



ST.JOSEPH'S COLLEGE OF ENGINEERING AND TECHNOLOGY , THANJAVUR

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CALCULATING FAMILY EXPENSES USING SERVICE NOW

TEAM ID: NM2025TMID07857

TEAM SIZE: 3

TEAM LEADER: SRUTHIKA S

TEAM MEMBER: SHANMUGA PRIYA S

TEAM MEMBER: SWATHY P

PROBLEM STATEMENT:

Managing household expenses manually often leads to errors, lack of clarity, and difficulty in budgeting. Families face challenges in tracking multiple categories of expenses such as food, rent, utilities, education, and healthcare. There is a need for a digital solution that automates expense calculation and provides real-time financial insights for better management.

OBJECTIVE:

The main objective of this project is to develop a comprehensive Family Expense Calculation System using ServiceNow.

This system aims to:

- Help users record, categorize, and track their daily or monthly expenses.
- Provide budget-setting and real-time expense tracking.
- Offer reporting and visualization tools for better financial decision-making.
- Ensure a user-friendly and scalable platform adaptable to families of different sizes and income levels.

SKILLS REQUIRED:

1. ServiceNow Development (Table creation, Form design, Flow Designer, UI Builder)
2. Scripting in ServiceNow using JavaScript/Glide Script
3. Data Management & Reporting (Report Designer, Performance Analytics)
4. Problem Solving and Analytical Thinking
5. User Interface Design

TASK INITIATION:

- ServiceNow Development (Table creation, Form design, Flow Designer, UI Builder)
- Scripting in ServiceNow using JavaScript/Glide Script
- Data Management & Reporting (Report Designer, Performance Analytics)
- Problem Solving and Analytical Thinking
- User Interface Design

FEATURES:

- Add, edit, and delete expense records.
- Categorize expenses (food, rent, travel, etc.).
- Automatic total expense calculation.
- Monthly and yearly expense reports.
- Budget limit alert system.
- Dashboard with charts and graphs.
- Secure data storage in ServiceNow.
- User-friendly interface for easy tracking.

MODEL IMPLEMENTED:

1. Data Model

- Table 1: u_family_expense → Stores family name, total expense, and monthly budget.
- Table 2: u_daily_expense → Stores date, amount, category, and description of each expense.
- Reference field links both tables (Family ID).
- Ensures structured data storage and easy retrieval.

2. Process Model

- User enters expenses through a ServiceNow form.
- Data is saved automatically in the u_daily_expense table.
- Flow Designer or Business Rule calculates total monthly expense.
- System compares total with budget and triggers alert if limit exceeded.
- Reports and dashboards update automatically with new data.

3. Logical Model

- Uses Flow Designer for automation and notifications.
- Business Rules handle calculations and validations.
- UI Policies control form visibility and mandatory fields.
- Client Scripts used for real-time calculations in the form.

4. Functional Model

- Expense entry and update.
- Monthly budget setup.
- Automatic total computation.
- Alert message or email when expenses exceed budget.
- Report and dashboard generation.

5. Output Model

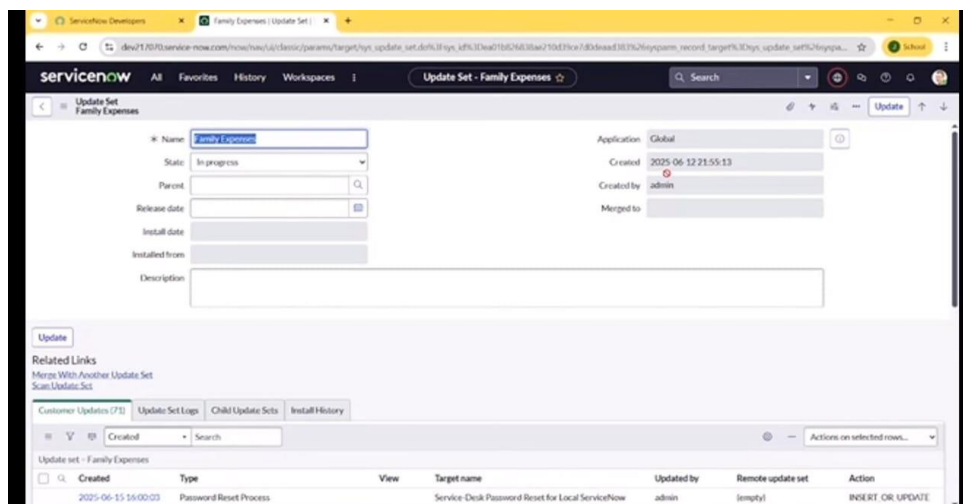
- Dashboard showing total and category-wise expenses.
- Pie chart and bar chart visualizations.
- Monthly and yearly summary reports.
- Budget alert notifications.

