Industrial Internship Report on

”BANKING INFORMATION SYSTEM”

Prepared by

TrishulNath

Pandey

Executive Summary

This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT).

This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time.

My project was Banking Information System which includes functionalities of a real banking systemt: User registration,account management,deposit and withdrawl,fund transfer,account statement.

1. **User Registration:**

- Form Creation: Create a user registration form that prompts users to input their personal details, such as name, address, contact information, and initial deposit amount.
- Output: Upon successful registration, the system will generate a unique account number for the user, and the user's details will be stored in the system's memory or File System.

2. **Account Management:**

- Form Creation: Develop an account management form that allows users to view and update their account information, such as name, address, contact details, and account settings.
- Output: After making any updates or changes, the system will display a confirmation message indicating that the account information has been successfully updated.

This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1. Preface

1. Project/Problem Statement:

- Explored the need for relevant internships in shaping robust career paths.
- Focused on the challenges individuals face in securing meaningful internships for career development.

2. Opportunity given by USC/UCT:

- Discussed the valuable opportunities provided by USC/UCT in the context of career development.
- Highlighted the program's commitment to hands-on experience and industry relevance.

3. Program Planning:

- Outlined the meticulous planning of the "Banking Information System with Java" project.
- Emphasized the alignment of the program with industry demands and career growth objectives.

4. Learnings and Overall Experience:

- Explored key learnings from the project, including technical skills in Java programming and system design.
- Reflected on the overall experience, highlighting challenges overcome and new skills acquired.

5. Acknowledgments:

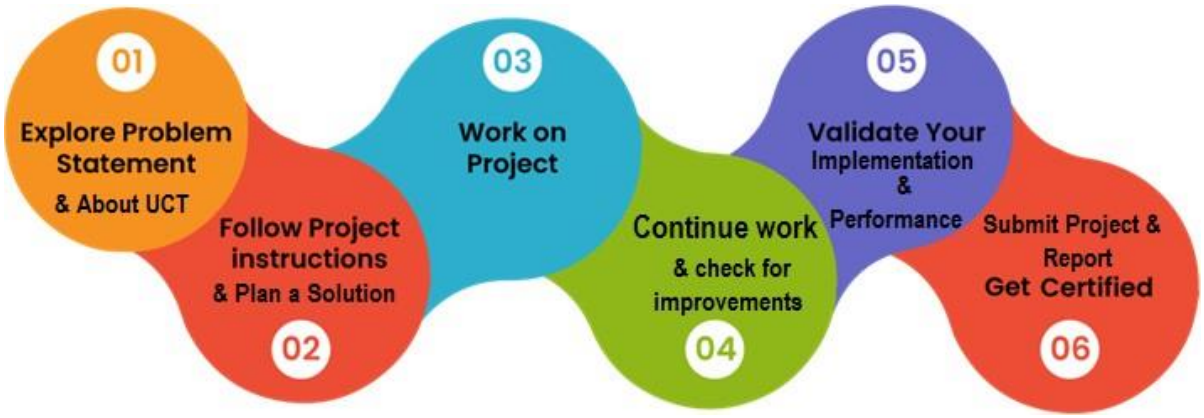
- Extended gratitude to all who directly or indirectly contributed to the project's success.
- Named individuals and expressed appreciation for their support, guidance, and collaboration.

6. Message to Juniors and Peers:

- Encouraged juniors and peers to embrace internship opportunities for practical skill development.
- Shared insights gained from the project and encouraged continuous learning and collaboration.

7. Closing Note:

- It concluded with a message of gratitude to the entire team and anticipation for future collaborations.
- Encouraged a culture of mentorship and support within the academic and professional community.


01
Explore Problem Statement
& About UCT
02
Follow Project instructions
& Plan a Solution
03
Work on Project
04
Continue work
& check for improvements
05
Validate Your Implementation
&
Performance
06
Submit Project &
Report
Get Certified

2. Introduction

2.1. About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in the Digital Transformation domain and providing Industrial solutions with a prime focus on sustainability and ROI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

i. UCT IoT Platform (UCT Insight)

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

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA

- It supports both cloud and on-premises deployments.

