

Calculating Family Expenses using Service Now

Team Members:

- Subarna sri.R (Team Leader)
- Rethna .R
- Sandhiya.B
- Srinidhi.S

Department: Information Technology

Course: ServiceNow Administrator

Institution: A.V.C. College of Engineering

Mentor : Mr.N.P.K.GaneshKumar, A/P, IT

Academic Year: 2025-2026

1. Abstract

The project aims to develop a comprehensive expense calculation system using ServiceNow. This system will enable users to track and manage family expenses efficiently. 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 end goal is to empower users with the tools they need to make informed financial decisions and promote financial well-being within the family unit.

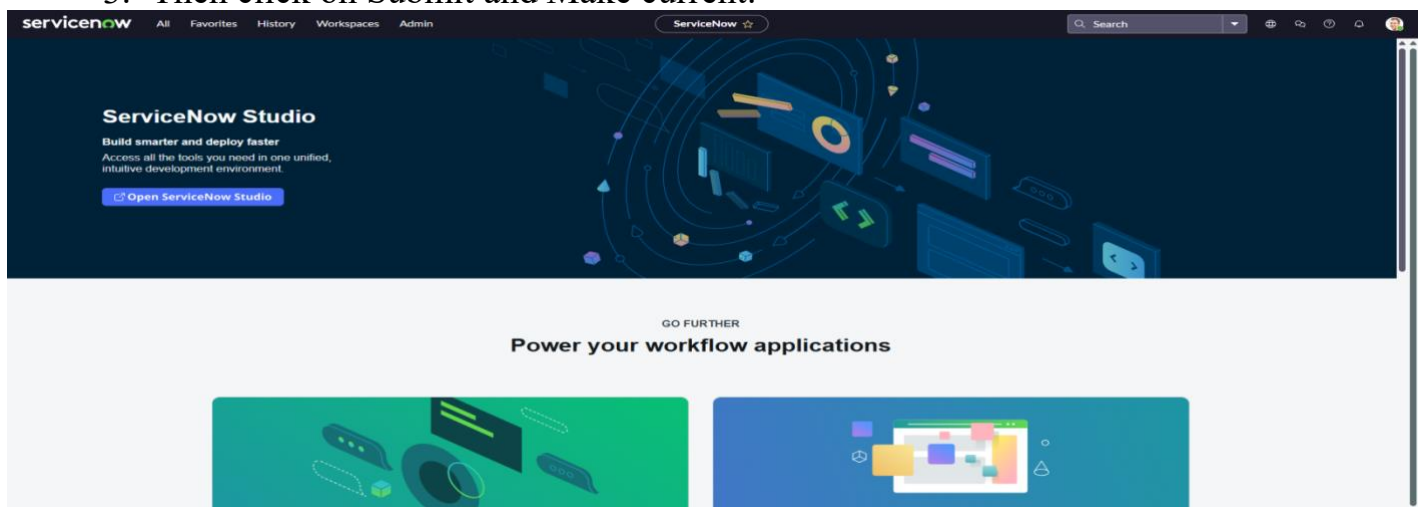
2. Tasks Dones With Explanation

Step 1: Setting up ServiceNow Instance

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

Step 2: 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.



ServiceNow Update Set - Create New Update Set

Navigation: All, local

Form Fields:

- Name: Family Expenses
- State: In progress
- Parent:
- Release date:
- Description:

Buttons: Submit, Submit and Make Current

Step 3: Creation of Table

1. Navigate 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. click on Save..

Table - New record

Form Fields:

- Label: Family Expenses
- Name: u_family_expenses
- Extends table:
- Application: Global
- Create module: ☒
- Create mobile module: ☒
- Add module to menu: -- Create new --
- New menu name: Family Expenditure

Column label	Type	Reference	Max length	Default value	Display
Insert a new row...					

Step 4: Creations of columns

- 1) Near Columns Double click near insert a new row.
 Give the details as:
 - 1) Column label : Number
Type : String
 - 2) Column label : Date
Type : Date

3) Column label : Amount

Type : Integer

4) Column label : Expense Details

Type : String

Max length : 800

2) click on Save.