IMPLEMENTATION STEPS:

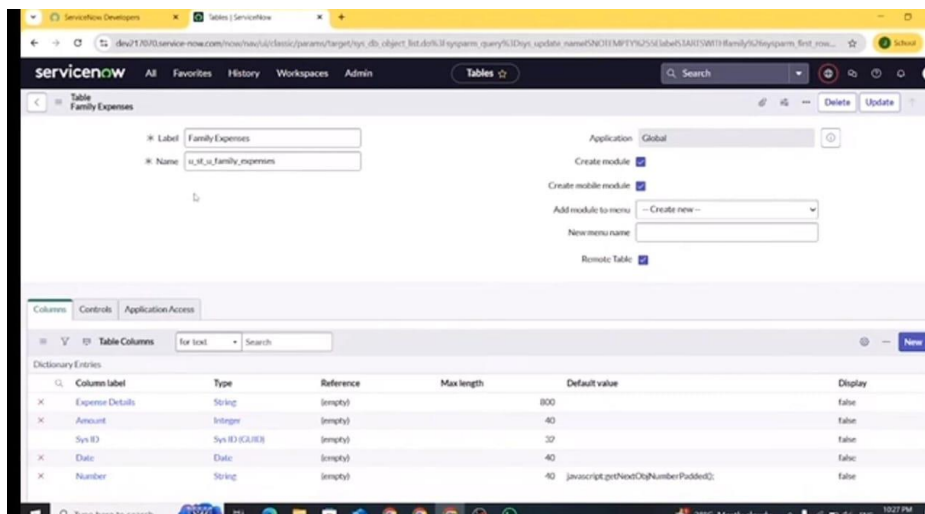
STEP 1: Creation of New Update Set:



STEP 2: Creation of Table:

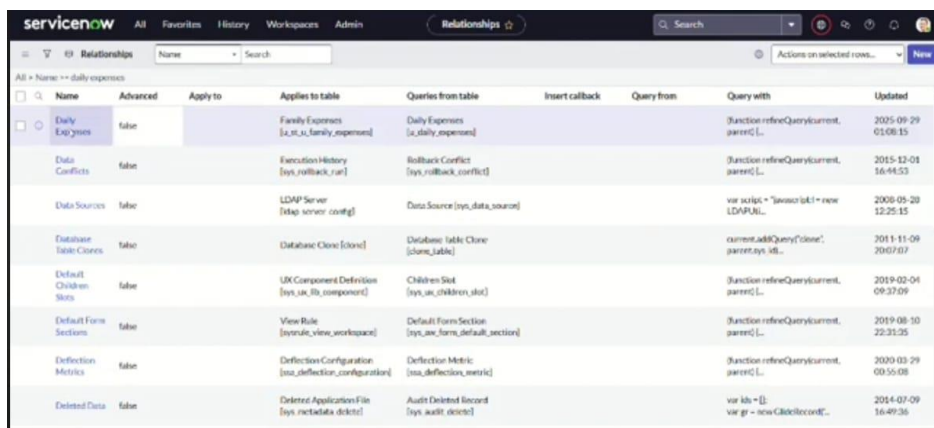


STEP 3:Creation of table(Family Expenses)



The screenshot shows the ServiceNow 'Table' configuration page for 'Family Expenses'. The 'Label' is 'Family Expenses' and the 'Name' is 'u_at_u_family_expenses'. The 'Application' is set to 'Global'. The 'Create module' checkbox is checked. The 'Create mobile module' checkbox is also checked. The 'Add module to menu' dropdown is set to 'Create new'. The 'New menu name' field is empty. The 'Remote Table' checkbox is checked. The 'Columns' tab is active, showing a list of columns with their labels, types, references, max lengths, default values, and display settings.

Column label	Type	Reference	Max length	Default value	Display
Expense Details	String	(empty)	800		false
Amount	Integer	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Date	Date	(empty)	40		false
Number	String	(empty)	40	javascript.getNextObjNumberPadded();	false

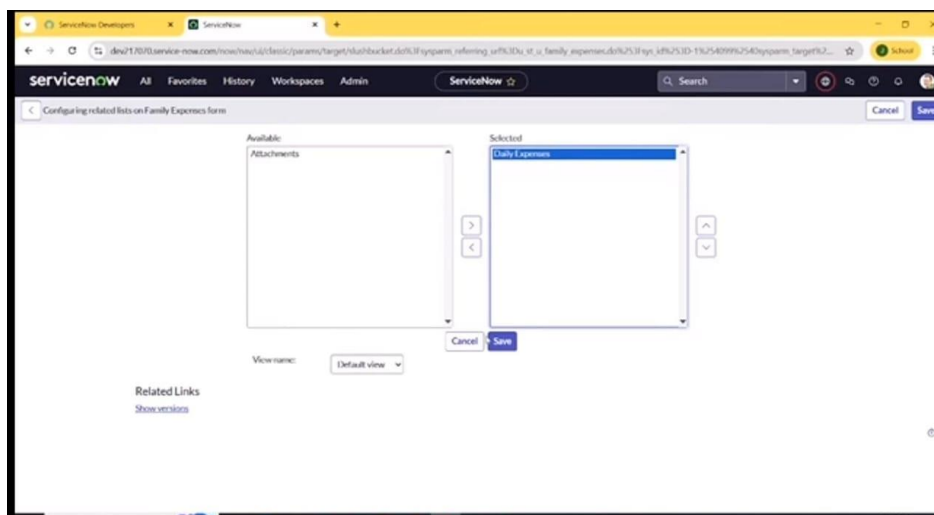


The screenshot shows the ServiceNow 'Relationships' table. The table lists various relationships between tables, including 'Daily Expenses', 'Data Conflicts', 'Data Sources', 'Database Table Clones', 'Default Children Skits', 'Default Form Sections', 'Deflection Metrics', and 'Deleted Data'. Each row includes columns for Name, Advanced, Apply to, Applies to table, Queries from table, Insert callback, Query from, Query with, and Updated.

