

Calculating Family Expenses Using Service Now

ServiceNow Team ID : NM2025TMID07316

Team Size : 4

Team Leader: Noorul Shafa M B

Team member: Vincy S

Team member: Subbu Lakshmi K

Team member: Sree Ranjini S

Problem Statement

In a project of managing family expenses is often challenging due to the lack of a centralization system for tracking income, expenses, and savings. Families typically rely on manual methods such as spreadsheets, paper records, or multiple apps, which can lead to errors, duplication, and difficulty in analysing financial patterns.

Objective

Simplify Expense Tracking- provide a centralization system for recording and categorising all family expenses

Improve Financial Awareness-offer real-time dashboards and reports to give clear insights into spending patterns

Enhance Budget Planning-support better financial planning by identifying high-spending categories and opportunities to save

Skills: Servicenow Development, Workflow&Automation, Scripting in Servicenow,Reporting&Dashboards, Integration Skills

TASK INITIATION

Update

Activity 1:

1. Open Service now

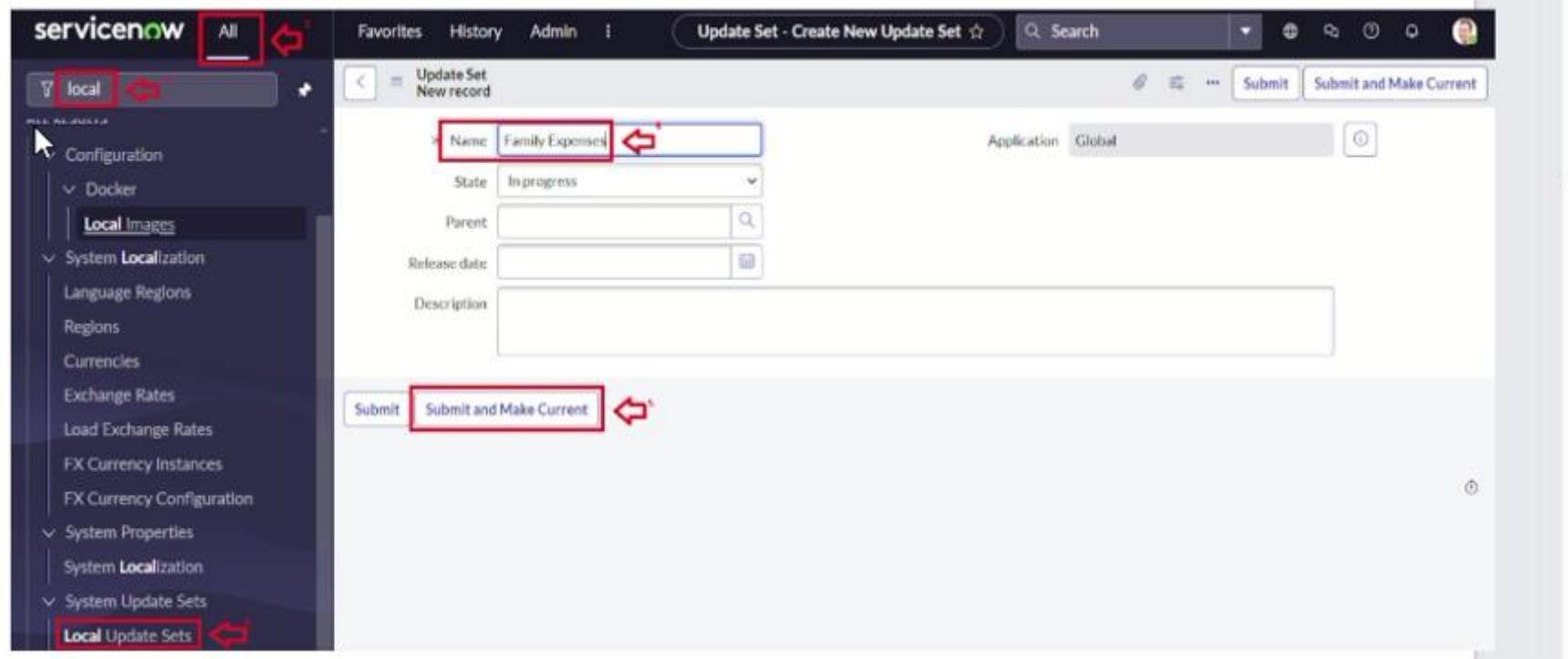
2. Go to All>>search for Local Update Set>Click on New



