

Calculating Family Expenses using ServiceNow

Team ID: NM2025TMID06530

Team Size: 5

Team Leader: SUDHARSHAN R

Team Member: SRI RENGANATHAN T

Team Member: SUDHAN K

Team Member: SUNDAR S

Team Member: SIVABALAN N

1. Ideation Phase

In the ideation phase, the team identified a common household problem — the lack of an efficient, automated way to manage and analyze family expenses. The idea was to create a centralized platform on **ServiceNow** to help users record, categorize, and analyze their spending. Brainstorming focused on improving financial visibility, preventing overspending, and providing real-time reports. The concept evolved into building a “**Family Expense Management System**”, enabling automation for recurring expenses and personalized dashboards for better decision-making.

2. Project Planning Phase

The planning phase involved defining the **project objectives, milestones, and roles**.

- **Tools used:** ServiceNow App Engine Studio, Flow Designer, and Reporting.
- **Milestones:**
 1. Setting up the ServiceNow instance.
 2. Creating update sets for version tracking.
 3. Designing the Family Expenses and Daily Expenses tables.
 4. Establishing relationships between tables.
 5. Implementing business rules for automation.
 6. Configuring related lists and reports.
- **Responsibilities:** Each team member was assigned tasks like table creation, form design, script development, and testing.
This phase ensured a clear roadmap and timeline for smooth implementation.

3. Project Design Phase

In this phase, the **architecture and interface design** were finalized.

- The team created **two core tables**:
 - *Family Expenses* (for monthly or weekly totals)
 - *Daily Expenses* (for individual expense entries).
 - **Relationships** were defined to link daily entries with family totals.
 - The **form design** was customized for user-friendly data entry, with fields like *Date*, *Amount*, *Expense Details*, and *Family Member Name*.
 - Business rules were scripted to **automatically update totals** in the Family Expenses table whenever new daily expenses were added.
- This phase focused on creating an intuitive user experience and logical database flow.

4. Requirement Analysis

The requirement analysis identified both **functional** and **technical** needs:

Functional Requirements:

- Users should be able to add and view daily and total expenses.
- The system must auto-update totals and expense details.
- The app should support multi-user roles for family members.
- Real-time reports and dashboards should be generated.

Technical Requirements:

- ServiceNow Developer Instance.
- App Engine Studio for table and form creation.
- Flow Designer for automation.
- Business Rules for backend scripting.
- Reporting & Dashboard module for visualization.

5. Performance Testing

After development, the project underwent **performance and functionality testing** to ensure reliability and efficiency.

- **Testing parameters included:**
 - Speed of data entry and update.
 - Accuracy of total calculations.
 - System responsiveness during concurrent user operations.
 - Validation of business rules and relationships.
 - Dashboard data refresh and report generation speed.

Problem Statement:

There is currently no efficient system in place to track, categorize, and analyze family expenses in a way that is integrated, automated, and accessible in real-time. This leads to poor budget planning, missed payments, overspending, and a lack of financial visibility.

Objective:

To design and implement a solution using **ServiceNow** that allows family members to:

- Input and categorize expenses** easily via a user-friendly interface or mobile portal.
- Automate recurring expenses** (e.g., rent, subscriptions).
- Track total and category-wise spending** on a monthly/weekly basis.
- Set spending limits and receive alerts** when thresholds are crossed.
- Generate reports and dashboards** to visualize trends and support better financial decisions.
- Share visibility and responsibilities** across family members via roles or user groups.

Skills:

