

CALCULATING FAMILY EXPENSES USING SERVICE NOW

Team Id: NM2025TMID18167

Team Members:

Team Leader: HARIPRASAD A

Team Member : RAGHUL S

Team Member : ERAIYANBU E

Team Member : HARISH K P

Problem Statement: 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.

Objectives: The objective of this project is to develop a family expense management system on the ServiceNow platform that enables efficient tracking, categorization, and budgeting of expenses. It aims to provide real-time monitoring, user-friendly dashboards, and insightful reports to support informed financial decisions. The system will ensure scalability for different family sizes, promote financial discipline through spending analysis, and maintain data security using ServiceNow's robust features.

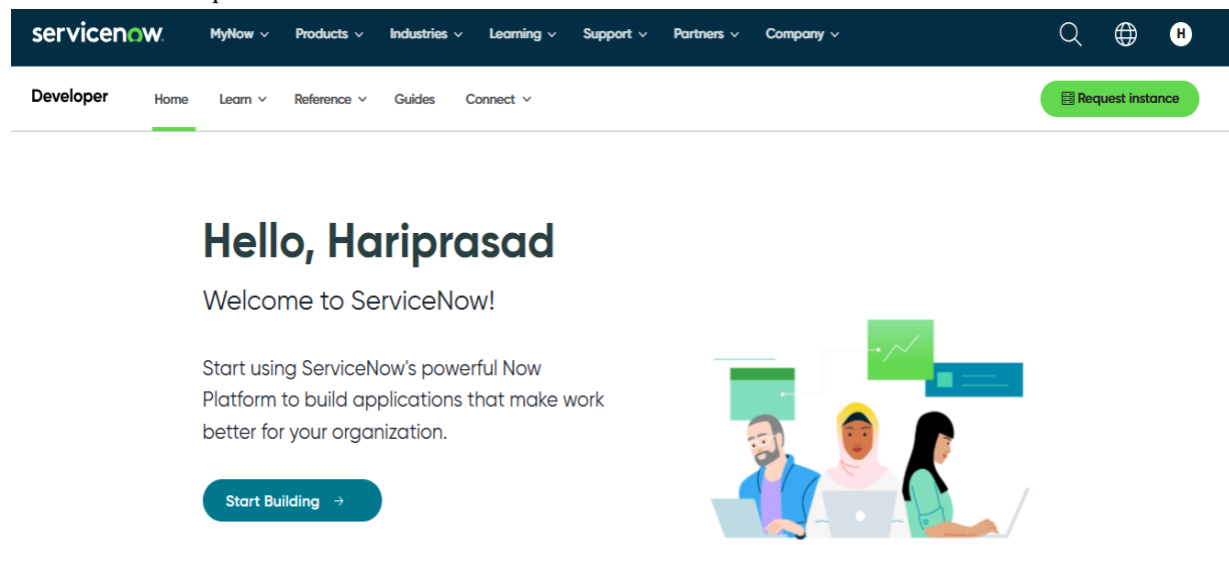
Skills:

TASK INITIATION

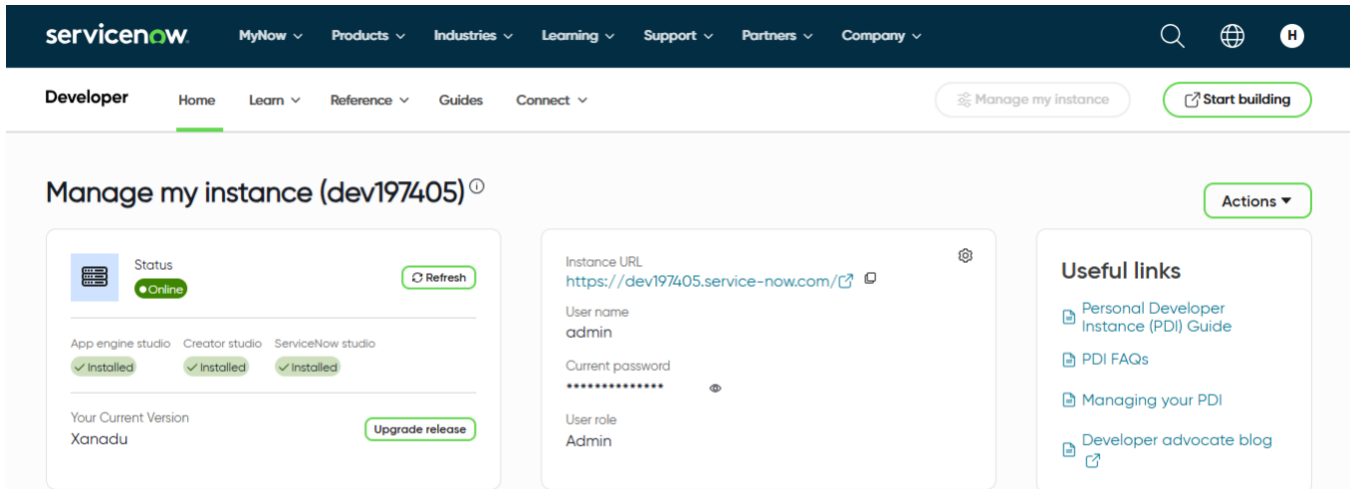
Milestone 1 : Setting up ServiceNow Instance

Activity : 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.



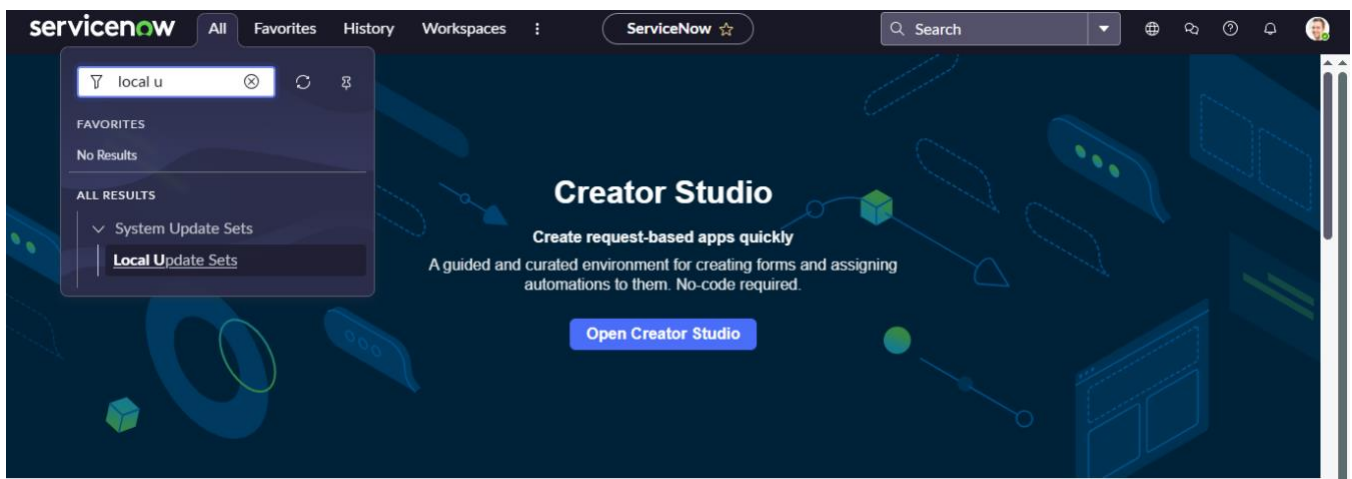
The screenshot shows the ServiceNow Developer portal for instance dev197405. The top navigation bar includes links for MyNow, Products, Industries, Learning, Support, Partners, and Company. Below this, a secondary bar has Developer, Home, Learn, Reference, Guides, and Connect. On the right, there are buttons for 'Manage my instance' and 'Start building'. The main content area is titled 'Manage my instance (dev197405)' and includes an 'Actions' dropdown. It features three panels: 1) Status panel showing 'Online' status and installed components (App engine studio, Creator studio, ServiceNow studio) with an 'Upgrade release' button for the current version 'Xanadu'. 2) Instance details panel showing the Instance URL, User name 'admin', Current password, and User role 'Admin'. 3) Useful links panel with links to the Personal Developer Instance (PDI) Guide, PDI FAQs, Managing your PDI, and Developer advocate blog.

