Project Design Phase

Technology Stack

Date: 02 NOV 2025

Team ID: NM2025TMID02325

Title: CRM APPLICATION FOR JEWEL MANAGEMENT

Maximum Marks: 4 Marks

1. Overview

The CRM Application for Jewel Management will be developed using a comprehensive and reliable technology stack designed to ensure optimal performance, robust security, excellent scalability, and simplified maintenance. The technology stack strategically combines both frontend and backend technologies, database solutions, integration tools, and modern deployment mechanisms to create a robust and efficient system.

2. Technology Stack Summary

Layer	Technology / Tool	Purpose
Frontend (User Interface)	HTML5, CSS3, JavaScript	Structure, style, and interactivity for the responsive CRM web interface.
	React.js / Angular	For creating responsive, dynamic, and reusable UI components with excellent performance.
Backend (Server-Side)	Node.js with Express / Java Spring Boot	Server logic, API handling, business process automation, and request processing.
Database	MySQL / PostgreSQL	Secure storage of customer, order, billing, and inventory data with reliability.
API Layer	RESTful APIs / GraphQL	Integration of billing modules, payment gateways, and SMS/email communication services.
Authentication & Security	JWT / OAuth 2.0	User login, session management, secure access control, and credential protection.
Hosting / Deployment	AWS / Google Cloud / Azure	Reliable cloud-based hosting, deployment, and auto-scaling capabilities.
Version Control	Git, GitHub / GitLab	Source code management, version tracking, and seamless team collaboration.
Communication Services	Twilio / SendGrid	Sending SMS, email notifications, and promotional alerts to customers.
Payment Integration	Razorpay / Stripe / PayPal APIs	Secure online payment processing and transaction tracking.

Layer	Technology / Tool	Purpose
Testing Frameworks	Jest / Mocha / Selenium	Unit testing, integration testing, and end-to-end testing for system reliability.
Analytics & Reporting	Power BI / Tableau / Metabase	Dashboard generation, visual analytics, and comprehensive performance reporting.
Containerization	Docker / Kubernetes	Container orchestration for consistent development, testing, and production environments.

3. Technology Stack Architecture

The system follows a **3-Tier Architecture** approach for scalability and maintainability:

Presentation Layer (Frontend/UI)

- Developed using **React.js** or **Angular** framework for dynamic, responsive interfaces
- Provides intuitive and user-friendly interfaces for administrators, sales teams, and end customers
- Implements responsive design for cross-device compatibility (desktop, tablet, mobile)
- Uses state management libraries (Redux/Context API) for efficient data handling

Application Layer (Backend/Logic)

- Implements comprehensive business logic, workflow automation, and data validation
- Uses **Node.js with Express** or **Java Spring Boot** for processing requests and managing APIs
- Handles authentication, authorization, and session management
- Implements microservices architecture for modularity and independent scaling
- Manages integration with external services and third-party APIs

Data Layer (Database/Storage)

- Uses **MySQL** or **PostgreSQL** for relational data storage and retrieval
- Includes robust data backup mechanisms and disaster recovery strategies
- Implements encryption at rest using industry-standard encryption protocols (AES)
- Maintains data integrity through transactions and constraint management
- Includes comprehensive audit logging for security and compliance

4. Integration Components

SMS & Email Services

- Twilio API for SMS notifications and customer communications
- SendGrid or AWS SES for email delivery and campaign management
- Support for scheduled and triggered notifications

Payment Gateways

- Integration with Razorpay for Indian market transactions

- Stripe integration for international payment processing

- PayPal support for alternative payment methods

- PCI DSS compliance for secure payment handling

External APIs

- Support for future integrations with accounting software (Tally, QuickBooks)

- Loyalty program integration capabilities

- Inventory synchronization with supplier systems

- Social media integration for marketing campaigns

Data Analytics

- Power BI or Tableau dashboards for executive reporting

- Real-time metrics visualization and KPI tracking

- Custom report generation with drill-down capabilities

5. Development & Deployment Tools

Category	Tools / Software
IDE / Code Editor	Visual Studio Code, IntelliJ IDEA, WebStorm
Version Control	GitHub, GitLab, Bitbucket
Project Management	Jira, Trello, Asana
API Testing	Postman, Insomnia, REST Client
Automated Testing	Jest, Mocha, Selenium, Cypress
CI/CD Pipelines	GitHub Actions, Jenkins, GitLab CI/CD
Documentation	Confluence, Notion, Swagger/OpenAPI
Monitoring & Logging	ELK Stack, CloudWatch, DataDog

6. Security Technologies

- **SSL/TLS Encryption** - All data transmission between client and server is encrypted using industry-standard protocols

- **Role-Based Access Control (RBAC)** - Different permission levels for Admin, Manager, Sales Executive, and Inventory Manager roles

- **Data Encryption at Rest** - Sensitive data stored in database is encrypted using AES-256 encryption standards

- **API Security** - Implementation of rate limiting, API authentication, and request validation

- **Regular Security Audits** - Periodic vulnerability assessments and penetration testing

- **Backup & Disaster Recovery** - Automated daily backups with point-in-time recovery capabilities

- **Compliance Standards** - Adherence to GDPR, data privacy regulations, and PCI-DSS for payment processing

7. Deployment & Maintenance

Cloud Hosting Infrastructure

- Deployed on AWS EC2 with auto-scaling for load management
- Multi-region deployment for high availability and disaster recovery
- Content Delivery Network (CDN) for optimal performance

Continuous Integration/Continuous Deployment (CI/CD)

- Automated testing on every code commit
- Automated deployment to staging and production environments
- Rollback capabilities for failed deployments

Monitoring & Performance Management

- Real-time application performance monitoring (APM) using tools like New Relic or Datadog
- Automated alerts for system failures, performance degradation, or security incidents
- Regular log analysis and performance optimization
- Server health checks and automatic restart mechanisms for failed services