment of the requirements for the Award of Degree of* **Bachelor of Technology in Computer Science and Engineering**

# by

|  |  |  |
| --- | --- | --- |
| Name of the Student | : | N. Siva Gayathri |
| Roll Number | : | 21KB1A05A5 |
| Organization | : | Slash Mark IT Start up |
| HR | : | P. Abhishek |

**M. B.K.R. INSTITUTE OF SCIENCE & TECHNOLOGY**

**(AUTONOMOUS)**

**Affiliated to JNTU ANANTHAPURAMU Accredited by NAAC with 'A' Grade Courses Accredited by NBA under TIER - I Vidyanagar - 524 413, Tirupathi Dist., A.P., 2024 – 2025**

# DECLARATION BY STUDENT

I N. Siva Gayathri, a student of N.B.K.R Institute of Science & Technology, Vidyanagar studying IV B.Tech., I semester, Computer Science and Engineering, bearing Roll Number 21KB1A05A5 hereby declare that I have completed the final semester internship from 30- 04-2024 to 31-08-2024.The report of the internship is prepared by me and submitted to the department.

Date: (Signature of Student)

# CERTIFICATE OF INTERNSHIP



**ACKNOWLEDGEMENT**

I would like to thank **Mr. P. Abhishek**, HR, Slash Mark IT Start-up, for giving me the opportunity to do an internship with the organization.

It is indeed with a great sense of pleasure and immense sense of gratitude that I acknowledge the help of the organization.

I am highly indebted to **Dr. V. Vijaya Kumar Reddy,** Director, N.B.K.R. Institute of Science andTechnology, for the facilities provided to accomplish this internship.

I would like to thank **Dr. A. Rajasekhar Reddy** Head of the Department, Computer Science and Engineering**,** N.B.K.R. Institute of Science and Technology, for his constructive criticism throughout my internship.

I would like to thank **Mr. P. Mutyalaiah** internship coordinator Department of Computer Science and Engineering, N.B.K.R. Institute of Science and Technology, for his support and advice to get and complete the internship in the above said organization.

I am extremely grateful to my department staff members and friends who helped me in successful completion of this internship.

**N.SIVA GAYATHRI (21KB1A05A5)**

# ABSTRACT

The Java-focused internship provided a well-rounded and structured journey through essential and advanced concepts of Java programming and object-oriented principles, carefully designed to build foundational skills before introducing more complex functionalities. Spanning 16 weeks, the program began with an introduction to Java syntax and setup, including the JDK, IDE configuration, and basic programming constructs such as variables, data types, and operators. Early weeks focused on the basics of Java programming and core object-oriented principles like encapsulation, inheritance, polymorphism, and abstraction.

As the internship progressed, the focus shifted to specific Java tasks and techniques for structuring programs effectively. We explored key OOP concepts further, delving into class hierarchies, method overloading and overriding, and the use of inheritance to create well-structured, reusable code. Practical examples were provided to demonstrate real-world applications of interfaces, abstract classes, and exception handling to manage errors gracefully. Throughout these exercises, we built hands-on experience, reinforcing an understanding of Java’s unique approach to modular and scalable coding.

In the later stages, the internship introduced advanced Java topics such as file handling, the Collections framework, JDBC for database connectivity, and multithreading for concurrent execution. We explored functional programming features in Java 8, such as lambdas and streams, which allowed for more efficient data processing and manipulation. Practical exercises included reading and writing files, working with various data structures, and establishing database connections to execute SQL commands. The program culminated in building a small project that integrated all the skills learned, providing real-world simulation through tasks like implementing multithreading, working with data, and using JDBC for database interactions. By the end, we gained a comprehensive, application-oriented understanding of Java, ready for real-world scenarios.

# COMPANY PROFILE

Slash Mark IT Startup is a fintech company that provides payment gateway integration, financial services, and KYC assistance. They partner with Razorpay to help businesses streamline transactions.

**Fintech**

Slash Mark offers solutions to help businesses streamline transactions and unlock their potential. They partner with Razorpay to provide payment gateway integration and other products.

**Internships**

Slash Mark offers unpaid internships that provide hands-on experience on projects. Interns can work on minor and major projects, and gain practical insights and build a portfolio.

**Culture**

Some say that Slash Mark has a positive hiring process, with interviewers who explain the company's culture and values in detail. They also say that the interviewers focus on the candidate's strengths and how they could fit into the team.

**ACHIEVEMENTS:**

From software development and digital transformation to consulting and support, slash mark empower organizations to thrive in the digital era and achieve their strategic objectives.