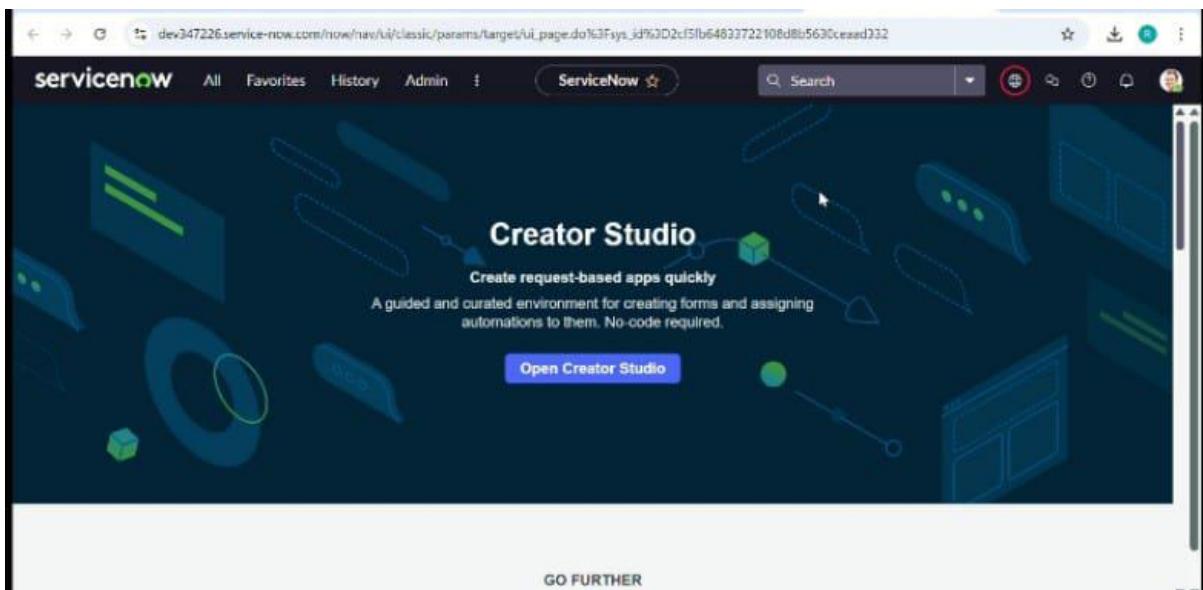
- ServiceNow App Development** (App Engine Studio)
- Custom Tables & Data Modeling**
- Form Design & UI Customization**
- Flow Designer** (for automation)
- Business Rules & Scripting**
- Reporting & Dashboards**
- User Roles & Access Control (ACLs)**

TASK INITIATION

Milestone 1 : Instance

Activity 1: Setting up ServiceNow instance

- Sign up for a developer account on the ServiceNow Developer site "<https://developer.servicenow.com>".
- Once logged in, navigate to the "Personal Developer Instance" section.
- Click on "Request Instance" to create a new ServiceNow instance.
- Fill out the required information and submit the request.
- You'll receive an email with the instance details once it's ready.
- Log in to your ServiceNow instance using the provided credentials.
- Now you will navigate to the ServiceNow.



Milestone 2: New Update Set

Activity 1: Creation of New Update Set

1. Go to All >> In the filter search for Local Update set > click on New.
2. Enter the Details as:
Name : Family Expenses
3. Then click on Submit and Make current.

The screenshot shows the 'Update Set - Create Educational org...' page. The 'Name' field is populated with 'Family Expenses'. The 'State' dropdown is set to 'In progress'. There are fields for 'Parent' and 'Release date'. A large 'Description' text area is empty. At the bottom, there are two buttons: 'Submit' and 'Submit and Make Current'.

Milestone 3: Table(Family Expenses)

Activity 1: Creation of Family Expenses Table

1. Go to All >> In the filter search for Tables > click on New.
2. Enter the Details:
Label : Family Expenses
Name : Auto-Populated
New menu name : Family Expenditure
3. Go to the Header and right click there >> click on Save.

The screenshot shows the 'Table - New Record' page. The 'Label' field is 'Family Expenses' and the 'Name' field is 'u_family_expenses'. There is a 'Extends table' field with a search icon. On the right, there are several checkboxes: 'Create module' (checked), 'Create mobile module' (checked), 'Add module to menu' (dropdown set to '-- Create new --'), 'New menu name' (text input 'Family Expenditure'), and 'Remote Table' (checkbox). A note at the top states: 'ServiceNow recommends creating custom tables in scoped applications. To learn more about creating scoped applications, click [here](#)'.

