

CALCULATING FAMILY EXPENCES USING SERVICE NOW

Team Id: NM2025TMID18161

Team Members:

Team Leader: THARUNKUMAR NV

Team Member : RAMKUMAR.S

Team Member : RANJITH .M

Team Member : SHUBAVARDHANA.M

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.

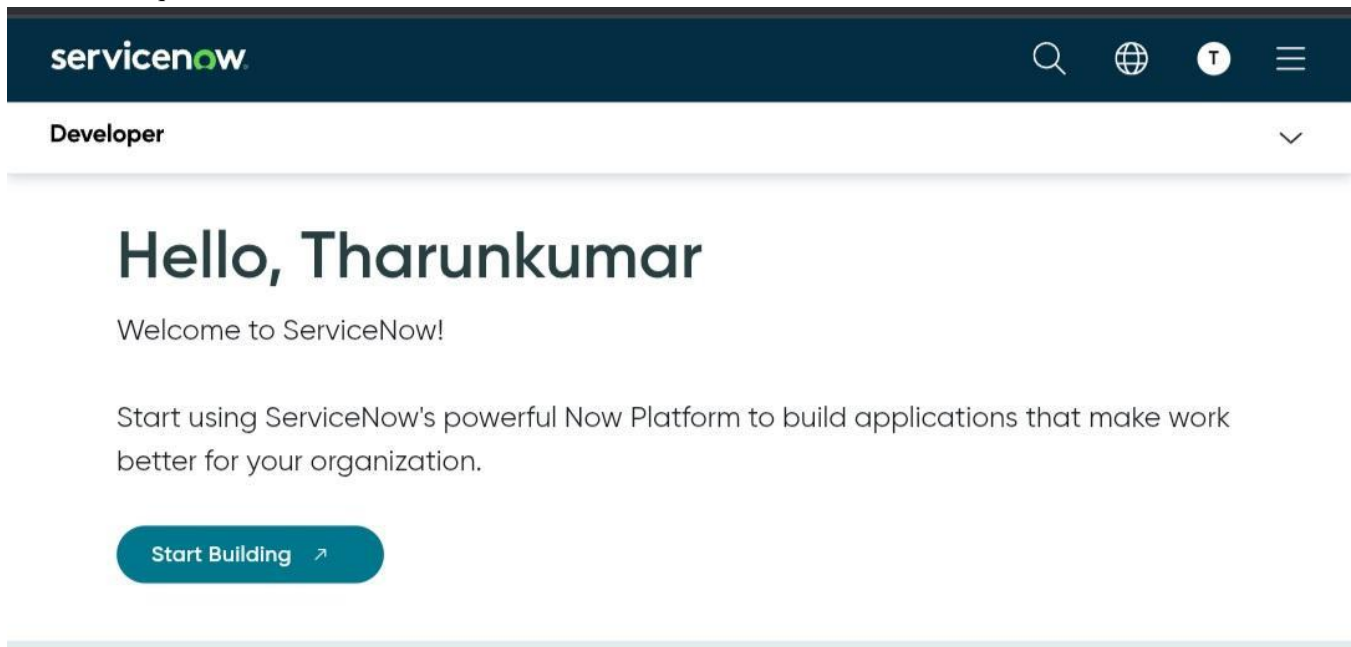
Objective: The objective of this project is to develop a family expense management system on the ServiceNow platform that enables efficient tracking, categorization, and budgeting of expenses. It aims to provide real-time monitoring, user-friendly dashboards, and insightful reports to support informed 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:

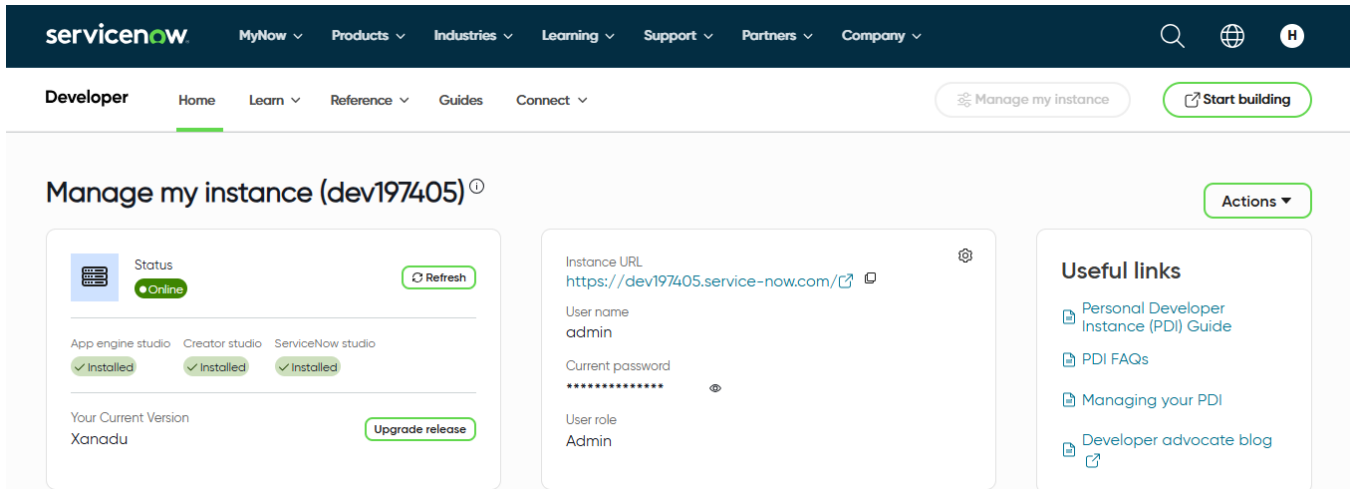
TASK INITIATION

Milestone 1 : Setting up ServiceNow Instance

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.

