Project Design Phase
Solution Architecture

Date	30 June 2025
Team ID	LTVIP2025TMID29074
Project Name	Educational Organisation Using ServiceNow
Maximum Marks	4 Marks

Solution Architecture:

The solution architecture for the **Educational Request Automation System** is designed to streamline and automate the handling of service requests (e.g., transcript requests, ID card reissues, course assistance) within an educational institution. It leverages ServiceNow's platform capabilities, rule-based logic, real-time workload dashboards, and optionally ML-based request prediction.

◇ Architecture Objectives

- Automate academic and administrative request routing
- Support dynamic workload-based assignment across departments
- Integrate SLA-based monitoring and escalation
- Enable optional ML-powered request categorization and routing
- Ensure scalability for multi-department or multi-campus environments

Component	Description / Department
User Interface	ServiceNow portal for students, faculty, and admins to raise/view requests and statuses
Assignment Rule Engine	Rule-based logic to determine which department or role should handle each request
Workload Monitor	Dashboards showing current team workload for balanced routing
ML Prediction Engine (Optional)	Model for predicting request category/assignee based on past data (hosted externally or via MID Server)
Ticket Database	Standard ServiceNow tables (`incident`, `task`, `student_request`, `sys_user`)
SLA Watcher	Monitors request resolution times, escalates or reassigns based on institutional SLAs
Integration Layer	Connects with external APIs or logic modules for enhanced routing and analytics

Architecture Flow

1. A student submits a request (e.g., for transcript, hostel issue, course help) via ServiceNow
2. Rule engine evaluates request metadata (e.g., category, urgency, student department)
3. Request is assigned based on rules or ML model output
4. Workload monitor prevents team overload and ensures fair distribution

5. SLA Watcher monitors status and escalates delayed requests
6. Notifications are sent to assigned staff/faculty via dashboard and email

Technology Stack

1. **Frontend**: ServiceNow Portal (Now Experience UI / Classic UI)
2. **Backend**: Flow Designer, Script Includes, Business Rules
3. **Data Storage**: ServiceNow tables (`incident`, `task`, `student_request`)
4. **ML Layer (Optional)**: Python model on AWS Lambda or Flask API
5. **Visualization**: ServiceNow Performance Analytics Dashboards, Reports

Development Phases

1. Problem & Workflow Definition for academic requests
2. Configuration of Rule Logic in ServiceNow
3. Dashboard and SLA Logic Integration
4. (Optional) ML Model Training using historical request dataAPI Integration and workflow automation
5. Testing and deployment

Scalability Considerations

- Modular rule engine to support academic, administrative, and technical departments
- Multi-campus request handling support via scoped apps
- Extensible APIs for ML, analytics, or ERP integrations
- SLA logic configurable per request type or department (e.g., Exams, Hostel, Library)

Solution Architecture Diagram: