<u>Calculating Family Expenses using Service Now</u>

Team ID: NM2025TMID01342

Team Size: 5

Team Leader: Anuja A J

Team member: Arasi A

Team member: Arul Selvam M

Team member: Ashick Vijin V S

Team member: Ashika Mol N S

AIM:

To perform calculating familiy expenses using ServiceNow Developer.

PROBLEM STATEMENT:

- Managing household finances is essential for every family to ensure financial stability and better planning. This project aims to develop a simple and efficient system using ServiceNow to record, track, and calculate monthly family expenses. ServiceNow, being a powerful cloudbased platform, offers custom application development that can be leveraged for personal finance tracking.

OBJECTVE:

➢ To build a Family Expense Management System using ServiceNow that allows users to:

➢ Enter daily or monthly expenses under different categories (e.g., groceries, rent, transport, education, etc.) ➢ View summary reports of total and category- wise expenses ➢ Set monthly budgets and receive alerts if spending exceeds limits

➢ Generate downloadable reports for any time period

SKILL:

1. ServiceNow Application Development ○ Creating custom tables ○ Designing forms and views ○ Using Flow Designer for automation
2. Workflow & Automation ○ Using ServiceNow Flow Designer to automate calculations and notifications
3. Database Management

- Creating and managing custom tables for storing expenses and budgets

4. Reporting & Dashboard Design
   - Designing ServiceNow reports

Building graphical dashboards for data visualization

<u>TASK INITIATION:</u>

STEP 1: Setting up ServiceNow Instance

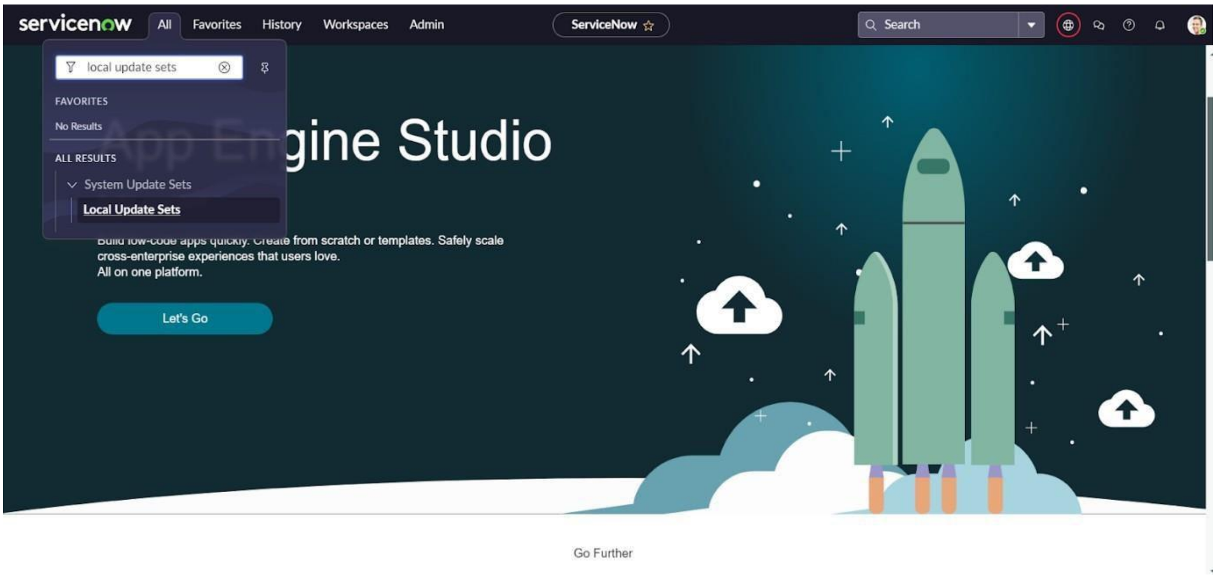
ACTIVITY 1: Setting up ServiceNow Instance

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.

