

REWORKFORCE

Human Resource Management

Project Duration: 12-Feb-2026 to 10-Feb-2026

2 0 2 6

 by: Mastan Sayyad

1

2

Hello,

I'm Mastan Sayyad

Java Developer

I am currently training as a Java Developer at Revature, with a strong foundation in Java, SQL, Frontend and backend development. Proficient in Git and Github, hands-on experience in technical writing, leading a team and delivering clean coding practices.





Agenda

1

2

- | | |
|---------------------------------|---------------------------------|
| 1. Introduction to RevWorkForce | 11. Live Demonstration |
| 2. Objective | 12. Loggers and Metrics |
| 3. Tech Stack | 13. Testing & Quality Assurance |
| 4. Methodology | 14. GitHub Walkthrough |
| 5. System Architecture | 15. Future Scope |
| 6. Design Patterns | 16. Key Learnings |
| 7. Database Design | 17. Conclusion |
| 8. Core Features and Modules | 18. Thank You! |
| 9. Leave Management Workflow | |
| 10. Security Features | |

1

2

3

4

5

6

7

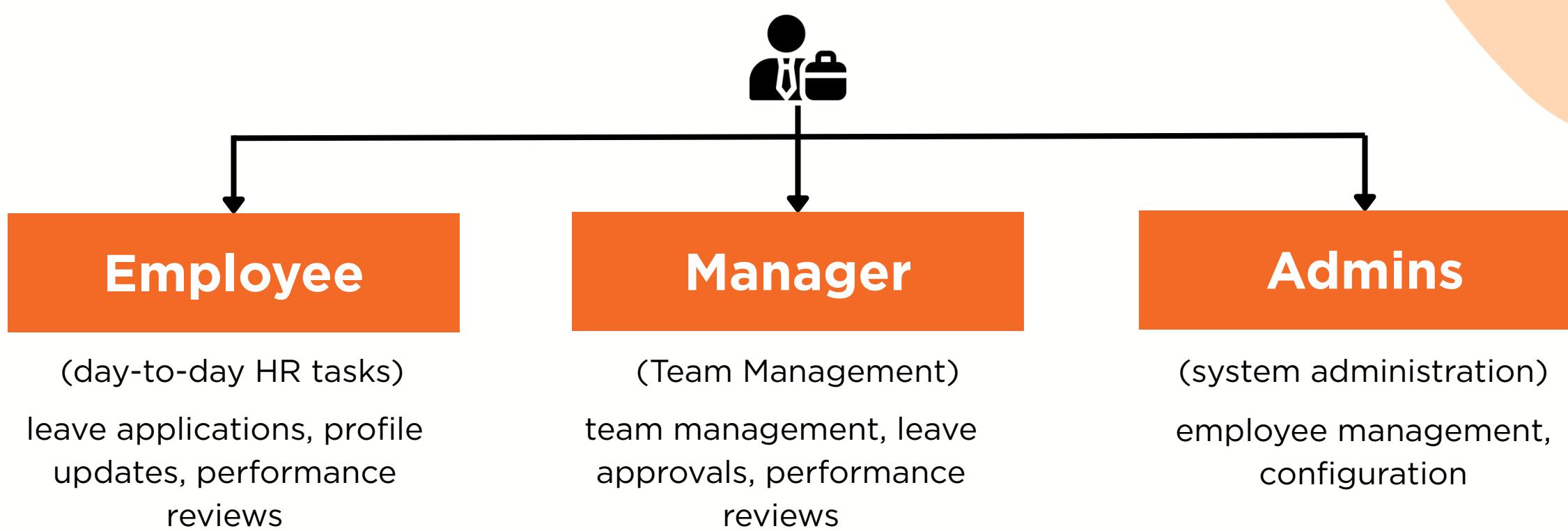
8

9

Introduction

What is RevWorkForce?

RevWorkForce is a comprehensive console-based Human Resource Management System designed to automate and streamline employee management, leave tracking, and performance review processes. The system implements role-based access control with three distinct user roles: **Employees**, **Managers**, and **Admins**.



...

1

2

3

4

5

6

7

8

9

Objectives

Primary goal:

1. Develop a scalable, role-based HRM system
2. Implement secure authentication & authorization
3. Automate leave management workflows
4. Enable structured performance management
5. Centralize employee information
6. Implement enterprise-grade security
7. Provide real-time notifications and audit trails

Role-based security

14,000+ LOC

14 Tables





Tech Stack

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

Core Technologies

Java 11 or Higher (LTS)

Oracle 19c Express Edition

JDBC for DB Connectivity

Maven 3.8+ (Build Tool)

Architecture Patterns

Layered Architecture

Libraries and Frameworks

JUnit 5 (Testing)

BCrypt (Hashing)

JaCoCo (Code Coverage)

Mockito (Mocking)

Log4j (Logging)

