

Calculating Family Expenses using Service Now

ServiceNow Virtual Internship - SmartBridge

Author: Arulkumar L

Date:09 December 2025

GitHub Link : <https://github.com/Arulkumar-L/Calculating-Family-Expenses-using-Service-Now/>

Abstract

This project implements a Family Expenses Calculation System on the ServiceNow platform. It enables families to record daily expenses, automatically aggregate totals per day, and provide a clear form-based UI for data entry. The system uses custom tables, business rules, and relationships to ensure that when daily expense records are inserted or updated, the corresponding aggregated family expense record is created or updated automatically. This reduces manual reconciliation and provides near real-time insights into household spending.

Problem Statement

Many households lack a centralized, low-effort way to record and aggregate daily expenses. Manual tracking is error-prone and time-consuming, often resulting in incomplete records and poor visibility into spending patterns. The goal is to provide a simple, maintainable system built on ServiceNow that enables easy recording of daily expenses and automated aggregation so families can monitor and manage their finances.

Objectives

- Provide a simple interface for recording daily family expenses.
- Automatically aggregate expenses by date to produce daily totals.
- Store descriptive comments and metadata (family member name, comments).
- Ensure the system is maintainable and extensible on the ServiceNow platform.

Methodology

The solution leverages ServiceNow's configuration capabilities: custom tables (Family Expenses and Daily Expenses), forms designed via the Form Designer, a Business Rule to perform aggregation logic on insert/update, and a Relationship / Related List to surface child daily expense records under their aggregated parent record. The development process included: requirements gathering, instance setup, table and field design, form configuration, scripting a server-side Business Rule, and testing.

Tables & Fields

Family Expenses (u_family_expenses)

Field	Type	Description
u_number	String	Auto-generated identifier, format prefix 'MFE' + padded number
u_date	Date	The date for which expenses are aggregated
u_amount	Integer	Total amount for the date
u_expense_details	String	Concatenated descriptions of daily expenses for quick review

Daily Expenses (u_daily_expenses)

Field	Type	Description
u_number	String	Auto-generated identifier
u_date	Date	Date of the expense
u_expense	Integer	Amount spent for the entry
u_family_member_name	Reference	Reference to a family member table or free text (max 800)
u_comments	String	Details or purpose of the expense

Form Design

Forms were created using ServiceNow's Form Designer. Key points:

- Number fields were set to read-only with dynamic default generation.
- Mandatory fields: Date and Amount for Family Expenses; Date and Family Member Name for Daily Expenses.
- Related List configuration allows Daily Expenses to be displayed within a Family Expenses record for easier navigation.

Business Rule

A server-side Business Rule named 'Family Expenses BR' runs on the Daily Expenses table for insert and update events. Its role is: when a daily expense is added or edited, find (or create) the Family Expenses record for the same date and update the aggregated total and expense details. This keeps the aggregate data consistent without manual intervention. The logic uses GlideRecord queries and handles both update and insert scenarios.

Business Rule Script

```
(function executeRule(current, previous /*null when async*/) { var FamilyExpenses = new
GlideRecord('u_family_expenses'); FamilyExpenses.addQuery('u_date', current.u_date);
FamilyExpenses.query(); if (FamilyExpenses.next()) { FamilyExpenses.u_amount += current.u_expense;
FamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.u_expense +
"/-"; FamilyExpenses.update(); } else { var NewFamilyExpenses = new
GlideRecord('u_family_expenses'); NewFamilyExpenses.u_date = current.u_date;
NewFamilyExpenses.u_amount = current.u_expense; NewFamilyExpenses.u_expense_details += ">" +
current.u_comments + ":" + "Rs." + current.u_expense + "/-"; NewFamilyExpenses.insert(); }
}) (current, previous);
```

Relationship

A Relationship was configured to link Daily Expenses records to Family Expenses aggregations using the date field as the connecting key. This allows the related list shown on the Family Expenses form to automatically display all Daily Expenses for that date. A refineQuery script example was used to add this matching logic.