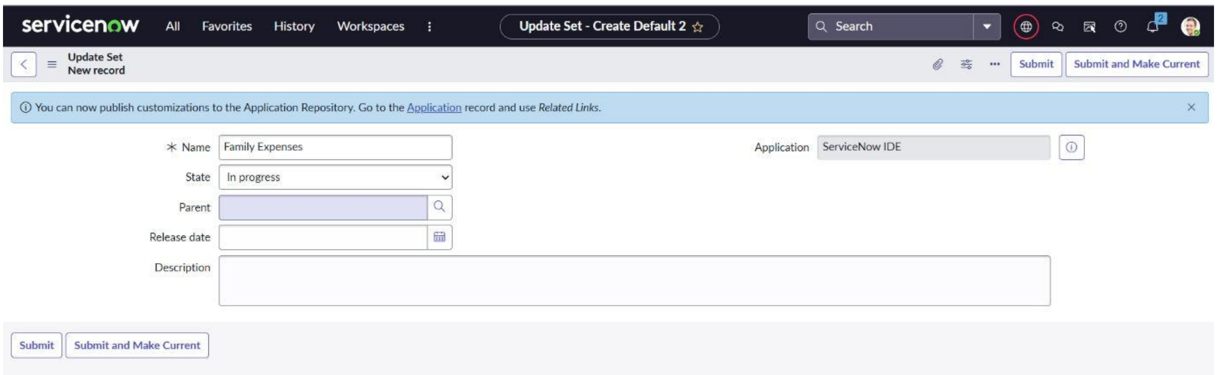
STEP 2: Creation of New Update Set

ACTIVITY 1: Creation of New Update Set

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

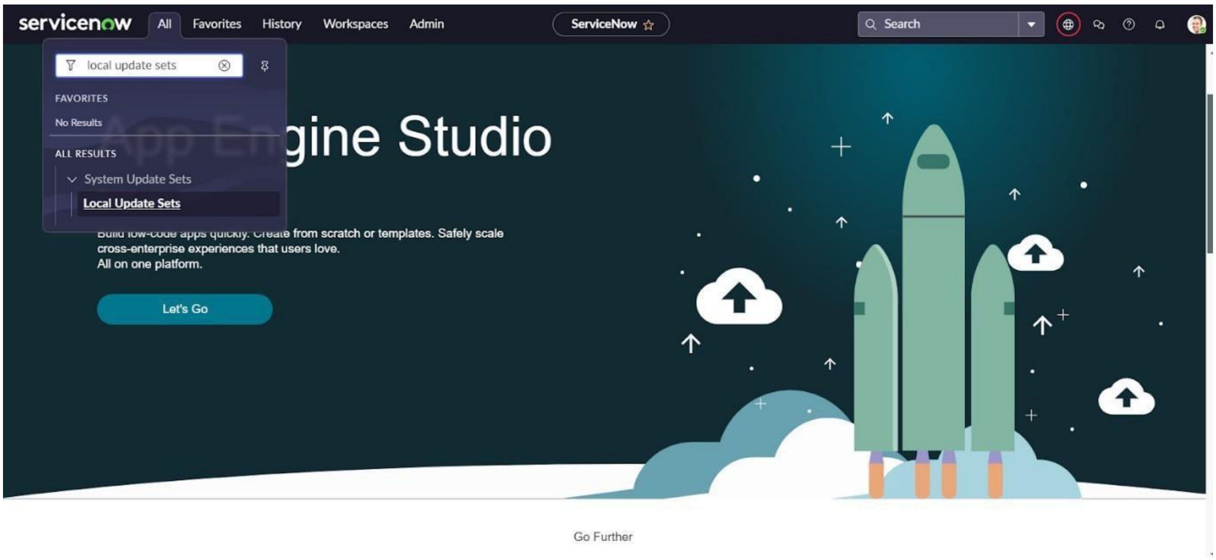


2. Enter the Details as: Name: Family Expenses
3. Then click on Submit and Make current.



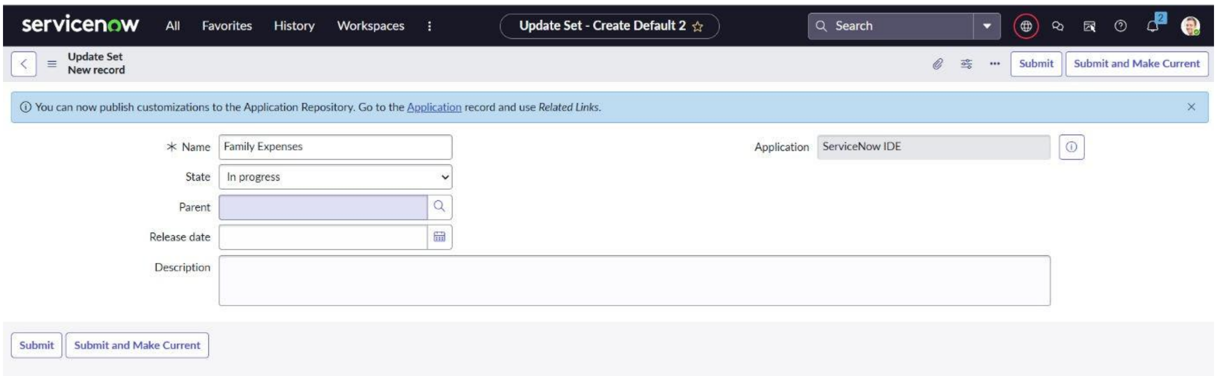
STEP 3: Creation of Table ACTIVITY 1: Creation of New Update Set