The screenshot shows the 'Table' configuration page in ServiceNow. The 'Label' is 'Family Expenses' and the 'Name' is 'u_family_expenses'. The 'Application' is 'Global'. The 'Create module' and 'Create mobile module' checkboxes are checked. The 'Add module to menu' dropdown is set to '-- Create new --' and the 'New menu name' is 'Family Expenditure'. Below this, there are tabs for 'Columns', 'Controls', and 'Application Access'. The 'Columns' tab is active, showing a table of dictionary entries.

	Column label	Type	Reference	Max length	Default value	Display
<input type="checkbox"/>	Number	String				false
<input type="checkbox"/>	Date	Date				false
<input type="checkbox"/>	Amount	Integer				false
<input type="checkbox"/>	Expense Details	String		800		false

Step 5: 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.

The screenshot shows the 'Dictionary Entry - Number' configuration page in ServiceNow. The 'Table' is 'Family Expenses [u_family_expenses]' and the 'Type' is 'String'. The 'Column label' is 'Number' and the 'Column name' is 'u_number'. The 'Max length' is 40. The 'Application' is 'Global'. The 'Active' checkbox is checked. The 'Function field' checkbox is unchecked. The 'Read only' checkbox is checked. The 'Mandatory' checkbox is unchecked. The 'Display' checkbox is unchecked. Below this, there is a section for 'Attributes'. The 'Default Value' tab is selected, showing the 'Use dynamic default' checkbox checked and the 'Dynamic default value' set to 'Get Next Padded Number'.

1. Go to All >> select Number Maintenance

2. Click on New.

Enter the below Details:

- Table : Family Expenses
- Prefix : MFE.

The screenshot shows the ServiceNow 'Number Maintenance' form for 'Number - MFE'. The form fields are: Table (Family Expenses), Prefix (MFE), Number (1,000), Application (Global), and Number of digits (7). There are 'Update' and 'Delete' buttons at the top right. Below the form, there is a 'Related Links' section with a 'Show Counter' link.

3. Click **Submit**.

The screenshot shows the ServiceNow 'Numbers' list view. The table has columns: Prefix, Number, Number of digits, Table, and Updated. The data row shows: MFE, 1,000, 7, Family Expenses, and 2025-10-24 23:36:50.

Prefix	Number	Number of digits	Table	Updated
MFE	1,000	7	Family Expenses	2025-10-24 23:36:50

Step 5: 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.

The screenshot shows the ServiceNow 'Form Design' editor for 'Family Expenses [u_family_expenses]'. The editor has a left sidebar with 'Fields' and 'Formatters' sections. The main area shows a form layout with two columns. The first column contains 'Number' and 'Date' fields. The second column contains 'Amount' and 'Expense' fields. Each field has a gear icon for configuration.

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.

servicenow All Favorites History Workspaces Admin Family Expenses - Create MFE0001016 Search

Family Expenses New record

Number MFE0001016 * Date * Amount Expense

Submit

Step 6: Creation of Daily Expenses Table

1. Go to **All** → **Search Tables** → click on **New**.
2. Enter the Details:
 - Label : Daily Expenses
 - Name : Auto-Populated
 - Add Module to menu : Family Expenditure
3. click on **Save**..

servicenow All Favorites History Workspaces Admin Table - Daily Expenses Search

Table Daily Expenses Delete Update Delete All Records

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](#)

* Label Daily Expense * Name u_daily_expenses Application Global

Columns Controls Application Access

Table Columns For text Search 1 to 11 of 11 New

Column label	Type	Reference	Max length	Default value	Display
Created by	String	(empty)	40		false
Comments	String	(empty)	800		false
Expense	Integer	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Created	Date/Time	(empty)	40		false
Family Member Name	Reference	User	32		false
Date	Date	(empty)	40		false
Updates	Integer	(empty)	40		false
Updated by	String	(empty)	40		false
Updated	Date/Time	(empty)	40		false
Number	String	(empty)	40	javascript:getNextObj(NumberPadded);	false
Insert a new row...					