- Now you will navigate to the ServiceNow.

Milestone 2 : Creation of New Update Set

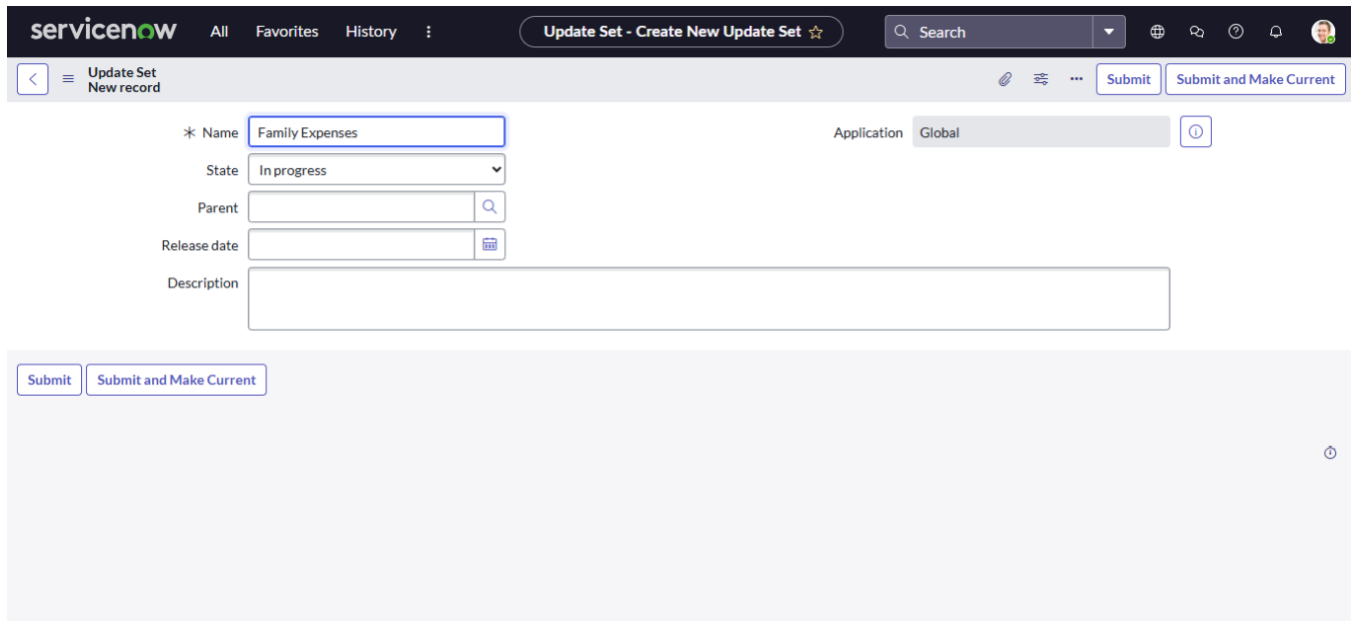
Activity: Creation of New Update Set

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



The screenshot shows the ServiceNow Creator Studio interface. The top navigation bar includes 'All', 'Favorites', 'History', 'Workspaces', and a 'ServiceNow' button. A search bar is on the right. A search filter 'local u' is applied, showing 'No Results' in the Favorites section and 'Local Update Sets' under 'ALL RESULTS'. The main content area is titled 'Creator Studio' and includes the text 'Create request-based apps quickly' and 'A guided and curated environment for creating forms and assigning automations to them. No-code required.' There is an 'Open Creator Studio' button.

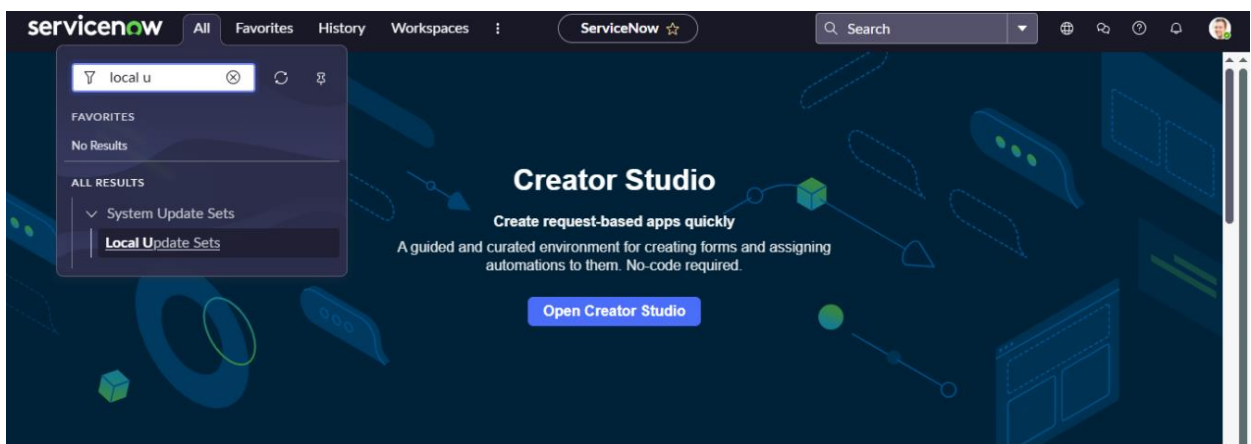
- Enter the Details as:
Name : Family Expenses
- Then click on Submit and Make current.



