

Calculating Family Expenses Using ServiceNow

Team ID: NM2025TMID15405

Team Members :

Team Leader: YUVASRI.M

Team Member: PATHMASRI.R

Team Member: UMA.S

Team Member: NAVINA.A

Problem Statement

The project aims to develop a comprehensive expense calculation system using ServiceNow. Managing Family expenses can be challenging due to scattered records, lack of categorization, and absence of real-time insights. This system will enable users to track, categorize, analyze and manage family expenses efficiently.

Objectives

- It will include features such as expense categorization, budget setting, real-time tracking, and reporting capabilities.
- Utilizing ServiceNow's robust platform, the project will ensure seamless integration, user-friendly interface, and scalability to accommodate varying family sizes and financial complexities.
- The goal is to empower users with the tools they need to make informed financial decisions and promote financial well-being within the family unit.

Skills

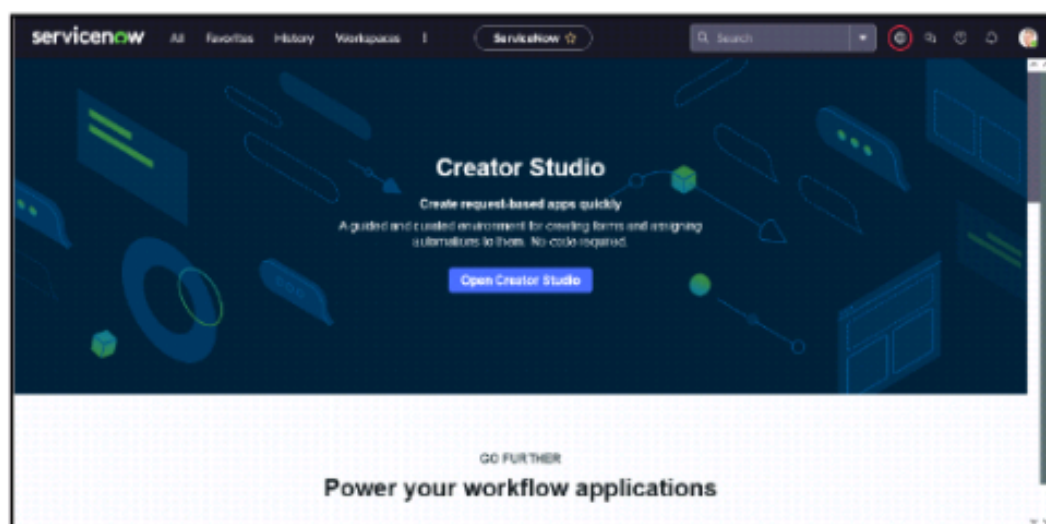
- Analytical Skills: Financial data analysis.
- Conceptual Skill : *IOT Open Hardware platforms (optional integration for smart tracking), Applying Data Structures for financial record-keeping.*

• **Technical Skills: *Data handling and reporting with ServiceNow.***

TASK INITIATION

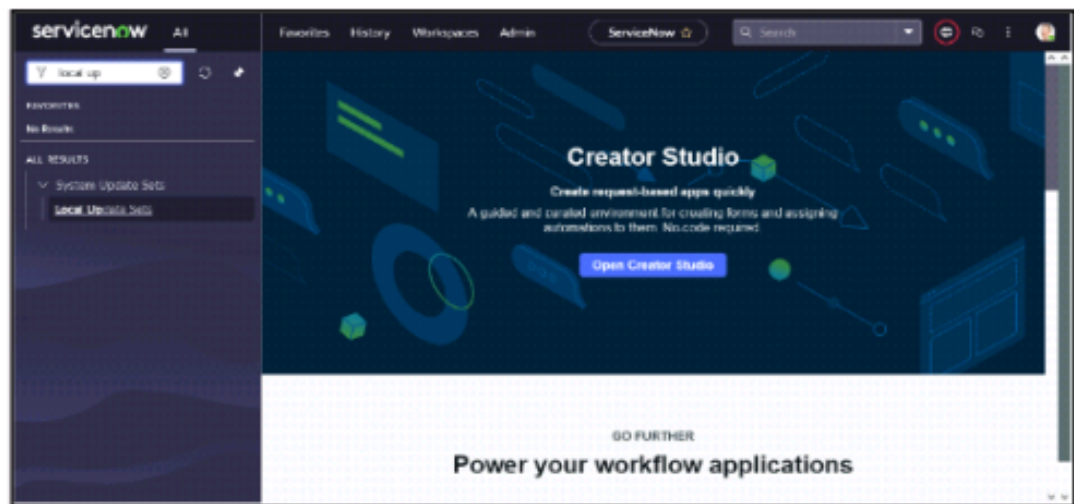
Milestone 1 : Setting up ServiceNow Instance

