

Calculating Family Expenses using ServiceNow

Team ID: NM2025TMID03321

Team Size: 4

Team Leader: Jasmine Hepsi Beula

Team Member: M Gopika

Team Member: M Karthiga

Team Member: I Muneeswari

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:

1. **Input and categorize expenses** easily via a user-friendly interface or mobile portal.
2. **Automate recurring expenses** (e.g., rent, subscriptions).
3. **Track total and category-wise spending** on a monthly/weekly basis.
4. **Set spending limits and receive alerts** when thresholds are crossed.
5. **Generate reports and dashboards** to visualize trends and support better financial decisions.
6. **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

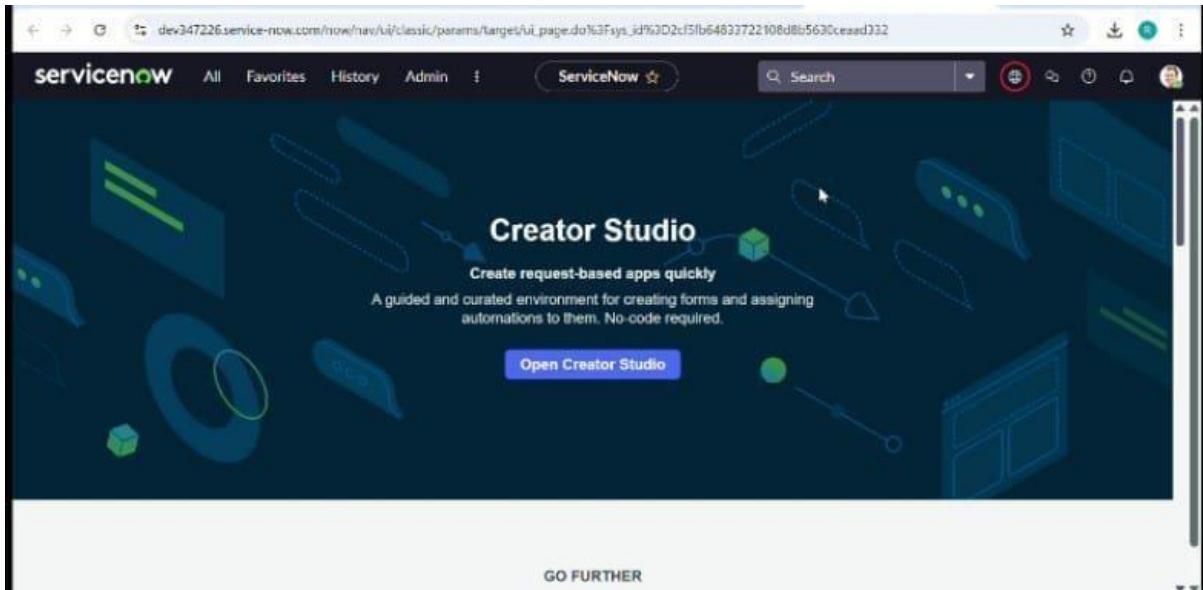
Milestone 1 : Instance

Activity 1: Setting up ServiceNow instance

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.



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.

A screenshot of a ServiceNow update set creation form. The title bar says "Update Set - Create Educational org...". The application scope is set to "Global" and the update set is "Default [Global]". The form has fields for Name (Family Expenses), State (In progress), Parent (empty), Release date (empty), and Description (empty). There are buttons for "Submit" and "Submit and Make Current".

Milestone 3: Table(Family Expenses) Activity 1: Creation of Family Expenses Table