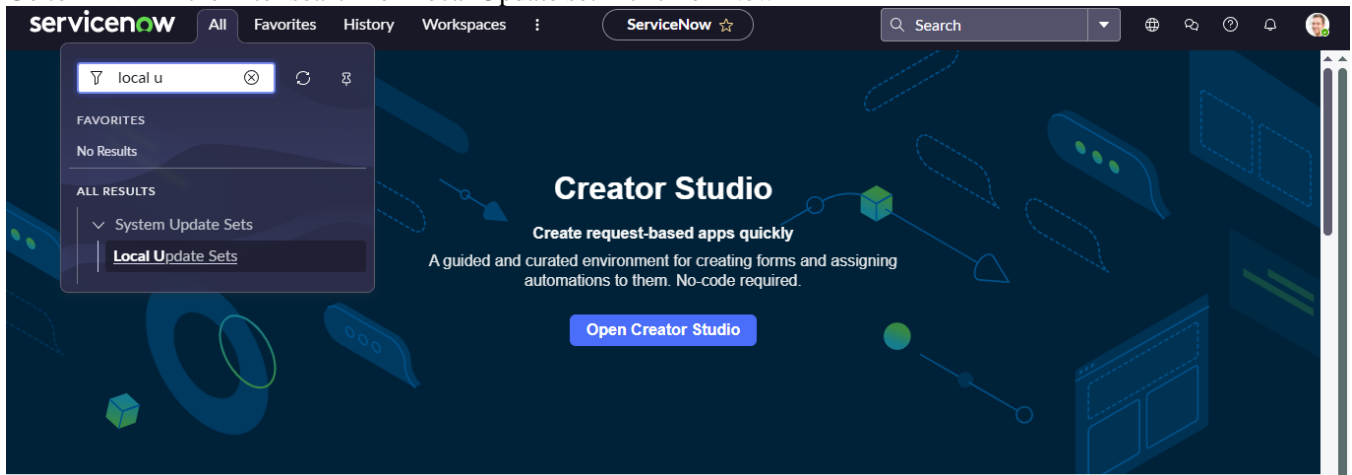


The screenshot shows the ServiceNow Developer portal for instance dev197405. The top navigation bar includes links for MyNow, Products, Industries, Learning, Support, Partners, and Company. Below this, a secondary bar has Developer, Home, Learn, Reference, Guides, and Connect. The main content area is titled 'Manage my instance (dev197405)' and features three panels. The left panel shows the instance status as 'Online' and lists installed components: App engine studio, Creator studio, and ServiceNow studio, all marked as 'Installed'. It also shows the current version as 'Xanadu' with an 'Upgrade release' button. The middle panel displays instance details: Instance URL (https://dev197405.service-now.com/), User name (admin), Current password (masked), and User role (Admin). The right panel, titled 'Useful links', provides links to the Personal Developer Instance (PDI) Guide, PDI FAQs, Managing your PDI, and the Developer advocate blog.

- Now you will navigate to the ServiceNow.

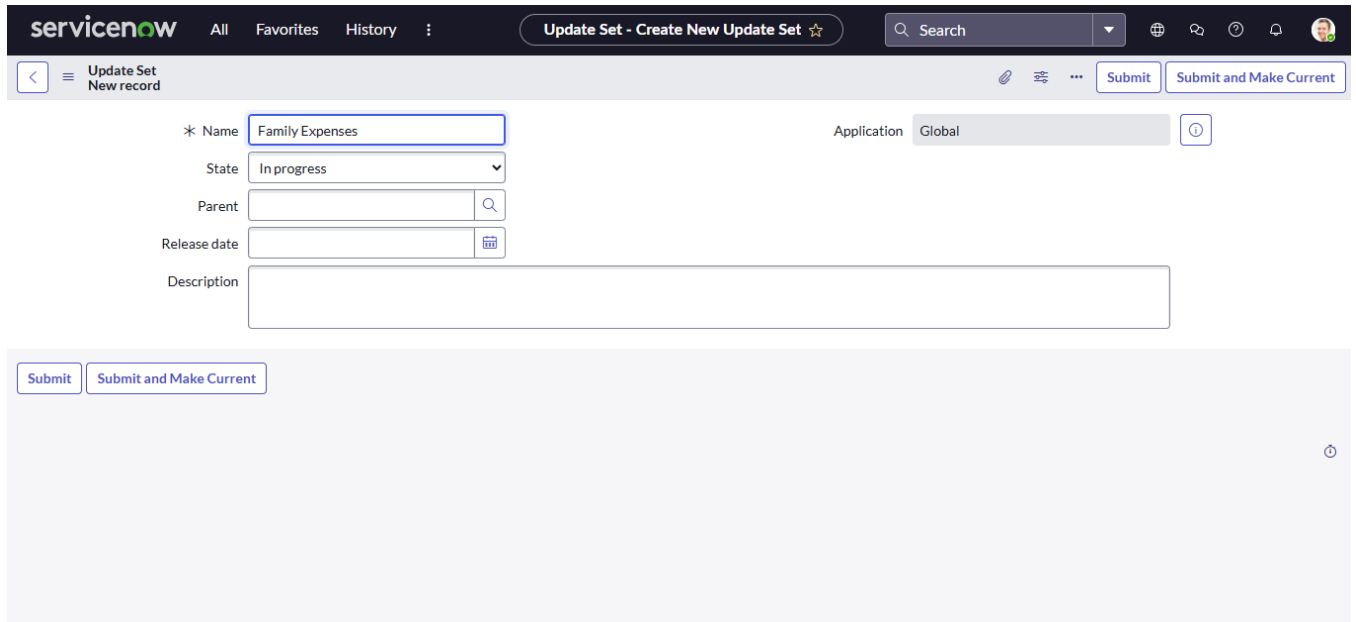
Milestone 2 : Creation of New Update Set

- 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 links for All, Favorites, History, Workspaces, and ServiceNow. A search bar is visible. A search filter is applied, showing 'local u' in the search box. The search results are displayed in a list under the heading 'ALL RESULTS'. The results show 'System Update Sets' with a sub-item 'Local Update Sets' highlighted. The main content area features a 'Creator Studio' banner with the text 'Create request-based apps quickly' and 'A guided and curated environment for creating forms and assigning automations to them. No-code required.' Below the banner is a button labeled 'Open Creator Studio'.

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



The screenshot shows the ServiceNow 'Update Set - Create New Update Set' form. The top navigation bar includes 'All', 'Favorites', 'History', and a search bar. The main form area contains the following fields:

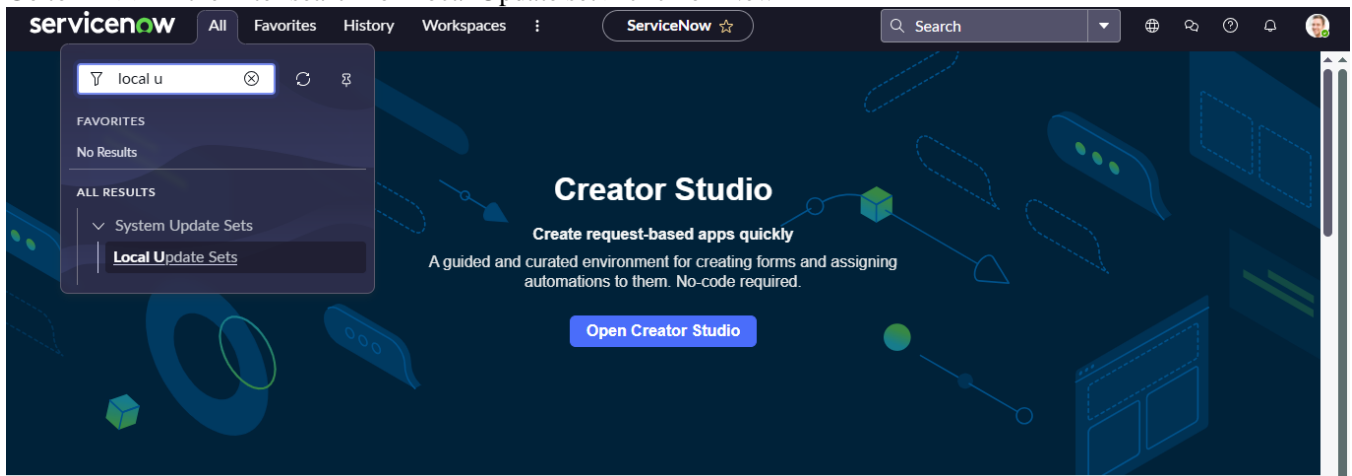
- Name:** A text input field containing 'Family Expenses'.
- State:** A dropdown menu currently set to 'In progress'.
- Parent:** A text input field with a search icon.
- Release date:** A date picker icon.
- Description:** A large text area for additional details.
- Application:** A dropdown menu set to 'Global'.

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

Milestone 3: Creation of Table

Activity: Creation of New Update Set

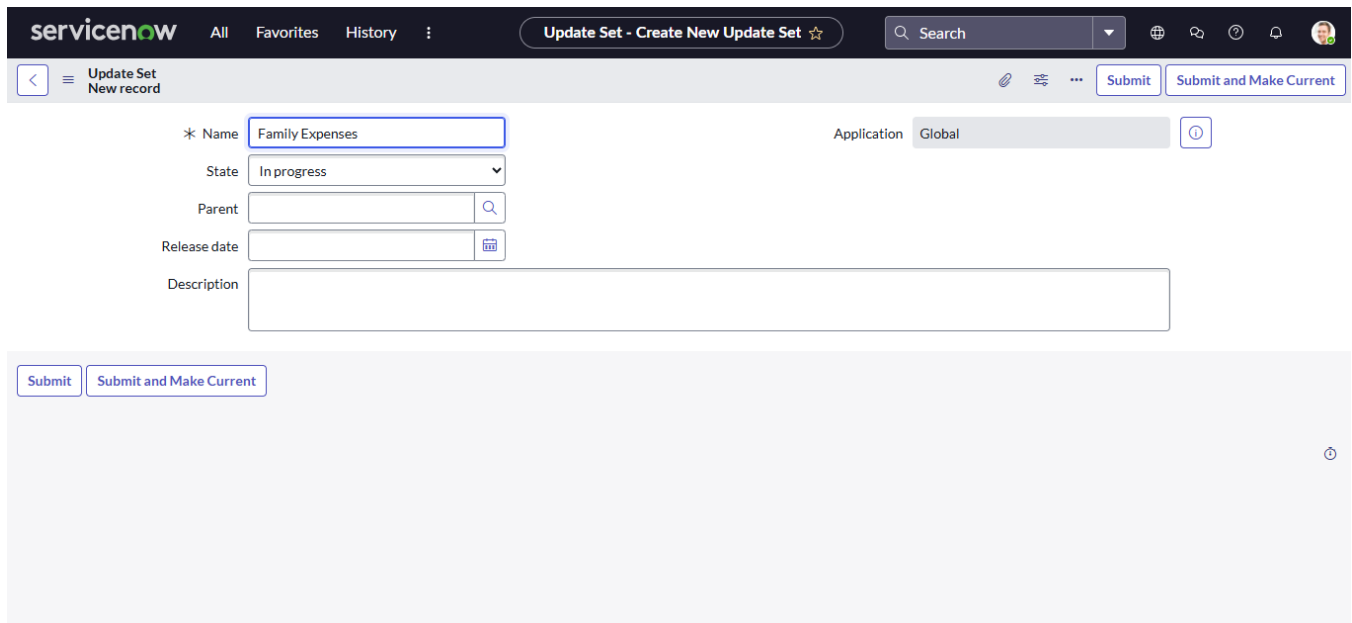
1. Go to All >> In the filter search for Local Update set > click on New



2. Enter the Details as:

Name : Family Expenses

3. Then click on Submit and Make current.



The screenshot shows the ServiceNow 'Update Set' form for 'Family Expenses'. The form is titled 'Update Set - Create New Update Set' and includes a search bar and navigation links. The form fields are as follows:

- Name:** Family Expenses
- State:** In progress
- Parent:** (empty field with a search icon)
- Release date:** (empty field with a calendar icon)
- Description:** (empty text area)
- Application:** Global

At the bottom of the form, 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.

Dictionary Entry Number View: Advanced

Display ☐

Alters the behavior of a field or functionality that depends on the field. [More Info](#)

