



Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

FACULTY KIT

Objective –

The purpose of this faculty kit is to provide a structured roadmap for guiding students in developing the *Miles Acquisition System (MAS)*. This system enables users to earn, track, and redeem miles through integrations with partner businesses such as airlines, e-commerce platforms, and banks. It includes modules for user registration, transaction tracking, partner rule processing, fraud detection, analytics, and reporting. The kit will help faculty oversee the design, development, testing, and deployment phases of the MAS project, while ensuring adherence to goals and quality benchmarks.

Requirements Specification –

The Miles Acquisition System will include:

- **User Management:** Secure authentication and role-based access for users (Customer, Partner, Admin).
- **Transaction Tracking:** Real-time capture and processing of user transactions across platforms.
- **Miles Calculation Engine:** Applies configurable partner-specific rules to calculate miles.
- **Redemption Module:** Allows users to redeem miles for various rewards.
- **Partner Integration:** API-based integration for transaction data ingestion and validation.
- **Admin Control Panel:** Admins manage partners, monitor fraud detection alerts, and oversee data quality.
- **Analytics & Reporting:** Dashboards for user and partner insights, including miles history and redemption trends.

Technology Familiarization –

The project will use Spring Boot for backend development, enabling scalable REST APIs and efficient database interaction. React.js will be used for the frontend to deliver a dynamic and responsive user interface. For the database layer, both SQL (MySQL) database will be used to manage structured and unstructured data such as transaction logs, user behavior, and partner rules.

Database Creation –

The MAS will utilize:

- SQL (PostgreSQL): For user profiles, transaction history, partner details, and redemption records.
- NoSQL (MongoDB): For logs, dynamic rules, and real-time processing data.

Main Collections/Tables:

- User: UserID, Name, Email, Role, MilesBalance
- Partner: PartnerID, Name, Type, Rules
- Transaction: TxnID, UserID, PartnerID, Amount, Timestamp, MilesEarned
- Redemption: RedemptionID, UserID, RewardType, MilesUsed, Date

High-Level and Detailed Design

System Layers:

1. Frontend (React.js)
 - Interfaces for registration, miles dashboard, transaction history, redemption, and partner portal.
2. Backend (Spring Boot)
 - Handles user authentication, miles processing logic, transaction validation, and REST APIs.
3. Database Layer (MySQL)
 - MySQL stores persistent structured data.
 - MySQL handles semi-structured logs, rule sets, and analytics.

Design Details:

- RESTful APIs will manage operations like user login, transaction submission, miles earning/redemption, and admin analytics.
- React components will consume these APIs using Axios to create an interactive UI.

Frontend Implementation

The Angular-based UI will include:

- Users: View miles, check transaction history, redeem points.
- Partners: Monitor submitted transactions, define miles rules, access dashboards.
- Admins: Manage users and partners, oversee fraud detection, generate reports.

Each role will have tailored interfaces and access controls.

Frontend-Backend Integration

Angular will communicate with the backend via REST APIs. Backend endpoints will handle CRUD operations on the PostgreSQL and MongoDB databases. Axios will be used in React to fetch or update user data, process transactions, and render dashboard updates in real-time.

Test Plan Review

Testing will be done in multiple phases:

- Unit Testing: Verifying API correctness and transaction logic.
- Integration Testing: Ensuring end-to-end flow between frontend and backend.
- UI Testing: Validating responsiveness and correctness of interfaces.
- Performance Testing: Testing the system under high transaction loads.

Final Review

A comprehensive review will ensure:

- Secure and correct user registration and login.
- Accurate miles calculation and redemption logic.
- Real-time partner integrations.
- Functional dashboards and reports. User feedback will be gathered to suggest improvements.

Documents/References

- Spring Boot Documentation (<https://spring.io/projects/spring-boot>)
- React.js Official Docs (<https://angular.org/>)
- MySQL Guides
- REST API Best Practices

Conclusion

This faculty kit provides a thorough framework to supervise the development of the Miles Acquisition System. With a focus on modular design, real-time data handling, and scalable architecture, MAS equips students with practical experience in building a real-world transactional system. It blends multiple technologies and disciplines, promoting hands-on learning in system integration, database design, and frontend-backend communication.

The comprehensive breakdown of requirements, design, testing, and implementation phases ensures faculty members can effectively track progress and guide students toward successful project delivery. As industries continue to shift toward customer loyalty and personalization, such systems represent the future of intelligent reward platforms. MAS not only provides technical exposure but also reflects the business impact of digital transformation in customer engagement strategies.