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



1. Enter the Details as: Name: Family Expenses

2.Then click on Submit and Make current



STEP 4: Creation of Table

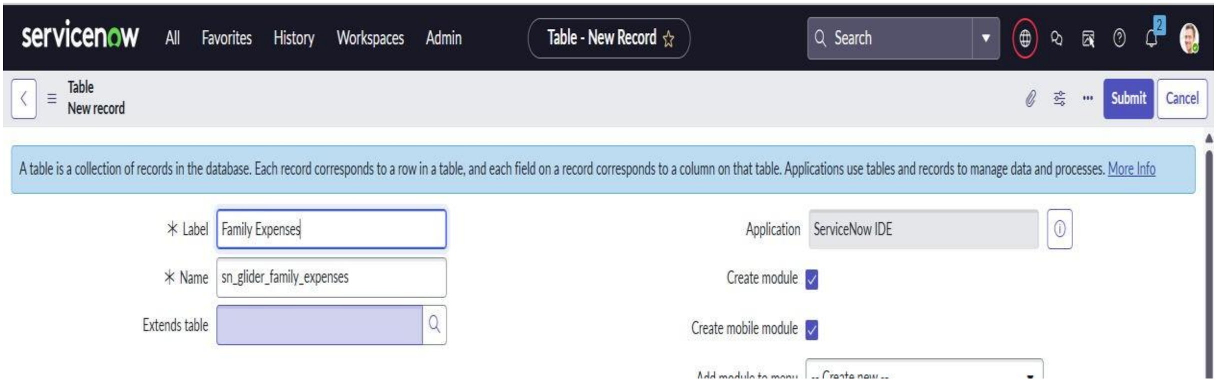
ACTIVITY 1: Creation of Family Expenses Table

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



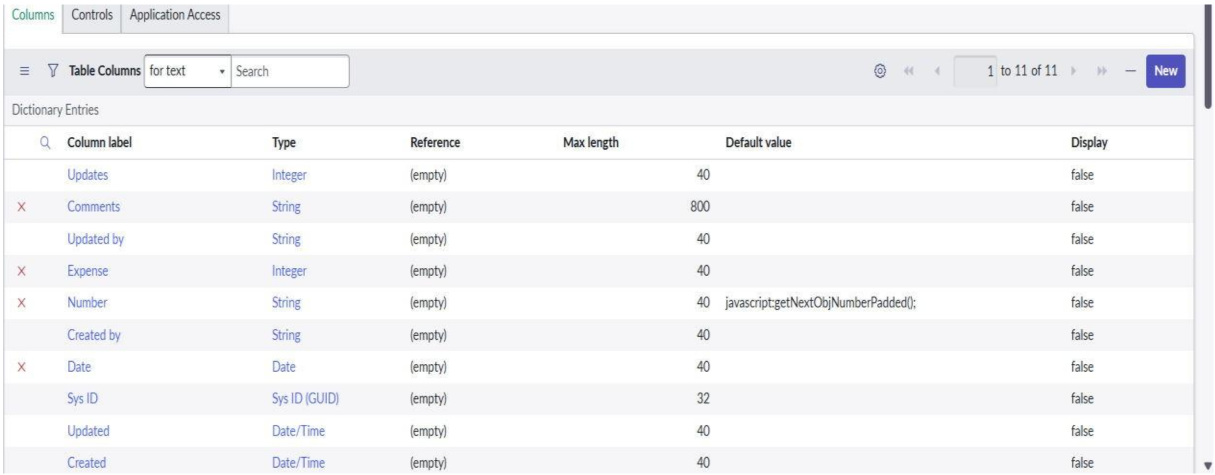
1. Go to the Header and right click there>> click on Save.

2. 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 clicks on insert a new row again
   4. Give the details as: Column label: Date Type: Date
   5. Double clicks on insert a new row again
   6. Give the details as: Column label: Amount Type: Integer
   7. Double clicks on insert a new row again
   8. Give the details as:

      Column label: Expense Details Type: String

      Max length: 800



Columns | Controls | Application Access

Table Columns | for text | Search | 1 to 11 of 11 | New

Dictionary Entries

	Column label	Type	Reference	Max length	Default value	Display
	Updates	Integer	(empty)	40		false
×	Comments	String	(empty)	800		false
	Updated by	String	(empty)	40		false
×	Expense	Integer	(empty)	40		false
×	Number	String	(empty)	40	javascript:getNextObjNumberPadded();	false
	Created by	String	(empty)	40		false
×	Date	Date	(empty)	40		false
	Sys ID	Sys ID (GUID)	(empty)	32		false
	Updated	Date/Time	(empty)	40		false
	Created	Date/Time	(empty)	40		false