Activity 2: Creation of Family Expenses Table

1. Near Columns Double click near insert a new row.

2. Give the details as:

 Column label : Number

 Type : String

3. Double click on insert a new row again.

4. Give the details as:

 Column label : Date

 Type : Date

5. Double click on insert a new row again

6. Give the details as:

 Column label : Amount

The screenshot shows the ServiceNow dictionary entry for 'Number'. The top navigation bar includes 'Dictionary Entry', 'Number View: Advanced', 'Delete Column', and 'Update'. Below the navigation is a 'Display' checkbox. A message bar states: 'Alters the behavior of a field or functionality that depends on the field. [More Info](#)'. The main area has an 'Attributes' section with a large empty text input. Below it is a tab bar with 'Choice List Specification', 'Calculated Value', and 'Default Value' (which is selected). A message says: 'The Default value specifies what value the field has when first displayed.' Underneath are fields for 'Use dynamic default' (checked), 'Dynamic default value' (set to 'Get Next Padded Number'), and 'Lookup using list' with search and refresh icons. At the bottom are 'Delete Column' and 'Update' buttons.

 Type : Integer

7. Double click on insert a new row again

8. Give the details as:

 Column label : Expense Details

 Type : String

 Max length : 800

9. Go to the Header and right click there>> click on Save.

The screenshot shows the 'Family Expenses' table in ServiceNow. The top navigation bar includes 'All', 'Favorites', 'History', 'Table - Family Expenses', 'Search', and various toolbar icons. The main area shows the 'Table Columns' view with a search bar. A message bar at the top says: 'Dictionary Entries'. Below is a table with columns: 'Column label', 'Type', 'Reference', 'Max length', 'Default value', and 'Display'. The table lists several columns: 'Updated' (Date/Time), 'Created by' (String), 'Sys ID' (Sys ID (GUID)), 'Created' (Date/Time), 'Updated by' (String), 'Updates' (Integer), 'Number' (String), 'Date' (Date), 'Amount' (Integer), and 'Expense Details' (String). The 'Max length' for 'Expense Details' is set to 800. At the bottom right of the table is a 'New' button.

Activity 3: Making Number Field an Auto-Number

1. Double click on the Number Field/Column.
2. Go down and double click on Advanced view
3. In Default Value:
 - Use dynamic default : check the box
 - Dynamic default value : Get Next Padded Number
4. Click on Update.
5. Go to All >> In the filter search for Number Maintenance >> select Number Maintenance
6. Click on New.
7. Enter the below Details:
 - Table : Family Expenses
 - Prefix : MFE
8. Click on Submit

The screenshot shows a configuration interface for a 'Number' field. At the top, there are buttons for back, forward, and search, followed by a 'Submit' button. Below these are several input fields:

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

At the bottom left is a 'Submit' button, and below it are 'Related Links' with a 'Show Counter' option. There are also some small icons at the bottom right.

Activity 4: Configure the Form

1. Go to All >> In the filter search for Family Expenses >> Open Family Expenses
2. Click on New
3. Go to the Header and right click there >> click on Configure >> Select Form Design
4. Customize or Drag Drop the form as per your requirement.
5. Make Number Read-Only Field by clicking on the gear icon and checking Read-Only
6. Make Date, Amount Mandatory Field by clicking on the gear icon and checking Mandatory
7. Click on Save.

The screenshot shows the ServiceNow Form Design interface. On the left, there's a sidebar with tabs for 'Fields' and 'Field Types', and sections for 'Filter', 'Created', 'Updated', and 'Formatters'. The main area is titled 'Form Design' and contains a section for 'Family Expenses [u_family_expenses]'. This section is set to a '2 Column' layout and contains three fields: 'Number', 'Amount', and 'Date'. Below this is another section titled 'Expense Details' with a '1 Column' layout.

Milestone 4: Table(Daily Expenses)

Activity 1: Creation ofTable(Daily Expenses)

1. Go to All > In the filter search for Tables > click on New.
2. Enter the Details:
Label : Daily Expenses
Name : Auto-Populated
Add Module to menu : Family Expenditure
3. Go to the Header and right click there>> click on Save.

The screenshot shows the ServiceNow 'Table - New Record' screen. At the top, it says 'Table - New Record' and has tabs for 'All', 'Favorites', 'History', and 'Admin'. Below that, it says 'Table New record'. There's a note: 'ServiceNow recommends creating custom tables in scoped applications. To learn more about creating scoped applications, click [here](#)'. A blue box below says: 'A table is a collection of records in the database. Each record corresponds to a row in a table, and each field on a record corresponds to a column on that table. Applications use tables and records to manage data and processes. [More Info](#)'.