Delete Update Delete All Records

Step 7: Creation of Columns

Near Columns Double click near insert a new row.

Give the details as:

- 1) Column label : Number
Type : String
- 2) Column label : Date
Type : Date
- 3) Column label : Expense
Type : Integer

4) Column label : Family Member Name

Type : Reference

Max length : 800

5) Column label : Comments

Type : String

Max length : 800

6) click on Save.

Table New record

Name: u_daily_expenses

Extends table:

Create module: ☒

Create mobile module: ☒

Add module to menu: Family Expenditure

Columns

Table Columns for text

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
Number	String				false
Family Member Name	Reference	User		false	false
Date	Date			false	false
Expense	Integer			false	false
Comments	String		800		false

Submit Cancel

Step 8: 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.

servicenow

Dictionary Entry - Number

Table: Daily Expenses [u_daily_expenses]

Type: String

Column label: Number

Column name: u_number

Max length: 40

Application: Global

Active: ☒

Function field: ☐

Read only: ☒

Mandatory: ☐

Display: ☐

Attributes:

Choice List Specification

Calculated Value

Default Value

The Default value specifies what value the field has when first displayed.

Use dynamic default: ☒

Dynamic default value: Get Next Padded Number

Delete Column Update

1. Go to **All >> Search Number Maintenance >> select Number Maintenance.**
2. Click on New.
3. Enter the below Details:

Table : Daily Expenses

Prefix : MFE

The screenshot shows the ServiceNow 'Number Maintenance' form for 'Number - MFE'. The form includes the following fields:

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

At the bottom, there are 'Update' and 'Delete' buttons, and a 'Related Links' section with a 'Show Counter' link.

4. Click on Submit.

Step 9: 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 Designer for 'Daily Expenses [u_daily_expenses]'. The form is configured with two columns:

- Column 1 (2 Column):** Contains four fields: 'Number', 'Family Member Name', 'Date', and 'Expense'. Each field has a gear icon for configuration.
- Column 2 (1 Column):** Contains one field: 'Comments'.

The form is currently in '2 Column' view, and the 'Comments' field is in '1 Column' view.

Step 10: 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. The form is titled 'Relationship - Daily Expenses'. It includes a search bar at the top right. The main form area has the following fields:

- Name:** Daily Expenses
- Advanced:** ☐
- Application:** Global
- Applies to table:** Family Expenses [u_family_expenses]
- Queries from table:** Daily Expenses [u_daily_expenses]

Below these fields is a blue informational bar: "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 expanded, showing a script editor with the following code:

```
1 (function refineQuery(current, parent) {
2
3   // Add your code here, such as current.addQuery(field, value);
4   current.addQuery('u_date', parent.u_date);
5   current.query();
6
7
8 })(current, parent);
```

At the bottom of the form are 'Update' and 'Delete' buttons.

Step 11: 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.

The screenshot shows the ServiceNow interface for configuring related lists on the 'Family Expenses' form. The title bar says 'Configuring related lists on Family Expenses form'. The main area is divided into two columns:

- Available:** Contains a list with 'Attachments'.
- Selected:** Contains a list with 'Daily Expenses'.

Between the columns are navigation buttons: a right arrow (>) to move items from Available to Selected, and a left arrow (<) to move items from Selected to Available. Below the Selected column are up and down arrow buttons for reordering.

At the bottom are 'Cancel' and 'Save' buttons. Below the 'View name:' dropdown (set to 'Default view') is a 'Related Links' section with a 'Show versions' link.

Step 11: 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
4. Check Advanced
5. In when to run Check Insert and Update.
6. Go to the Header and right click there >> click on Save.