3. Enter the Details as:

Name: Family Expenses

4. And then click on Submit and make current



Table

Activity 2:

1. Open Service now
2. Go to all>search for Tables>Click on New
3. Enter the Details:

Label: Family Expenses

Name: Auto-Populated

New menu name: Family Expenditure

The screenshot shows a form configuration interface. On the left, there is a table with two rows. The first row has a column labeled 'Label' with the value 'Family Expenses'. The second row has a column labeled 'Name' with the value 'u_st_family_expenses'. Both rows are highlighted with a red border. On the right, there are several settings: 'Application' is set to 'Global', 'Remote Table' is checked, 'Create module' is checked, and 'Create mobile module' is checked. There is also a 'Save' button at the bottom right.

4. Click on save

Activity 3:

1. Open Service now

2. Near Columns Double-click near insert a new row

Give the details as:

Column label: Number

Type: String

3. Double-click on insert a new row again

Give the details as:

Column label:Date

Type:Date

4.Double click on insert a new row again

Give the details as:

Column label:Amount

Type:Integer

5 .Double click on insert a new row again

Give the details as:

Column label:Expense Details

Type:String

Max length:800



6.Click on save

Activity 4:

1.Open Service now

2.Double click on the Number field/Column

3.Go down and double click on Advanced view

4.In Default Value:

Use dynamic default:check the box

Dynamic default value:Get Next Padded Number

5.click on update

6.Go to all>>search for Number Maintenance>>select number maintenance

7.click on new

8.Enter the details:

Table:Family Expenses

Prefix:MFE

< Number MFE Update Delete

* Table Family Expenses

Prefix MFE

* Number 1,000

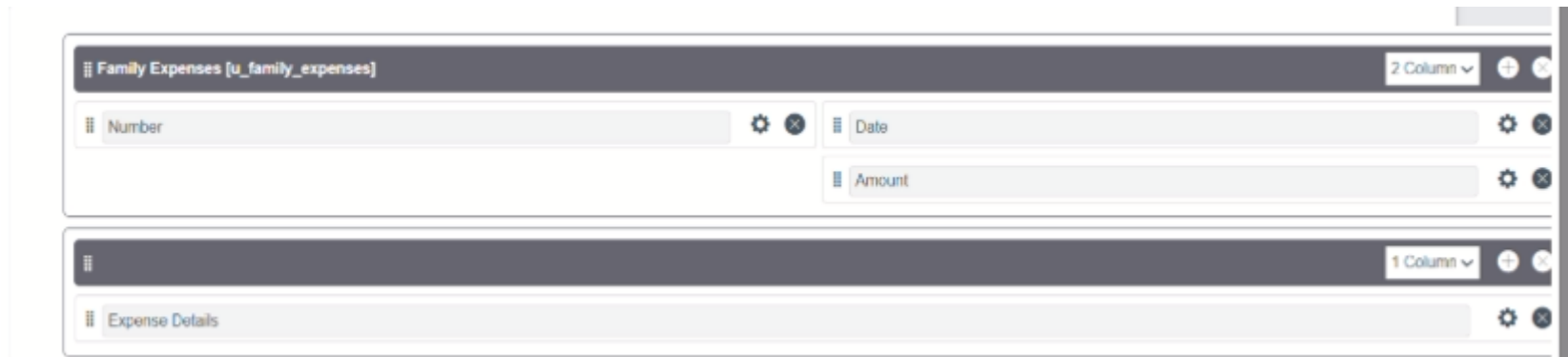
Application Global

Number of digits 7

8. Click on Submit

Activity 5:

1. Open Service now
2. Go to all >> search for Family Expenses >> Open family Expenses
3. Click on new
4. Go to header and click there >> click on configure >> Select Form Design
5. Drag Drop the form as requirement



6. Make Number Read-Only Field by Clicking

7. Make Date, Amount Mandatory field by Clicking

8. Click on save

Table(Daily Expenses)

Activity 6:

1. Open Service now

2. Go to all>search for Tables>Click on New

3. Enter the Details:

Label: Daily Expenses

Name: Auto-Populated

Add Module to menu : Family Expenditure

The screenshot shows a configuration window for a module. On the left, there are three input fields: 'Label' with the value 'Daily Expenses' (marked with a red box and a red arrow labeled '1'), 'Name' with the value 'u_daily_expenses' (marked with a red box and a red arrow labeled '2'), and 'Extends table' which is empty. On the right, there are checkboxes for 'Create module' and 'Create mobile module', both of which are checked. Below these is a button labeled 'Application Menu'. At the bottom, there is a section titled 'Add module to menu' which contains a dropdown menu currently showing 'Family Expenditure' (marked with a red box and a red arrow labeled '3').