Attributes

Choice List Specification Calculated Value **Default Value**

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

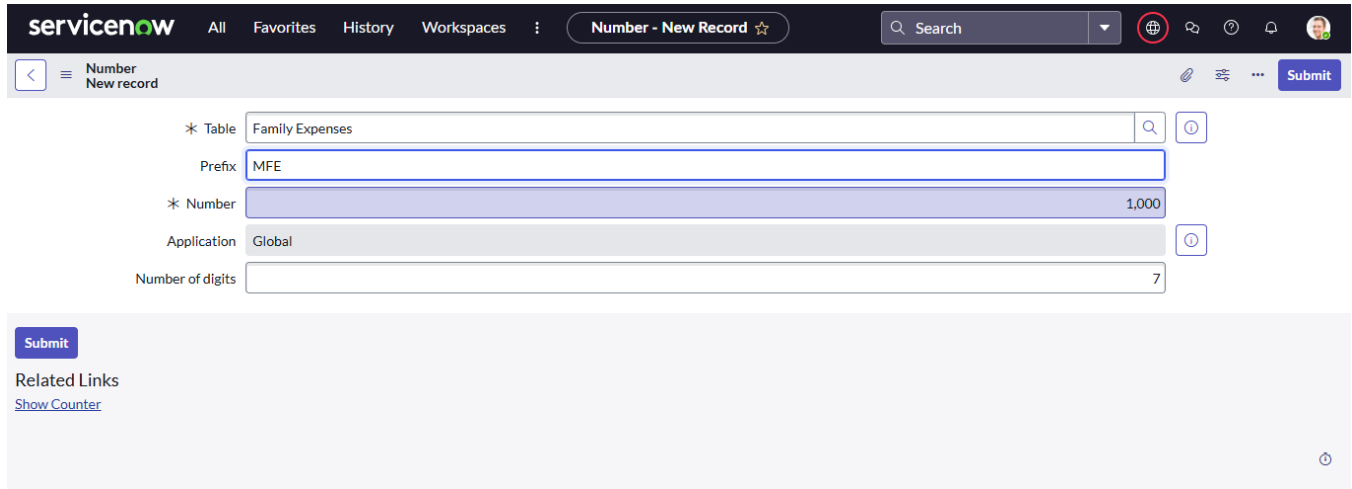
Use dynamic default ☒

Dynamic default value Get Next Padded Number

Delete Column Update

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

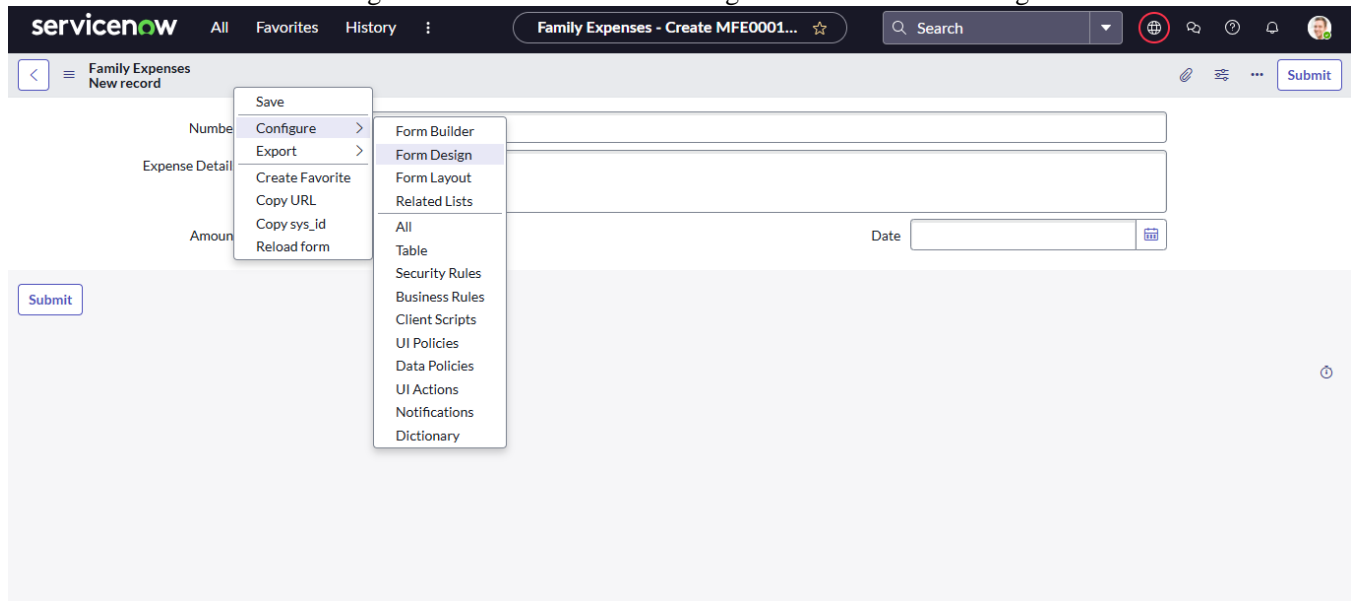
7. Enter the below Details:
 - i. Table : Family Expenses
 - ii. Prefix : MFE



8. Click on Submit.

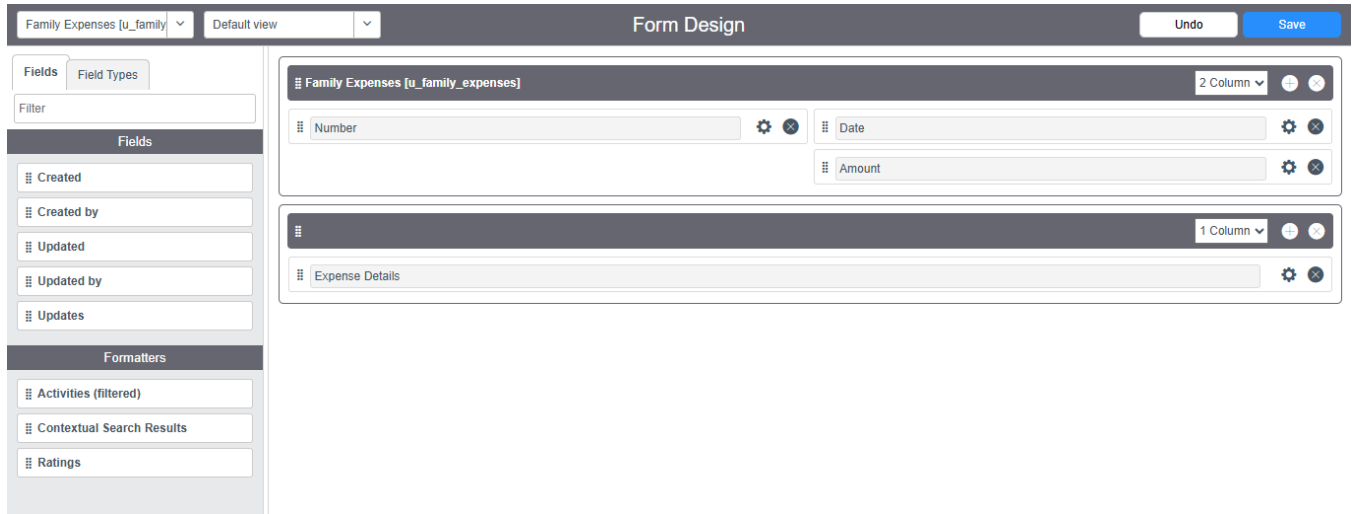
Activity 4: Configure the Form

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



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.