Development Tools

Eclipse/IntelliJ idea

Git and Github

SQL Developer

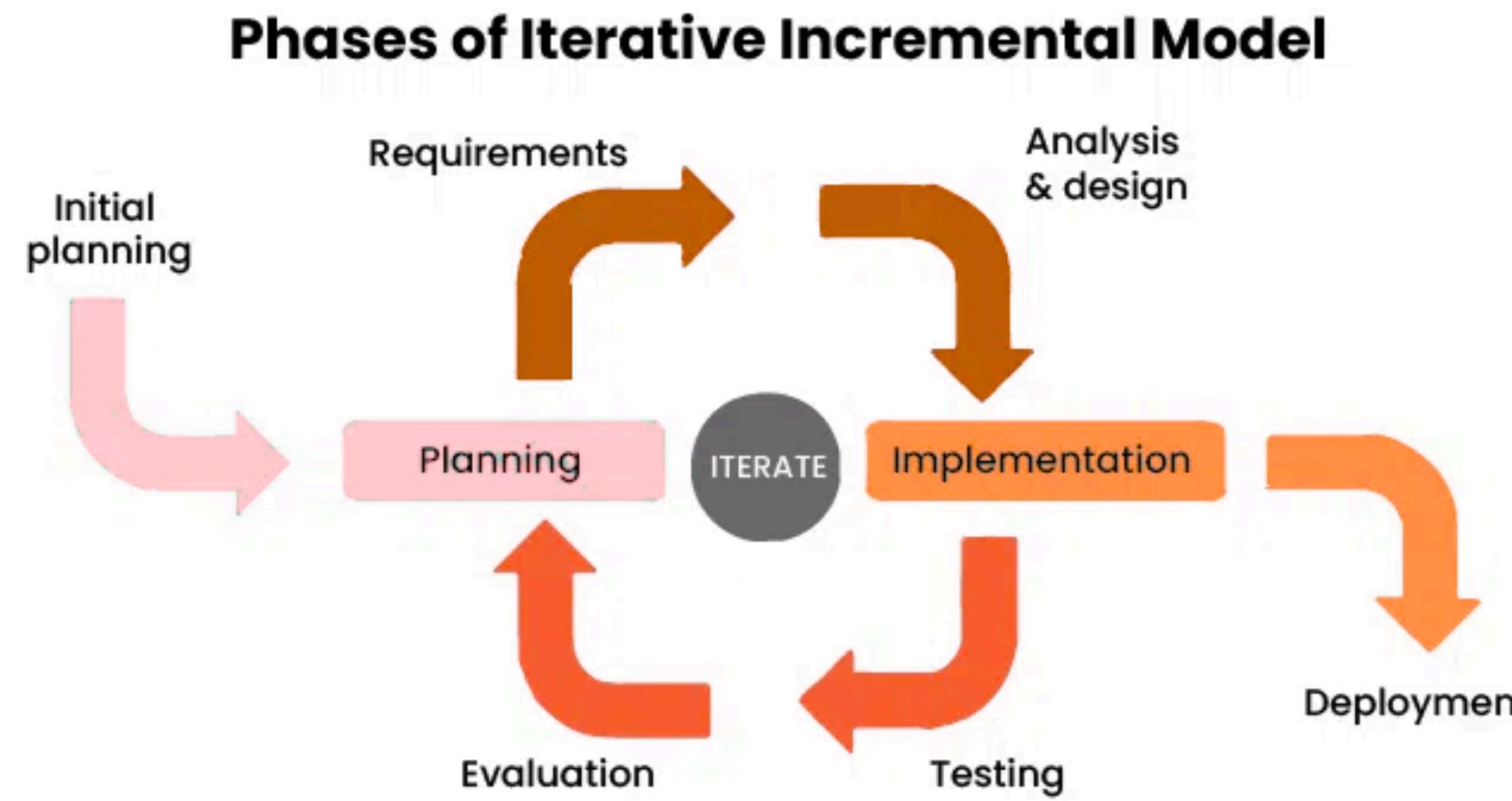
Github Actions

Methodology



Methodology