Milestone 3: Creation of Table

Activity: Creation of New Update Set

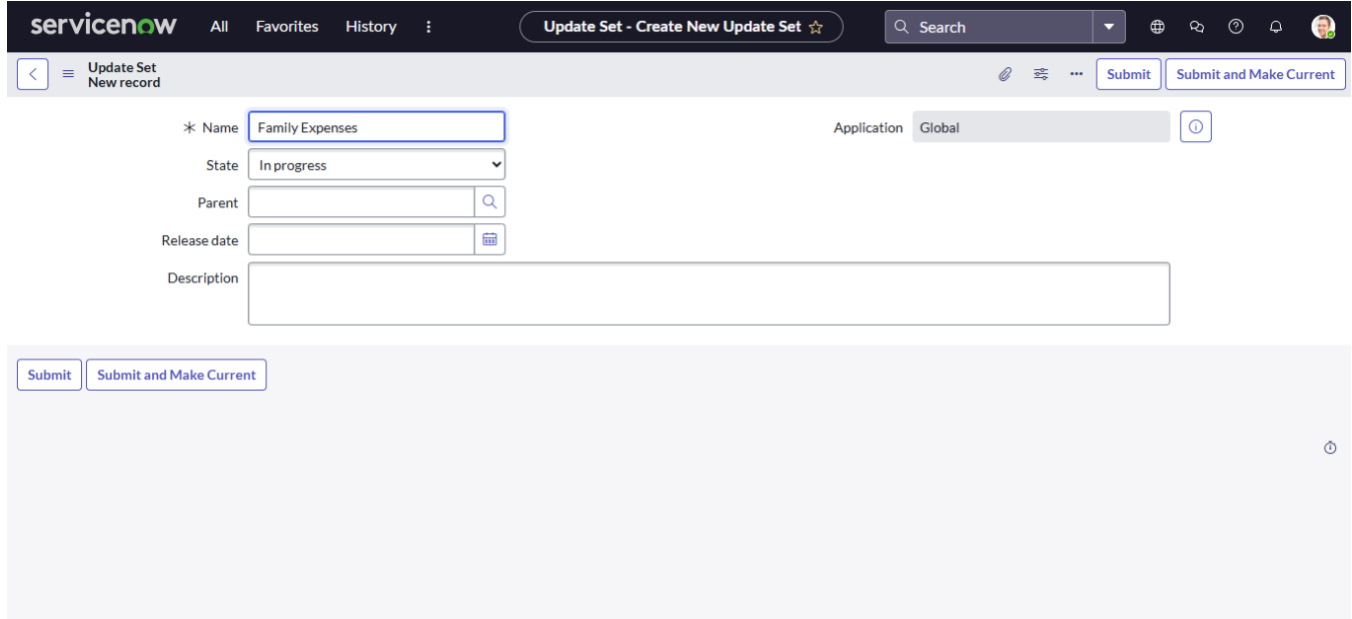
1. Go to All >> In the filter search for Local Update set > click on New



2. Enter the Details as:

Name : Family Expenses

- Then click on Submit and Make current.



The screenshot shows the ServiceNow 'Update Set' form. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', and a search bar. The main header shows 'Update Set - Create New Update Set' with a star icon. Below this, there's a sub-header 'Update Set' and 'New record'. The form fields are as follows:

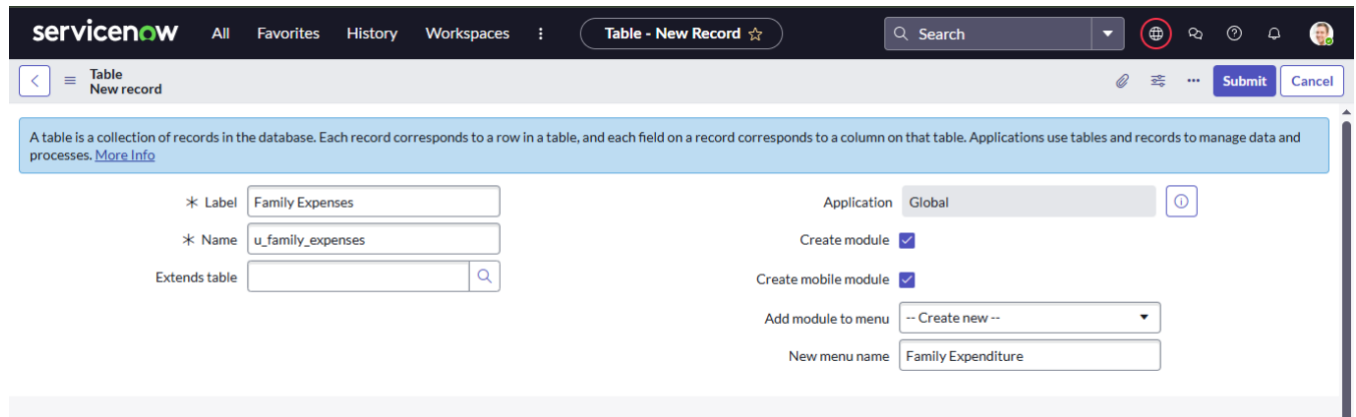
- * Name:
- State:
- Parent:
- Release date:
- Description:
- Application:

At the bottom, there are two buttons: 'Submit' and 'Submit and Make Current'.

Milestone 4: Creation of Table

Activity 1: Creation of Family Expenses Table

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



servicenow All Favorites History Workspaces **Table - New Record** Search

Table New record Submit Cancel

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 Application

* Name Create module ☒

Extends table Create mobile module ☒

Add module to menu

New menu name

Activity 2: Creation of Columns(Fields)

1. Near Columns Double click near insert a new row.
2. Give the details as:
 - i. Column label : Number
 - ii. Type : String
3. Double click on insert a new row again
4. Give the details as:
 - i. Column label : Date
 - ii. Type : Date
5. Double click on insert a new row again
6. Give the details as:
 - i. Column label : Amount
 - ii. Type : Integer
7. Double click on insert a new row again
8. Give the details as:
 - i. Column label : Expense Details
 - ii. Type : String
 - iii. Max length : 800\

servicenow All Favorites History Admin Table - Family Expenses Search

Table Columns for text Search

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
Updated by	String	(empty)		40	false
Updates	Integer	(empty)		40	false
Updated	Date/Time	(empty)		40	false
Created by	String	(empty)		40	false
Created	Date/Time	(empty)		40	false
Sys ID	Sys ID (GUID)	(empty)		32	false
Number	String				false
Date	Date				false
Amount	Integer				false
Expense Details	String		800		false
Insert a new row...					

