

9530

St. MOTHER THERESA ENGINEERING COLLEGE
COMPUTER SCIENCE ENGINEERING

NM-ID:B61DCBAF5F9AEADA4E54997D903710

29 REG NO :953023104047

DATE15-09-2025

Your paragraph text

Completed the project named as

Phase 2

FRONT END TECHNOLOGY
NEWS FEED APPLICATION

SUBMITTED BY:

KIRTHIC

PRAKASH .S

6382471983

Phase 2 – Solution Design and Architecture

❖ System Overview :

The IBM-FE News Feed Application will be designed as a web-based platform that provides users with real-time access to IBM-FE announcements, events, and updates. The architecture follows a modular client-server model to ensure scalability, real-time performance, and maintainability.

The application will have three core layers:

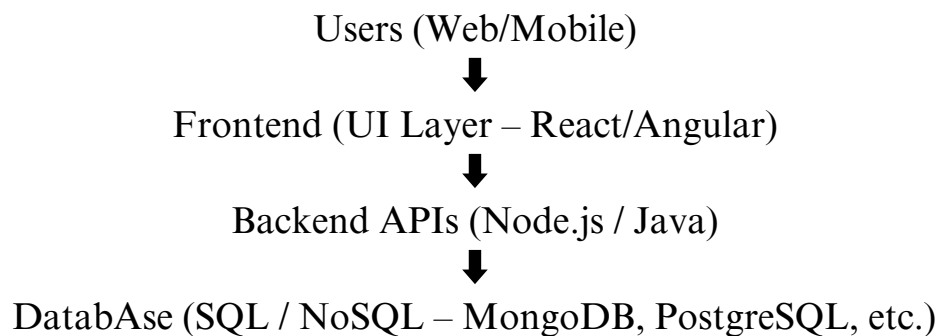
- Frontend (User Interface Layer): Provides a responsive, intuitive interface for viewing and interacting with news posts.
- Backend (Application Logic Layer): Handles data retrieval, real-time updates, authentication, and business logic.
- Database (Data Storage Layer): Stores all posts, categories, user preferences, and feed history.

❖ Solution Design

- Frontend (React/Angular/IBM-FE preferred framework):
 - Components for feed display, categories, post detail view.
 - Real-time updates using WebSockets or polling.
 - Responsive design for desktop and mobile.
 - Accessibility compliance (WCAG standards).
- Backend (Node.js/Java/Spring Boot):
 - RESTful API for retrieving posts, categories, and user settings.
 - WebSocket server for push notifications and live updates.
 - Authentication & authorization (role-based access if required).
 - Logging & monitoring for performance tracking.
- Database (SQL/NoSQL):
 - Posts table/collection (title, description, timestamp, category, author).
 - Categories table/collection (event, announcement, update).
 - User data for personalization (favorites, read/unread status).

- . Indexed queries for fast retrieval.

❖ Architecture Diagram (Conceptual) :



- . WebSockets / Push Notification Service → Enables real-time updates.
- . Caching layer (Redis/Memcached)→ Speeds up repeated feed retrieval.

❖ Key Design Considerations

- Scalability: The architecture supports horizontal scaling for handling large numbers of users.
- . Performance: Real-time updates with minimal latency through WebSockets.
- . Security: HTTPS, role-based authentication, and secure API endpoints.
- . Maintainability: Modular components for easier debugging and upgrades.
- . Reliability: Database replication and backup for continuous availability.

❖ Acceptance Criteria for Architecture

- . Users can view posts with low latency (<2s load time).
- . Feed auto-refreshes or updates via WebSockets without manual reload.
- . System can support concurrent users without performance drops.
- . Data consistency across frontend, backend, and database is maintained.