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

ACTIVITY 3: Making Number Field an Auto- Number

1.Double click on the Number Field/Column

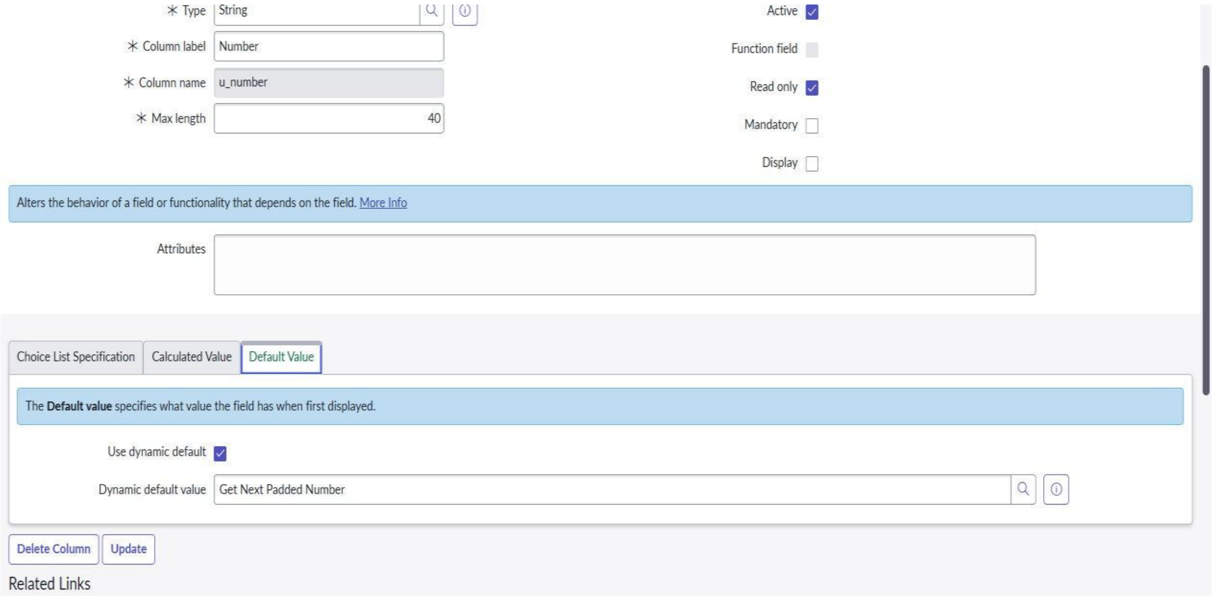
2.Go down and double click on Advanced view

3.In Default Value:

Use dynamic default: check the box Dynamic default

value: Get Next Padded Number

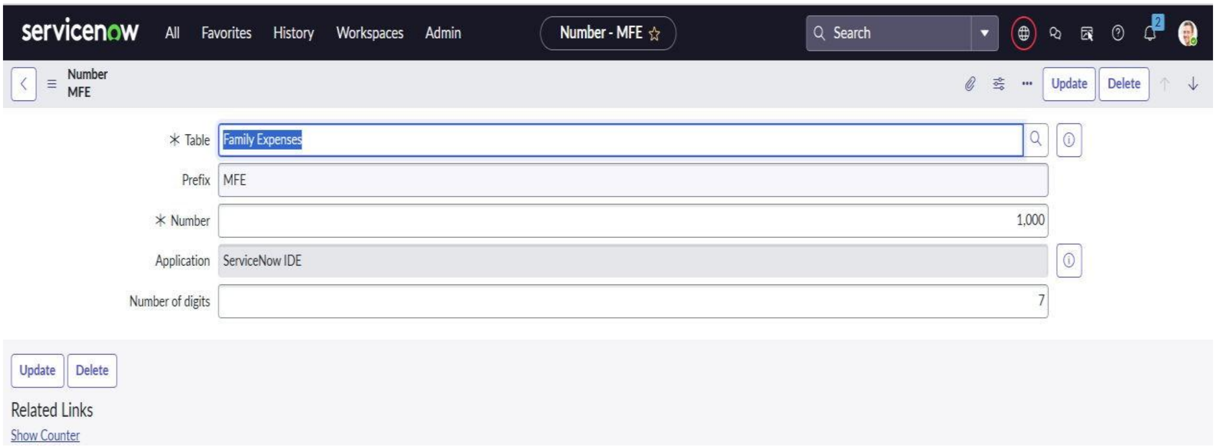
1. Click on Update.



1. Go to All >> In the filter search for Number Maintenance>>select Number Maintenance 2.Click on New

3.Enter the below Details: Table:

Family Expenses Prefix: MFE



9.Click on Submit.