Name	Advanced	Apply to	Applies to table	Queries from table	Insert callback	Query from	Query with	Updated
Daily Expenses	false		Family Expenses [u_at_u_family_expenses]	Daily Expenses [u_at_u_family_expenses]			function refineQuery(current, parent) {	2025-09-29 01:08:15
Data Conflicts	false		Execution History [sys_rollback_rar]	Rollback Conflict [sys_rollback_conflict]			function refineQuery(current, parent) {	2025-12-01 16:44:53
Data Sources	false		LDAP Server [ldap_server_config]	Data Source [sys_data_source]			var script = "jsscsc {url = new LDAPURL,	2008-05-20 12:25:15
Database Table Clones	false		Database Clone [clone]	Database Table Clone [clone_table]			current.addQuery('clone', parent.sys_id,	2011-11-09 20:07:07
Default Children Skits	false		UX Component Definition [sys_ux_its_component]	Children Skit [sys_ux_children_skit]			function refineQuery(current, parent) {	2019-02-04 09:37:09
Default Form Sections	false		View Rule [sysrule_view_workspace]	Default Form Section [sys_ux_form_default_section]			function refineQuery(current, parent) {	2019-08-10 22:31:25
Deflection Metrics	false		Deflection Configuration [sys_deflection_configuration]	Deflection Metric [sys_deflection_metric]			function refineQuery(current, parent) {	2020-03-29 00:55:08
Deleted Data	false		Orchestration Application File [sys_mactadata_delete]	Audit Deleted Record [sys_audit_delete]			var kb = {} var gr = new GlobalRecord();	2014-07-09 16:49:36

STEP 3:Creation of relationship

Configuring Related List on Family Expense



The screenshot shows the 'Configuring related lists on Family Expenses form' dialog. The 'Available' list contains 'Attachments'. The 'Selected' list contains 'Daily Expenses'. The 'View name' dropdown is set to 'Default view'. The 'Cancel' and 'Save' buttons are visible at the bottom right.

STEP 4: Creation of business roles

The screenshot shows the ServiceNow Business Rule configuration page for a rule named 'Family Expenses BR'. The rule is set to 'Global' application, 'Active', and 'Advanced'. The 'Table' field is set to 'Daily Expenses [u:daily_expenses]'. The 'When to run' section is set to 'Before' with an 'Order' of 500. The 'Filter Conditions' section is empty, and the 'Web Services' section is also empty. The 'Actions' tab is selected, showing a list of actions: 'Insert' (checked), 'Update' (checked), 'Delete' (unchecked), and 'Query' (unchecked). The 'Rule conditions' section is empty.

STEP 5: Configure the Relationship

The screenshot shows the ServiceNow Relationship configuration page for a relationship named 'Daily Expenses'. The relationship is set to 'Global' application, 'Advanced', and 'Applies to table' is set to 'Family Expenses [u:it.u:family_expenses]'. The 'Queries from table' field is set to 'Daily Expenses [u:daily_expenses]'. A note states: 'This script refines the query in current that will populate the related list. For more information about it, its parameters and control variables, see the documentation. See also the article about the recommended form of the script.' The 'Query with' section is set to 'Turn on ECMAScript 2021 (ES12) mode'. The script content is as follows:

```
1 (function refineQuery(current, parent) {
2
3
4 // Add your code here, such as current.addQuery(field, value);
5
6 current.addQuery("u:state", parent.u:state);
7
8 current.query();
9
10
11 })(current, parent);
12
```

The 'Update' and 'Delete' buttons are visible at the bottom of the page. The 'Related Links' section is empty.

OUTCOME:

The project “Calculation of Family Expenses using ServiceNow” successfully developed an automated system to manage and monitor family expenses efficiently. The application allows users to record daily expenses, categorize them, and automatically calculate monthly and yearly totals. Reports and dashboards were created to display category-wise spending and overall budget usage. The system also provides alerts when expenses exceed the set budget limit, helping users control their spending. This project improved the accuracy, speed, and organization of expense tracking while providing a user-friendly interface. It also enhanced practical knowledge of ServiceNow tools such as table creation, form design, Flow Designer, and report generation.

CONCLUSION:

The Family Expense Calculation System on ServiceNow offers an easy and efficient way to manage household finances. With features like expense tracking, budget setting, and reporting, it helps users monitor spending and make informed financial decisions. The system's scalability and user-friendly interface ensure it supports families of all sizes, promoting better financial planning and control.