It has features to

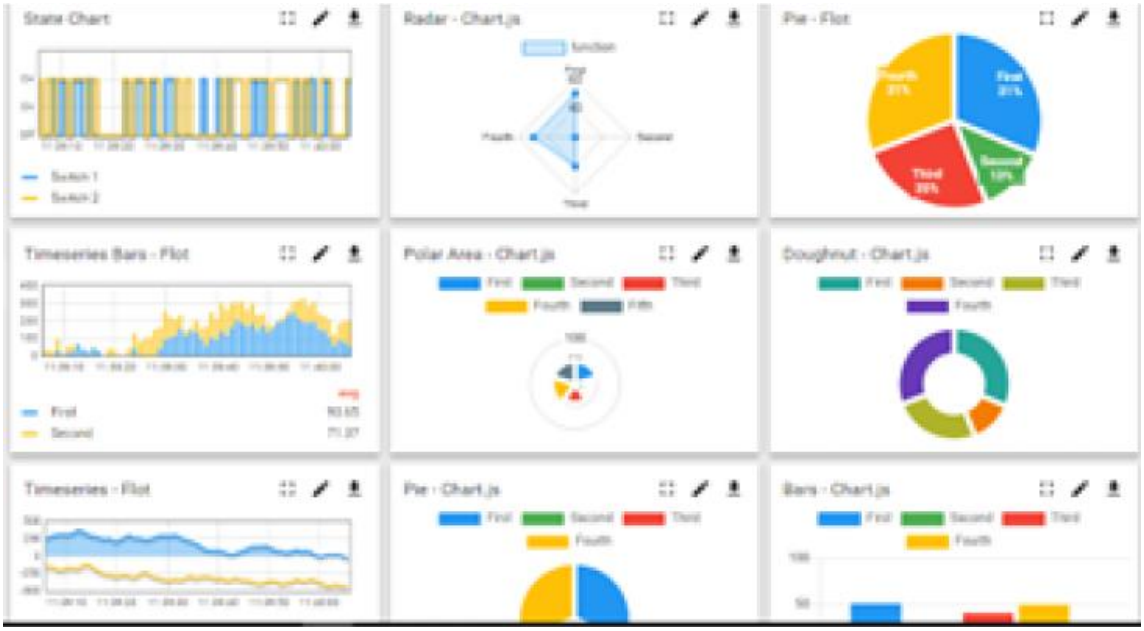
• Build Your dashboard

• Analytics and Reporting

• Alert and Notification

• Integration with third-party applications (Power BI, SAP, ERP)

• Rule Engine



ii. Smart Factory Platform (FACTORY WATCH)

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.

- To unleash the true potential of the data that their machines are generating and help to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users save time, cost and money.

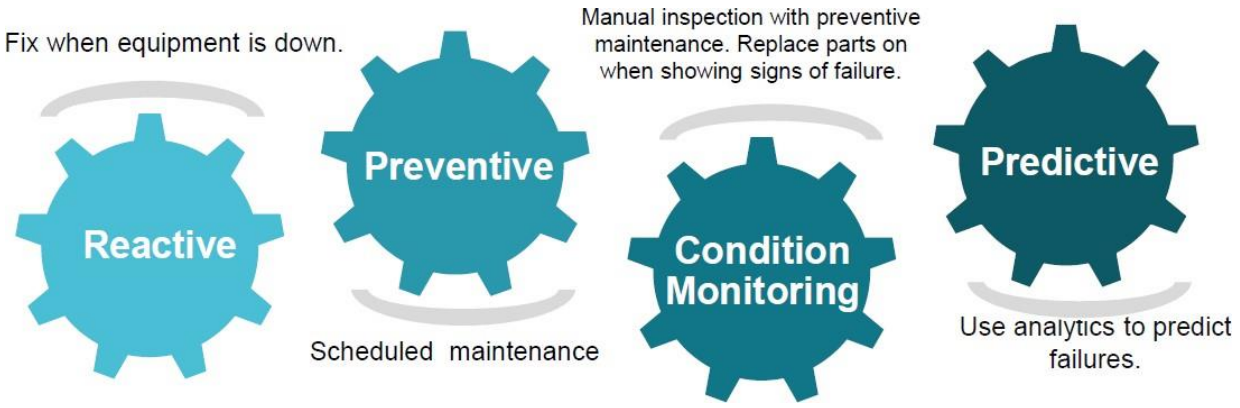
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1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

iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN technology and provides solutions in agriarch, Smart Cities, Industrial Monitoring, Smart Street lights, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