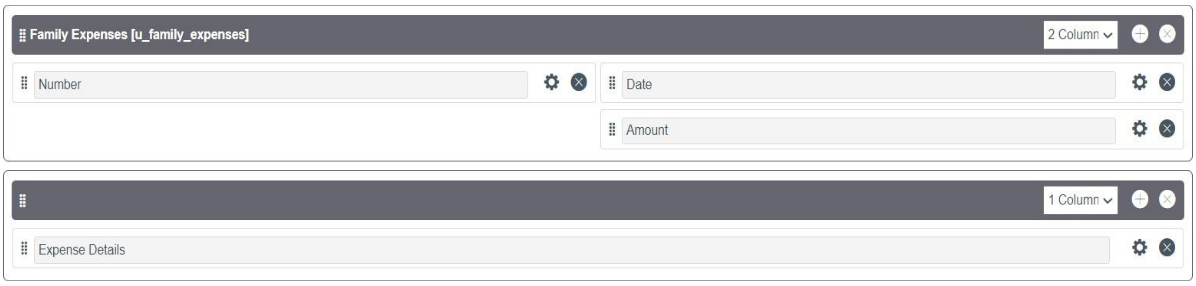
ACTIVITY 4: Configure the Form

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.

STEP 5: Creation of Table (Daily Expenses)

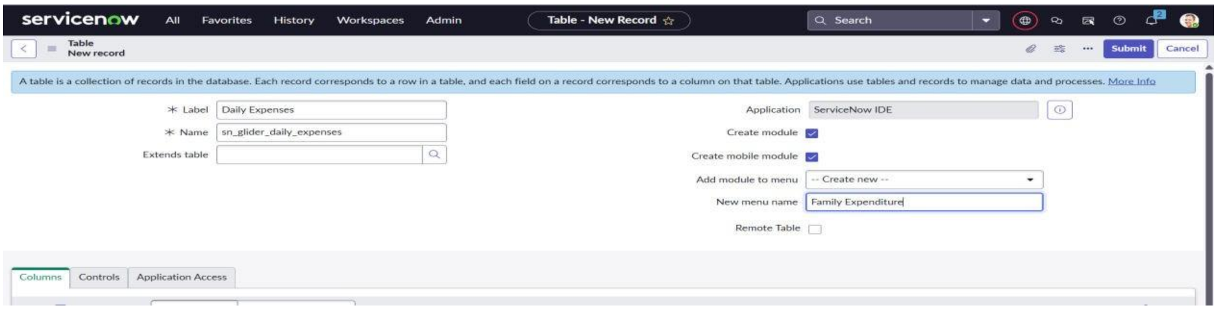
ACTIVITY 1: Creation of Daily Expenses Table

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



1. Double clicks on insert a new row again

2. Give the details as: Column label: Expense Type: Integer 3.

   Double click on insert a new row again

4. Give the details as:

   Column label: Family Member Name Type:

   Reference

   Max length: 800

5. Double clicks on insert a new row again

6. Give the details as: Column label: Comments Type: String Max length: 800

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

ACTIVITY 3: Making Number Field an Auto- Number

1. Double clicks on the Number Field/Column.

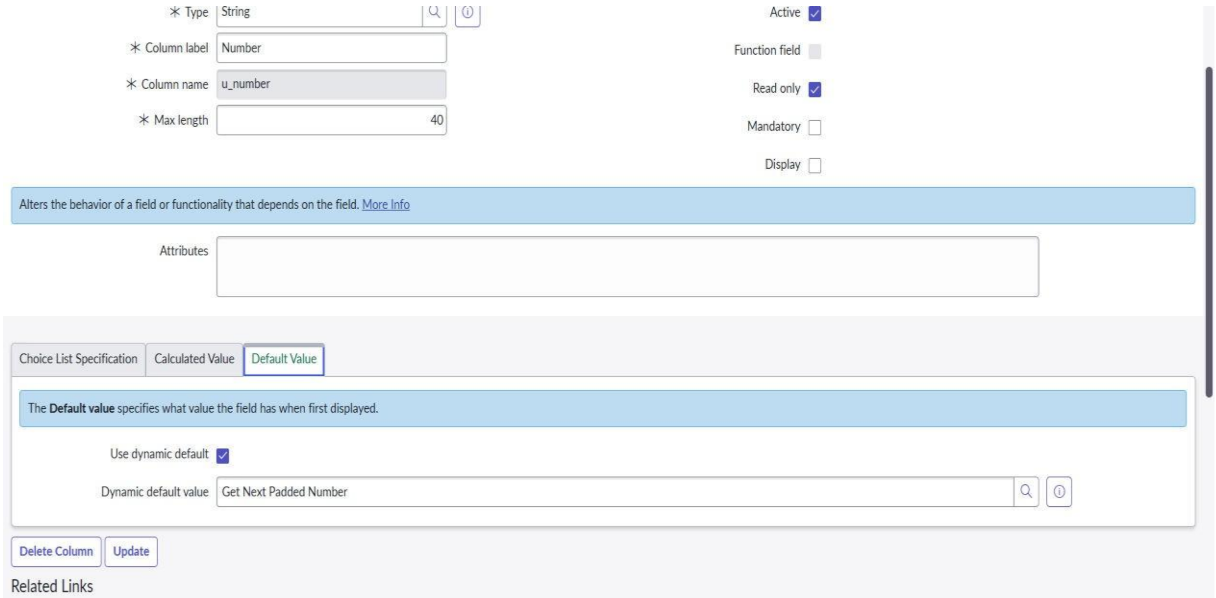
2. Go down and double click on Advanced view 3. In Default