On the right, there are several input fields and checkboxes:

- * Label: Daily Expenses
- * Name: u_daily_expenses
- Extends table: (empty)
- Application: Global
- Create module:
- Create mobile module:
- Add module to menu: Family Expenditure
- Remote Table:

Activity 2: Creation of Columns(Fields)

1. Near Columns Double click near insert a new row.
2. Give the details as:
 Column label : Number
 Type : String
3. Double click on insert a new row again
4. Give the details as:
 Column label : Date
 Type : Date
5. Double click on insert a new row again
6. Give the details as:
 Column label : Expense
 Type : Integer
7. Double click on insert a new row again
8. Give the details as:
 Column label : Family Member Name
 Type : Reference
 Max length: 800
9. Double click on insert a new row again
10. Give the details as:
 Column label : Comments
 Type : String
 Max length : 800

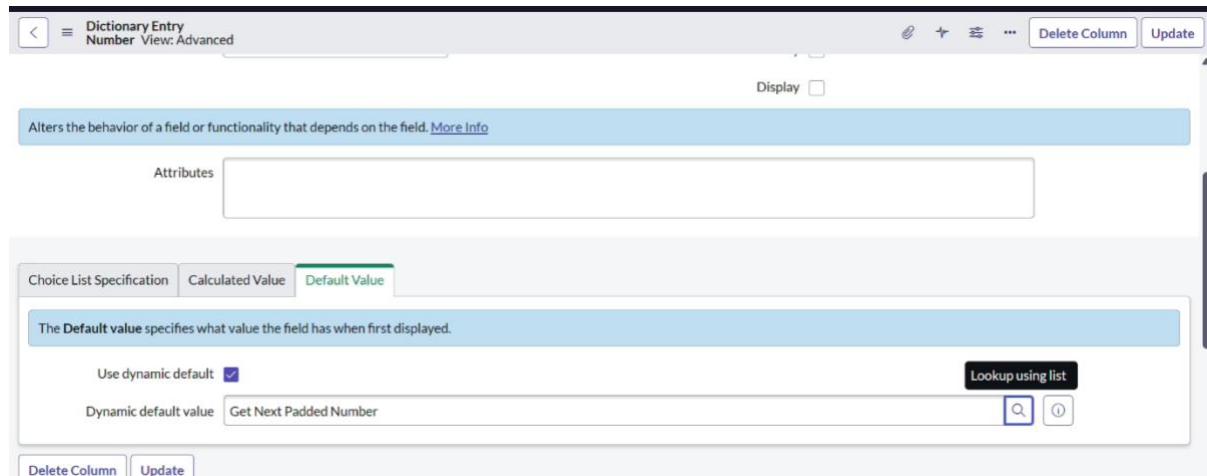
11. Go to the Header and right click there>> click on Save.

The screenshot shows the ServiceNow Table Design interface for the 'Daily Expenses' table. The 'Table - Daily Expenses' tab is selected. In the 'Dictionary Entries' section, a new column 'Number' has been added, highlighted with a blue border. The table structure includes columns for Updated by, Updates, Updated, Created by, Created, Sys ID, Number, Date, Expense, Family Member Name, and Comments. The 'Comments' column has a maximum length of 800. The 'Number' column is currently selected, as indicated by the blue border around its row.

Column label	Type	Reference	Max length	Default value
Updated by	String	(empty)	40	
Updates	Integer	(empty)	40	
Updated	Date/Time	(empty)	40	
Created by	String	(empty)	40	
Created	Date/Time	(empty)	40	
Sys ID	Sys ID (GUID)	(empty)	32	
Number	String			
Date	Date			
Expense	Integer			
Family Member Name	Reference		800	
Comments	String		800	

Activity 3: Making Number Field an Auto-Number

1. Double click on the Number Field/Column.
2. Go down and double click on Advanced view
3. In Default Value:
 - Use dynamic default : check the box
 - Dynamic default value : Get Next Padded Number
4. Click on Update.