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9



Agile Practices

Product Backlog

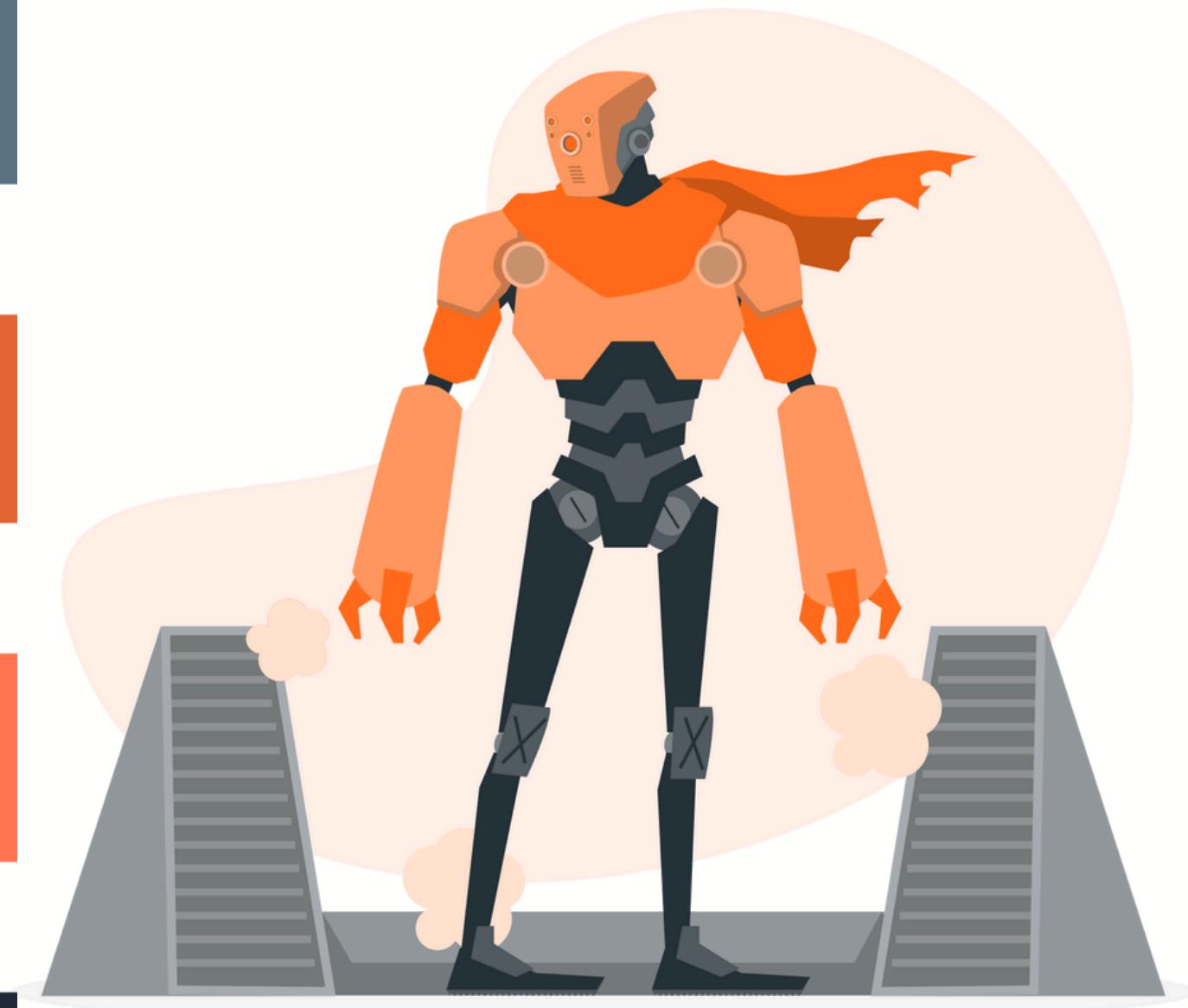
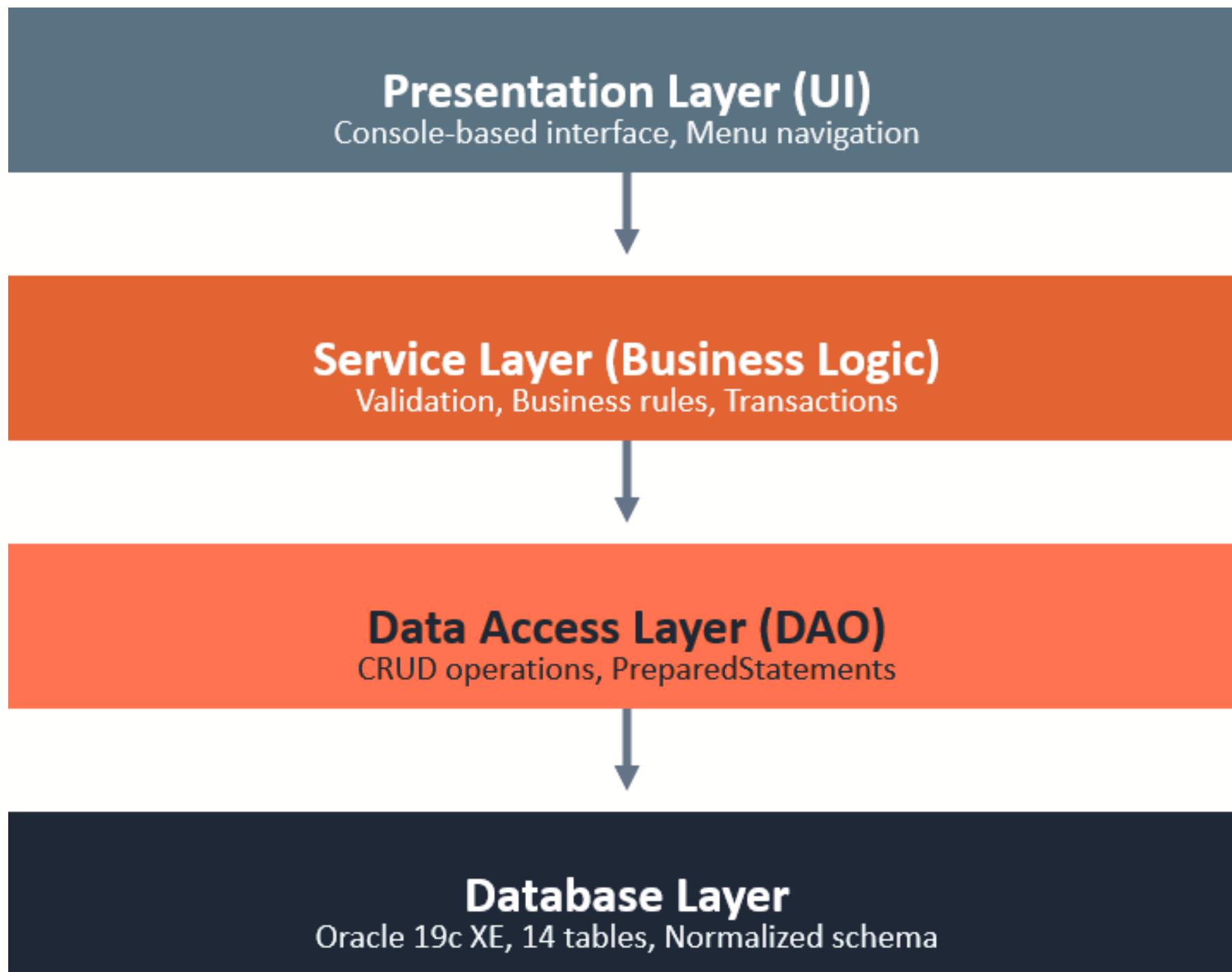
Sprint Planning/Board