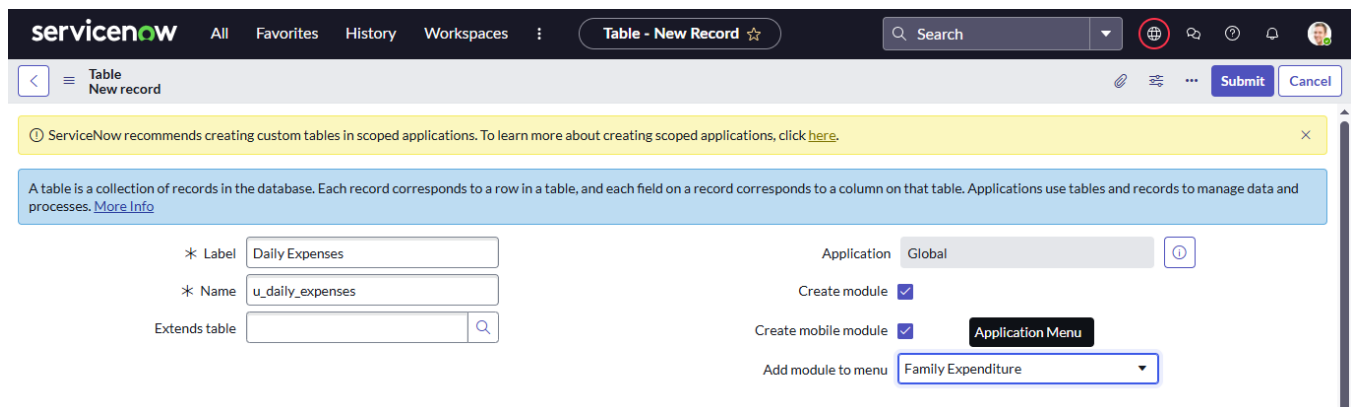


The screenshot shows the 'Form Design' interface for a table named 'Family Expenses [u_family_expenses]'. The interface has a top bar with 'Family Expenses [u_family]' and 'Default view' dropdowns, and 'Undo' and 'Save' buttons. On the left, there are tabs for 'Fields' and 'Field Types', a 'Filter' input, and sections for 'Fields' (listing Created, Created by, Updated, Updated by, Updates) and 'Formatters' (listing Activities (filtered), Contextual Search Results, Ratings). The main area shows the table structure with two sections: the first has 2 columns (Number, Date, Amount) and the second has 1 column (Expense Details). 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.



The screenshot shows the 'Table - New Record' form in ServiceNow. The top bar includes the ServiceNow logo, navigation tabs (All, Favorites, History, Workspaces), and a 'Table - New Record' button. Below the top bar, there's a search bar and a 'Submit' button. A yellow banner message states: 'ServiceNow recommends creating custom tables in scoped applications. To learn more about creating scoped applications, click [here](#).' Below this, a blue box explains: '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](#)'.

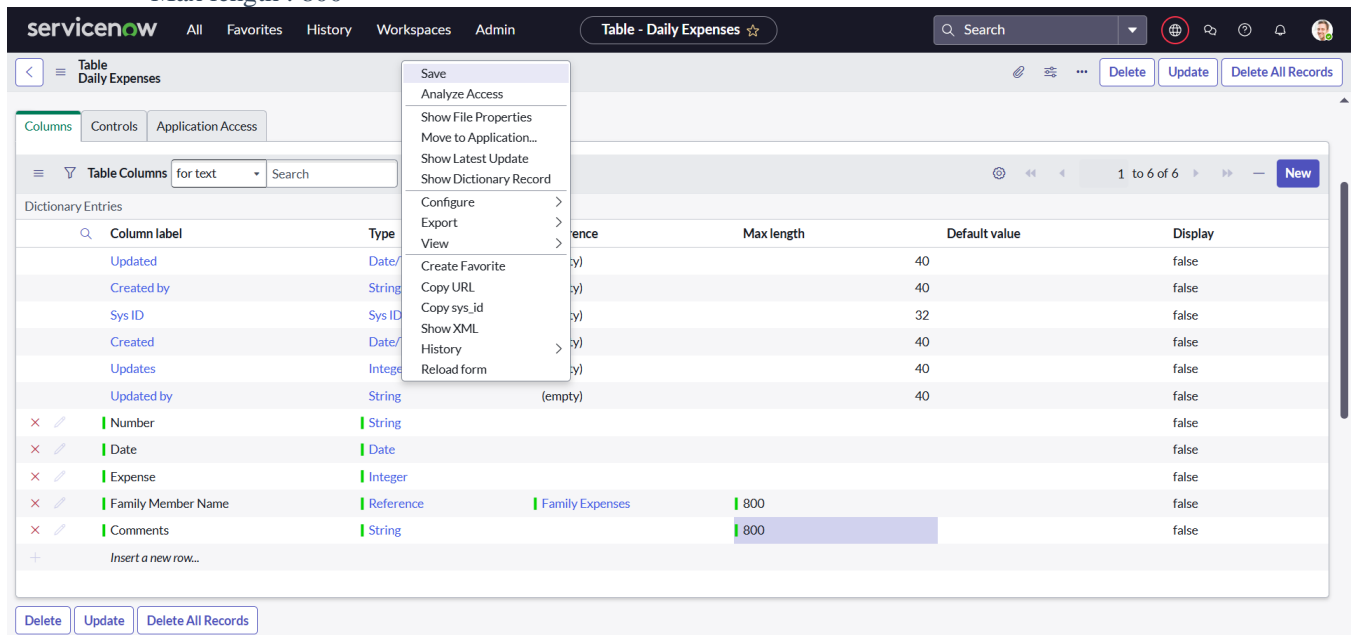
The form fields are as follows:

- * Label:
- * Name:
- Extends table:
- Application:
- Create module: ☒
- Create mobile module: ☒
- Add module to menu:

There is also an 'Application Menu' button and a 'Submit' button.

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
4. Double click on insert a new row again
Give the details as:
Column label : Family Member Name
Type : Reference
Max length : 800
5. Double click on insert a new row again
Give the details as:
Column label : Comments
Type : String
Max length : 800

