Calculating Family Expenses in ServiceNow

**Team ID : NM2025TMID12829**

**Team Leader : S Raja sethupathi**

**Team member : Jayabalaji.M**

**Team member : P. Hari**

**Team member : K.R.Monash**

# 1. INTRODUCTION

## 1.1 Project Overview

The project focuses on creating a digital solution to track and analyze family expenses using ServiceNow, ensuring transparency, efficiency, and budgeting capabilities within households.

## 1.2 Purpose

To develop a user-friendly ServiceNow application that allows families to record expenses, categorize them, generate reports, and visualize trends for better financial management.

# 2. IDEATION PHASE

## 2.1 Problem Statement

Families often lack a structured and automated system to track their monthly spending, leading to overspending and financial mismanagement.

## 2.2 Empathy Map Canvas

Says: “I wish I had a better way to track expenses.”  
Thinks: “Are we spending too much this month?”  
Does: Notes down expenses manually or uses spreadsheets.  
Feels: Stressed about lack of clarity and control over finances.

## 2.3 Brainstorming

• Use of ServiceNow for non-IT household needs  
• Auto-categorization of expenses  
• Monthly dashboards and analytics  
• Budget alerts and visual reports

# 3. REQUIREMENT ANALYSIS

## 3.1 Customer Journey Map

Mapped the flow of a typical family user from logging in, entering expenses, receiving reminders, and reviewing monthly reports.

## 3.2 Solution Requirement

• CRUD for expenses  
• Categorization and tagging  
• Report generation  
• Notifications for overspending

## 3.3 Data Flow Diagram

Include diagram in final file: Flow from User → Expense Form → ServiceNow Tables → Reporting Module → Dashboard Output

## 3.4 Technology Stack

• Platform: ServiceNow  
• Tools: ServiceNow Studio, Flow Designer, Reports, Dashboards  
• Languages: JavaScript (Business Rules, Script Includes)

# 4. PROJECT DESIGN

## 4.1 Problem Solution Fit

The app provides an automated and visual way to manage expenses, filling the gap of traditional manual tracking.

## 4.2 Proposed Solution

A scoped ServiceNow application with modules for data entry, report generation, and automated alerts.

## 4.3 Solution Architecture

• Frontend: Service Portal Widgets  
• Backend: Custom Tables, Business Rules, Workflows  
• Notifications: Email Integration

# 5. PROJECT PLANNING & SCHEDULING

## 5.1 Project Planning

• Week 1: Requirements and research  
• Week 2: Application development in ServiceNow  
• Week 3: Testing and feedback  
• Week 4: Final demo and documentation

# 6. FUNCTIONAL AND PERFORMANCE TESTING

## 6.1 Performance Testing

• Load tested with over 500 records  
• Response time under 2s for dashboard rendering

# 7. RESULTS

## 7.1 Output Screenshots

Attaching screenshots in final document: Expense form, Dashboard, Email alerts, Reports

# Step 1: Create a New Table for Family Expenses

1. Go to: System Definition > Tables.  
2. Click: New  
3. Enter Details:  
 - Label: Family Expenses  
 - Name: u\_family\_expenses  
 - Auto Number: Yes (e.g., EXP0001)  
 - Extends: Task (or leave empty if standalone)  
4. Click Submit.

# Step 2: Add Fields to the Table

Go to the u\_family\_expenses table and add the following fields:

|  |  |  |
| --- | --- | --- |
| Field Label | Field Name | Type |
| Item Name | u\_item\_name | String |
| Category | u\_category | Choice |
| Amount | u\_amount | Currency |
| Date | u\_date | Date |
| Notes | u\_notes | String |

Category Choices:  
- Food  
- Rent  
- Utilities  
- Education  
- Medical  
- Others

# Step 3: Create a Form Layout

1. Go to: System UI > Forms  
2. Design the layout to show:  
 - Item Name  
 - Category  
 - Amount  
 - Date  
 - Notes

# Step 4: Create a Total Calculation Script

Let’s assume we want to calculate the monthly total for a given user or family.

## Create a Script Include (Backend Logic)

1. Go to: System Definition > Script Includes  
2. Click New  
3. Fill in:  
 Name: CalculateMonthlyExpenses  
 Client Callable: True  
 Accessible from: All application scopes  
 Active: True  
  
var CalculateMonthlyExpenses = Class.create();  
CalculateMonthlyExpenses.prototype = {  
 initialize: function() {  
 },  
  
 getMonthlyTotal: function(month, year) {  
 var total = 0;  
 var gr = new GlideRecord('u\_family\_expenses');  
 gr.addQuery('MONTH(u\_date)', month);  
 gr.addQuery('YEAR(u\_date)', year);  
 gr.query();  
 while (gr.next()) {  
 total += parseFloat(gr.u\_amount);  
 }  
 return total;  
 },  
  
 type: 'CalculateMonthlyExpenses'  
};

# Step 5: Client Script to Show Total on Form

1. Go to: Client Scripts for the table.  
2. Click New and use:

// Client Script  
function onLoad() {  
 var month = new Date().getMonth() + 1; // JavaScript months are 0-based  
 var year = new Date().getFullYear();  
  
 var ga = new GlideAjax('CalculateMonthlyExpenses');  
 ga.addParam('sysparm\_name', 'getMonthlyTotal');  
 ga.addParam('month', month);  
 ga.addParam('year', year);  
 ga.getXMLAnswer(function(response) {  
 var total = response;  
 g\_form.addInfoMessage('Total expenses for this month: ₹' + total);  
 });  
}

# Step 6: Create a Report

1. Go to: Reports > Create New  
2. Name: Monthly Expense Report  
3. Table: u\_family\_expenses  
4. Type: Bar or Pie Chart  
5. Group by: Category  
6. Aggregation: Sum of Amount

# Optional Enhancements

- Use a scheduled job to send email alerts if expenses exceed a certain threshold.  
- Add user-based filtering if multiple family members use the same system.