Relationship refineQuery Script

```
(function refineQuery(current, parent) { current.addQuery('u_date', parent.u_date);
current.query(); }) (current, parent);
```

Outputs & Test Cases

Example test cases executed:

- Insert a Daily Expense for 2025-12-01 of Rs. 150 with comment 'Grocery' -> Family Expenses record for 2025-12-01 created with total 150.
- Insert another Daily Expense for the same date of Rs. 50 with comment 'Snacks' -> Family Expenses u_amount updated to 200 and u_expense_details appended.
- Update an existing Daily Expense amount -> Family Expenses aggregation logic adjusts totals accordingly (additional logic can be added to handle subtraction on update).

The provided screenshots show table definitions, sample records, number maintenance configuration, relationship setup, and business rule evidence.

Screen Shots :

Daily Expenses number maintainence

This screenshot shows the ServiceNow configuration interface for a 'Number' object named 'DFE'. The configuration details are as follows:

- * Table: Daily Expenses
- Prefix: DFE
- * Number: 1,000
- Application: Global
- Number of digits: 7

Below the configuration, there are 'Update' and 'Delete' buttons. Under 'Related Links', there is a link to 'Show Counter'. The bottom of the screen shows a Windows taskbar with various pinned icons and system status.

Daily Expenses record

This screenshot shows the ServiceNow list view for 'Daily Expenses'. The table displays the following data:

Number	Comments	Date	Expense	Family Member Name
DFE0001010	Tutition Fees	2025-12-08	2,000	Adela Cervantsz
DFE0001012	Groceries	2025-12-08	500	Andrew Och
DFE0001009	Fuel for bike	2025-12-08	500	Andrew Och
DFE0001011	Electricity Bill	2025-12-08	1,000	Andrew Och

The bottom of the screen shows a Windows taskbar with various pinned icons and system status.

Daily Expenses table

The screenshot shows the 'Daily Expenses - Create DFE0001013' page. The 'Number' field contains 'DFE0001013'. The 'Date' field is empty. The 'Family Member Name' field is empty. The 'Expense' field is empty. There is a 'Comments' text area below the date field. A 'Submit' button is at the bottom left.

Family expenses output

The screenshot shows the 'Family Expenses' list page. It displays a single expense record with Number 'MFE0001010', Date '2025-12-08', and Amount '4,000'. The 'Expense Details' section lists: >Fuel for bike: Rs.500/-, >>Tuition Fees: Rs.2000/-, >Electricity Bill: Rs.1000/-, >Groceries: Rs.500/-. Below this, a table lists four daily expenses:

Number	Comments	Date	Expense	Family Member Name
DFE0001010	Tuition Fees	2025-12-08	2,000	Adela Cervantsz
DFE0001012	Groceries	2025-12-08	500	Andrew Och
DFE0001009	Fuel for bike	2025-12-08	500	Andrew Och
DFE0001011	Electricity Bill	2025-12-08	1,000	Andrew Och

Family expenses record

The screenshot shows a ServiceNow web interface for managing family expenses. The title bar has multiple tabs open, including 'ServiceNow Developers', 'Family Expenses BR | Business Unit', 'Family Expenses | ServiceNow', 'MFE0001010 | Family Expenses', and 'ServiceNow'. The main content area displays a list of expenses with the following columns: Number, Amount, Date, and Expense Details. One expense is listed:

Number	Amount	Date	Expense Details
MFE0001010	4,000	2025-12-08	>Fuel for bike: Rs.500/- >Tuition Fees...

At the bottom, there is a navigation bar with icons for search, file, and other applications, along with system status indicators like weather (24°C, sunny) and date (09-12-2025).

Family expenses table

The screenshot shows a ServiceNow web interface for creating a new family expense record. The title bar has tabs for 'ServiceNow Developers', 'Family Expenses BR | Business Unit', 'Create MFE0001011 | Family Expenses', 'MFE0001010 | Family Expenses', and 'ServiceNow'. The main content area is a form titled 'Family Expenses New record' with fields for Number (MFE0001011), Date (marked with a red asterisk), and Amount (marked with a red asterisk). There is also a large 'Expense Details' text area and a 'Submit' button at the bottom left.