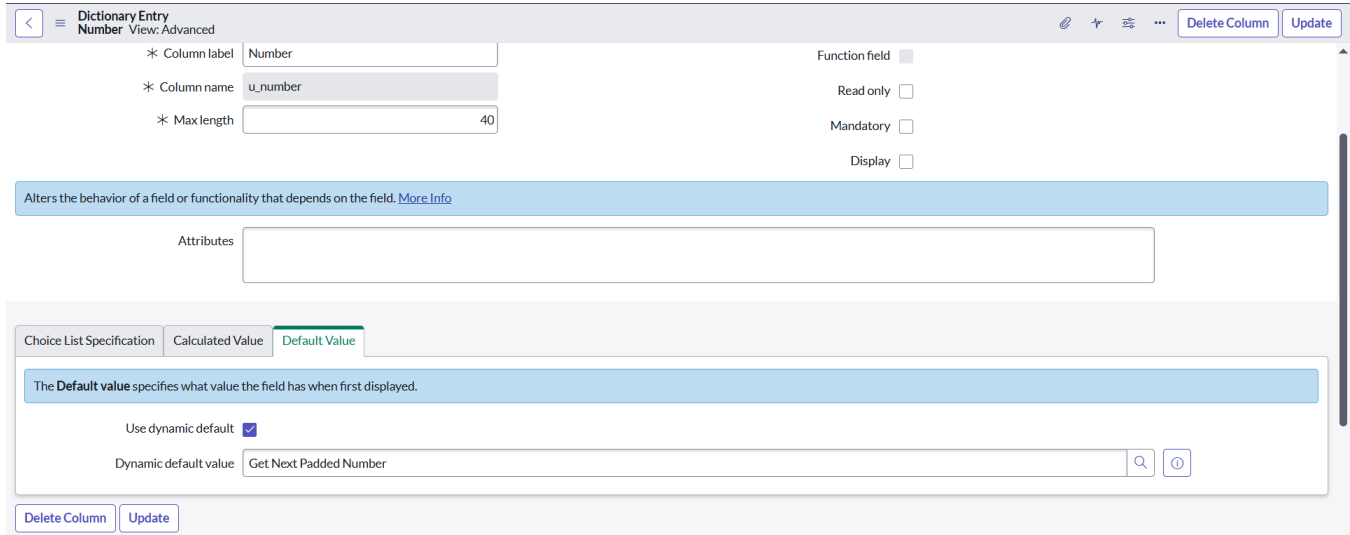


Column label	Type	Max length	Default value	Display
Updated	Date/Time			false
Created by	String			false
Sys ID	Sys ID			false
Created	Date/Time			false
Updates	Integer			false
Updated by	String			false
Number	String			false
Date	Date			false
Expense	Integer			false
Family Member Name	Reference	800	Family Expenses	false
Comments	String	800		false
Insert a new row...				

6. 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



Dictionary Entry
Number View: Advanced

* Column label: Number

* Column name: u_number

* Max length: 40

Function field: ☐

Read only: ☐

Mandatory: ☐

Display: ☐

Attributes:

Choice List Specification | Calculated Value | **Default Value**

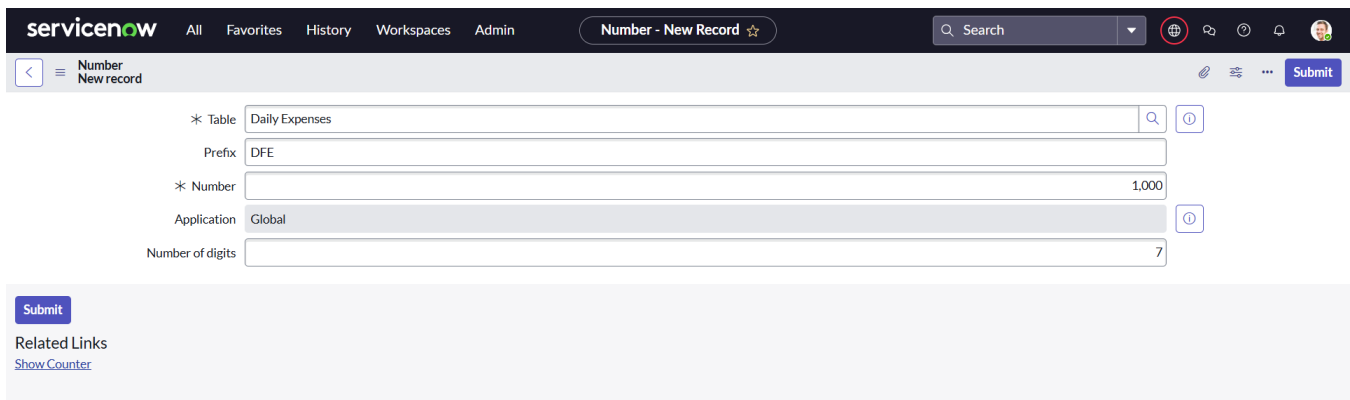
The **Default value** specifies what value the field has when first displayed.

Use dynamic default: ☒

Dynamic default value: Get Next Padded Number

Delete Column | Update

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



servicenow All Favorites History Workspaces Admin Number - New Record

Search

Number - New record

* Table: Daily Expenses

Prefix: DFE

* Number: 1,000

Application: Global

Number of digits: 7

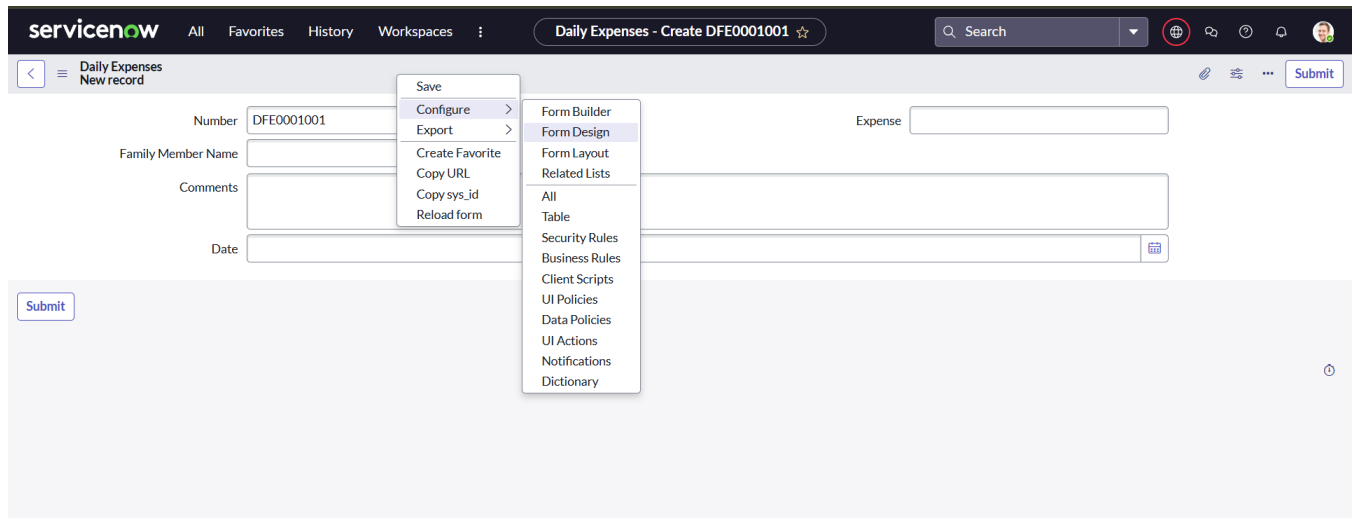
Submit

Related Links
[Show Counter](#)