4.click on save

Activity 7:

1.Open Service now

2.Near Columns Double Click near insert a new row

Give the details as:

Column label:Number

Type:String

3.Double click on insert a new row again

Give the details as:

Column label:Date

Type:Date

4. Double click on insert a new row again

Column label:Amount

Type:Integer

5 .Double click on insert a new row again

Column label:Expense Details

Type:String

Max length:800

6.Double click on insert a new row again

Give the details as:

Column label:Comments

Type:String

Max length:800

6.Click on save

Activity 8:

1.Open Service now

2.Double click on the Number field/Column

3.Go down and double click on Advanced view

4.In Default Value:

Use dynamic default:check the box

Dynamic default value:Get Next Padded Number

5.click on update

Choice List Specification Calculated Value **Default Value** 1

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

Use dynamic default ☒ 2

Dynamic default value: 3

Data Column **Data** 4

6.Go to all>>search for Number Maintenance>>select number maintenance

7.click on new

8.Enter the details:

Table:Daily Expenses

Prefix:MFE

< **Number**
New record Submit

* Table: 1

* Prefix: 2

* Number:

Application:


Number of digits:

Submit 3

9.click on submit

Activity 9:

1. Open Service now
2. Go to all>>search for Daily Expenses>>Open Daily Expenses
3. Click on new
4. Go to header and click there>>click on configure>>Select Form Design
5. Drag Drop the form as requirement



The screenshot shows the ServiceNow form design interface for the 'Daily Expenses [u_daily_expenses]' form. The form is configured with a 2-column layout. The first section contains four fields: 'Number', 'Date', 'Family Member Name', and 'Expense'. The second section contains a single field: 'Comments'. Each field has a configuration icon (gear) and a mandatory icon (checkmark in a circle).

6. Make Number Read-Only Field by Clicking
7. Make Date, Family Member Name Mandatory field by Clicking
8. Click on save

Relationship

Activity 10:

1. Open Service now

2.Go to all>>search for relationships>>open relationships

3.click on new

4.Enter the details:

Name:Daily Expenses

Applies to table:select Family Expenses

Daily Expenses:select Daily Expenses

5.click save

Configuring Related List

Activity 11:

1.open Service now

2. .Go to all>>search for Family Expenses>>Open FamilyExpenses

3.Click on new

4.Go to header and click there>>click on configure>>select Related Lists

5.Add Daily Expenses to selected area

6.click on save



Business rule

Activity 12:

- 1.Open Service Now
- 2.Go to all>>search for Business Rules
- 3.Under system definition select business rules then click new
- 4.Enter the Details:
Name:Family Expenses BR
Table:Select Daily Expenses
Check Advanced

Business Rule
New record

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.

Name: Family Expenses BR ← 1

Table: Daily Expenses [u: daily_expenses] ← 2

Application: Global ⓘ

Active ☒

Advanced ☒ ← 3

5. In when the run check insert and update

When to run ← 1 Advanced

Specify whether the business rule should run on **Insert** or **Update**. Use **Filter Conditions** to specify under which conditions.