- 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: Creation Of New Update Set

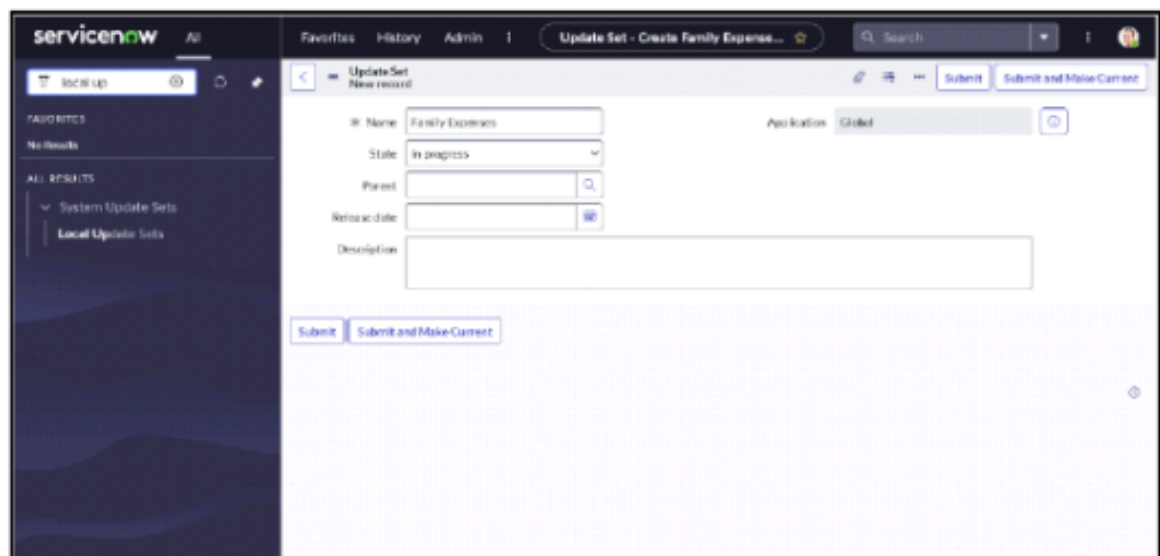
- Go to All >> In the filter search for Local Update set > click on New



- Enter the Details as:

Name : Family Expenses

- Then click on Submit and Make current.



Milestone3: Creation of Table

Activity 1 : Creation of Family Expenses Table

- Go to All > In the filter search for Tables > click on New.
- Enter the Details:

Label : Family Expenses

Name : Auto-Populated

New menu name : Family Expenditure

The screenshot shows the ServiceNow 'Table - New Record' form. The left sidebar contains a navigation menu with options like 'Table', 'Table Titles', 'System Class', 'Table Definition', 'Table Columns', 'Table Rotations', 'Decision Tables', 'Remote Tables', 'Table Definitions', 'System Diagnostics', and 'State'. The main form area has a header with 'Table - New Record' and a search bar. Below the header, there's a description: '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](#)'. The form fields include: 'Label' (Family Expenses), 'Name' (u_family_expenses), 'Extendable' (checkbox), 'Application' (Global), 'Create module' (checkbox), 'Create mobile module' (checkbox), 'Add module to menu' (dropdown), 'New menu name' (Family Expenditure), and 'Remote table' (checkbox). At the bottom, there's a 'Columns' tab with a table showing one column: 'u_family_expenses'.

Column label	Type	Reference	Max length	Default value	Display
u_family_expenses					

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

Activity 2 : Creation of Columns (Fields)

- Near Columns Double click near insert a new row.
- Give the details as:

Column label : Number

Type : String

- Double click on insert a new row again
- Give the details as:

Column label : Date

Type : Date

- Double click on insert a new row again
- Give the details as:

Column label : Amount

Type : Integer

- Double click on insert a new row again
- Give the details as:

Column label : Expense Details

Type : String

Max length : 800

The screenshot shows the ServiceNow Table Editor interface for a table named "Family Expenses". The interface includes a sidebar with navigation options and a main area for editing the table structure. The "Table Columns" tab is active, displaying a list of columns with their respective properties.