   Value:

   Use dynamic default: check the box

   Dynamic default value: Get Next Padded Number

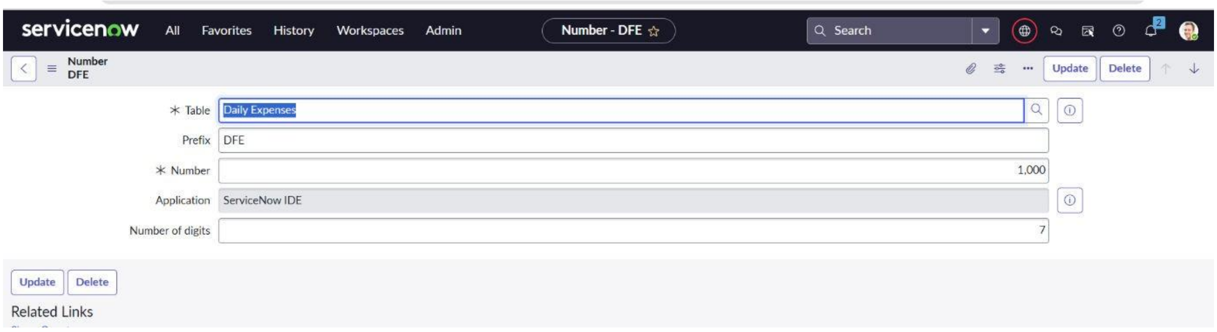
4. Click on Update.



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



9. Click on Submit.

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

5. Make Date, Family Member Name Mandatory Field by clicking on the gear icon and checking Mandatory 6. Click on Save.

STEP 6: Creation of Relationship

ACTIVITY 1: Creation of Relationship between Family Expenses and Daily Expenses table

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

   Name : Daily Expenses

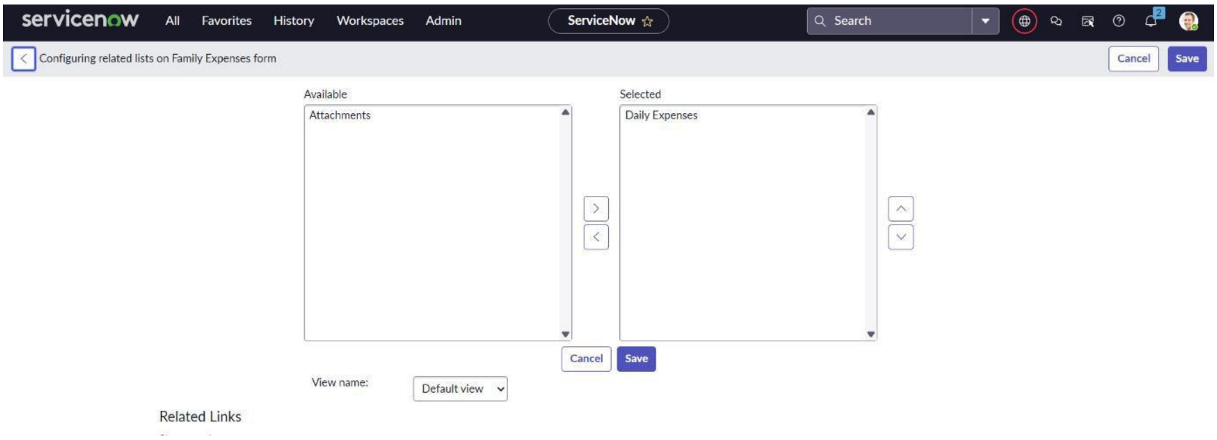
   Applies to table : Select Family Expenses

