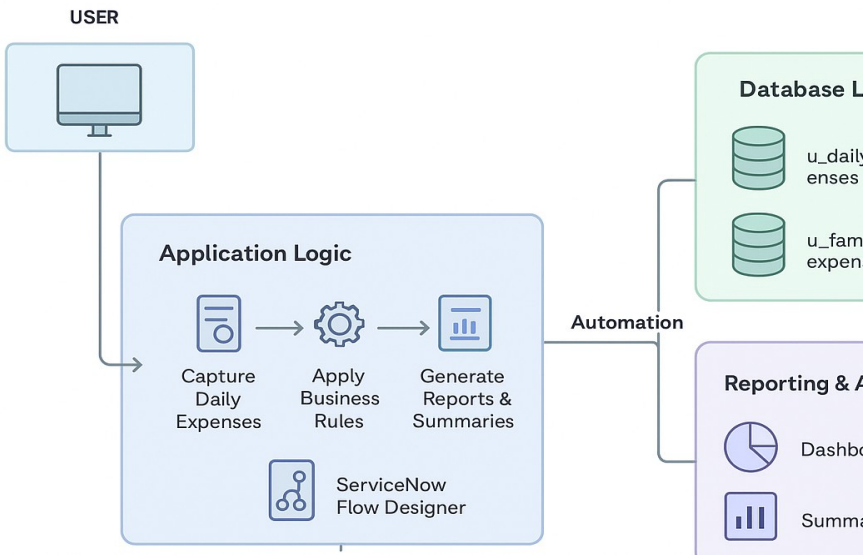
Project Design Phase-II

Technology Stack (Architecture & Stack)

Date	02 NOV 2025
Team ID	NM2025TMID01562
Project Name	Calculating Family Expenses Using ServiceNow
Maximum Marks	4 Marks

Technical Architecture:

This project uses ServiceNow's cloud-based low-code platform to automate family expense management. It includes two primary tables — Family Expenses and Daily Expenses — linked through relationships and maintained by Business Rules and Workflows for automated calculations. The system allows users to input expenses, categorize them, and visualize totals via dashboards.



Flow Summary:

1. User submits daily expense → stored in Daily Expenses table.
2. Business Rule triggers → updates Family Expenses total.
3. Workflow checks budget limit → triggers notification if exceeded.
4. Data stored securely in ServiceNow's cloud database.
5. Dashboard displays total and categorized expense summary.

Table-1: Components & Technologies

S.No	Component	Description	Technology
1	User Interface	Interface for adding and viewing expenses	ServiceNow Web UI
2	Application Logic-1	Creates and updates Family Expenses automatically	ServiceNow Business Rule
3	Application Logic-2	Validates mandatory fields and budget thresholds	ServiceNow Flow Designer
4	Application Logic-3	Handles categorization and data validation	ServiceNow Form Design
5	Database	Stores Family and Daily Expense data	ServiceNow Tables
6	Cloud Hosting	Hosted and managed in the ServiceNow Cloud	ServiceNow SaaS Platform
7	Notifications	Sends alerts when expenses exceed budget	ServiceNow Notification Service
8	External APIs	Optional integration with bank or finance apps	REST API in ServiceNow

Table-2: Application Characteristics

S.No	Characteristic	Description	Technology
1	Security	Role-based access for family members	ACLs, Scoped Apps
2	Scalability	Can handle multiple users and large data	ServiceNow Cloud Architecture
3	Availability	Always available through ServiceNow hosting	Load-balanced Instances
4	Usability	Simple and clear UI for all users	Form-based Design
5	Performance	Optimized with background scripts and indexed tables	GlideRecord, Background Scripts