

Calculating Family Expenses Using ServiceNow

Introduction

This project develops a robust family expense calculation and management system using ServiceNow. The solution enables users to efficiently log, categorize, and analyze daily and overall family expenditures. With features like real-time tracking, expense categorization, automated calculations, and reporting, the platform supports families in making informed financial decisions and maintaining their financial health.

Project Objectives

- Build and configure tables for Family Expenses and Daily Expenses in ServiceNow
- Establish automated and relational data handling (auto-numbering, lookups, related lists)
- Implement business rules and relationships for streamlined operations
- Enable transparent reporting for budget and expense analysis

Tools & Technologies

- ServiceNow Personal Developer Instance
- ServiceNow Tables, Fields, Forms, Relationships
- Update Sets for project configuration management
- Business Rules (automation)
- Related Lists and reporting features

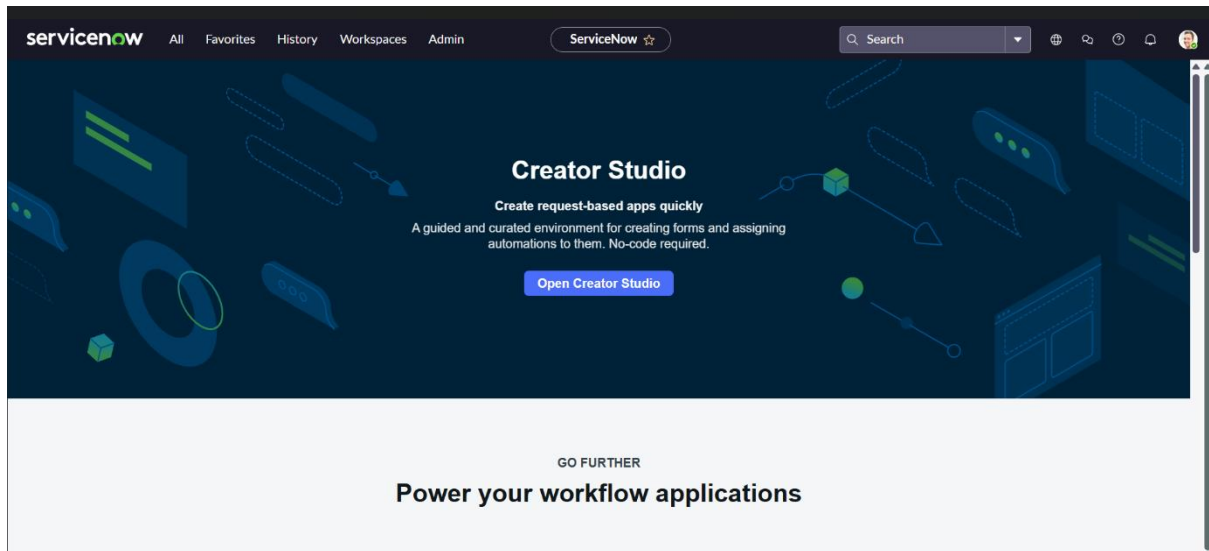
Stepwise Implementation

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.

Setting up ServiceNow Instance

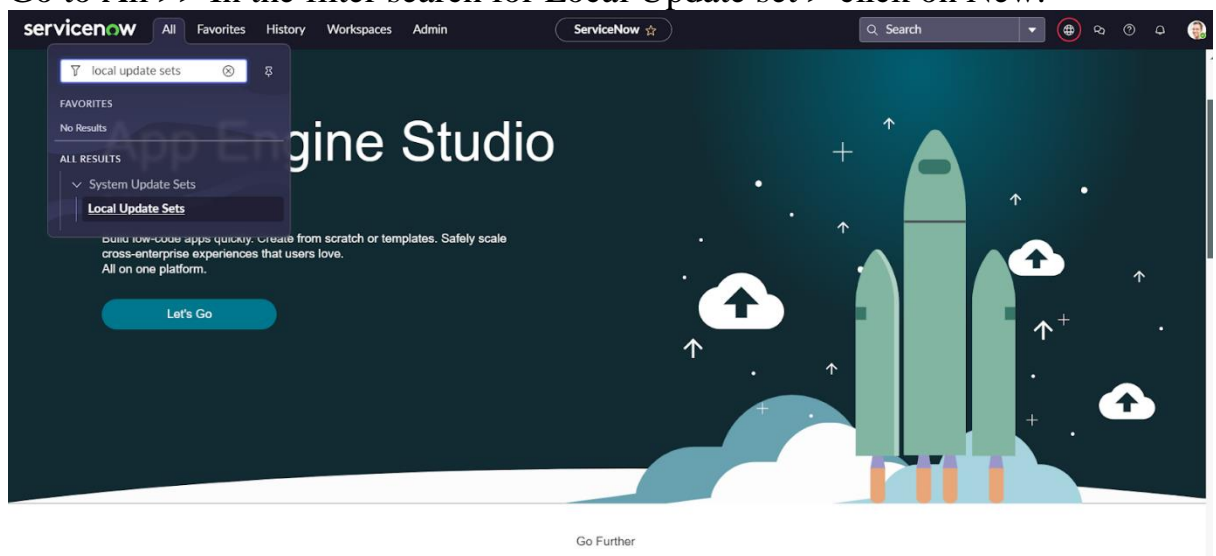
1. Sign up for a developer account on the ServiceNow Developer site "<https://developer.servicenow.com>".
2. Once logged in, navigate to the "Personal Developer Instance" section.
3. Click on "Request Instance" to create a new ServiceNow instance.
4. Fill out the required information and submit the request.
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.