Column label	Type	Reference	Max length	Default value	Display
Date	Date	(empty)	40		false
Number	String	(empty)	40	{java.lang.Integer@NumberPadded0}	false
Sys ID	String	(empty)	32		false
Expense Details	String	(empty)	800		false
Amount	Integer	(empty)	40		false

Below the table, there are "Delete" and "Update" buttons, and a "Related Links" section with links to "Form Builder", "Design Form", and "Insert Entry".

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

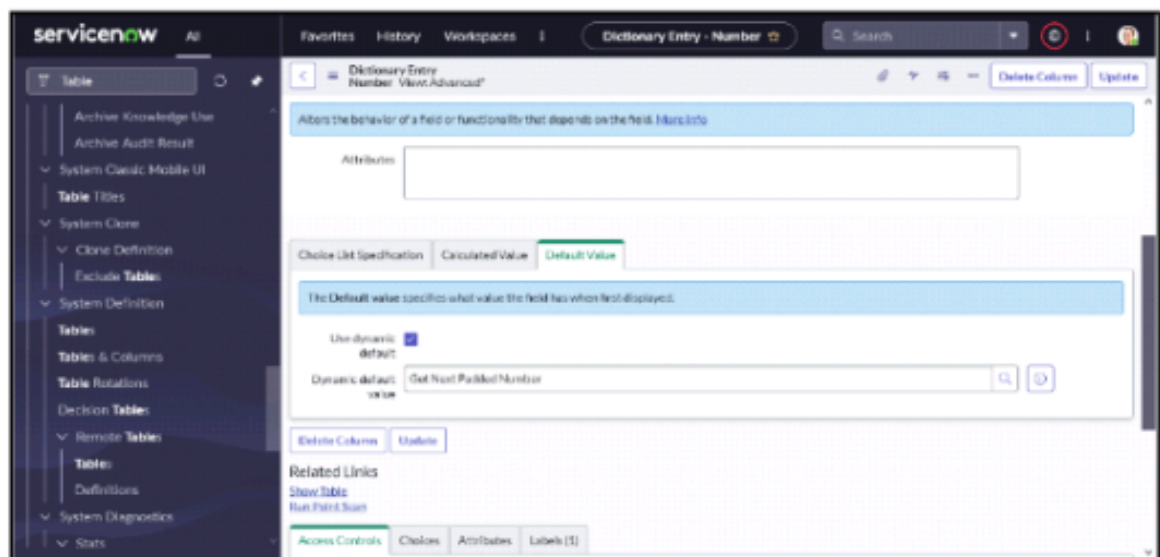
Activity 3 : Making Number Field an Auto-Number

- Double click on the Number Field/Column.
- Go down and double click on Advanced view
- In Default Value:

Use dynamic default : check the box

Dynamic default value : Get Next Padded Number

- Click on Update.



- Go to All >> In the filter search for Number Maintenance >> select
NumberMaintenance
- Click on New.
- Enter the below Details:

Table : Family Expenses

Prefix : MFE

The screenshot shows a configuration window for a table named 'Family Expenses'. The window has a dark header bar with tabs for 'All', 'Favorites', 'History', and 'Workspaces'. Below the tabs, there's a search bar and a 'Number - MFE' label. The main area contains a form with the following fields: 'Table' (set to 'Family Expenses'), 'Prefix' (set to 'MFE'), 'Number' (set to '1'), and 'Application' (set to 'Global'). There are also icons for undo, redo, and a 'Update' button.

- Click on Submit.

Activity 4 : Configure the Form

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

The screenshot shows the form design interface for 'Family Expenses'. The form is divided into two sections. The top section is titled 'Family Expenses [u_family_expenses]' and contains three fields: 'Number', 'Date', and 'Amount'. The bottom section is titled 'Expense Details' and contains a single field. The interface includes a '2 Column' dropdown and a '1 Column' dropdown, indicating different layout options.

- Make Number Read-Only Field by clicking on the gear icon and checking Read-Only
- Make Date, Amount Mandatory Field by clicking on the gear icon and checking Mandatory
- Click on Save.