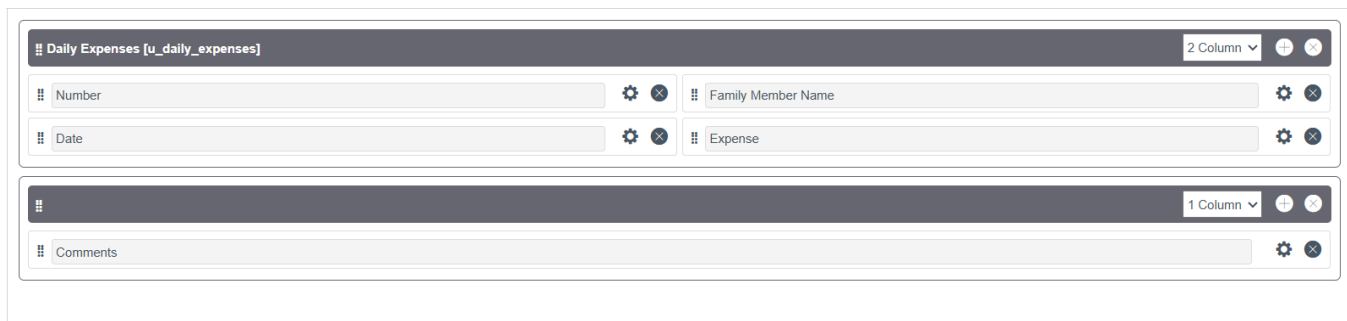
8. Click on Submit

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



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

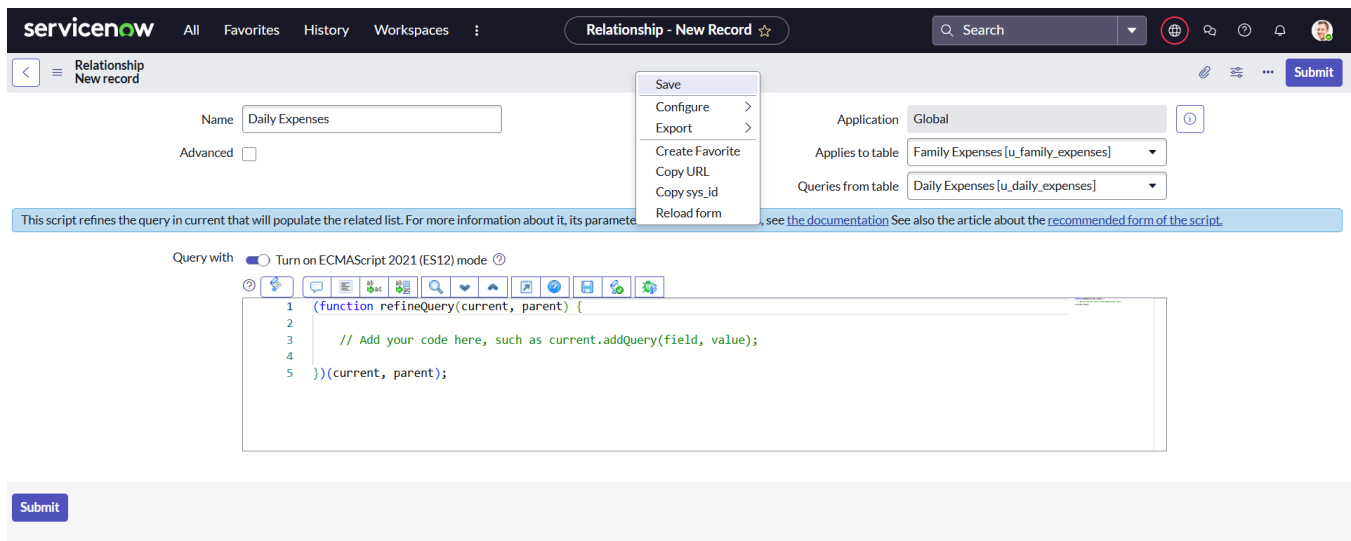


7. Click on Save.

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

Relationship New record

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);
```

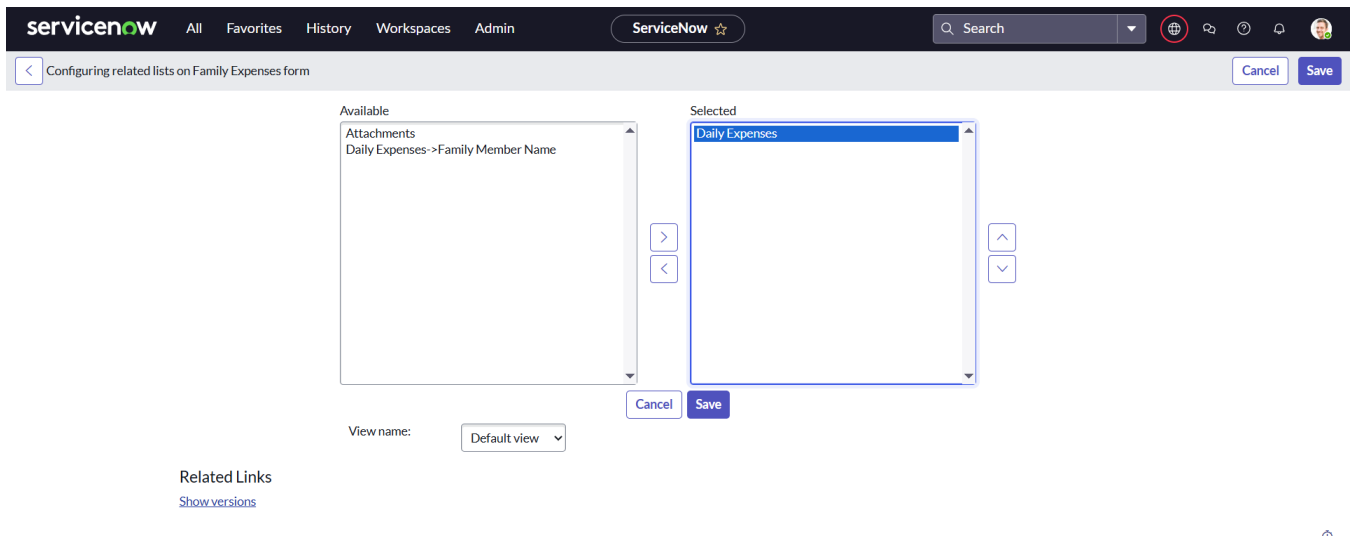
Submit

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.



servicenow All Favorites History Workspaces Admin ServiceNow Search

< Configuring related lists on Family Expenses form Cancel Save

Available
Attachments
Daily Expenses -> Family Member Name

Selected
Daily Expenses

View name: Default view

Related Links
[Show versions](#)

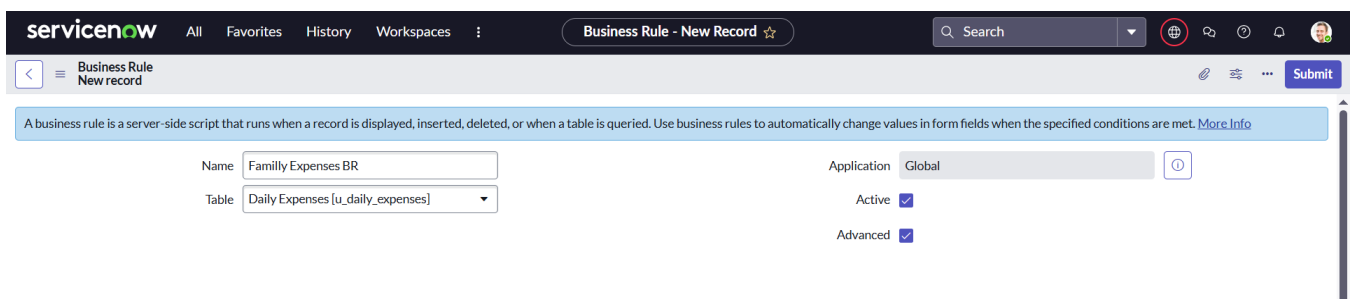
5. Click on Save

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