   Daily Expenses : Select Daily Expenses

4. Click Save.

STEP 7: 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

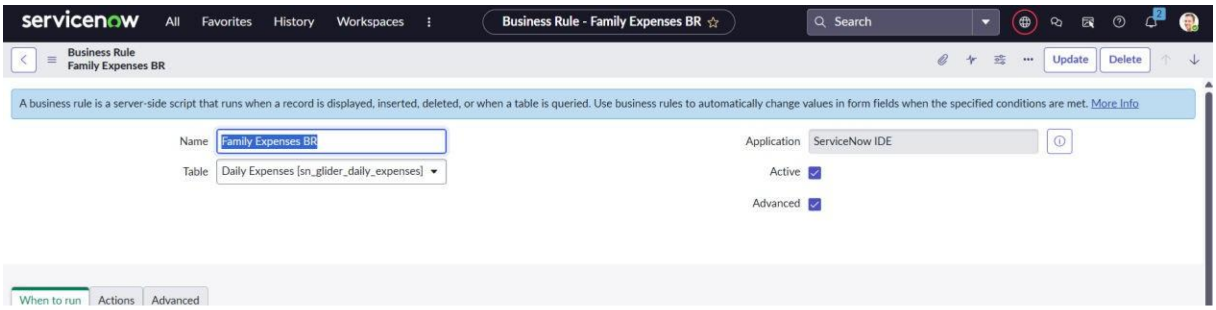


STEP 8: Creation of Business Rules

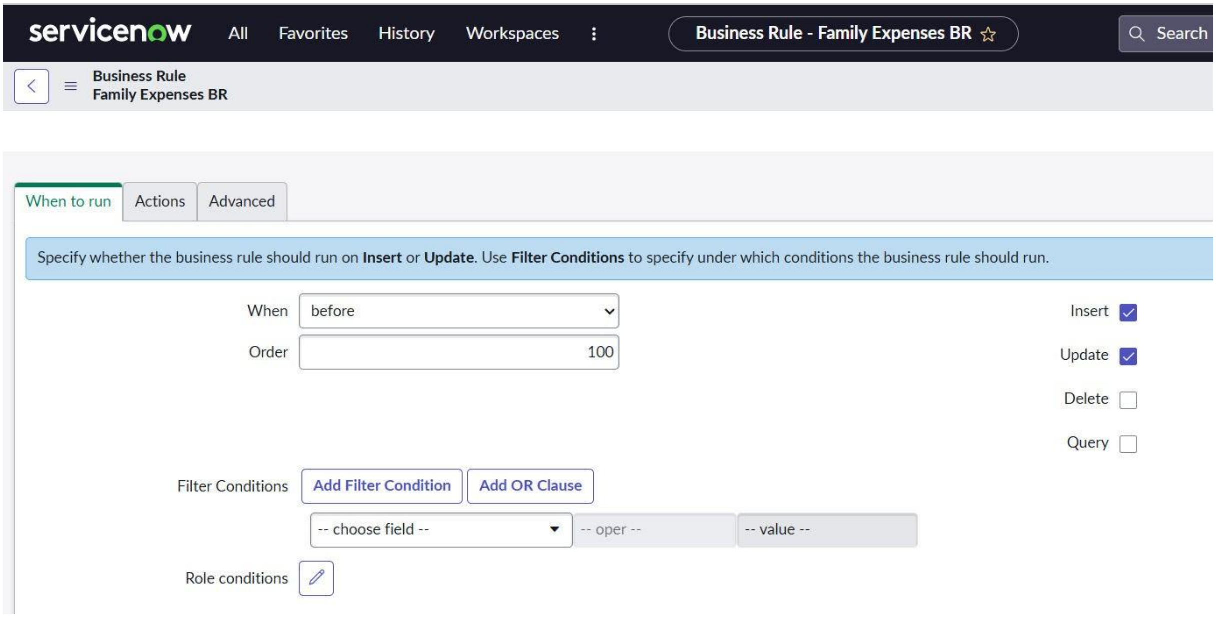
1. Go to All >> In the filter search for Business Rules.
2. Under System Definition Select Business Rules then click on New.
3. Enter the Details:

Name : Family Expenses BR

Table : Select Daily Expenses Check Advanced



4.In when to run Check Insert and Update



1.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_ex pense+"/-";
   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_ex pense+"/-";
   NewFamilyExpenses.insert();}