Delete Update Delete All Records

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

Activity 3: Making Number Field an Auto-Number

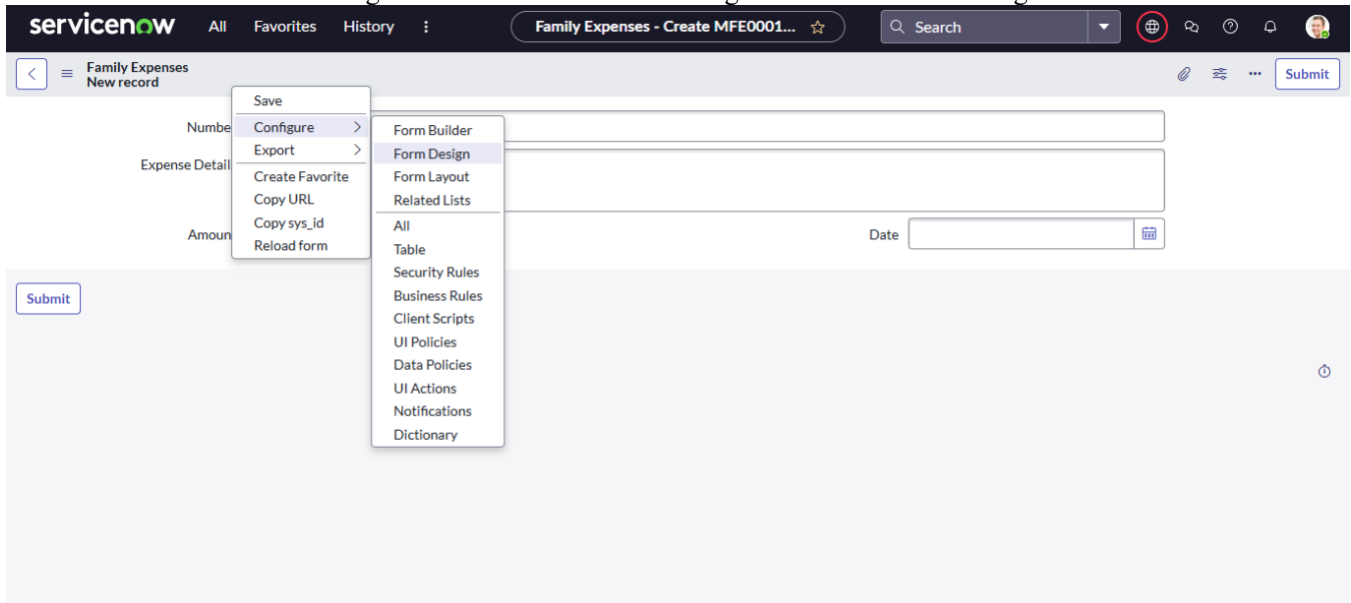
- Double click on the Number Field/Column.
- Go down and double click on Advanced view
- In Default Value:
 - Use dynamic default : check the box
 - Dynamic default value : Get Next Padded Number
- Click on Update.

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

8. Click on Submit.

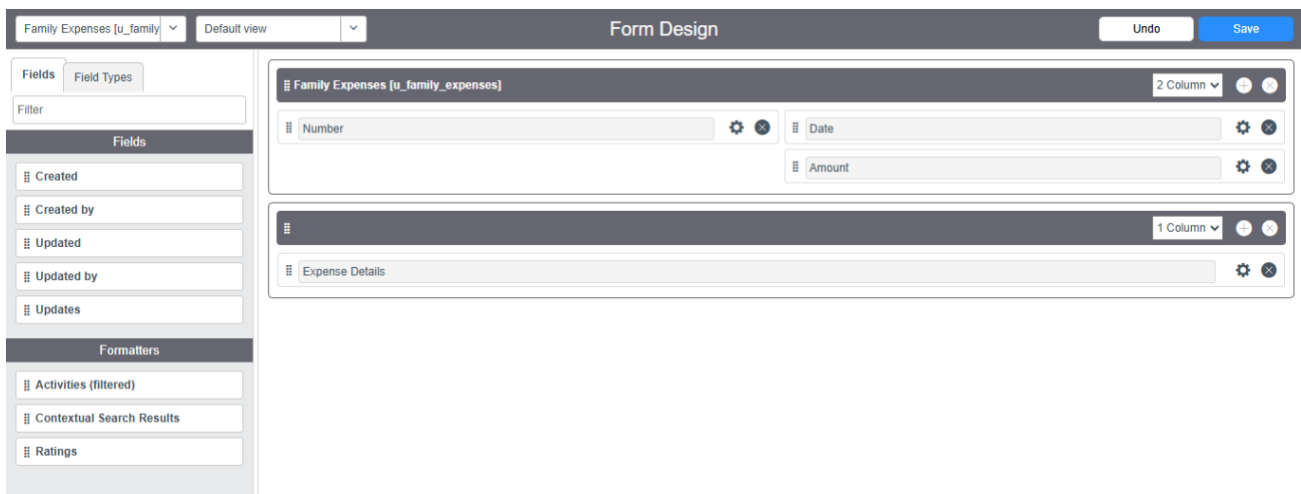
Activity 4: Configure the Form

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



The screenshot shows the ServiceNow interface for creating a new record in the 'Family Expenses' table. The header bar includes the 'servicenow' logo, navigation tabs (All, Favorites, History), and a search bar. The main content area displays the form fields: 'Number', 'Expense Detail', and 'Amount'. A right-click context menu is open over the form header, showing options like 'Save', 'Configure', 'Export', 'Create Favorite', 'Copy URL', 'Copy sys_id', and 'Reload form'. The 'Configure' option is selected, and a sub-menu is displayed with options: 'Form Builder', 'Form Design' (highlighted), 'Form Layout', 'Related Lists', 'All', 'Table', 'Security Rules', 'Business Rules', 'Client Scripts', 'UI Policies', 'Data Policies', 'UI Actions', 'Notifications', and 'Dictionary'.

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.