Milestone 4 : Creation of Table(DailyExpenses)

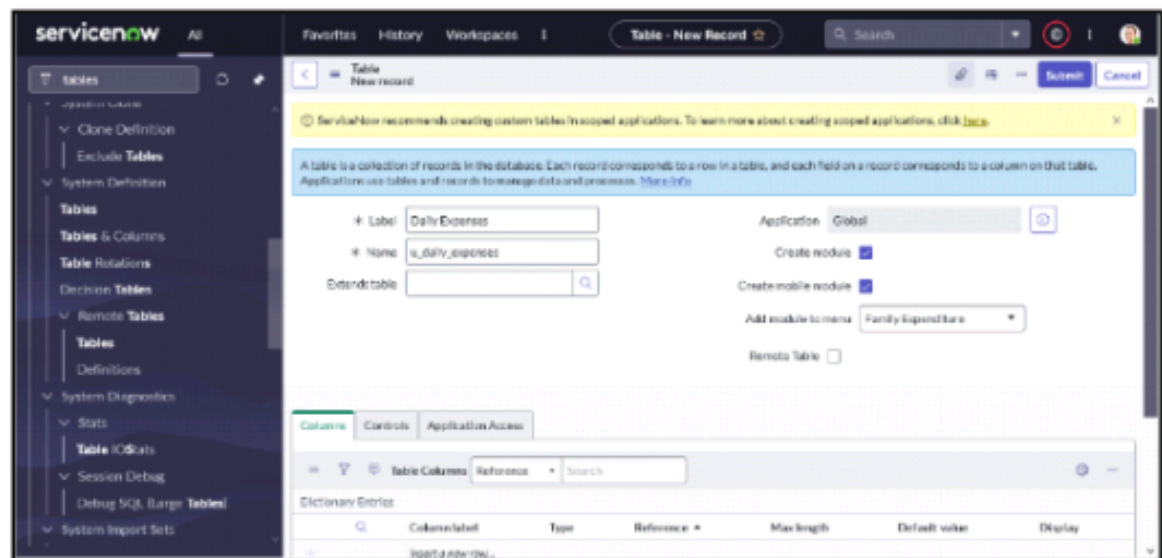
Activity 1: Creation of Table(Daily Expenses)

- Go to All > In the filter search for Tables > click on New.
- Enter the Details:

Label : Daily Expenses

Name : Auto-Populated

Add Module to menu : Family Expenditure



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

Activity2: Creation of Columns(Fields)

- Near Columns Double click near insert a new row.
- Give the details as:

Column label : Number

Type : String

- Double click on insert a new row again
- Give the details as:

Column label : Date

Type : Date

- Double click on insert a new row again
- Give the details as:

Column label : Expense

Type : Integer

- Double click on insert a new row again
- Give the details as:

Column label : Family Member Name

Type : Reference

Reference:User

Max length : 32

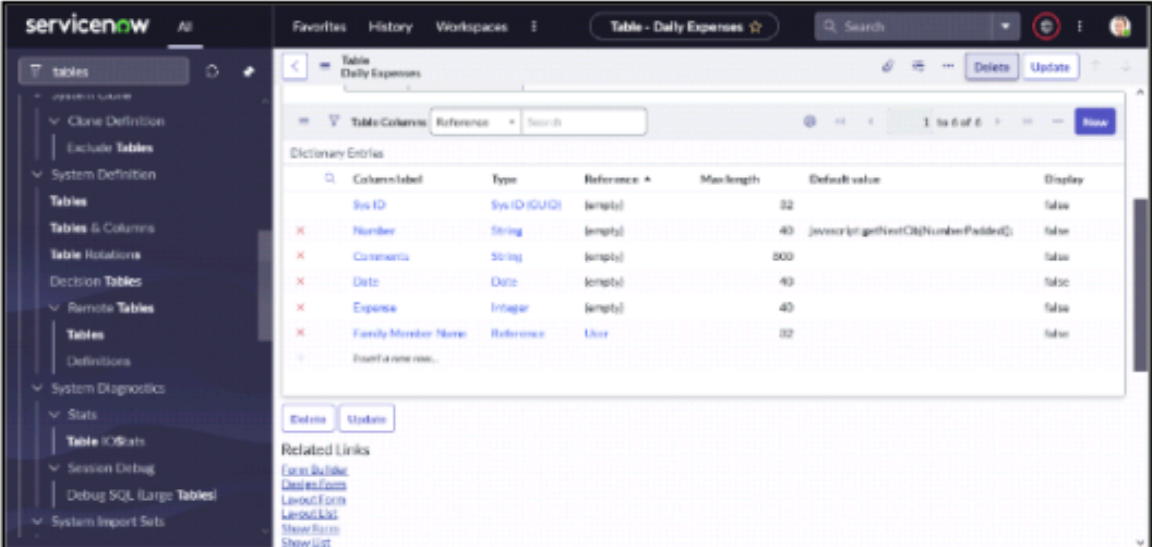
- Double click on insert a new row again
- Give the details as:

Column label : Comments

Type : String

Max length : 800

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



The screenshot shows the ServiceNow interface for configuring a table. The left sidebar contains navigation links for 'tables', 'Clone Definition', 'Exclude Tables', 'System Definition', 'Tables', 'Tables & Columns', 'Table Rotations', 'Decision Tables', 'Remote Tables', 'Table Definitions', 'System Diagnostics', 'Stats', 'Table Objects', 'Session Debug', 'Debug SQL (Large Tables)', and 'System Import Sets'. The main area displays the 'Table Columns' configuration for the 'Daily Expenses' table. The table has the following columns:

Column label	Type	Reference	Max length	Default value	Display
Sys ID	Sys ID (GUID)	[empty]	32		false
Number	String	[empty]	40	javascript:parseInt(0)(Number.Padded(0))	false
Comments	String	[empty]	800		false
Date	Date	[empty]	40		false
Expense	Integer	[empty]	40		false
Family Member Name	Reference	User	32		false

Below the table, there are buttons for 'Delete' and 'Update'. A 'Related Links' section is also visible, containing links for 'Form Builder', 'Design Form', 'Layout Form', 'Layout List', 'Show Form', and 'Show List'.

Activity 3: Making Number Field an Auto-Number

- Double click on the Number Field/Column.
- Go down and double click on Advanced view

In Default Value:

Use dynamic default : check the box

Dynamic default value : Get Next Padded Number

- Click on Update.

The screenshot displays the ServiceNow interface for configuring a Dictionary Entry for a Number field. The left sidebar shows the navigation menu with 'Tables' selected. The main content area shows the 'Default Value' tab for the 'Number' field. The 'Default Value' tab is active, showing a text input field for the default value. Below this, there are sections for 'Dynamic default' and 'Dynamic default value'. The 'Dynamic default' section has a checkbox labeled 'Use dynamic default' which is checked. The 'Dynamic default value' section has a text input field containing 'Get Next Padded Number'. At the bottom, there are buttons for 'Delete Columns' and 'Update'. The 'Related Links' section shows 'show table' and 'Run/Paint Run'.

- Go to All >> In the filter search for Number Maintenance >> select

NumberMaintenance

- Click on New.

- Enter the below Details:

Table : Daily Expenses

Prefix : DFE

- Click on Submit.

The screenshot shows the ServiceNow configuration page for a Number (DSE). The fields are as follows:

Field	Value
Title	Daily Expenses
Prefix	DSE
Number	1,000
Application	Global
Number of digits	7

Buttons: Update, Delete

Related Links: [Show Config](#)

Activity 4 : Configure the Form

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

The screenshot shows the ServiceNow Form Design interface for the 'Daily Expenses' form. The form is titled 'Daily Expenses [dse_form_expenses]' and has a 'Comments' field. The form is designed with a header section and a main body section. The header section contains the following fields:

Field	Type
Number	Text
Date	Text
Family member name	Text
Expense	Text

The main body section contains the following fields:

Field	Type
Comments	Text

Buttons: UNDO, SAVE

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

Milestone 5: Creation of Relationship

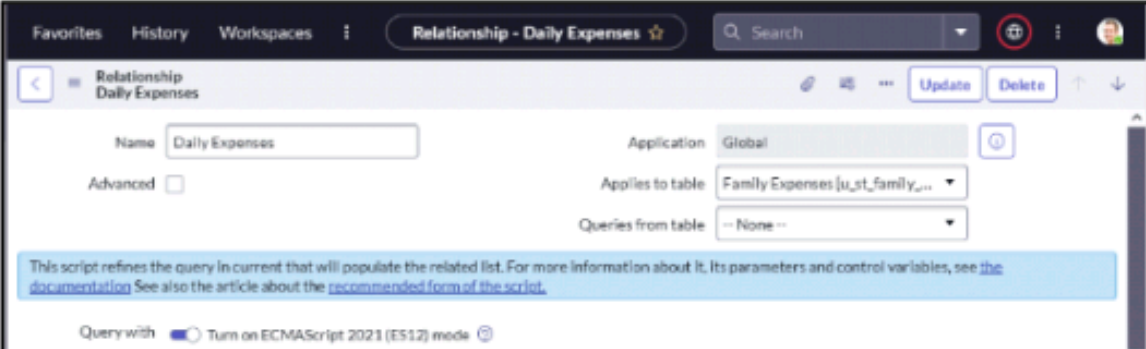
Activity: Creation of Relationship between Family Expenses and Daily Expenses tables

- Go to All >> In the filter search for Relationships >> Open Relationships
- Click on New.
- Enter the details:

Name : Daily Expenses

Applies to table : Select Family Expenses

- Click Save.



The screenshot shows the 'Relationship - Daily Expenses' configuration form. The form has a header bar with 'Favorites', 'History', 'Workspaces', and a search bar. Below the header, the form is titled 'Relationship - Daily Expenses'. It contains the following fields:

- Name:** A text input field containing 'Daily Expenses'.
- Advanced:** A checkbox that is currently unchecked.
- Application:** A dropdown menu set to 'Global'.
- Applies to table:** A dropdown menu set to 'Family Expenses [u_st_family...]'.
- Queries from table:** A dropdown menu set to '-- None --'.

At the bottom of the form, there is a blue informational box with the text: '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 this box, there is a 'Query with' section with a radio button and the text 'Turn on ECMAScript 2021 (ES12) mode'.

Milestone 6: Configuring Related List on Family Expenses

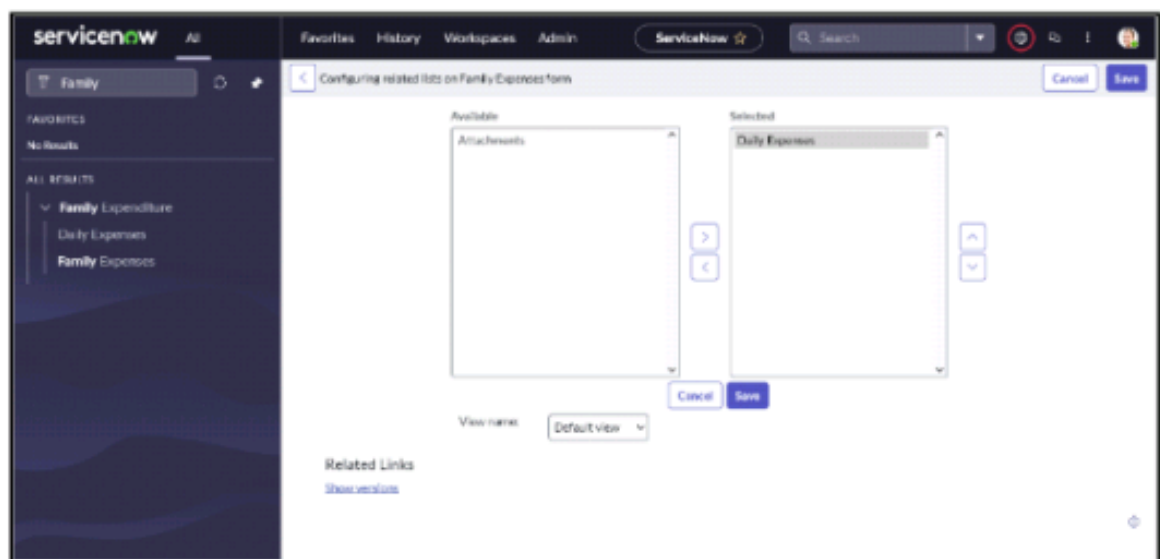
- Go to All >> In the filter search for Family Expenses >> Open Family

Expenses

- Click on New
- Go to the Header and right click there>> click on Configure >> Select