  })(current, previous);
```

Script

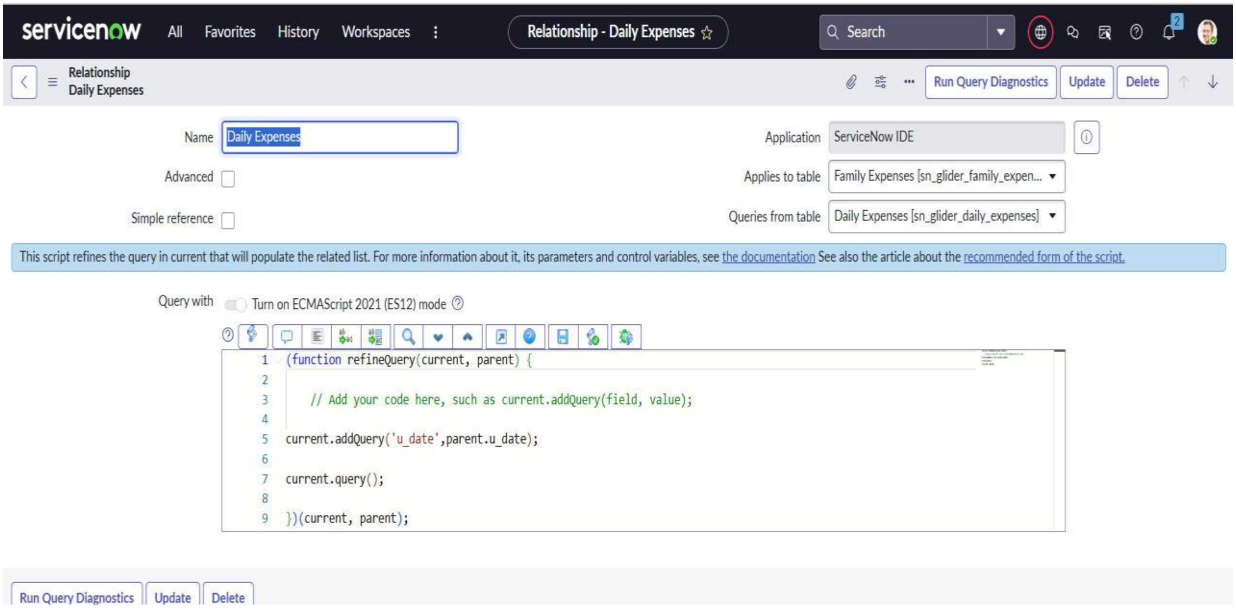
```
(function executeRule(current, previous /*null when async*/) {

    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);
```

STEP 9: 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) {

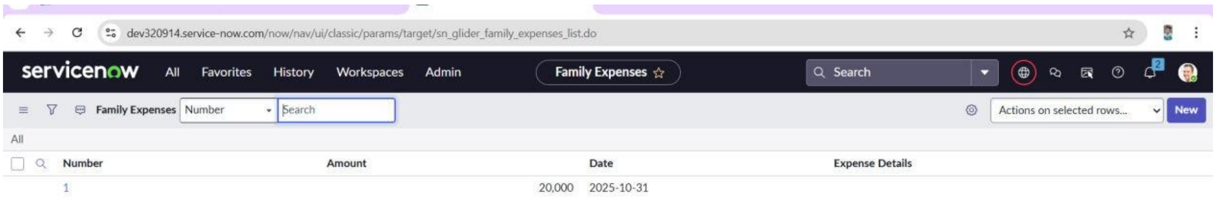
// Add your code here, such as current.addQuery(field, value); current.addQuery('u_date',parent.u_date); current.query();

})(current, parent);

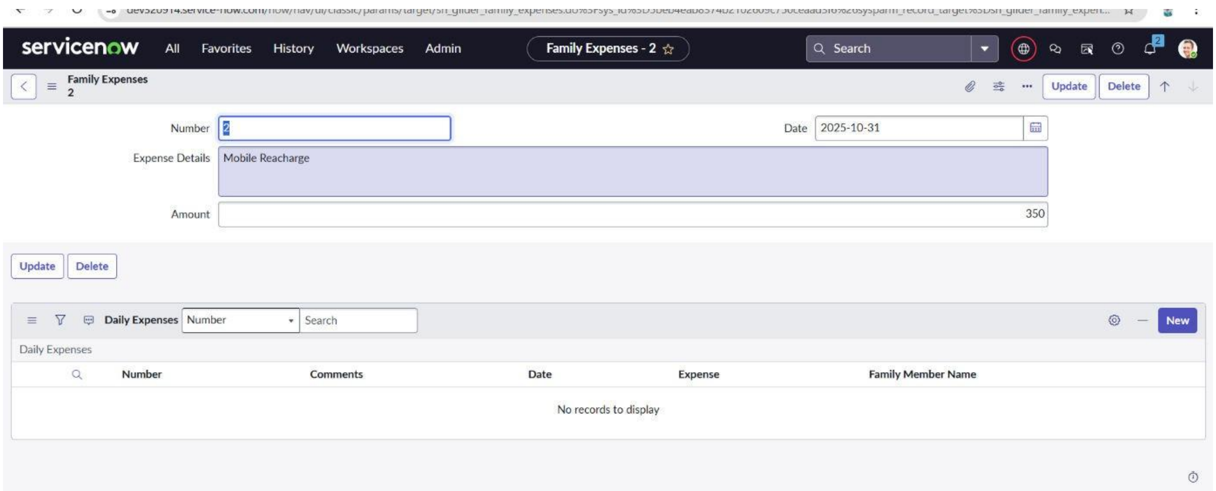


5.Click on Update.

Example Output:01



Example Output:02



Conclusion:

The Family Expense Calculation System on ServiceNow offers an easy and efficient way to manage household finances. With features like expense tracking, budget setting, and reporting, it helps users monitor spending and make informed financial decisions. The system’s scalability and user-friendly interface ensure it supports families of all sizes, promoting better financial planning and control.