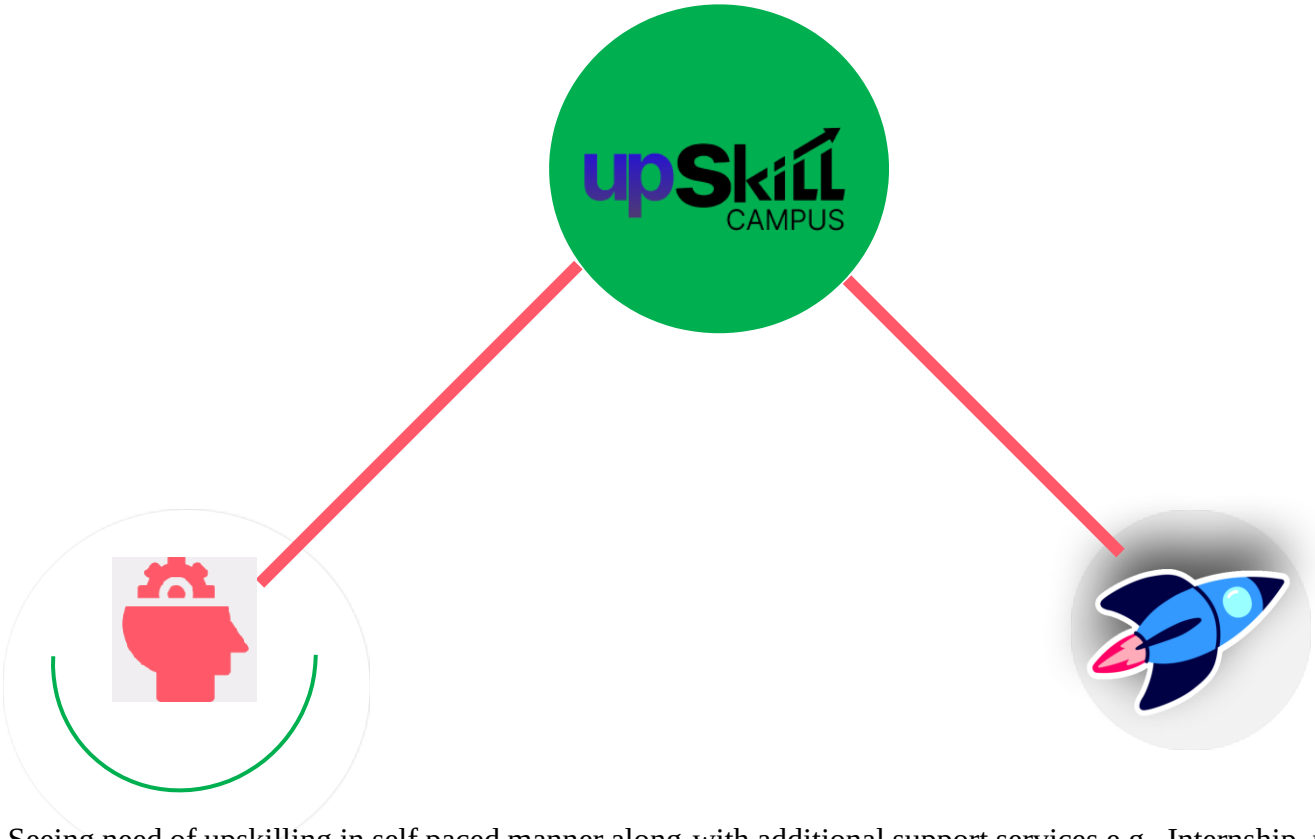
UCT is providing Industrial Machine health monitoring and Predictive maintenance solutions leveraging Embedded systems, Industrial IoT and Machine Learning Technologies by finding the Remaining useful lifetime of various Machines used in the production process.