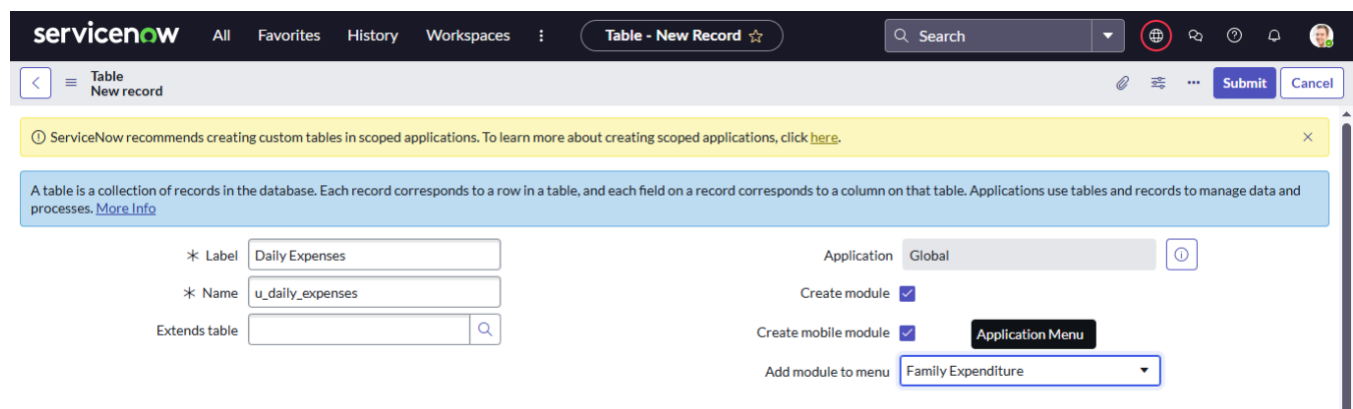


The screenshot shows the 'Form Design' interface in ServiceNow. The top bar includes the 'Family Expenses [u_family]' dropdown, 'Default view' dropdown, and 'Form Design' title. The left sidebar has 'Fields' and 'Field Types' tabs. The 'Fields' section lists various system fields like 'Created', 'Created by', 'Updated', 'Updated by', and 'Updates'. The 'Formatters' section lists 'Activities (filtered)', 'Contextual Search Results', and 'Ratings'. The main area displays the form layout for 'Family Expenses [u_family_expenses]'. It shows a '2 Column' layout with fields 'Number', 'Date', and 'Amount'. Below this is a '1 Column' layout with the 'Expense Details' field. Each field has a gear icon for configuration.

Milestone 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:
 - i. Label : Daily Expenses
 - ii. Name : Auto-Populated
 - iii. Add Module to menu : Family Expenditure
3. Go to the Header and right click there>> click on Save.



Activity 2: Creation of Columns(Fields)

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

Give the details as:

Column label : Expense

Type : Integer

- Double click on insert a new row again

Give the details as:

Column label : Family Member Name

Type : Reference

Max length : 800

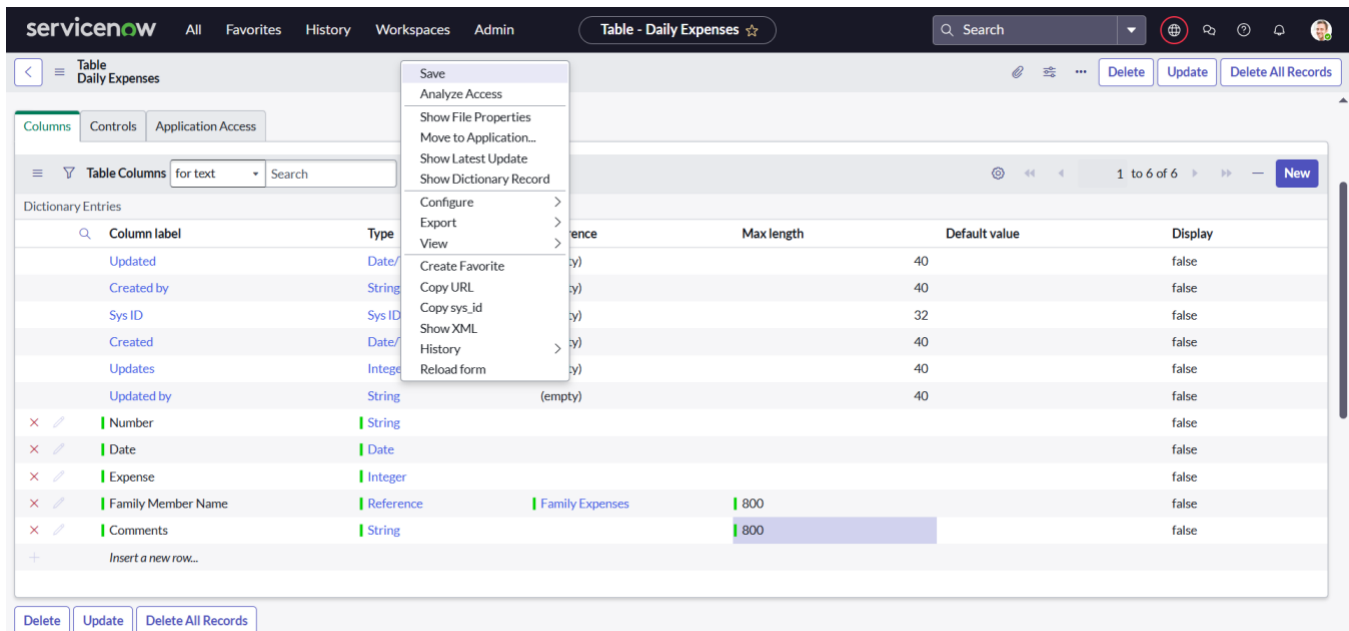
- Double click on insert a new row again

Give the details as:

Column label : Comments

Type : String

Max length : 800



The screenshot shows the ServiceNow configuration page for the 'Table - Daily Expenses'. The 'Columns' tab is selected, displaying a list of columns. A context menu is open over the 'Expense' column, with the 'Save' option highlighted. The table shows columns for 'Expense' (Integer, 800), 'Family Member Name' (Reference, 800), and 'Comments' (String, 800). The 'Family Expenses' column is highlighted in blue.

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

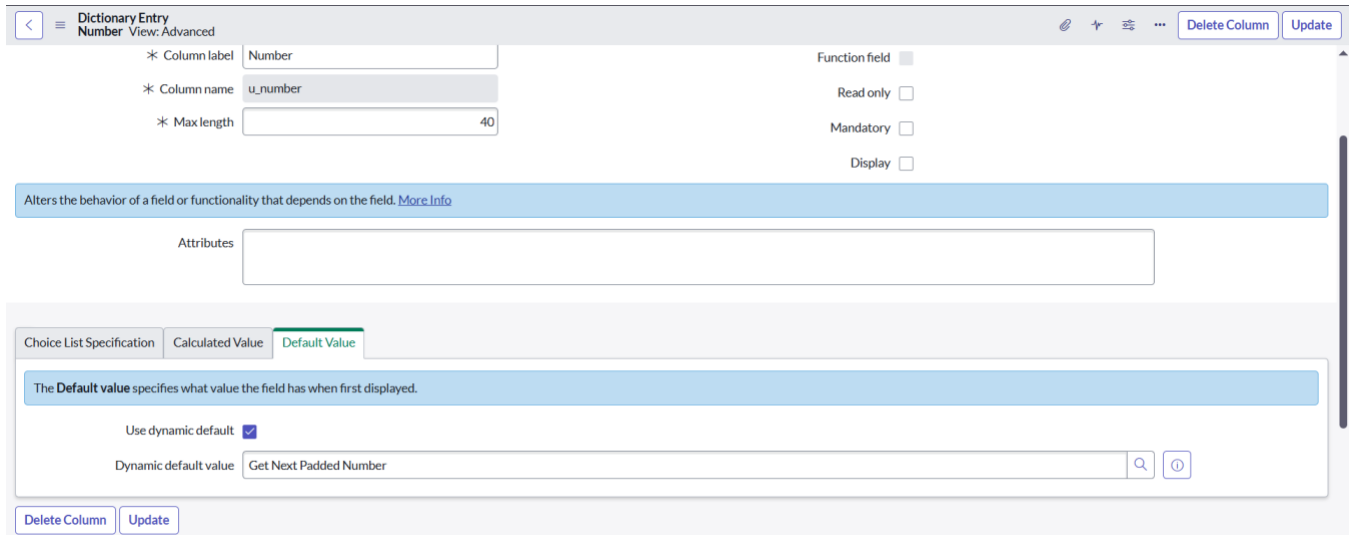
- Go to the Header and right click there>> click on Save.