Burn Up Charts

- **Modular Nature:** The project has clearly defined modules (Authentication, Leave Management, Performance Management) that can be developed incrementally
- **Time Constraints:** 10-day timeline requires working software at each iteration
- **Risk Mitigation:** Early iterations help identify technical challenges early
- **Feedback Integration:** Each iteration can be tested and refined
- **Priority-Based Development:** Core features can be developed first, additional features later

System Architecture

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9



Design Patterns

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

Singleton Pattern

DatabaseUtil, SessionManager
Ensures single instance

DAO Pattern

EmployeeDAO, LeaveDAO, etc.
Separates persistence logic

Service Layer

LeaveService, PerformanceService
Encapsulates business logic

Factory Pattern

Menu creation
Centralizes object creation

Dependency Injection

Services inject DAOs
Promotes loose coupling

DRY (Don't Repeat Yourself)

Meaningful naming conventions

Exception handling

Logging

Clean Code principles

Database Design

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

14 Tables | 6 Sequences | Normalized to 3NF

Employee Management

- EMPLOYEES
- DEPARTMENTS
- DESIGNATIONS
- ROLES

Leave Management

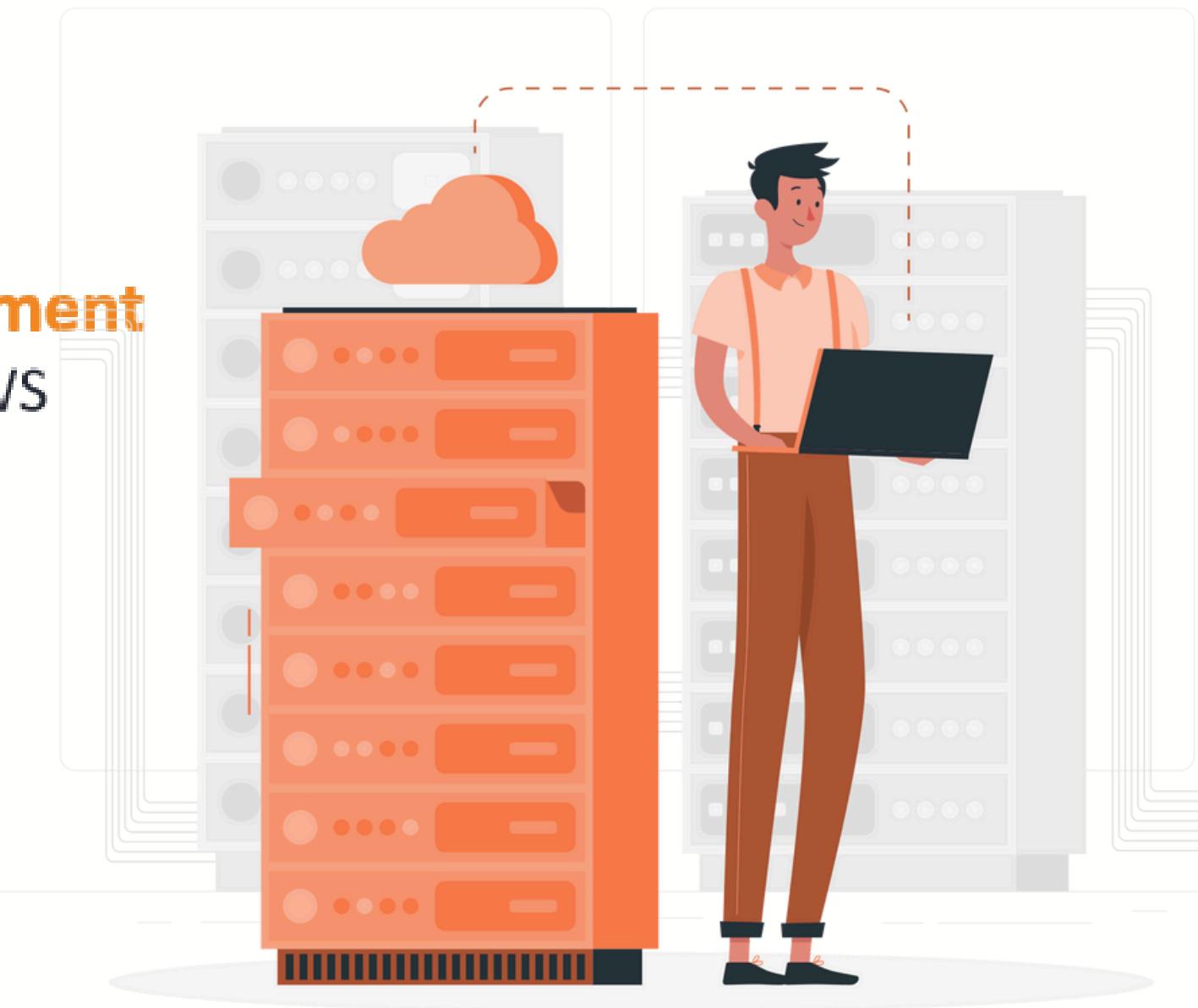
- LEAVE_APPLICATIONS
- LEAVE_TYPES
- LEAVE_BALANCES
- HOLIDAYS