* 38 Projects Completed
* 67 Happy Clients
* 2 Business Partners
* 116 IT Consultants

# WEEKLY OVERVIEW OF INTERNSHIP ACTIVITIES

## Week 1: Introduction to Java Programming

**Focus**: Introduce the fundamental concepts of programming in Java.

## Activities:

* Overview of Java
* Setting up the Java development environment (JDK, IDE)
* Basic syntax: variables, data types, and operators

## Week 2: Understanding Object-Oriented Programming (OOP)

**Focus**: Learn the basics of Object-Oriented Programming, the core of Java.

## Activities:

* Introduction to OOP principles: Encapsulation, Inheritance, Polymorphism, and Abstraction
* Understanding Classes and Objects
* Creating and using methods

## Week 3: Deep Dive into Classes and Objects

**Focus**: Expand understanding of classes, methods, and constructors.

## Activities:

* Constructors and Overloading
* this keyword and method overloading
* Introduction to static keyword and access modifiers

## Week 4: Inheritance in Java

**Focus**: Learn how inheritance works in Java and how to use it to build more complex structures.

## Activities:

* Types of inheritance: Single, Multilevel, Hierarchical
* Use of super keyword
* Method overriding and understanding polymorphism

## Week 5: Abstract Classes and Interfaces

**Focus**: Explore abstract classes and interfaces for advanced OOP techniques.

## Activities:

* Differences between abstract classes and interfaces
* Implementing multiple interfaces
* Using abstract classes in applications

## Week 6: Exception Handling

**Focus**: Understand how to handle errors in Java applications.

## Activities:

* Introduction to exception handling
* Try-catch-finally blocks, throw and throws keywords
* Creating custom exceptions

**Week 7: Understanding Packages and Access Modifiers Focus**: Learn to organize code using packages and control access. **Activities**:

* Java package structure and how to create packages
* Using public, protected, private, and package-private access modifiers

## Week 8: Working with Java Collections

**Focus**: Get familiar with Java Collections framework for handling data.

## Activities:

* Overview of collections: List, Set, Map
* ArrayList, LinkedList, HashSet, HashMap examples
* Iterator and for-each loop

## Week 9: Working with File Handling

**Focus**: Learn how to handle files and perform basic I/O operations.

## Activities:

* Reading from and writing to files
* BufferedReader and BufferedWriter
* File operations: create, delete, rename

## Week 10: Basic JDBC and Database Connection

**Focus**: Understand how to connect Java applications with databases.

## Activities:

* Introduction to JDBC
* Setting up a database connection
* Executing basic SQL commands from Java

## Week 11: Advanced JDBC Operations

**Focus**: Learn advanced database operations in Java.

## Activities:

* Prepared Statements and Callable Statements
* Handling transactions
* Batch processing and stored procedures

## Week 12: Introduction to Multithreading

**Focus**: Understand the basics of multithreading and concurrency.

## Activities:

* Creating and starting threads
* Thread lifecycle and Thread methods
* Synchronization basics

## Week 13: Advanced Multithreading

**Focus**: Explore more advanced topics in multithreading.

## Activities:

* Inter-thread communication
* Thread synchronization with synchronized block and methods
* Deadlock, wait(), notify(), and notifyAll() methods

## Week 14: Working with Lambdas and Streams (Java 8)

**Focus**: Learn functional programming features introduced in Java 8.

## Activities:

* Understanding lambdas
* Working with Java Streams for data processing
* Using functional interfaces: Predicate, Function, Consumer, etc.

## Week 15: Building a Mini Java Project

**Focus**: Apply the skills learned so far by building a small project.

## Activities:

* Define a project scope and requirements
* Implementing classes and methods
* Integrating file handling, JDBC, and multithreading concepts