At the bottom, there is a navigation bar with icons for search, file, and other applications, along with system status indicators like weather (24°C, sunny) and date (09-12-2025).

Business rule

The screenshot shows the ServiceNow Business Rule - Family Expenses BR page. The title bar has tabs for ServiceNow Developers, Family Expenses BR | Business Rule, MFE0001010 | Family Expenses, MFE0001010 | Family Expenses, and ServiceNow. The main content area is titled "Business Rule - Family Expenses BR". It shows a script editor with the following code:

```
1 (function executeRule(current, previous /*null when async*/) {
2
3     var FamilyExpenses = new GlideRecord('u_family_expenses');
4     FamilyExpenses.addQuery('u_date', current.u_date);
5     FamilyExpenses.query();
6
7     var line = "\n" + current.u_comments + ": Rs." + current.u_expense + "-";
8
9     if (FamilyExpenses.next()) {
10        FamilyExpenses.u_amount += current.u_expense;
11        FamilyExpenses.u_expense_details += line;
12        FamilyExpenses.update();
13    } else {
14        var NewFamilyExpenses = new GlideRecord('u_family_expenses');
15        NewFamilyExpenses.u_date = current.u_date;
16        NewFamilyExpenses.u_amount = current.u_expense;
17        NewFamilyExpenses.u_expense_details = line; // initialize instead of +=
18        NewFamilyExpenses.insert();
19    }
20
21 })(current, previous);
```

The "When to run" tab is selected. The "Actions" and "Advanced" tabs are also visible. The bottom status bar shows a trending video notification for "The Hunger Games" and system status indicators.

family expenses number maintainence

The screenshot shows the ServiceNow Number - MFE page. The title bar has tabs for ServiceNow Developers, MFE | Number | ServiceNow, Family Expenses | ServiceNow, MFE0001010 | Family Expenses, Daily Expenses | ServiceNow, and ServiceNow. The main content area is titled "Number - MFE". The configuration details are as follows:

- * Table: Family Expenses
- Prefix: MFE
- * Number: 1,000
- Application: Global
- Number of digits: 7

Buttons for "Update" and "Delete" are at the bottom left. A "Related Links" section includes a "Show Counter" link. The bottom status bar shows a battery icon with 3 LCI -1.03% and system status indicators.

Relationship

The screenshot shows the ServiceNow interface for managing relationships between tables. The top navigation bar includes tabs for 'ServiceNow Developers', 'Daily Expenses | Relationship', 'Family Expenses | ServiceNow', 'MFE0001010 | Family Exp...', and 'Daily Expenses | ServiceNow'. The main title is 'Relationship - Daily Expenses'. The page displays a form for creating a new relationship named 'Daily Expenses' with 'Global' application, 'Family Expenses [u_family_expenses]' as the target table, and 'Daily Expenses [u_daily_expenses]' as the source table. A note at the bottom 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](#)'. Below the note is a code editor containing the following ECMAScript code:

```
1 (function refineQuery(current, parent) {  
2   // Add your code here, such as current.addQuery(field, value);  
3   current.addQuery('u_date',parent.u_date);  
4   current.query();  
5 })(current, parent);
```

At the bottom of the page are 'Update' and 'Delete' buttons, along with a 'Related Links' section containing a 'Run Point Scan' link. The status bar at the bottom right shows system information: LCI -1.03%, ENG IN, 10:32, and the date 09-12-2025.

Conclusion

The Family Expenses Calculation System demonstrates how ServiceNow can be extended beyond ITSM to deliver simple, practical business applications. It reduces manual work, provides a single source of truth for daily household spending, and is suitable for further enhancements such as user authentication, family member catalog, detailed reporting, and dashboards.

Future Scope

Potential improvements:

- Add transaction-level edit tracking to adjust aggregated totals correctly on update/delete.
- Implement a Family Members table and reference field to enforce data integrity.
- Create scheduled reports and Performance Analytics widgets for visualization.
- Integrate export options (CSV/PDF) and notifications for budget thresholds.
- Add validation rules and UI Policies to improve data quality.