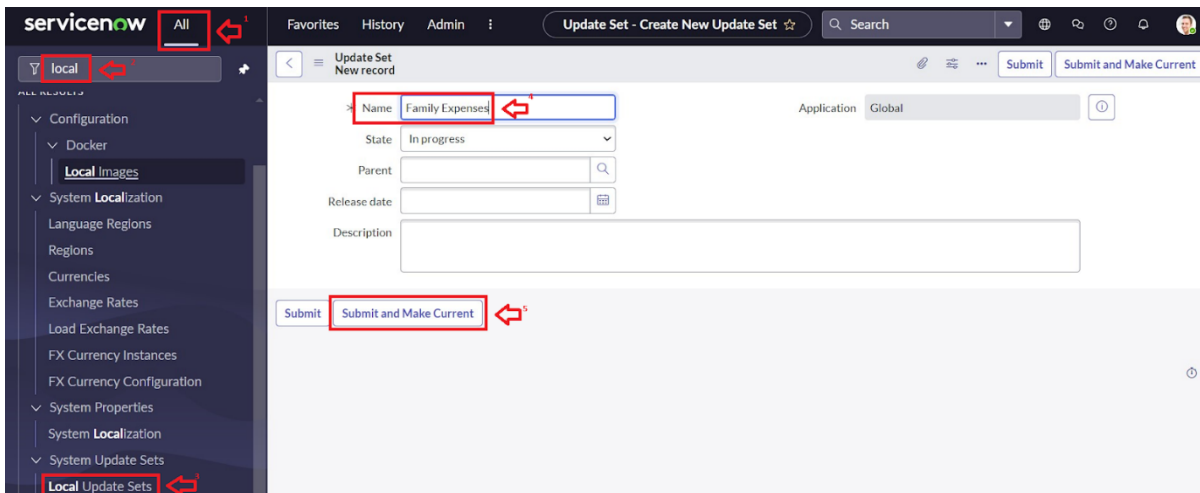


Creation of New Update Set

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.



Creation of Table 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

	Column label	Type	Reference	Max length	Default value	Display
	Number	String				false
	Date	Date				false
	Amount	Integer				false

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

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 : Amount

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

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

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

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.
6. Go to All >> In the filter search for Number Maintenance >> select Number Maintenance
7. Click on New.
8. Enter the below Details:
Table : Family Expenses
Prefix : MFE

9. Click on Submit.

Configure the Form

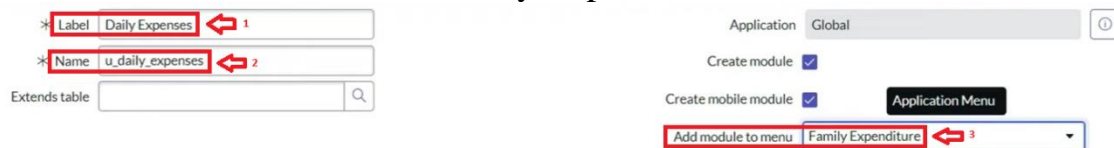
1. Go to All >> In the filter search for Family Expenses >> Open Family Expenses
2. Click on New
3. Go to the Header and right click there>> click on Configure >> Select Form Design
4. Customize or Drag Drop the form as per your requirement.

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.

Creation of Daily Expenses Table

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



The screenshot shows the 'New Table' form. On the left, there are three input fields: 'Label' with the value 'Daily Expenses' (marked with a red box and arrow 1), 'Name' with the value 'u_daily_expenses' (marked with a red box and arrow 2), and 'Extends table' which is empty. On the right, there are checkboxes for 'Create module' and 'Create mobile module', both checked. Below these is an 'Application Menu' dropdown menu, which is currently set to 'Family Expenditure' (marked with a red box and arrow 3). The 'Application' dropdown at the top is set to 'Global'.

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

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.

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.