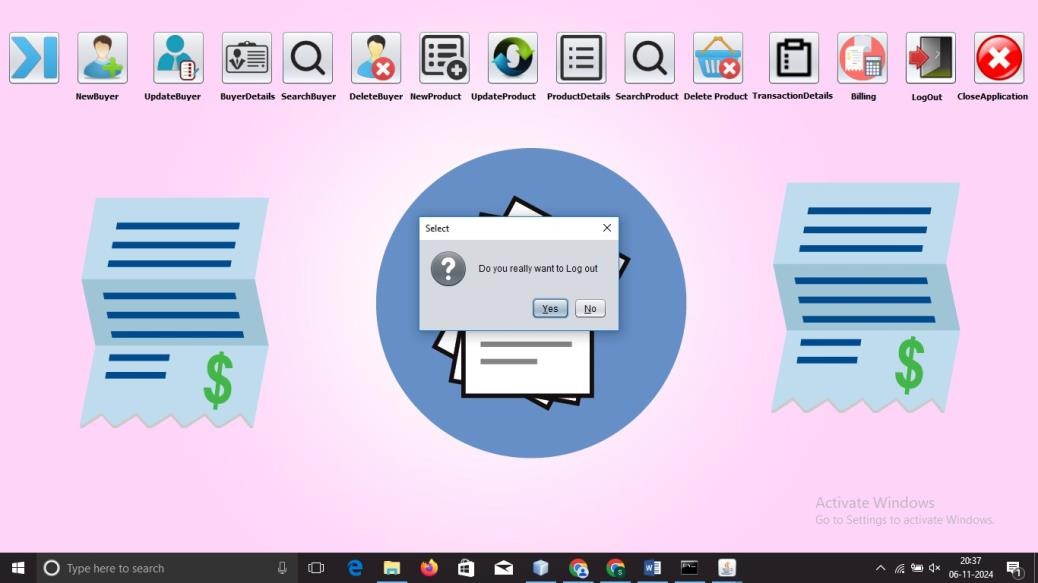
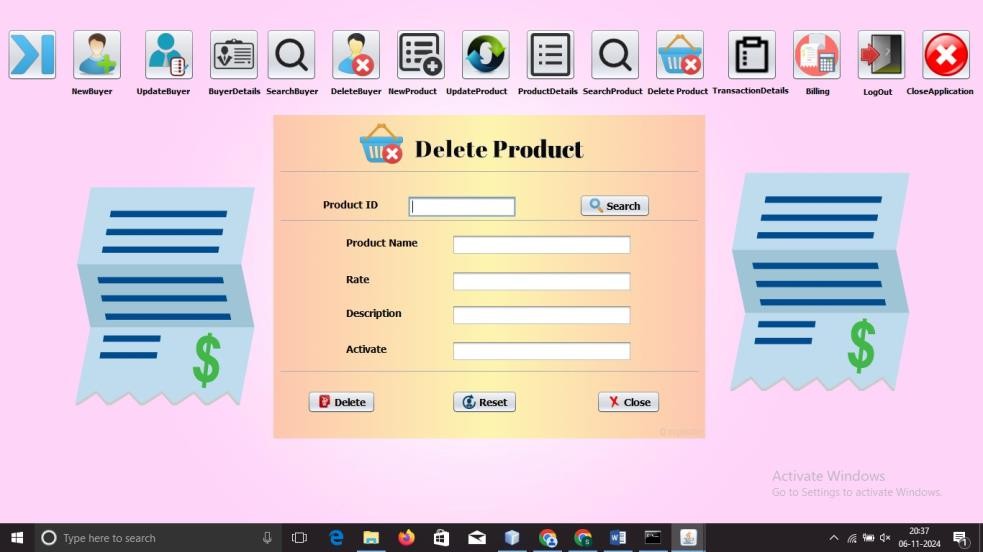
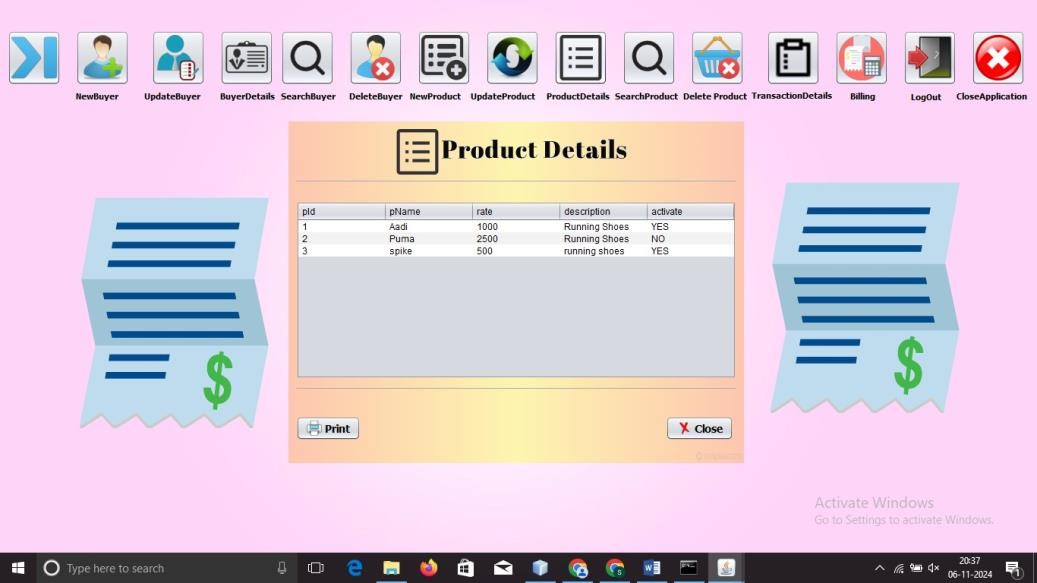
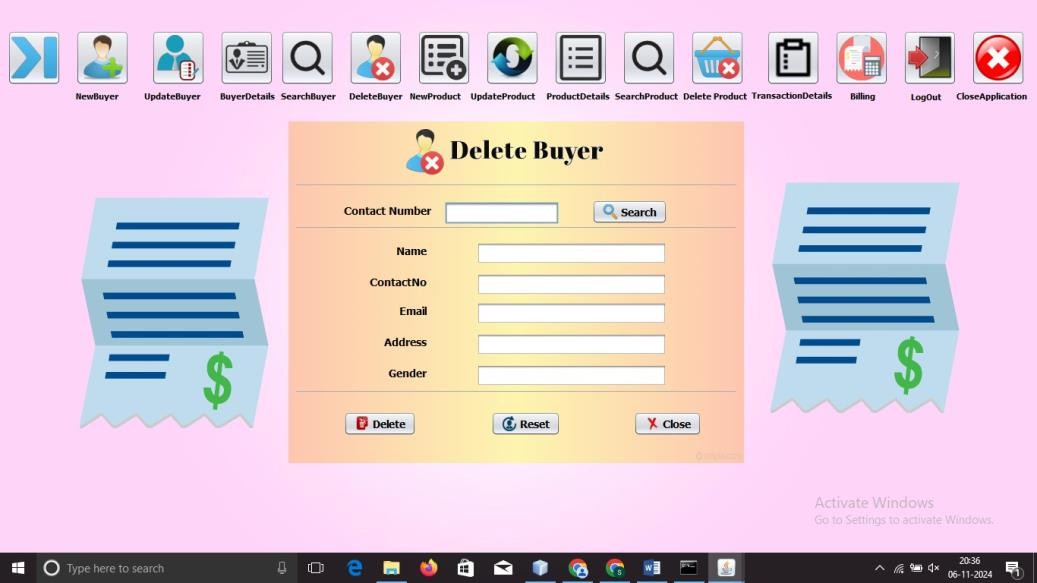
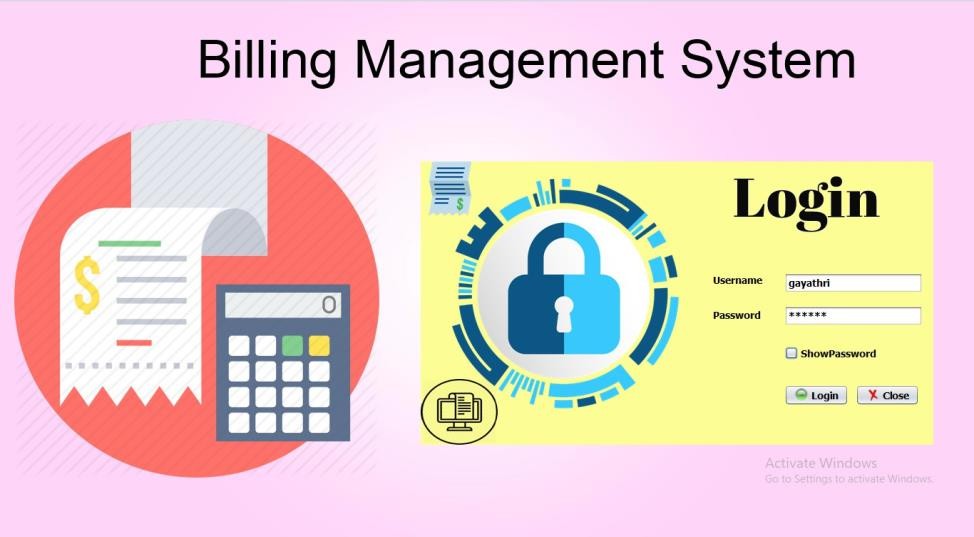
## Week 16: Final Project Review and Best Practices

**Focus**: Review the project, focus on best coding practices, and finalize the internship.

## Activities:

* Project testing and debugging
* Code optimization techniques
* Final project presentation and review

# OUTCOME



**CONCLUSION**

The Java internship equipped me with a strong foundation in Java programming and object-oriented principles, moving from fundamental concepts to advanced features. Beginning with Java syntax, I quickly learned to write and structure code, focusing on best practices in object-oriented design with classes, methods, and key OOP principles like encapsulation and inheritance. As I progressed, I gained confidence in handling real-world tasks such as file I/O operations, managing data with collections, and connecting applications to databases using JDBC.

Through hands-on exercises, I developed practical skills in error handling, multithreading, and advanced Java features like lambdas and streams, which enabled efficient data processing. Each week, I built on prior knowledge, reinforcing my understanding of Java’s potential in developing scalable, maintainable applications. The final project solidified my skills by integrating core concepts into a cohesive application, providing a glimpse into real-world application development.

Overall, this experience has given me a solid base for future roles involving Java development. I now feel prepared to tackle various programming challenges with confidence and precision, equipped with practical skills that are valuable in both small projects and larger, more complex applications.