2.2. About upskill Campus (USC)

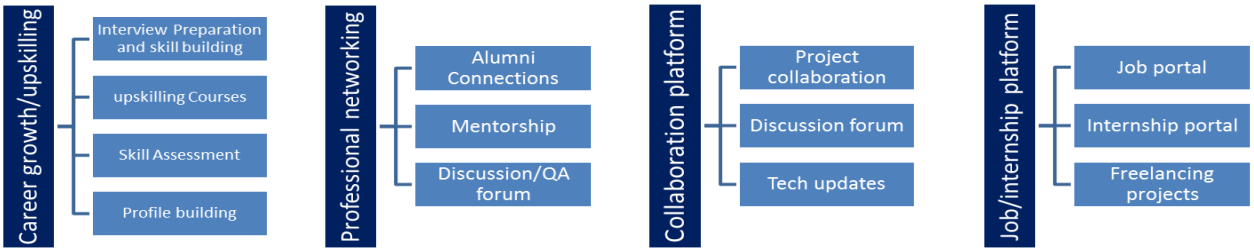
upskill Campus along with The IoT Academy and in association with Uniconverge Technologies has facilitated the smooth execution of the complete internship process.

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



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth ServicesupSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



2.3. The IoT Academy

The IoT academy is the EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

2.4. Objectives of this Internship program

The objective of this internship program was to

- ☛ get practical experience working in the industry.
- ☛ to solve real-world problems.
- ☛ to have improved job prospects.
- ☛ to have an Improved understanding of our field and its applications.
- ☛ to have Personal growth like better communication and problem-solving.

2.5. Reference

1.VS code - Official website of VS IDE. Available at: https://code.visualstudio.com/download

2.GitHub - Official website of GitHub(Code Submission). Available at: https://github.com/

2.6. Glossary

Terms	Acronym
IDE	Integrated Development Environment
VS	Visual Studio

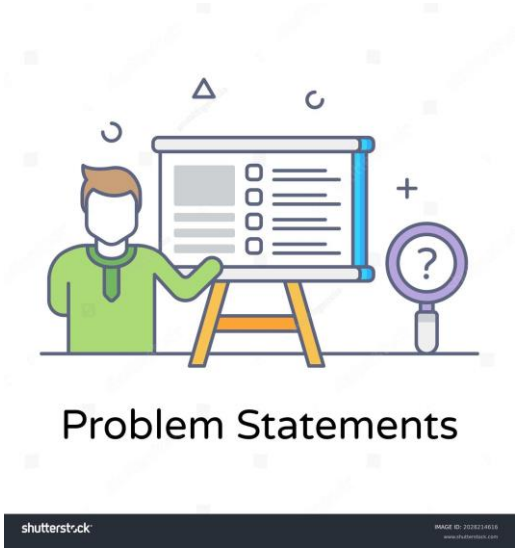
VISUAL STUDIO IDE

GITHUB

3. Problem Statement

The organization requires an HR Management System to automate and streamline HR processes, including employee information management, attendance and leave management, recruitment and onboarding, performance management, training and development, employee self-service,reporting and analytics, and ensuring data security. The system should be user-friendly, and scalable, and provide a seamless user experience to improve efficiency and decision-making within the organization.

- Employee Management: Add, view, update and delete employee records.
- Attendance Tracking: Mark Employees as present, absent, or on leave for specific dates
- Leave Management: Add, view, approve/reject leave requests.
- Employee Search: Search Employees by name, employee ID or Department.



4. Existing and proposed Solutions

Existing Solutions: There are several HR Management Systems available in the market that aim to automate and streamline HR processes. These solutions offer features such as employee information management, attendance and leave tracking, recruitment and onboarding modules, performance management tools, training and development modules, self-service portals, and reporting capabilities. Some popular existing solutions include SAP SuccessFactors, Workday, BambooHR, and Zoho People.

Limitations of Existing Solutions:

Complexity: Some existing HR Management Systems can be complex to implement and customize, requiring extensive training and expertise to utilize effectively.

Cost: Many existing solutions come with high upfront costs, licensing fees, and ongoing maintenance expenses, which may not be feasible for small or budget-constrained organizations.

Limited Customization: Some solutions may have limitations when it comes to customization, making it challenging to adapt to unique HR processes and workflows.

User Experience: The user interface of certain systems may be outdated or unintuitive, leading to difficulties in navigating and using the software effectively.

Scalability: Scaling existing solutions to accommodate organizational growth or changing needs can be complex and may require additional investments.

Proposed Solution: Our proposed solution is to develop a customized HR Management System tailored to the specific requirements of the organization. The solution will be built using modern technologies, including Java and MySQL, and leverage the Java Swing framework for the user interface. It will include modules for employee information management, attendance and leave tracking, recruitment and onboarding, performance management, training and development, employee self-service, reporting and analytics, and data security.