Performance Management

- PERFORMANCE_REVIEWS
- GOALS

System & Audit

- NOTIFICATIONS
- ANNOUNCEMENTS
- AUDIT_LOGS



Primary Keys (Sequences) | Foreign Keys | Indexes | ACID Compliance

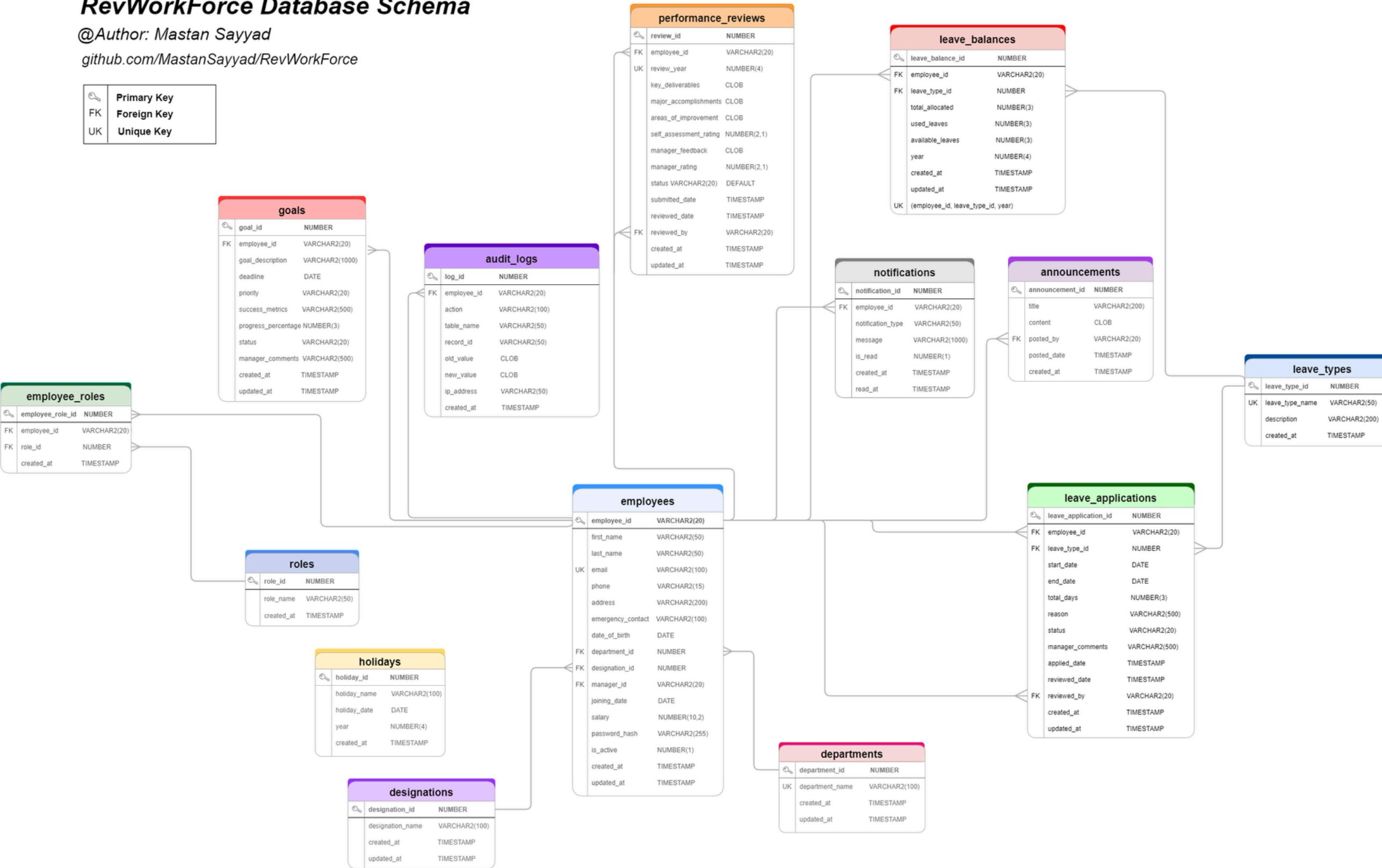
Database Design

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

RevWorkForce Database Schema

@Author: Mastan Sayyad
github.com/MastanSayyad/RevWorkForce

	Primary Key
	Foreign Key
	Unique Key



Core Features and Modules