Related Lists

- Add Daily Expenses to the Selected Area.
- Click on Save



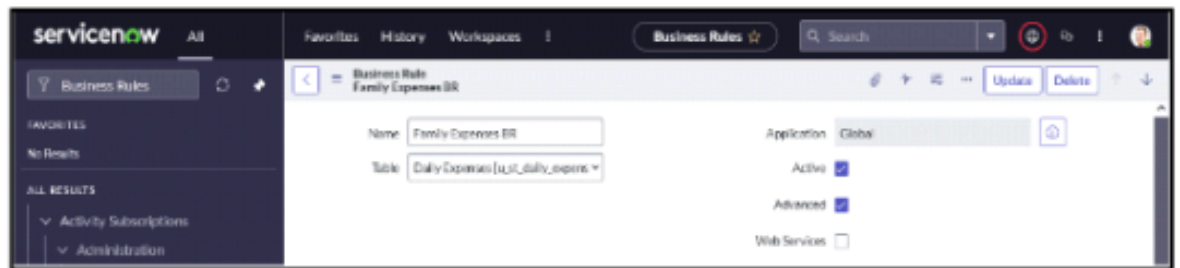
Milestone 7: Creation of Business Rules

- Go to All >> In the filter search for Business Rules.
- Under System Definition Select Business Rules then click on New.
- Enter the Details:

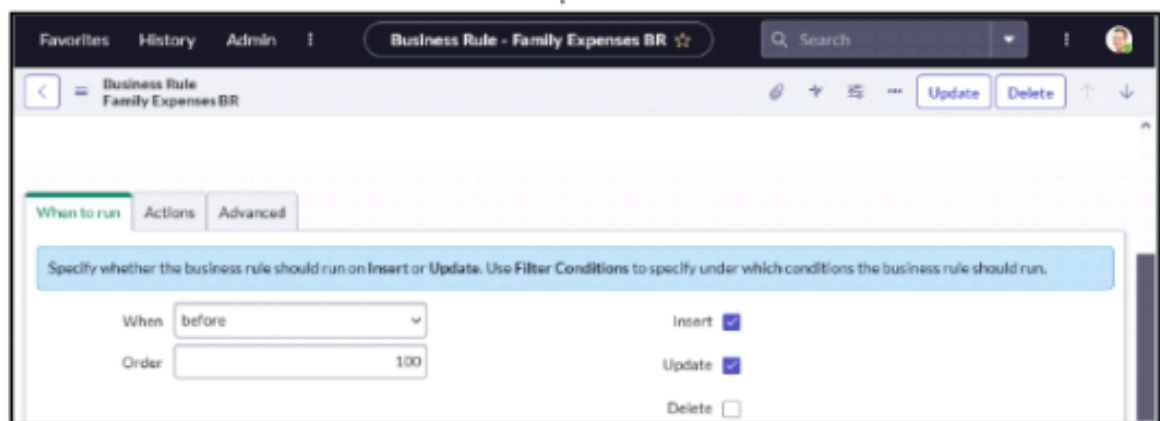
Name : Family Expenses BR

Table : Select Daily Expenses

Check Advanced



- In when to run Check Insert and Update.



- 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())

{

Fam ilyExpenses.u_am ount += 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 +=

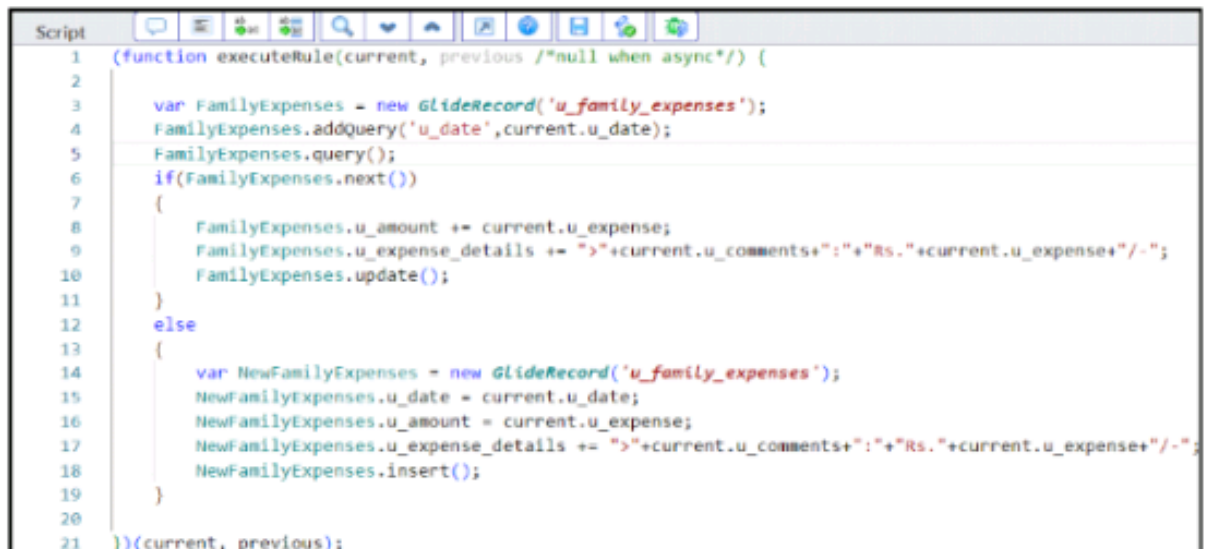
">" + current.u_comments + ":" + "Rs." + current.u_expense + "/-";