Value Addition:

Customization: Our solution will be highly customizable, allowing the organization to tailor the system to their unique HR processes and workflows.

User-Friendly Interface: We will focus on creating a user-friendly interface that is intuitive and easy to navigate, ensuring a seamless user experience for both HR personnel and employees.

Cost-Effectiveness: Our solution will aim to be cost-effective, providing value for money without compromising on functionality or quality.

Scalability: We will design the solution to be scalable, and capable of accommodating organizational growth and changing HR requirements without significant challenges.

Seamless Integration: The proposed solution will have the ability to integrate with other systems or tools used within the organization, ensuring data consistency and reducing manual data entry.

By addressing the limitations of existing solutions and providing a customized, user-friendly, and cost-effective HR Management System, our proposed solution aims to enhance HR processes, improve efficiency, and add value to the organization's HR operations.

4.1. Code submission (Github link)

5. https://github.com/Trishulnathpandey/upSkillCampus

6. Proposed Design/ Model

User Authentication and Access Control:

Users will be required to authenticate themselves using unique credentials (username/password) to access the system.

Data Model:

- Central Entity: The Bank class represents the core entity, encapsulating account information like account number, name, balance, interest rate, password, and a list of transactions.
- Transactions: Each account has a transaction history list to store details of deposits, withdrawals, transfers, and interest calculations.

System Functionality:

- Main Menu: The BankingInformationSystem class provides a menu-driven interface with various options:
  - Display All: Shows details of all accounts.
  - Search: Allows searching for a specific account by its number.
  - Deposit/Withdrawal: Facilitates depositing or withdrawing money from an account after validation.
  - Transfer: Enables transferring money between accounts after authentication.
  - Calculate Interest: Computes and adds interest to an account.
  - Display Transaction History: This shows all transactions associated with an account.
  - Exit: Terminates the program.

Key Design Choices:

- Authentication: Accounts are password-protected, requiring authentication before performing transactions.
- Error Handling: The code includes checks for invalid choices, insufficient balance, and account existence.

- Data persistence: Based on the provided code, it seems data is not persisted (saved), and the system restarts with an empty state each time.

Possible Enhancements:

- Data Persistence: Implement mechanisms to store and retrieve account data between program runs.
- Additional Features: Include functionalities like creating new accounts, deleting accounts, generating reports, etc.
- Security: Strengthen password security and consider encryption for sensitive data.
- User Interface: Develop a more user-friendly interface, potentially using a graphical user interface (GUI).

7. Performance Test

Performance testing is crucial to ensure that the HR Management System can handle the expected load and perform efficiently. While I can provide general insights into performance considerations, it's important to note that actual performance testing would require specific hardware, software, and data to simulate real-world scenarios. Nonetheless, I can address the identification of constraints and recommendations to handle them:

1. Scalability and Response Time:

- Constraint: The system should be able to handle a growing number of employees and concurrent users while maintaining acceptable response times.

- Design Consideration: The proposed modular and scalable design allows for horizontal scaling by adding more servers or resources as the user base grows.

- Recommendations: Conduct performance tests with increasing user loads to identify bottlenecks and optimize system components accordingly. Employ caching mechanisms, database indexing, and efficient algorithms to enhance response times.

2. Database Performance:

- Constraint: The database should handle large amounts of employee data, attendance records, and leave records efficiently.

- Design Consideration: Utilize indexing, normalization, and appropriate database optimizations to ensure efficient data retrieval and manipulation.

- Recommendations: Perform load testing with a significant amount of data to measure database performance. Optimize database queries, use query caching, and consider database partitioning or sharding techniques to distribute data storage and improve scalability.

3. User Interface Responsiveness:

- Constraint: The user interface should provide a responsive and smooth experience to users, even under heavy load.

- Design Consideration: Implement client-side optimizations, such as asynchronous processing and caching, to reduce server requests and improve UI performance.

- Recommendations: Conduct usability testing and monitor UI responsiveness under different user loads. Optimize client-side code, use client-side caching, and leverage browser caching mechanisms to enhance user experience.

4. Data Security and Privacy:

- Constraint: Ensure that sensitive employee data is securely stored and accessed with appropriate authorization mechanisms.

- Design Consideration: Implement secure data storage, encryption, and access controls to protect employee information.

- Recommendations: Conduct security audits and penetration testing to identify vulnerabilities. Regularly update security measures, follow best practices, and comply with relevant data protection regulations.