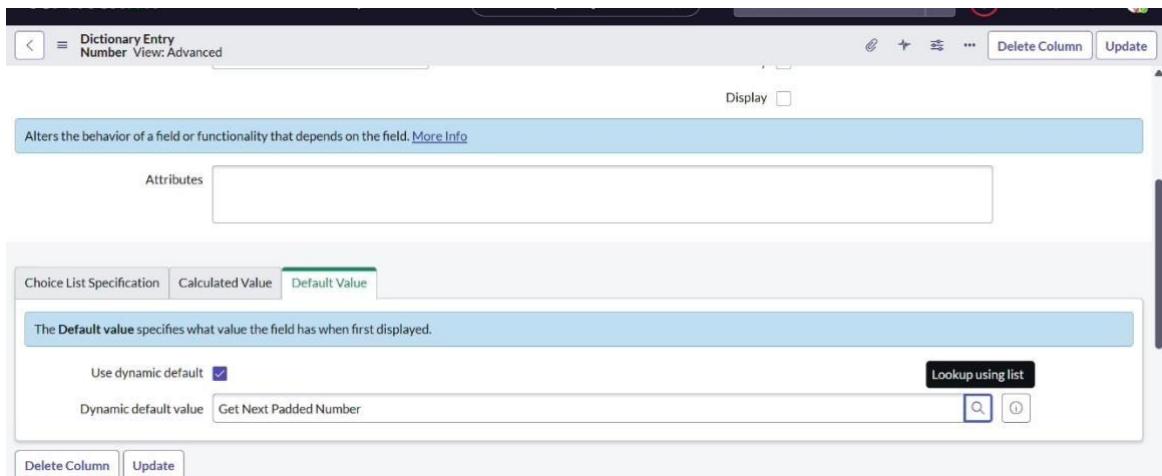
1. Go to All > In the filter search for Tables > click on New.
2. Enter the Details:
 - Label : Family Expenses
 - Name : Auto-Populated
 - New menu name : Family Expenditure
3. Go to the Header and right click there>> click on Save.

The screenshot shows the ServiceNow 'Table - New Record' page. At the top, there are tabs for 'All', 'Favorites', 'History', and 'Admin'. The main title is 'Table - New Record'. Below the title, it says 'Table New record'. A yellow banner at the top states: 'ServiceNow recommends creating custom tables in scoped applications. To learn more about creating scoped applications, click [here](#)'. A blue info bar below it 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.' There are several input fields and checkboxes:

- * Label: Family Expenses
- * Name: u_family_expenses
- Extends table: (empty)
- Application: Global
- Create module:
- Create mobile module:
- Add module to menu: -- Create new --
- New menu name: Family Expenditure
- Remote Table:

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
 - 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.

Table - Family Expenses						
Dictionary Entries		Column label	Type	Reference	Max length	Default value
		Updated	Date/Time	(empty)	40	false
		Created by	String	(empty)	40	false
		Sys ID	Sys ID (GUID)	(empty)	32	false
		Created	Date/Time	(empty)	40	false
		Updated by	String	(empty)	40	false
		Updates	Integer	(empty)	40	false
x	/	Number	String			false
x	/	Date	Date			false
x	/	Amount	Integer			false
x	/	Expense Details	String		800	false

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 table named 'Family Expenses'. The 'Prefix' field is set to 'MFE'. A mandatory 'Number' field is present with a value of '1,000'. The 'Application' is set to 'Global', and the 'Number of digits' is set to '7'. Below the form, there is a 'Submit' button and a 'Related Links' section with 'Show Counter'.

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 'Form Design' interface for the 'Family Expenses' table. The main area displays two columns: 'Number' and 'Amount' in the top section, and 'Date' in the bottom section. The 'Expense Details' section is collapsed. On the left, there is a sidebar with tabs for 'Fields' and 'Field Types', and lists for 'Created', 'Created by', 'Updated', 'Updated by', and 'Updates'. The 'Formatters' section includes 'Activities (filtered)' and 'Contextual Search Results'.

Milestone 4: Table(Daily Expenses)

Activity 1: Creation of Table(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' page. At the top, it says 'Table - New Record' and has tabs for 'All', 'Favorites', 'History', and 'Admin'. Below that is a search bar and a toolbar with icons for search, refresh, and other actions. The main area has a yellow header bar with a note about creating custom tables in scoped applications. A blue info bar below it explains what a table is. On the left, there are fields for 'Label' (set to 'Daily Expenses'), 'Name' (set to 'u_daily_expenses'), and 'Extends table' (empty). On the right, there are checkboxes for 'Application' (set to 'Global'), 'Create module' (checked), 'Create mobile module' (checked), and 'Add module to menu' (set to 'Family Expenditure'). There is also a 'Remote Table' checkbox which is unchecked. At the bottom right are 'Submit' and 'Cancel' buttons.