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

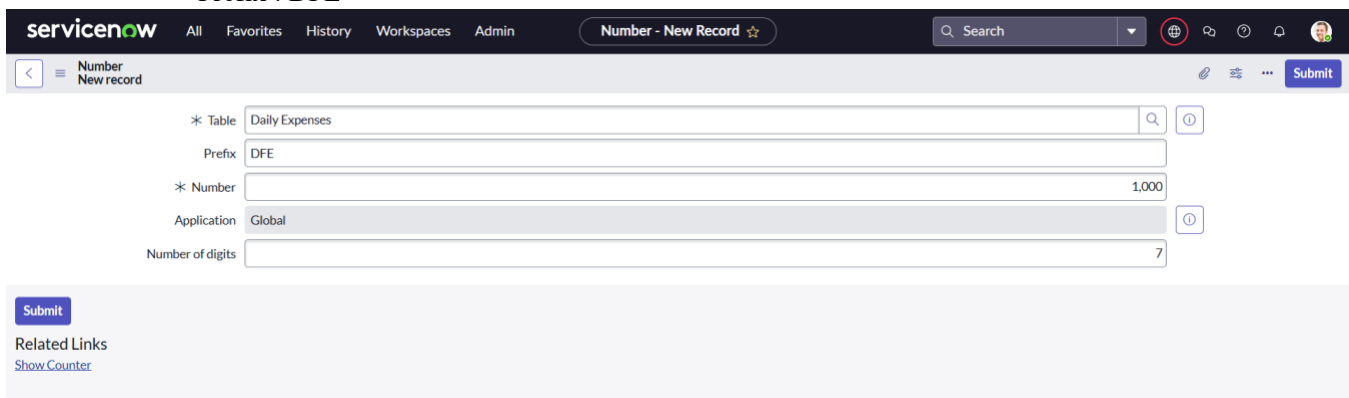
Dynamic default value : Get Next Padded Number



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 : Daily Expenses

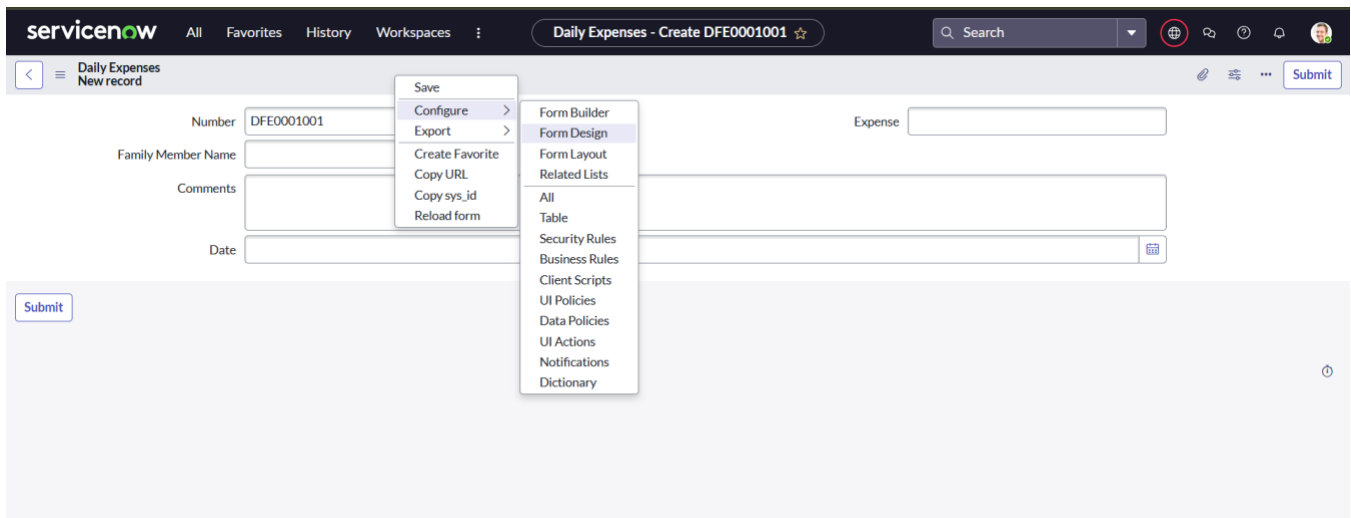
Prefix : DFE



8. Click on Submit

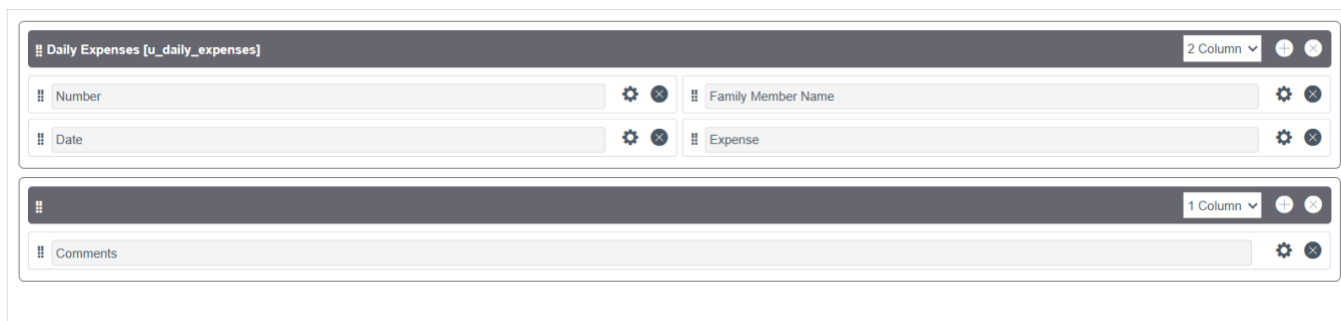
Activity 4 : Configure the Form

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



The screenshot shows the ServiceNow interface for creating a new record in the 'Daily Expenses' table. The form is titled 'Daily Expenses - Create DFE0001001'. The 'Configure' menu is open, and the 'Form Design' option is selected. The form fields include 'Number' (DFE0001001), 'Family Member Name', 'Comments', and 'Date'. The 'Expense' field is also visible. The 'Submit' button is at the bottom left.