The screenshot shows the ServiceNow interface for creating a Business Rule. The header includes navigation tabs (All, Favorites, History, Workspaces, Admin) and a search bar. The main title is 'Business Rule - Family Expenses BR'. Below the title, there's a description: 'A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. [More info](#)'. The configuration fields include: Name (Family Expenses BR), Table (Daily Expenses [u_daily_expenses]), Application (Global), Active (checked), and Advanced (checked). The 'When to run' tab is selected, showing a condition field and a script editor. The script is written in ECMAScript 2021 (ES12) mode and contains 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   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  }(current, previous);
21
22
```

Step 11: 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 Query.

The screenshot shows the ServiceNow interface for configuring a Relationship. The header includes navigation tabs (All, Favorites, History, Workspaces, Admin) and a search bar. The main title is 'Relationship - Daily Expenses'. Below the title, there's a description: '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 configuration fields include: Name (Daily Expenses), Application (Global), Applies to table (Family Expenses [u_family_expenses]), and Queries from table (Daily Expenses [u_daily_expenses]). The 'Query with' tab is selected, showing a script editor. The script is written in ECMAScript 2021 (ES12) mode and contains the following code:

```
1 (function refineQuery(current, parent) {
2
3   // Add your code here, such as current.addQuery(field, value);
4   current.addQuery('u_date', parent.u_date);
5   current.query();
6
7
8  })(current, parent);
```

At the bottom, there are buttons for 'Update' and 'Delete', and a section for 'Related Links' with a link to 'Run Point Scan'.

5. Click on Update.

Result / Output

1. Enter the details and then submit it.

servicenow All Favorites History Workspaces Admin Daily Expenses - Create MFE0001025 Search

Daily Expenses New record

Number MFE0001025 * Family Member Name Waldo Sisk

* Date 2025-10-01 Expense 15,000

Comments Courses fee

Submit

2. Detail will be updated on **Daily Expenses** .

servicenow All Favorites History Workspaces Admin Daily Expenses Search

Daily Expenses Number Search Actions on selected rows... New

Number	Comments	Date	Expense	Created by
MFE0001018	party	2025-10-15	13,333	admin
MFE0001015	EMI	2025-10-23	5,000	admin
MFE0001017	college fees	2025-10-22	20,000	admin
MFE0001013	traveling	2025-10-30	500	admin
MFE0001020	petrol	2025-11-05	500	admin
MFE0001016	mobile	2025-10-23	10,000	admin
MFE0001025	Courses fee	2025-10-01	15,000	admin
MFE0001014	Netflix	2025-10-21	2,500	admin

3. **Daily Expenses Details** will be automatically updated on **Family Expenses** .

servicenow All Favorites History Workspaces Admin Family Expenses Search

Family Expenses Number Search Actions on selected rows... New

Number	Amount	Date	Expense
MFE0001018	15,000	2025-10-01	
MFE0001010	2,500	2025-10-21	
MFE0001011	15,000	2025-10-23	
MFE0001009	1,000	2025-11-20	subscription
MFE0001012	20,000	2025-10-22	
MFE0001013	13,333	2025-10-15	
MFE0001008	500	2025-10-30	travelling
MFE0001014	500	2025-11-05	

4. Output will be successfully updated.

The screenshot displays the ServiceNow user interface for managing family expenses. At the top, the navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. The current page is titled 'Family Expenses - MFE0001018'. Below the header, there is a form for entering expense details. The 'Number' field is populated with 'MFE0001018'. The 'Date' field is set to '2025-10-01'. The 'Amount' field is set to '15,000'. The 'Expense' field is currently empty. Below the form, there is a table titled 'Daily Expenses'. The table has columns for 'Number', 'Comments', 'Date', and 'Expense'. A single row is visible with the number 'MFE0001025', the comment 'Courses fee', the date '2025-10-01', and the amount '15,000'. The table includes pagination controls at the bottom, showing '1 to 1 of 1'.

Number	Comments	Date	Expense
MFE0001025	Courses fee	2025-10-01	15,000

Conclusion

In conclusion, the “**Calculating Family Expenses using ServiceNow**” project effectively demonstrates how ServiceNow can be used to manage and track family expenses in an organized way. By creating custom tables, relationships, and business rules, the system allows users to record daily expenses and automatically update total family expenses. Features like auto-numbering, mandatory fields, and form customization make the application accurate and user-friendly. Overall, the project provides a simple and efficient solution for monitoring family budgets and supports better financial planning through automation on the ServiceNow platform.