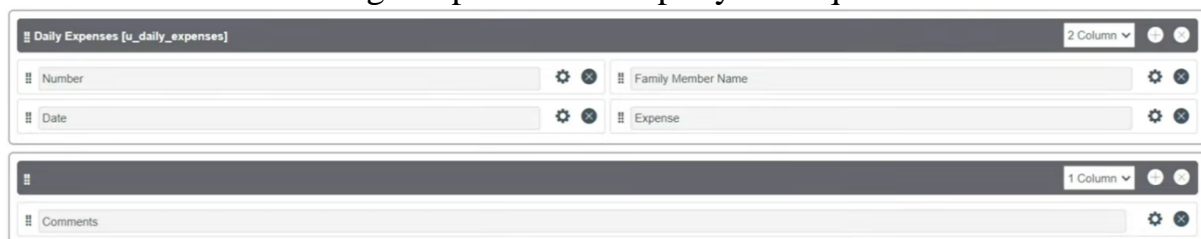
- 5.
6. Go to All >> In the filter search for Number Maintenance >> select Number Maintenance
7. Click on New.
8. Enter the below Details:
Table : Family Expenses
Prefix : MFE



9. Click on Submit.

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.

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.

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



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

Business Rule
New record

ss 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 


Table Daily Expenses [u.daily_expenses] 

Application Global 


Active ☒

Advanced ☒ 

4. In when to run Check Insert and Update

When to run  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 ☒

Delete ☐

Query ☐

Filter Conditions [Add Filter Condition](#) [Add "OR" Clause](#)

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

Role conditions 

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

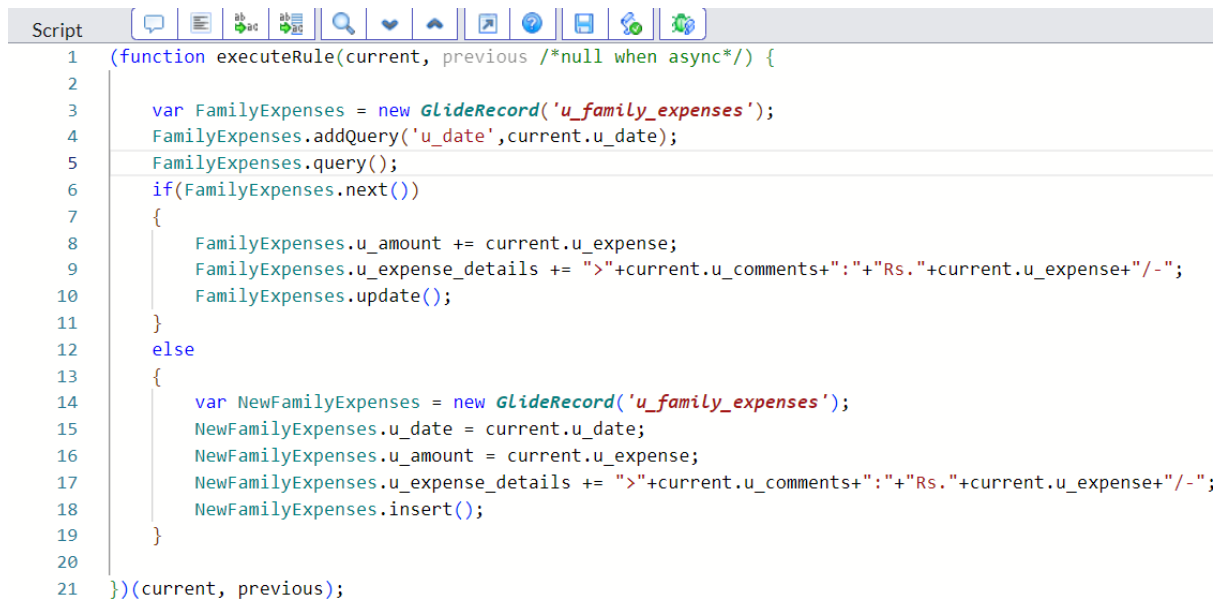
```
(function executeRule(current, previous /*null when async*/) {
    var FamilyExpenses = new GlideRecord('u_family_expenses');
    FamilyExpenses.addQuery('u_date',current.u_date);
    FamilyExpenses.query();
    if(FamilyExpenses.next())
    {
        FamilyExpenses.u_amount += current.u_expense;
        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);

```



```

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.

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.

```

(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);

```

5. Click on Update.

Relationship
Daily Expenses
Update
Delete

Name
Daily Expenses
Application
Global

Advanced
☐
Applies to table
Family Expenses [u_family_expenses]

Queries from table
Daily Expenses [u_daily_expenses]

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);

```

Update
Delete

Conclusion

In conclusion, implementing the "Calculation of Family Expenses Using ServiceNow" project offers numerous benefits to the household. By leveraging the robust capabilities of ServiceNow, family members can efficiently submit, track, and manage their expenses in a centralized and streamlined manner.