When: before

Order: 100

Insert ☒

Update ☒ → 2

Delete ☐

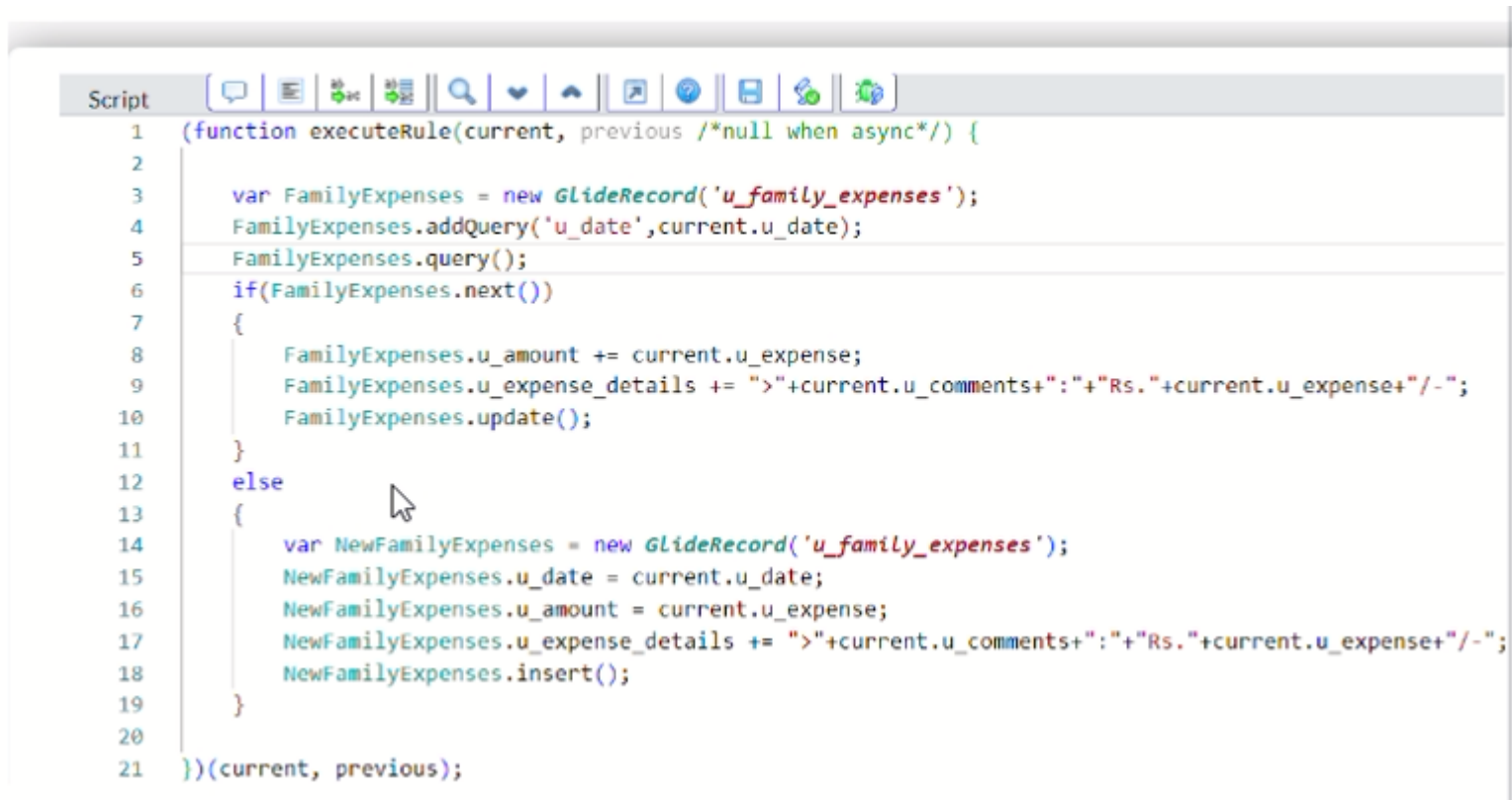
Query ☐

Filter Conditions: Add Filter Condition Add "OR" Clause

-- choose field -- -- oper -- -- value --

Role conditions ✎

6. In Advance (we write the code)



```
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);
```

7.click on save

Configure The Relationship

Activity 13:

- 1.Open Service now
- 2.Go to all>>search for relationships>>open relationships

3. In open Daily Expenses relationship

For Applies to table: select Family Expenses

In Query with: write the below Query

4. click on update

Relationship
Daily Expenses

Name:

Application:

Advanced: ☐

Applies to table: ← 1

Queries from table:

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.

Query with:

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

 ← 2

Update Delete

Update ← 3

Conclusion

The Family Expense Calculation System developed on Service Now provides a centralized, automated, and user-friendly platform for managing and tracking household expenses. By leveraging ServiceNow's capabilities, users can record, categorize, and analyze expenses in real time, ensuring better financial planning and transparency. The system not only simplifies routine expense tracking but also helps families identify spending patterns, plan budgets effectively, and make informed decisions.

Overall, the project demonstrates how automation and digital platforms like ServiceNow can be effectively utilized to streamline personal finance management, making it accurate, efficient, and accessible.