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



The screenshot shows the ServiceNow form design canvas. The form is titled 'Daily Expenses [u_daily_expenses]' and is in '2 Column' layout. The fields are arranged in a grid: 'Number' and 'Family Member Name' in the first row, 'Date' and 'Expense' in the second row, and 'Comments' in the third row. Each field has a gear icon for configuration.

7. Click on Save.

Milestone 6: Creation of Relationship

Activity : Creation of Relationship between Family Expenses and Daily Expenses tables.

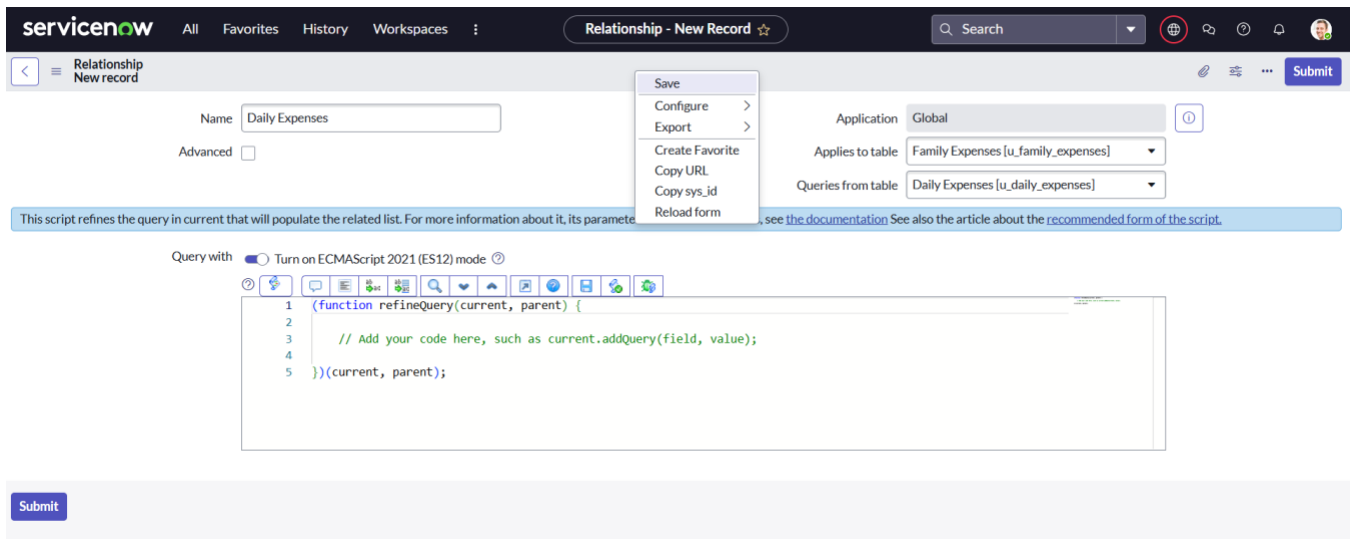
1. Go to All >> In the filter search for Relationships >> Open Relationships
2. Click on New.
3. Enter the details:

Name : Daily Expenses

Applies to table : Select Family Expenses

Daily Expenses : Select Daily Expenses

4. Click Save.



servicenow All Favorites History Workspaces Relationship - New Record

Search

Name: Daily Expenses

Advanced ☐

Application: Global

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, 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);
```

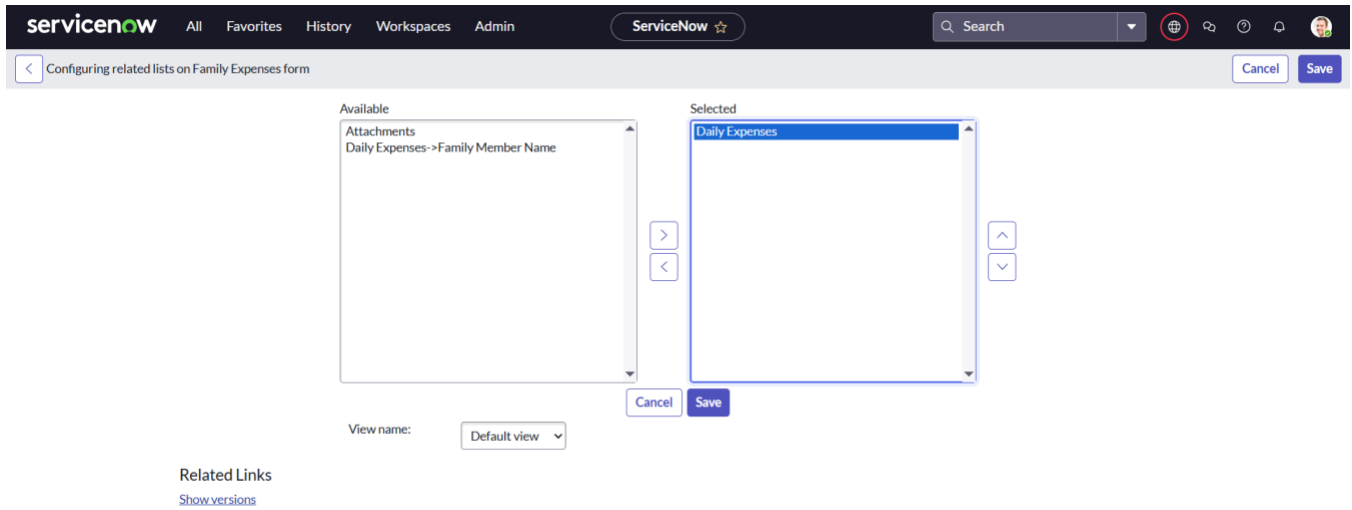
Submit

Milestone 7: Configuring Related List on Family Expenses

Activity : 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 8: Creation of Business Rules

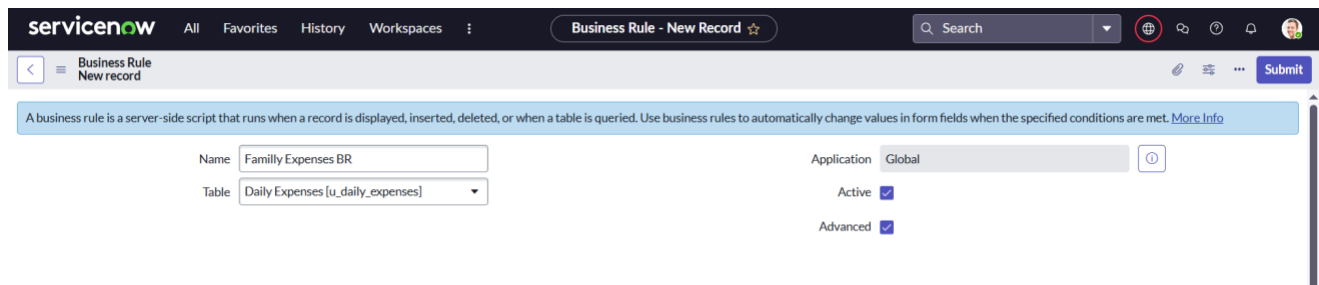
Activity : Creation of Business Rules

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

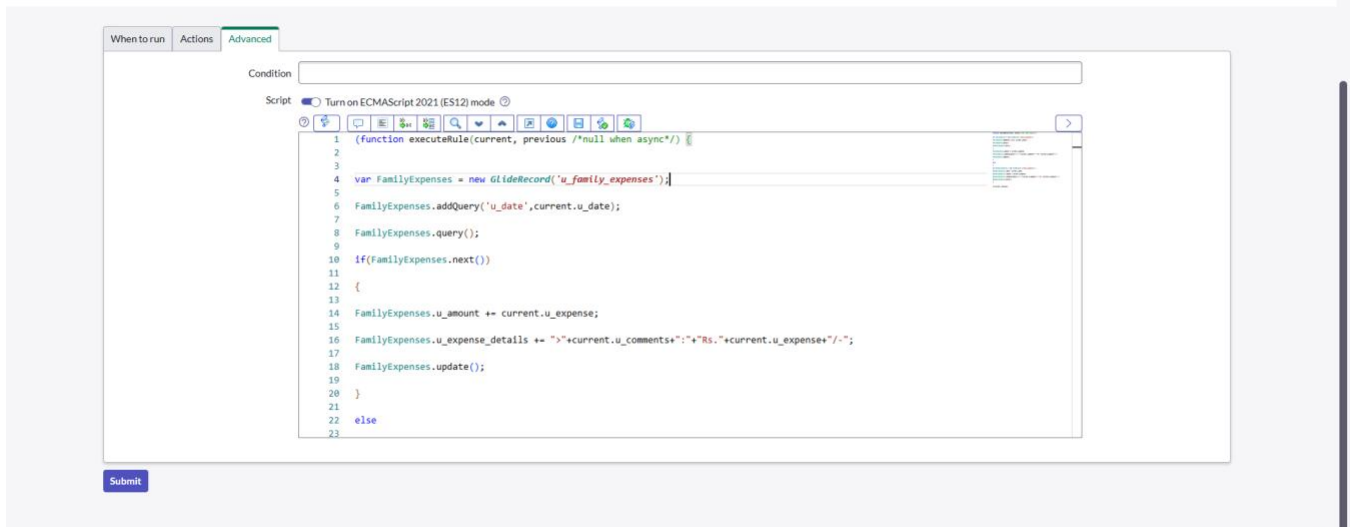
Table : Select Daily Expenses

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  
    {  
        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);
```