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

Employee Management

- CRUD operations
- RBAC (3 roles)
- Department mapping
- Lifecycle tracking

Leave Management

- Multiple leave types
- Balance tracking
- Working days calc
- Approval workflow

Performance Mgmt

- Annual reviews
- Self-assessment
- Goal tracking
- Rating (1.0-5.0)

Notifications

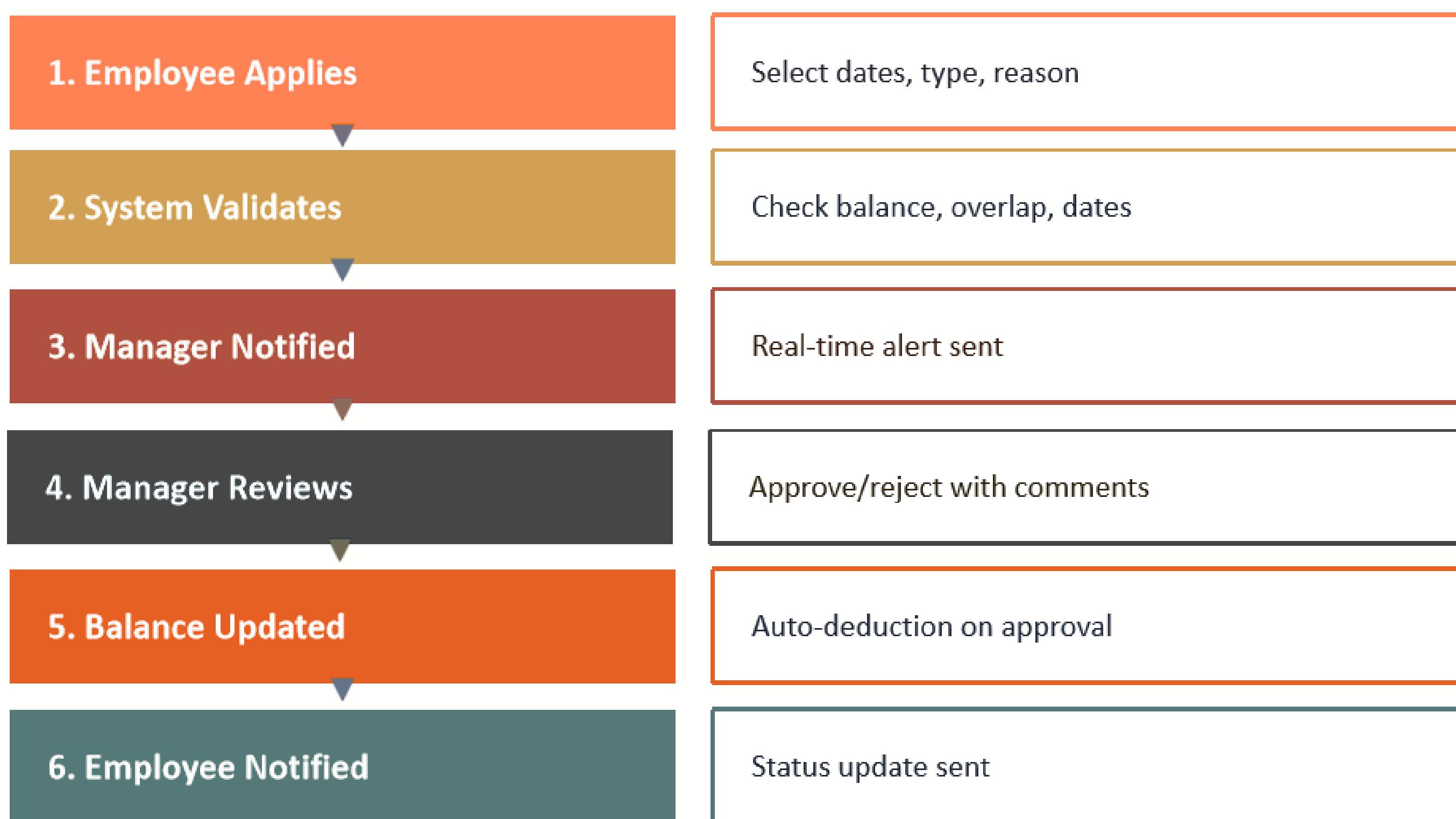
- Real-time alerts
- Announcements
- Audit trail
- Birthday reminders