NewFamilyExpenses.insert();

}

})(current, previous);

```



```
Script
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     if(FamilyExpenses.next())
7     {
8         FamilyExpenses.u_amount += current.u_expense;
9         FamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.u_expense + "/-";
10        FamilyExpenses.update();
11    }
12    else
13    {
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 += ">" + current.u_comments + ":" + "Rs." + current.u_expense + "/-";
18        NewFamilyExpenses.insert();
19    }
20
21 })(current, previous);
```

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

Milestone 8: Configure the Relationship

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

```
(function refineQuery(current, parent) {
```

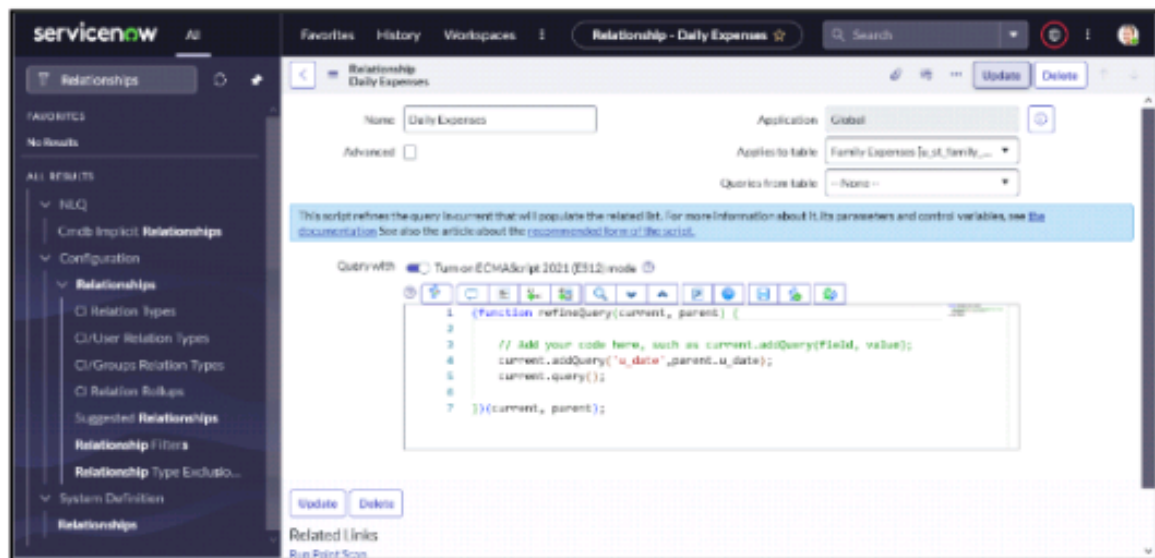
```
// Add your code here, such as current.addQuery(field, value);
```

```
current.addQuery('u_date', parent.u_date);
```

```
current.query();
```

```
})(current, parent);
```

- Click on Update.



Conclusion

The Project demonstrates how ServiceNow can be effectively adapted for personal financial management by building a scalable, automated expense tracking system. It not only simplifies day-to-day expense management but also supports long-term financial planning.

By leveraging ServiceNow's robust features, families can gain better control over their finances, avoid unnecessary overspending , and make smarter financial decisions, ultimately enhancing financial well-being.