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

Milestone 9 : Configure the Relationship

Activity : Configure the Relationship

1. 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.

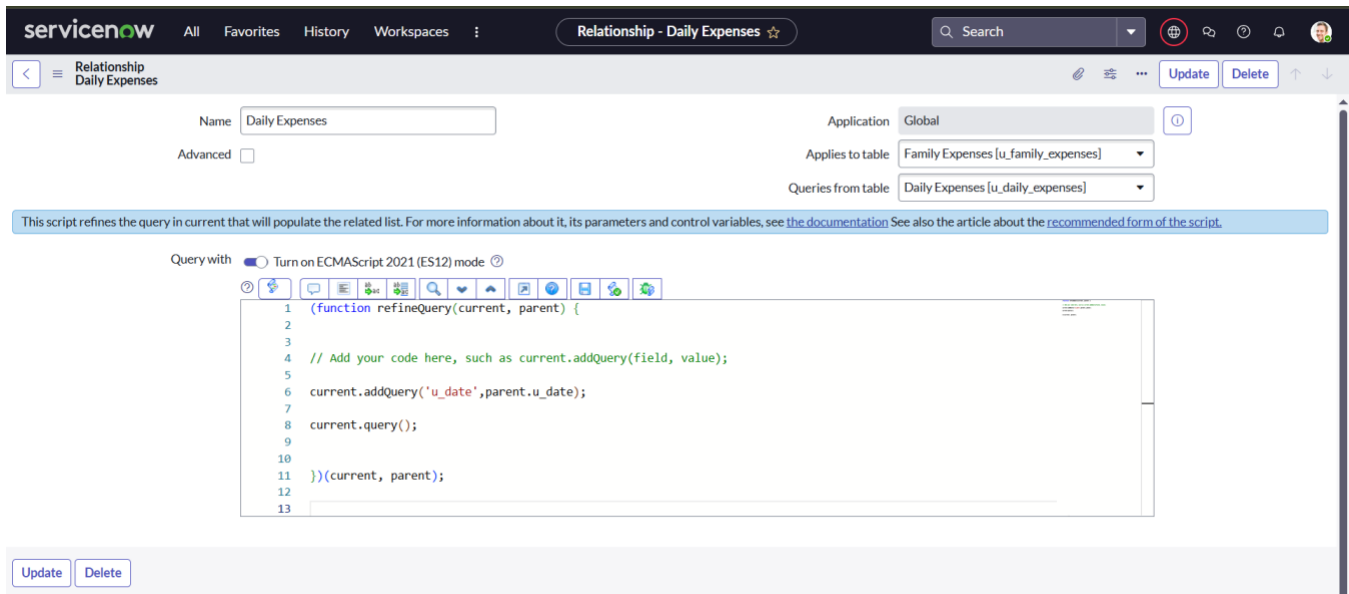
```
(function refineQuery(current, parent) {
```

```
// Add your code here, such as current.addQuery(field, value);
```

```
current.addQuery('u_date',parent.u_date);
```

```
current.query();
```

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



servicenow All Favorites History Workspaces Relationship - Daily Expenses Search

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 ☒ Turn on ECMAScript 2021 (ES12) mode

```
1 (function refineQuery(current, parent) {
2
3
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);
12
13
```

Update Delete

5. Click on Update.

Conclusion :

The project **“Calculating Family Expenses using ServiceNow”** successfully demonstrates how technology can simplify and streamline the process of managing household finances. By leveraging the features of ServiceNow, the system enables effective expense tracking, categorization, and reporting, while maintaining scalability and security. Through the creation of customized tables, forms, relationships, and business rules, the project provides a user-friendly solution that empowers families to monitor their spending patterns and make informed financial decisions. Ultimately, this initiative not only promotes financial discipline but also highlights the potential of ServiceNow as a platform for developing practical, real-world applications beyond its traditional IT service management scope.