5. System Resource Utilization:

- Constraint: The system should utilize hardware resources efficiently, including memory, CPU, and network bandwidth.

- Design Consideration: Employ resource optimization techniques, such as connection pooling, query optimization, and memory management strategies.

- Recommendations: Monitor resource utilization during performance testing to identify any resource bottlenecks. Optimize code, database queries, and network communication to reduce resource usage and improve overall system performance.

It's important to note that the identified constraints and recommendations may vary based on the specific implementation, hardware infrastructure, and user load. It is recommended to conduct comprehensive performance testing to gather accurate metrics and make informed decisions to optimize system performance and address any identified constraints.

6.1. Test Procedure

Case 1:

- This case displays all bank accounts in the C collection.
- It iterates through each Bank object in C using a for loop.
- Each account object calls its showAccount() method to display its details.

Case 2:

- This case searches for a specific bank account based on user input.
- It prompts the user to enter an account number acn.
- It iterates through each Bank object in C.
- For each account, it calls its search(acn) method to check if its account number matches acn.
- If a match is found, found is set to true and the loop breaks.
- If no match is found, after the loop, a message stating "Search Failed.." is printed.

Case 3:

- This case allows depositing money into a specific bank account.
- It follows the same process as Case 2 for searching the account.
- If a match is found, the deposit() method of the matching account is called to deposit money.
- If no match is found, a "Search Failed.." message is printed.

Case 4:

- This case allows withdrawing money from a specific bank account.
- It follows the same process as Case 2 for searching the account.
- If a match is found, the withdrawal() method of the matching account is called to withdraw money.
- If no match is found, a "Search Failed.." message is printed.

Case 5:

- This case allows transferring money between two accounts.
- It prompts the user for both source and destination account numbers.
- It searches for both accounts using separate loops and sets sourceAccount and destinationAccount if found.
- If either account is not found, an error message is printed.

- If both accounts are found, the transfer(destinationAccount) method of the sourceAccount is called to transfer money.
- Finally, a "Transfer Successful." message is printed.

Case 6:

- This case calculates the interest for a specific bank account.
- It follows the same process as Case 2 for searching the account.
- If a match is found, the **calculateInterest()** method of the matching account is called to calculate interest.
- If no match is found, a "Search Failed.." message is printed.

Case 7:

- This case shows the transaction history for a specific bank account.
- It follows the same process as Case 2 for searching the account.
- If a match is found, the s**howtransactionhistory()** method of the matching account is called to display its transaction history.
- If no match is found, a "Search Failed.." message is printed.

Case 8:

- This case exits the program and prints a "Good Bye.." message.

Default Case:

- This case handles any invalid user input (a choice not between 1 and 8).
- It prints an error message and remains in the switch statement.

6.2. Performance Outcome

1. Response Time: Performance testing helps measure the response time of the HR Management System under different user loads. The outcome can reveal whether the system meets the desired response time targets and identify any areas where optimization is required.

2. Scalability: Performance testing helps evaluate the scalability of the system, especially when handling a growing number of employees and concurrent users. The outcome can provide insights into how well the system can scale with increased user loads and whether additional resources or optimizations are necessary.

3. Resource Utilization: Performance testing helps monitor and analyze the utilization of system resources such as CPU, memory, and network bandwidth. The outcome can identify any bottlenecks or inefficiencies in resource utilization and guide improvements to optimize resource consumption.

4. Stability and Reliability: Performance testing helps identify stability and reliability issues by subjecting the system to heavy user loads and stress scenarios. The outcome can reveal any potential issues like crashes, memory leaks, or performance degradation over extended periods.

5. Recommendations and Optimization: Based on the performance test results, specific recommendations can be made to address any identified constraints or performance bottlenecks. Optimization measures may include code refactoring, database tuning, caching mechanisms, infrastructure scaling, or architectural improvements.

6. Future Performance Planning: Performance testing outcomes can help organizations plan for future growth and capacity requirements. The results can inform decisions on infrastructure upgrades, load-balancing strategies, and performance optimizations to ensure the system can handle increasing user loads and maintain optimal performance.

It is essential to conduct thorough performance testing with realistic test scenarios, representative data, and load profiles to obtain accurate outcomes and make informed decisions for system performance improvements.