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.

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		600	
Comments	String		600	

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.

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

Use dynamic default [Lookup using list](#)

Dynamic default value [Search](#) [Help](#)

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 ServiceNow interface for creating a new record. The top navigation bar includes links for All, Favorites, History, and a search bar. The main title is "Number - New Record". The form itself has several input fields: "Table" set to "Daily Expenses", "Prefix" set to "DFE", "Number" set to "1,000", "Application" set to "Global", and "Number of digits" set to "7". Below the form is a "Submit" button and a "Related Links" section containing a link to "Show Counter".

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 tool. The main area displays a grid layout for the "Daily Expenses [u_daily_expenses]" table. It consists of two columns: the first column contains fields for "Number" and "Family Member Name"; the second column contains fields for "Date" and "Expense". Below this grid is a single-column section containing a field for "Comments". On the left side, there is a sidebar titled "Fields" which lists various fields such as "Created", "Updated", and "Activities (filtered)". At the top right, there are "Undo" and "Save" buttons.

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.

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  Turn on ECMAScript 2021 (ES12) mode
1 (function refineQuery(current, parent) {
2
3     // Add your code here, such as current.addQuery(field, value);
4
5 })(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+"/-";
    }
})
```

$+=$

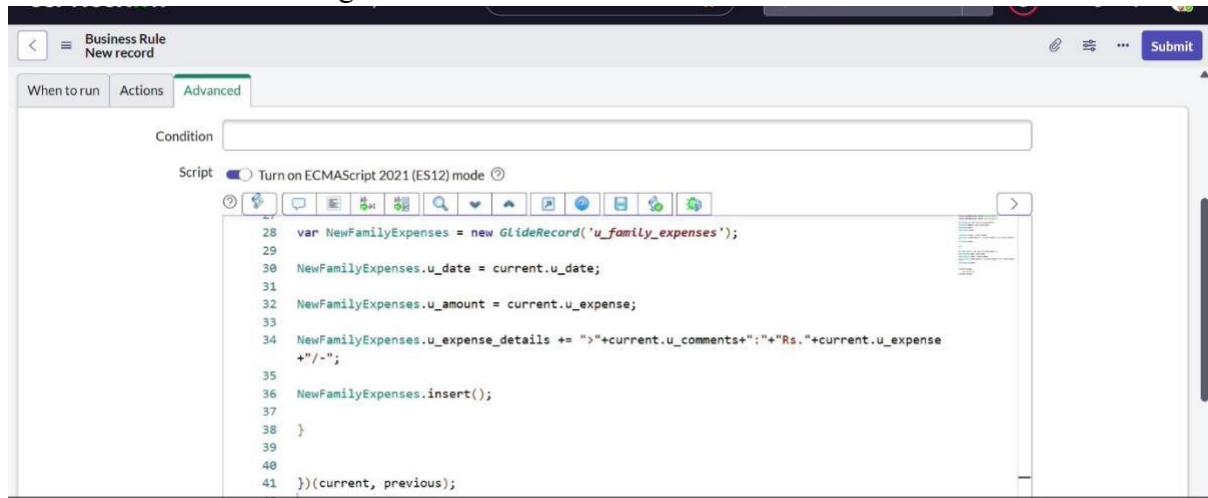
```

FamilyExpenses.update();
}
else
{
varNewFamilyExpenses = 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.

```
(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.
```

Servicenow

All Favorites History Relationship - Daily Expenses Search

Relationship Daily Expenses

Name: Daily Expenses Application: Global

Advanced: Applies to table: Family Expenses [u_family_expe...]

Queries from table: Daily Expenses [u_daily_expens...]

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 Turn on ECMAScript 2021 (ES12) mode

```
4 // Add your code here, such as current.addQuery(field, value);
5
6 current.addQuery('u_date',parent.u_date);
7
8 current.query();
9
10
11 })(current, parent);
```