Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	29 October 2025
Team ID	**NM2025TMID06552**
Project Name	Optimizing User, Group, and Role Management with Access Control and Workflows
Maximum Marks	4 Marks

Technical Architecture

The project architecture focuses on creating a **secure, automated, and scalable environment** for managing users, groups, and roles through **access control and workflow automation**.

It combines frontend interfaces, workflow engines, backend logic, and access control layers to ensure secure and efficient role management.

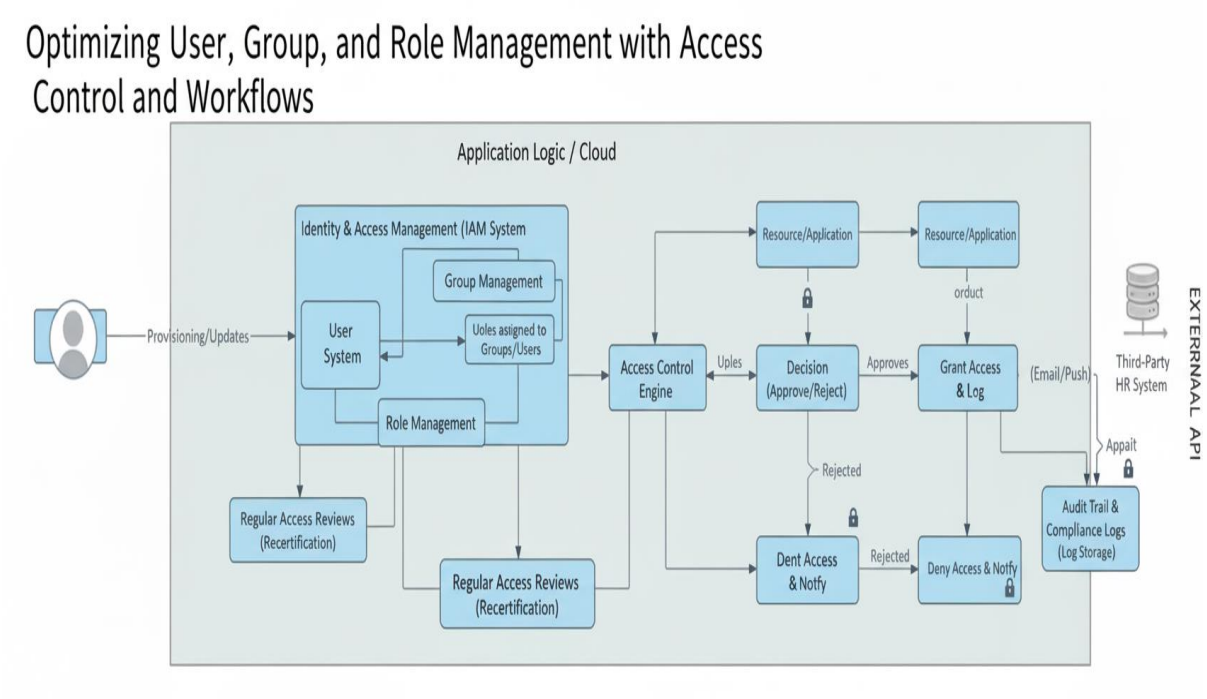

Table 1: Components & Technologies

S.No	Component Description	Technology Used
1.	**User Interface (UI)** – Admin and users interact through a responsive dashboard for user, group, and role operations.	React.js / HTML / CSS / JavaScript
2.	**Application Logic Layer – 1:** Handles user creation, validation, and group-role assignment.	Node.js / Express.js
3.	**Application Logic Layer – 2:** Implements dependency validation between user, group, and role relationships.	Server-side scripting (JavaScript / Node.js)
4.	**Workflow Engine:** Automates approval processes for user access requests.	ServiceNow Flow Designer / Node Workflow Library
5.	**Access Control Engine (RBAC):** Enforces access restrictions and role-based permissions dynamically.	Role-Based Access Control (RBAC) Logic / Middleware
6.	**Database Layer:** Stores user profiles, group details, roles, and access history.	MySQL / PostgreSQL
7.	**Cloud Infrastructure:** Provides scalability, hosting, and backup support for the application.	AWS / Azure Cloud Services
8.	**API Gateway:** Facilitates communication between frontend, backend, and external integrations.	RESTful APIs
9.	**Audit and Logging Service:** Tracks workflow and access control events for monitoring and compliance.	MongoDB / Elasticsearch / ServiceNow Logs
10.	**Notification System:** Sends alerts and status updates after workflow approval or access changes.	NodeMailer / Firebase Cloud Messaging (FCM)
11.	**External Integration (Optional):** Connects with third-party HRMS or LDAP for user verification.	LDAP Integration / REST API
12.	**Infrastructure (Deployment):** Deployed on scalable cloud environments for continuous availability.	AWS EC2 / Azure App Service

Table 2: Application Characteristics

S.No	Characteristic	Description	Technology / Implementation
1.	**Open-Source Frameworks**	Uses open-source frameworks for scalability and ease of customization.	Node.js, React.js
2.	**Security Implementations**	Incorporates Role-Based Access Control (RBAC), secure session handling, and encrypted communications.	JWT Authentication, ACLs, HTTPS
3.	**Scalable Architecture**	Built with a modular and microservice-ready structure to handle large-scale enterprise workloads.	Node.js, AWS Auto Scaling
4.	**Availability**	High availability ensured through load balancing and failover redundancy.	AWS Elastic Load Balancer / Azure Cloud Load Balancing
5.	**Performance**	Optimized through asynchronous workflows, caching, and efficient database indexing.	Redis Cache / Indexed SQL Queries
6.	**Auditability**	Each user action, role change, or workflow approval is logged for compliance and traceability.	MongoDB Logs / CloudWatch
7.	**Maintainability**	Designed with modular components for easy updates and debugging.	Docker Containers / CI-CD Pipelines
8.	**Interoperability**	Supports integration with identity management systems and third-party applications.	REST APIs / LDAP / OAuth 2.0