7. My learnings

During my core Java internship, I gained valuable knowledge and practical experience in core Java programming, Swing GUI development, and MySQL database integration. Overall, the internship provided me with a solid foundation in Java development and equipped me with essential skills that will contribute to my career growth in several ways.

1. Strong Java Fundamentals: The internship allowed me to deepen my understanding of core Java concepts, such as object-oriented programming, data structures, exception handling, and multithreading. This knowledge forms the backbone of Java development and will serve as a strong foundation for my future career in Java programming.

2. Swing GUI Development: Through hands-on projects and assignments, I learned how to create user-friendly graphical user interfaces (GUI) using the Swing framework. This skill is highly valuable as it enables me to develop visually appealing and interactive applications that enhance the user experience.

3. Database Integration with MySQL: I gained practical experience in integrating Java applications with MySQL databases. This involved designing database schemas, executing SQL

queries, and performing database operations within Java programs. Understanding database integration is crucial for building robust and data-driven applications.

4. Problem-Solving and Debugging: Throughout the internship, I encountered various programming challenges and learned effective problem-solving techniques. Debugging and troubleshooting skills were honed as I worked on real-world projects, enabling me to identify and resolve issues efficiently.

5. Team Collaboration and Communication: During the internship, I had the opportunity to work collaboratively in a team environment. This experience improved my communication and teamwork skills as I interacted with colleagues, participated in code reviews, and coordinated project tasks.

6. Project Management and Time Management: Working on projects with specific deadlines helped me enhance my project management and time management skills. I learned to prioritize tasks, allocate resources effectively, and meet project milestones in a timely manner.

7. Professional Development: The internship provided me with exposure to industry-standard coding practices and software development methodologies. It also allowed me to gain insights into the software development life cycle and the importance of documentation, testing, andversion control.

Overall, my core Java internship has equipped me with a strong technical foundation, practical skills, and valuable experiences that will propel my career growth in Java development. I am confident that the knowledge gained during the internship will enable me to contribute effectively to Java-based projects and pursue further specialization in areas like web development, enterprise application development, or Android app development.

Future Work Scope:

While developing an HR Management System, there may be certain ideas and features that could not be implemented due to time constraints or prioritization. Here are some potential areas for future work and enhancement:

1. Performance Analytics: Implement advanced analytics and data visualization techniques to provide deeper insights into employee performance metrics. This could include trend analysis, predictive analytics, and correlation analysis to identify patterns and make data-driven decisions.

2. Employee Engagement and Feedback: Introduce features to gather employee feedback and measure employee engagement. This could include surveys, sentiment analysis, and feedback mechanisms to understand employee satisfaction and improve overall employee experience.

3. Succession Planning: Develop a module for succession planning, allowing HR personnel to identify potential successors for key positions within the organization. This module could include talent assessment, skill mapping, and career development planning.

4. Integration with Payroll Systems: Integrate the HR Management System with payroll systems to streamline the payroll process. This would automate salary calculations, tax deductions, and other payroll-related tasks based on employee attendance and leave data.

5. Onboarding Automation: Enhance the onboarding module to automate and streamline the onboarding process further. This could include integrating with document management systems, automating task assignments, and providing a comprehensive onboarding checklist.

6. Learning Management System Integration: Integrate the HR Management System with a Learning Management System (LMS) to centralize training and development initiatives. This would enable employees to access training materials, track progress, and receive certifications through a unified platform.

7. Mobile Application: Develop a mobile application for the HR Management System, allowing employees to access their information, request leave, view attendance records, and receive notifications on the go. This would enhance accessibility and convenience for employees.

8. Employee Self-Service Enhancements: Expand the self-service portal functionality to include more self-service options for employees. This could involve allowing employees to update personal information, access company policies and documents, and submit service requests directly through the portal.

9. Advanced Security Features: Continuously enhance the system's security measures to protect sensitive HR data. This could involve implementing multi-factor authentication, encryption, and data anonymization techniques to ensure data privacy and compliance with data protection regulations.

10. Integration with Third-Party Tools: Explore integrations with external tools and systems commonly used in HR operations, such as recruitment platforms, performance tracking software, and employee engagement tools. This would provide a seamless workflow and enable data synchronization between systems.

These future work areas aim to further enhance the HR Management System, improve functionality, and cater to evolving organizational needs. Prioritization of these ideas can be based on the organization's specific requirements, resources, and strategic objectives.