5. Go to All >> In the filter search for Number Maintenance >> select Number Maintenance
6. Click on New.
7. Enter the below Details:
 - Table : Daily Expenses
 - Prefix : DFE
8. Click on Submit.

The screenshot shows the 'Number - New Record' form for 'Daily Expenses'. The 'Table' field is set to 'Daily Expenses', and the 'Prefix' field is set to 'DFE'. The 'Number' field contains '1,000'. Other fields include 'Application: Global' and 'Number of digits: 7'. At the bottom, there is a 'Submit' button with a hand icon, and a 'Related Links' section with a 'Show Counter' link.

Activity 4: Configure the Form

1. Go to All >> In the filter search for Daily Expenses >> Open Daily Expenses
2. Click on New
3. Go to the Header and right click there>> click on Configure >> Select Form Design
4. Customize or Drag Drop the form as per your requirement.

5. Make Number Read-Only Field by clicking on the gear icon and checking Read-Only
 6. Make Date, Family Member Name Mandatory Field by clicking on the gear icon and checking Mandatory
 7. Click on Save.

The screenshot shows the ServiceNow Form Design interface for the 'Daily Expenses [u_daily_expenses]' form. The top navigation bar includes the URL 'dev347226.service-now.com/\$ng_fd.do?sysparm_attributes=startTable:"u_daily_expenses"%2CstartView:"Default%20view"', a star icon, and a refresh icon. Below the URL are buttons for 'Undo' and 'Save'. The main area is titled 'Form Design'.

Fields tab is selected. On the left, a sidebar lists fields under 'Fields' and 'Formatters' categories:

- Fields: Created, Created by, Updated, Updated by, Updates
- Formatters: Activities (filtered), Contextual Search Results, Ratings

The main workspace displays the 'Daily Expenses [u_daily_expenses]' table structure with two columns:

Number	Family Member Name
Date	Expense

A third column is partially visible at the bottom, labeled 'Comments'.

Milestone 5: Creation of Relationship

Activity 1: Creation of Relationship between Family Expenses and Daily Expenses tables

1. Go to All >> In the filter search for Relationships >> Open Relationships
 2. Click on New.
 3. Enter the details:
Name : Daily Expenses
Applies to table : Select Family Expenses
Daily Expenses : Select Daily Expenses
 4. Click Save.

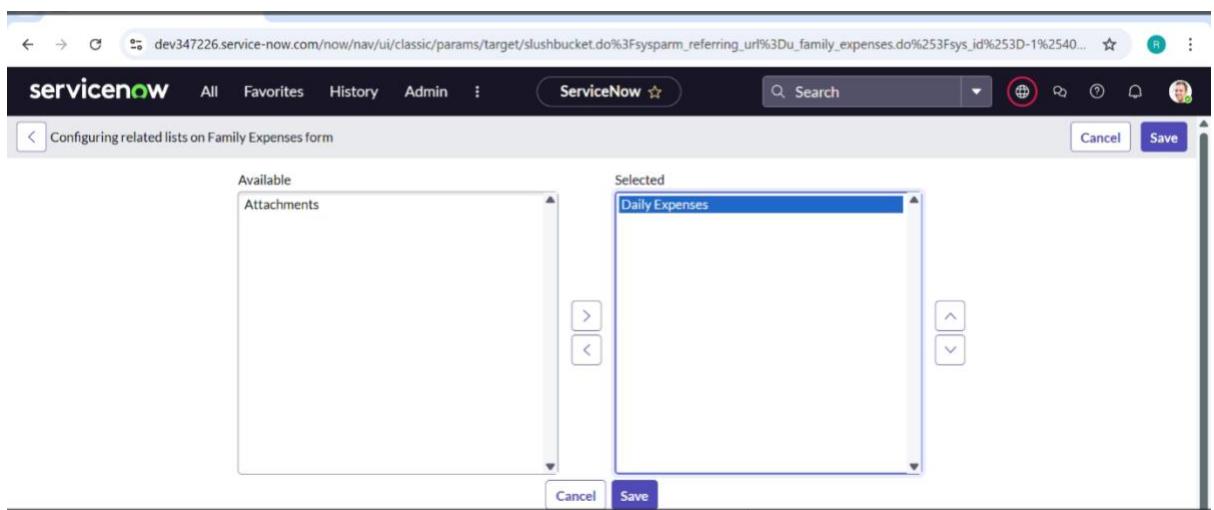
The screenshot shows the ServiceNow interface for creating a new relationship record. The top navigation bar includes links for All, Favorites, History, and a search bar. The main title is "Relationship - New Record". On the left, there's a sidebar with a back arrow, a "Relationship" section, and a "New record" button. The main form has fields for "Name" (set to "Daily Expenses"), "Application" (set to "Global"), and checkboxes for "Advanced" and "Scripted". Below these are dropdowns for "Applies to table" (set to "Family Expenses [u_family_expe...]" with a dropdown arrow) and "Queries from table" (set to "Daily Expenses [u_daily_expens...]" with a dropdown arrow). A note at the bottom explains the purpose of the script. The bottom section contains a "Query with" toggle, a "Turn on ECMAScript 2021 (ES12) mode" button, and a script editor window with the following code:

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

Milestone 6: Configuring Related List on Family Expenses

Activity 1: Configuring Related List on Family Expenses

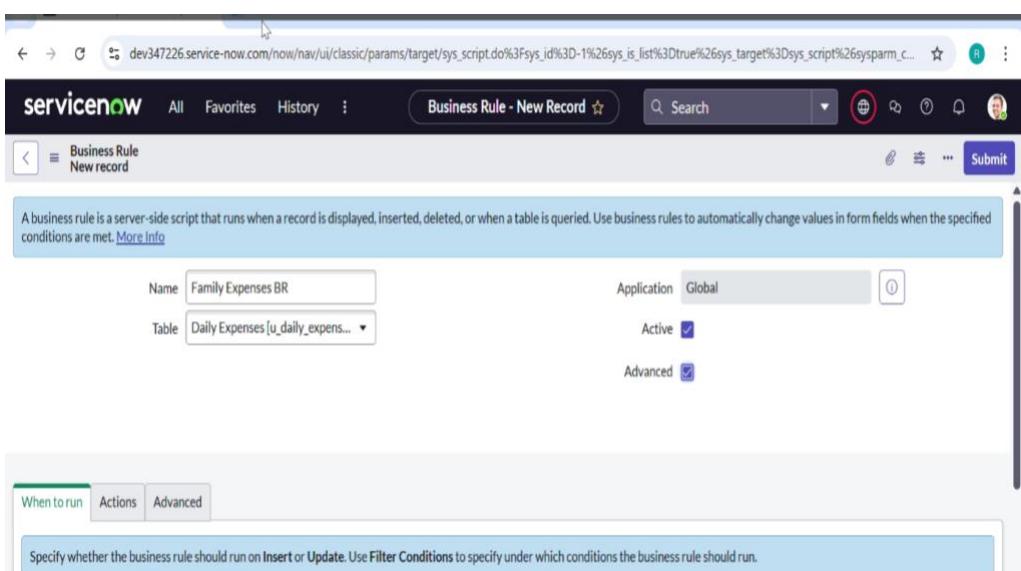
1. Go to All >> In the filter search for Family Expenses >> Open Family Expenses
2. Click on New
3. Go to the Header and right click there >> click on Configure >> Select Related Lists
4. Add Daily Expenses to the Selected Area.
5. Click on Save