70% Time Reduction | 100% Accuracy | Real-time Visibility



Leave Management Workflow

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9



Security Features

10

11

12

13

14

15

16

17

Password Security

- BCrypt hashing (12 rounds)
- Minimum 8 characters
- Salt generation

Access Control

- Role-based permissions
- 3-tier hierarchy
- Menu restrictions

Session Management

- 30-min timeout
- Single active session
- Secure storage

Audit Trail

- All actions logged
- User/action tracking
- Compliance support



Security Features

10

11

12

13

14

15

16

17

Data Protection

Input Validation & Sanitization

- All user inputs are validated for format, length, and mandatory fields before processing.

SQL Injection Prevention (PreparedStatements)

- SQL queries are precompiled and user inputs are bound as parameters, not concatenated.
- This completely prevents attackers from injecting malicious SQL code.

Sensitive Data Encryption (Passwords)

- Passwords are never stored in plain text.
- Secure hashing (BCrypt) is used before storing credentials in the database.





Live Demonstration

- Employee login and authentication
- User Features
- Leave application workflow
- Manager approval process
- Performance review creation
- Real-time notifications
- Audit Logs



10

11

12

13

14

15

16

17

Logger and Metrics

10

11

12

13

14

15

16

17

Log4j2 Implementation

ERROR

WARN

INFO

DEBUG

Critical errors

Warnings

Events

Debug info

- Logs are written automatically based on the log level used.
- Helps monitor application flow, capture errors, and debug issues without using System.out.println

Project Metrics

14,899 +

Lines of Code

62 +

Java Files

121+

Test Cases

60%+

Code Coverage

Testing & Quality Assurance

10

11

12

13

14

15

16

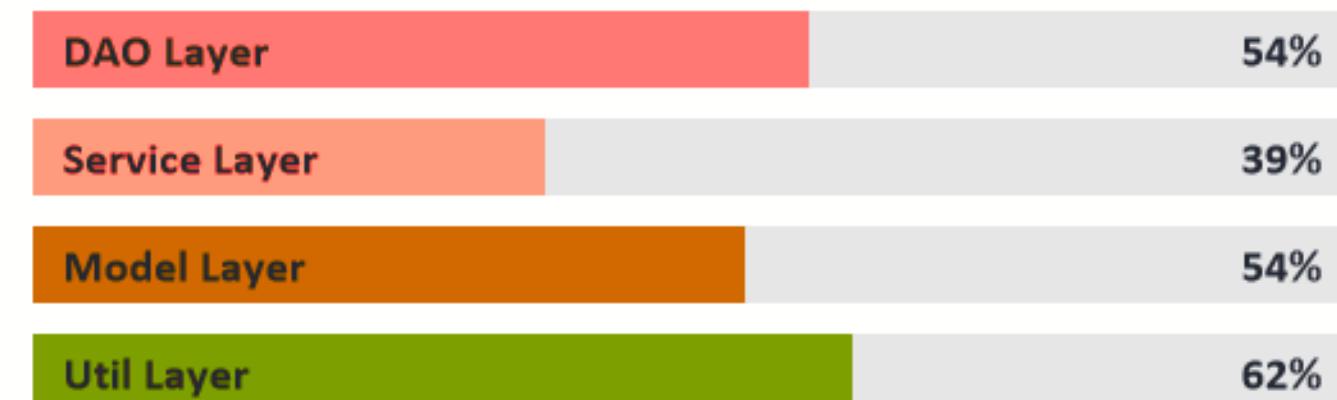
17

The project follows **Test-Driven Development (TDD)** approach, test cases are written before implementing business logic.

Testing Framework

- JUnit 5 - Unit Testing
- Mockito - Mocking (Future Enhancement)
- JaCoCo - Coverage
- Manual Testing (100% Pass)

Coverage by Layer



Test Coverage

Unit Tests

160+

Integration Tests

30+

DAO Tests

50+

✓ All Tests Pass ✓ 60%+ Coverage ✓ Zero Bugs ✓ Clean Code

GitHub Walkthrough



10

11

12

13

14

15

16

17

Repository Highlights

- Well-organized structure
- Comprehensive README
- DB setup scripts
- Maven configuration
- JaCoCo reports
- documentation
- Meaningful commits
- Issue Management