servicenow All Favorites History Workspaces Business Rule - New Record Search

< Business Rule New record Submit

A 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. [More Info](#)

Name: Family Expenses BR Application: Global

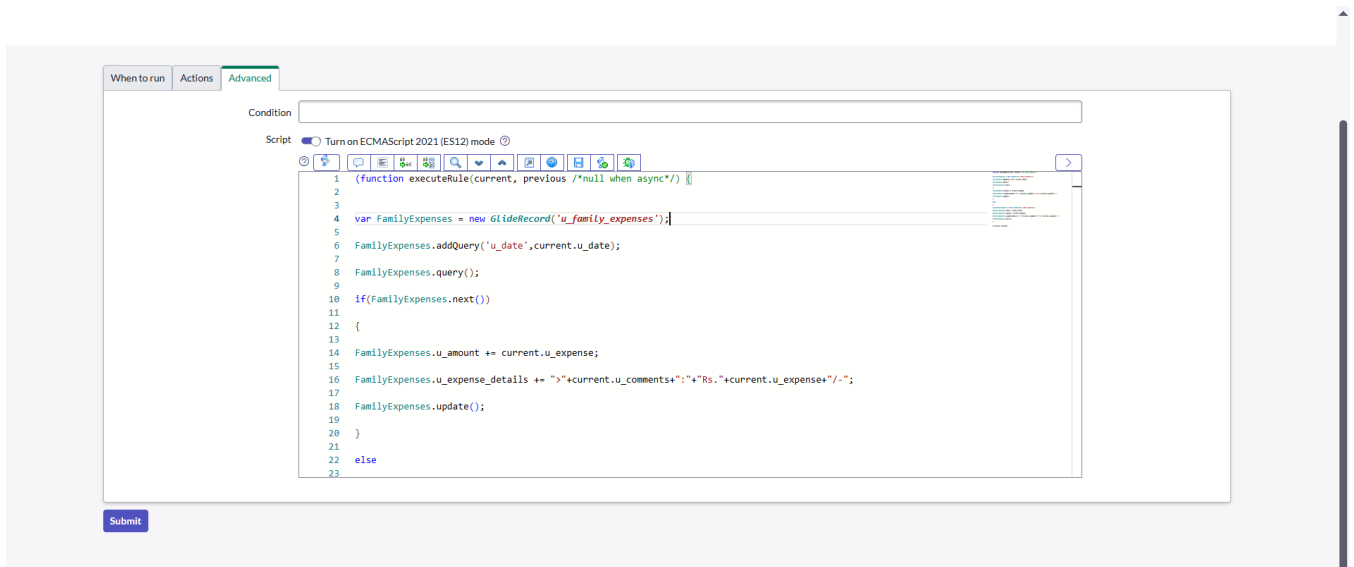
Table: Daily Expenses [u_daily_expenses] Active: ☒ Advanced: ☒

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);
```



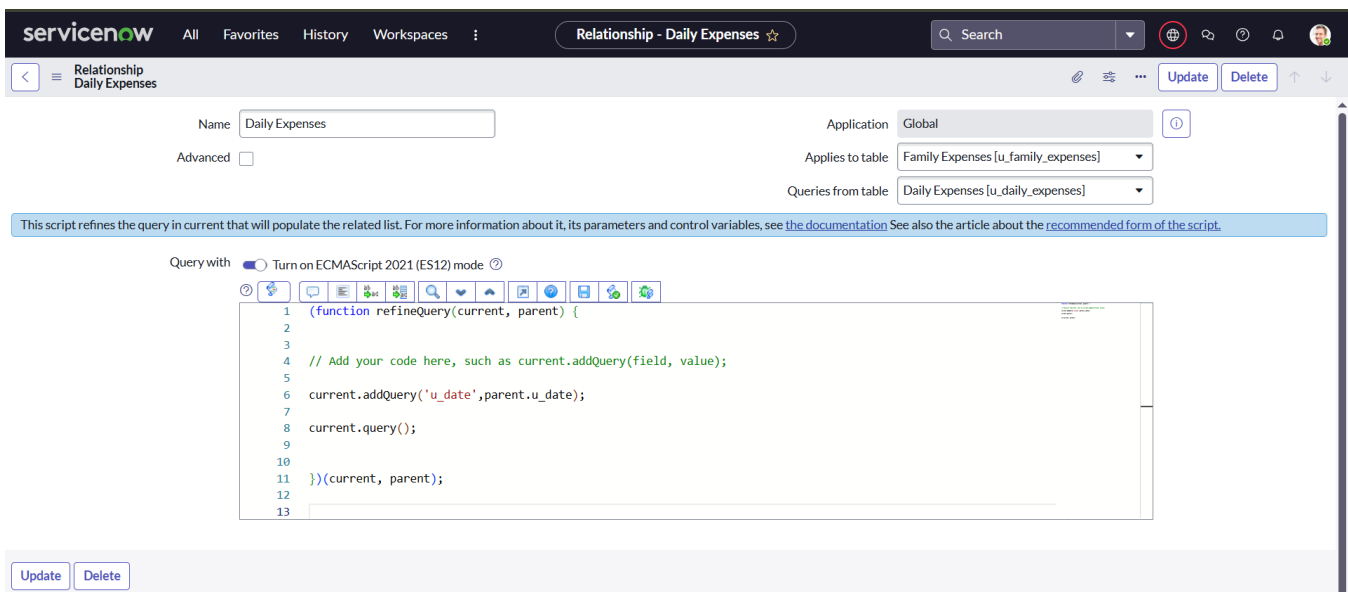
6. Go to the Header and right click there>> click on Save.

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 effectively demonstrates how ServiceNow can be used to manage and track family expenses with real-time monitoring, automated calculations, and user-friendly dashboards. It simplifies expense tracking, promotes financial discipline, and highlights the flexibility of ServiceNow for solving practical, real-world problems.