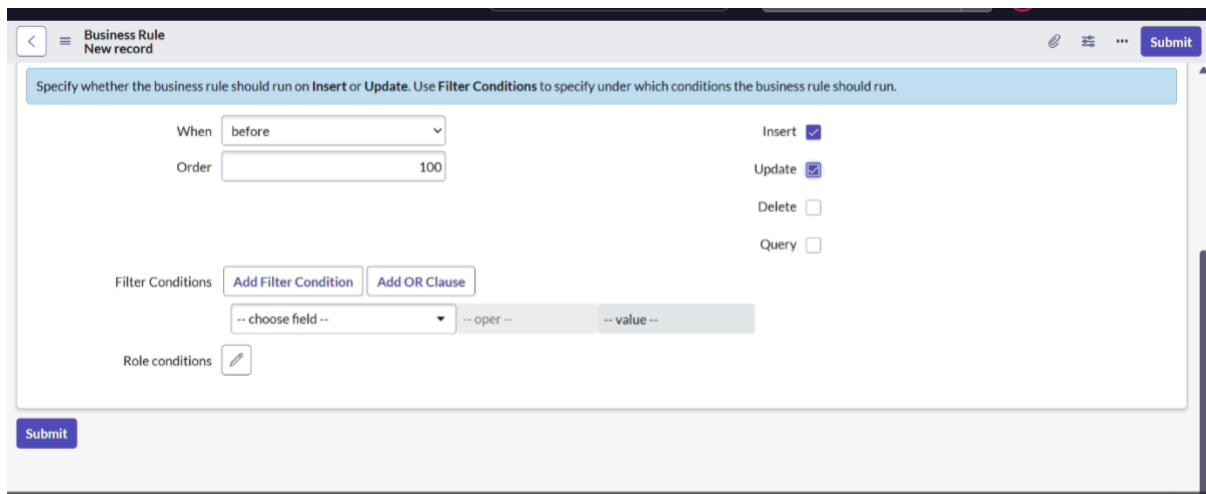


Milestone 7: Business Rules

Activity 1: Creation of Business Rules

1. Go to All >> In the filter search for Business Rules.
2. Under System Definition Select Business Rules then click on New.
3. Enter the Details:
Name : Family Expenses BR
Table : Select Daily Expenses
Check Advanced
4. In when to run Check Insert and Update



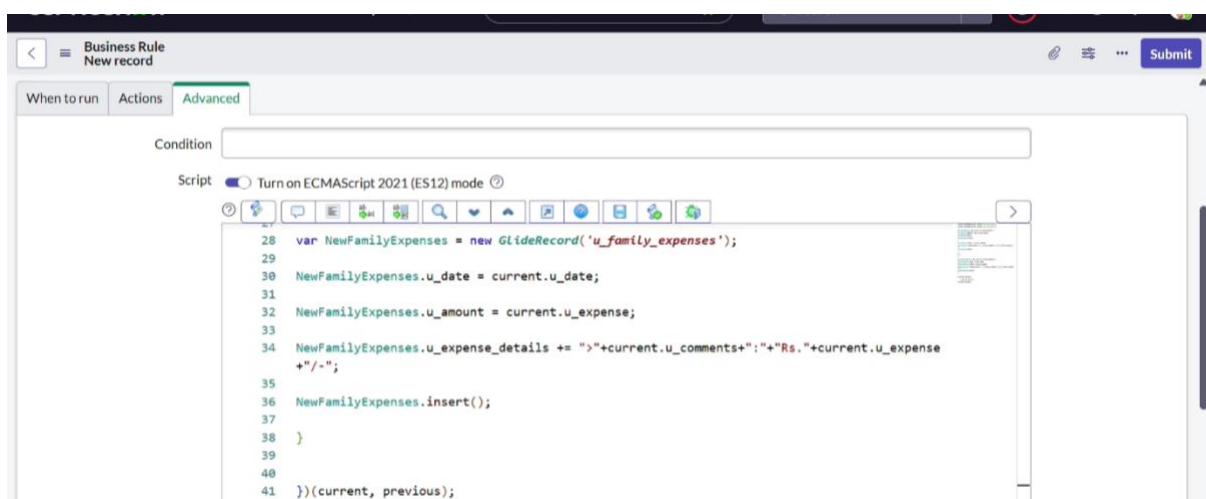


5. In Advance(we write the code): Write the below code >>

```
(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);
```

6. Go to the Header and right click there>> click on Save.



Milestone 8: Relationship

Activity 1: Configure the Relationship

1. Go to All >> In the filter search for Relationships >> Open Relationships.
2. In that, open Daily Expenses Relationship.
3. For Applies to table : Select Family Expenses.
4. In Query with : write the below Query.

```
(functionrefineQuery(current, parent) {  
  
    // Add your code here, such as current.addQuery(field, value);  
    current.addQuery('u_date',parent.u_date);  
    current.query();  
  
})(current, parent);
```

5. Click on Update.