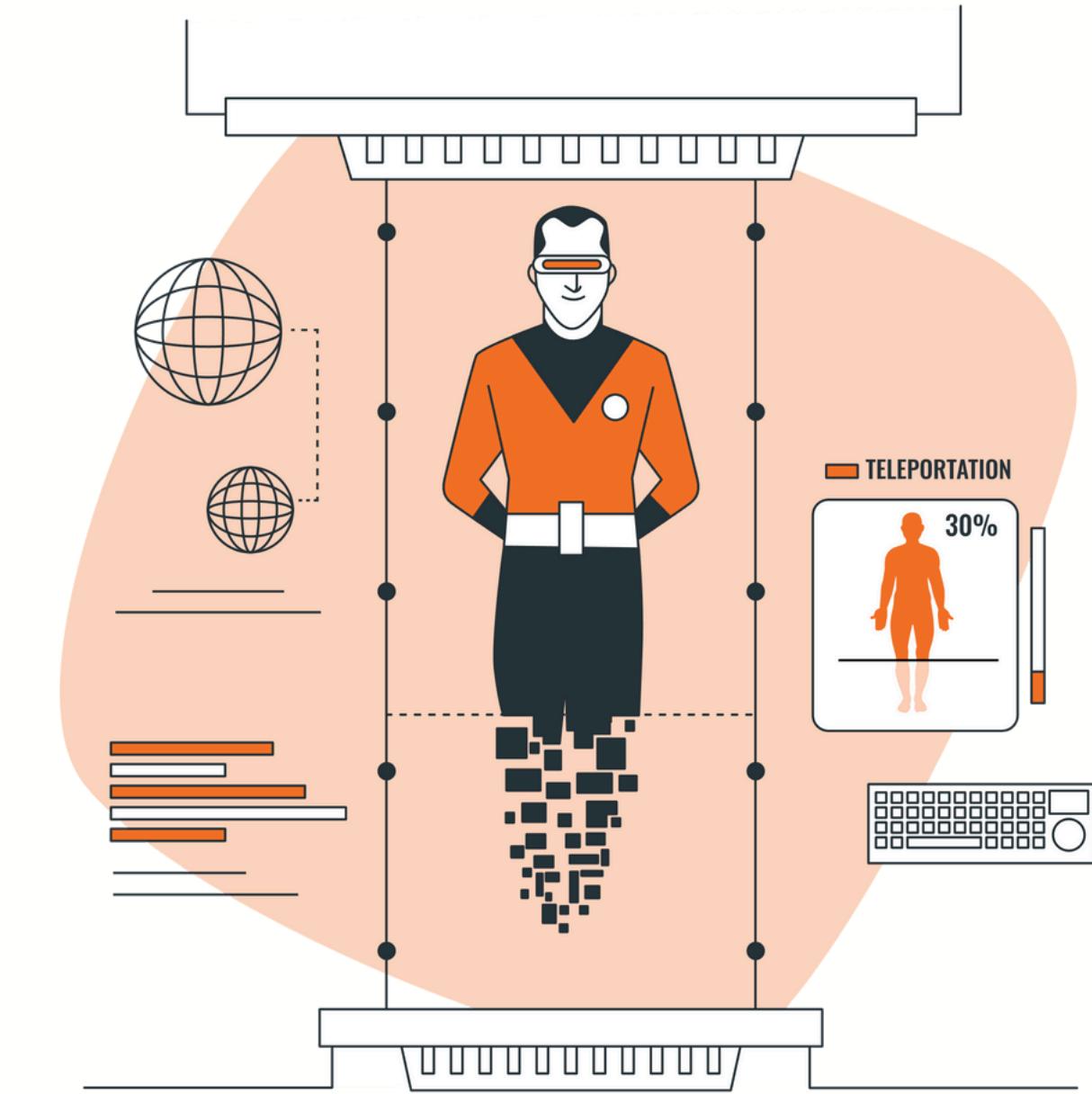
Development Workflow

- Feature branches
- Pull requests
- Main for releases
- GitHub Actions CI



Future Scope

- Migrate to Spring Boot
- Microservices architecture
- Spring MVC (Model-View-Controller)
- Docker containerization
- Event-driven design
- RESTful API development
- Email notifications
- Training & Development Module
- Thymeleaf spring boot integration
- Jenkins Continuous Integration (CI) and Continuous Delivery/Deployment



10

11

12

13

14

15

16

17

Conclusion

10

11

12

13

14

15

16

17

WHAT WE BUILT:

- ✓ Comprehensive HRM System
- ✓ 14,899+ lines of quality code
- ✓ 14-table database schema
- ✓ 160+ test cases (60% coverage)
- ✓ Complete documentation

WHAT WE ACHIEVED:

- ✓ Solved real HR management challenges
- ✓ Implemented enterprise architecture
- ✓ Applied industry best practices
- ✓ Delivered production-ready system
- ✓ Created scalable foundation

***Learnings: Software Engineering • Database Design • Security • Testing • Agile
Version Control • Project Management • TDD • Code review• Documentation***

10

11

12

13

14

15

16

17

Thank You

REVWORKFORCE

Human Resource Management

 **Mastan Sayyad**

 sayyadmastan78@gmail.com

 github.com/MastanSayyad

 <https://www.linkedin.com/in/mastan-sayyad>



(Scan for Github Repo)

Project Resources:

- Source Code: <https://github.com/MastanSayyad/RevWorkForce>
- Documentation: [RevWorkForce/Docs](#)
- User Manual: [User Manual](#)
- Presentation: [PPT](#)
- Testing Report: [JUnit Test Report](#)

