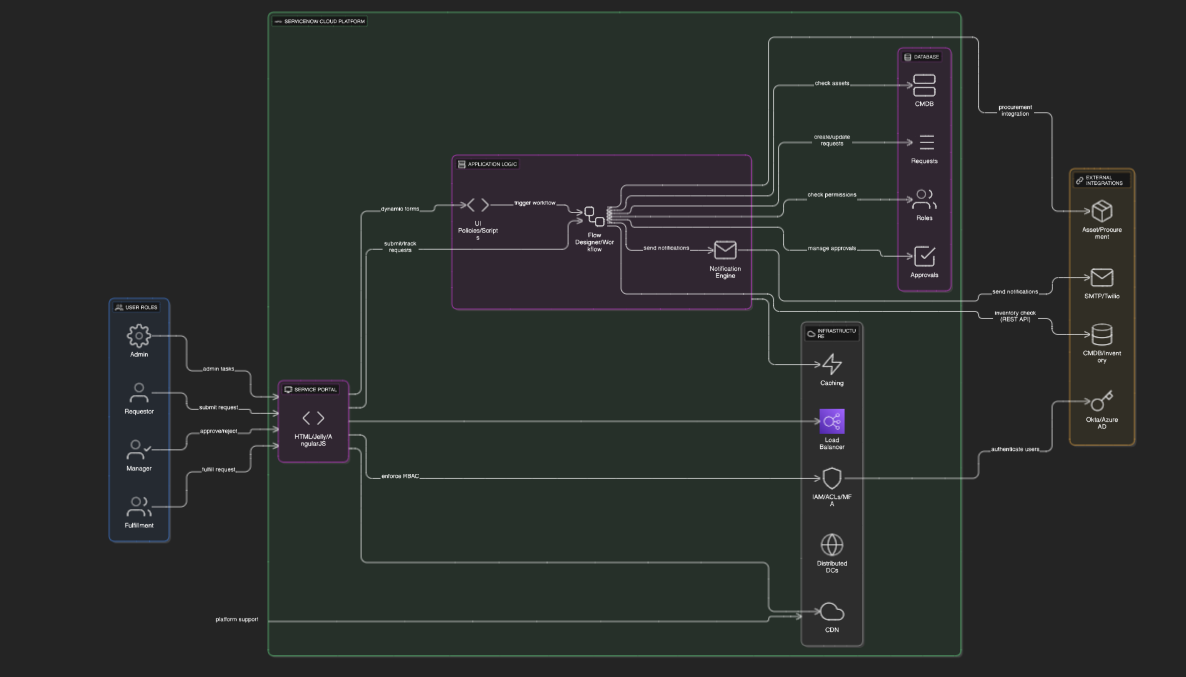
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 31 October 2025 |
| Team ID | NM2025TMID08347 |
| Project Name | LAPTOP REQUEST CATALOG ITEM |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | Web-based ServiceNow Service Catalog portal to submit requests | ServiceNow UI Framework (HTML, Jelly, AngularJS inside ServiceNow) |
|  | Application Logic-1 | Dynamic form behavior using UI Policies and Client Scripts | ServiceNow Server-side scripts (JavaScript, Glide API) |
|  | Application Logic-2 | Workflow automation and approval processing | ServiceNow Flow Designer / Workflow Engine |
|  | Application Logic-3 | Email notification and alert logic | ServiceNow Notification Engine |
|  | Database | Stores laptop request, approval, and user data | ServiceNow CMDB, Task and Request tables (MySQL-based backend) |
|  | External API-1 | Integration to fetch laptop availability from CMDB | ServiceNow REST API or OOTB CMDB APIs |
|  | Infrastructure (Server / Cloud) | ServiceNow Platform hosted on Cloud | ServiceNow Cloud SaaS Platform |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | ServiceNow platform leverages open-source components like AngularJS for UI, and JavaScript for client/server scripting. | AngularJS, JavaScript |
|  | Security Implementations | Role-based access control, multi-factor authentication, encryption of data at rest and in transit, OWASP best practices. | ServiceNow IAM, HTTPS, TLS, ACLs, OAuth2 |
|  | Scalable Architecture | ServiceNow’s multi-tenant cloud architecture supports scalability with modular components and APIs following microservices principles. | ServiceNow Cloud Platform, REST APIs |
|  | Availability | High availability with geographically distributed data centers, automated failover, scheduled maintenance windows. | ServiceNow Cloud Infrastructure, Load Balancers |
|  | Performance | Optimized workflows, caching mechanisms within ServiceNow, and CDN services for static assets to ensure low latency and high throughput. | ServiceNow Caching Layer, CDN (Akamaized or similar) |