Calculating Family Expenses using ServiceNow

Team ID: NM2025TMID06530
Team Size: 5
Team Leader: SUDHARSAN R
Team Member: SRI RENGANATHAN T
Team Member: SUDHAN K
Team Member: SUNDAR S
Team Member: SIVABALAN N

1. Ideation Phase

In the ideation phase, the team identified a common household problem — the lack of an efficient, automated way to manage and analyze family expenses. The idea was to create a centralized platform on **ServiceNow** to help users record, categorize, and analyze their spending. Brainstorming focused on improving financial visibility, preventing overspending, and providing real-time reports. The concept evolved into building a **"Family Expense Management System"**, enabling automation for recurring expenses and personalized dashboards for better decision-making.

2. Project Planning Phase

The planning phase involved defining the **project objectives, milestones, and roles**.

- **Tools used:** ServiceNow App Engine Studio, Flow Designer, and Reporting.
- **Milestones:**
    1. Setting up the ServiceNow instance.
    2. Creating update sets for version tracking.
    3. Designing the Family Expenses and Daily Expenses tables.
    4. Establishing relationships between tables.
    5. Implementing business rules for automation.
    6. Configuring related lists and reports.
- **Responsibilities:** Each team member was assigned tasks like table creation, form design, script development, and testing.
  This phase ensured a clear roadmap and timeline for smooth implementation.

3. Project Design Phase

In this phase, the **architecture and interface design** were finalized.

- The team created **two core tables**:
  - *Family Expenses* (for monthly or weekly totals)
  - *Daily Expenses* (for individual expense entries).
- **Relationships** were defined to link daily entries with family totals.
- The **form design** was customized for user-friendly data entry, with fields like *Date*, *Amount*, *Expense Details*, and *Family Member Name*.
- Business rules were scripted to **automatically update totals** in the Family Expenses table whenever new daily expenses were added.
  This phase focused on creating an intuitive user experience and logical database flow.

4. Requirement Analysis

The requirement analysis identified both **functional** and **technical** needs:

Functional Requirements:

- Users should be able to add and view daily and total expenses.
- The system must auto-update totals and expense details.
- The app should support multi-user roles for family members.
- Real-time reports and dashboards should be generated.

Technical Requirements:

- ServiceNow Developer Instance.
- App Engine Studio for table and form creation.
- Flow Designer for automation.
- Business Rules for backend scripting.
- Reporting & Dashboard module for visualization.

5. Performance Testing

After development, the project underwent **performance and functionality testing** to ensure reliability and efficiency.

- **Testing parameters included:**
  - Speed of data entry and update.
  - Accuracy of total calculations.
  - System responsiveness during concurrent user operations.
  - Validation of business rules and relationships.
  - Dashboard data refresh and report generation speed.

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:

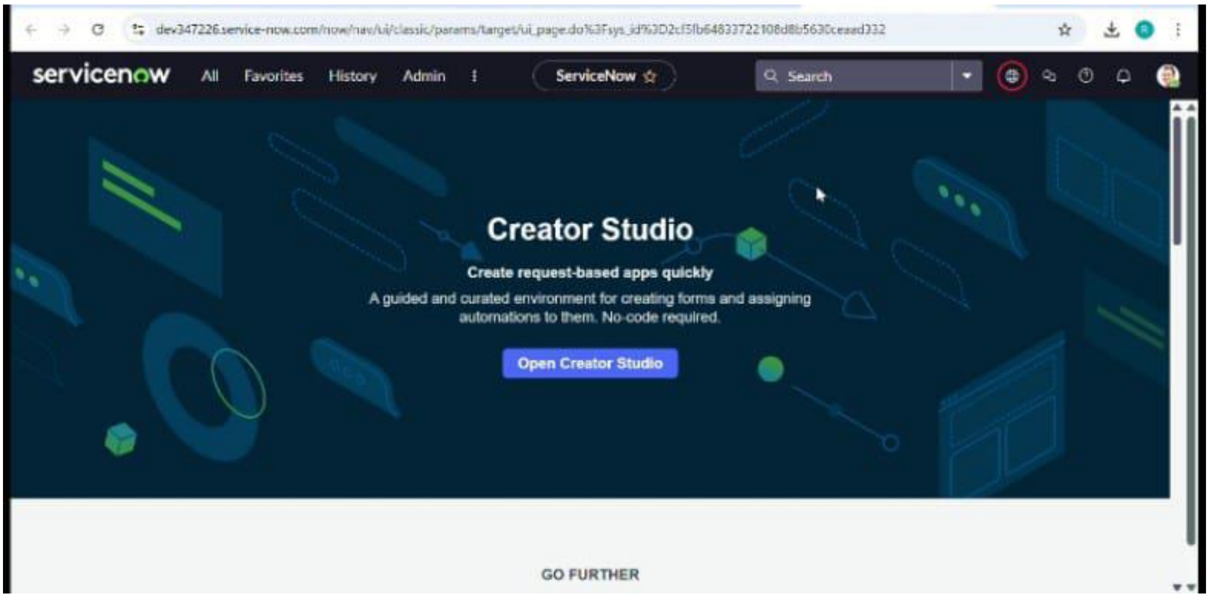
- **ServiceNow App Development** (App Engine Studio)
- **Custom Tables & Data Modeling**
- **Form Design & UI Customization**
- **Flow Designer** (for automation)
- **Business Rules & Scripting**
- **Reporting & Dashboards**
- **User Roles & Access Control (ACLs)**

TASK INITIATION

Milestone 1 : Instance

Activity 1: Setting up ServiceNow instance

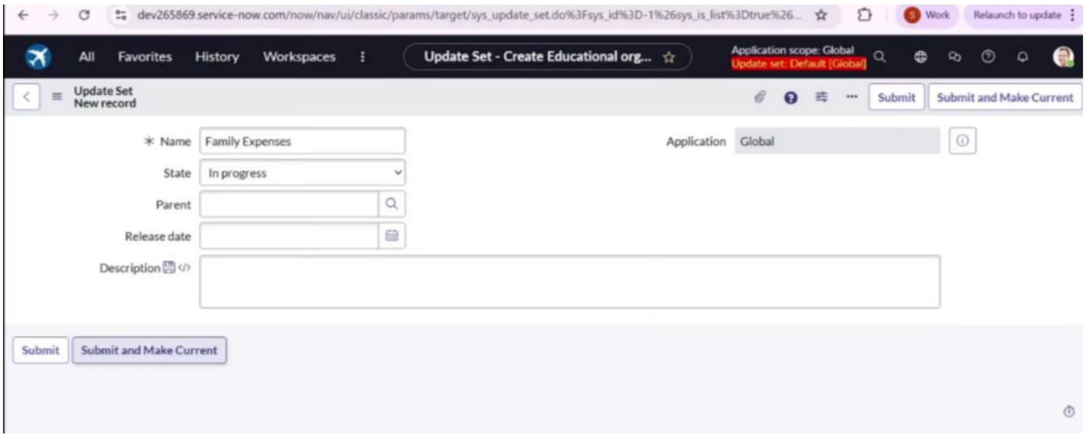
1. Sign up for a developer account on the ServiceNow Developer site "https://developer.servicenow.com".
2. Once logged in, navigate to the "Personal Developer Instance" section.
3. Click on "Request Instance" to create a new ServiceNow instance.
4. Fill out the required information and submit the request.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

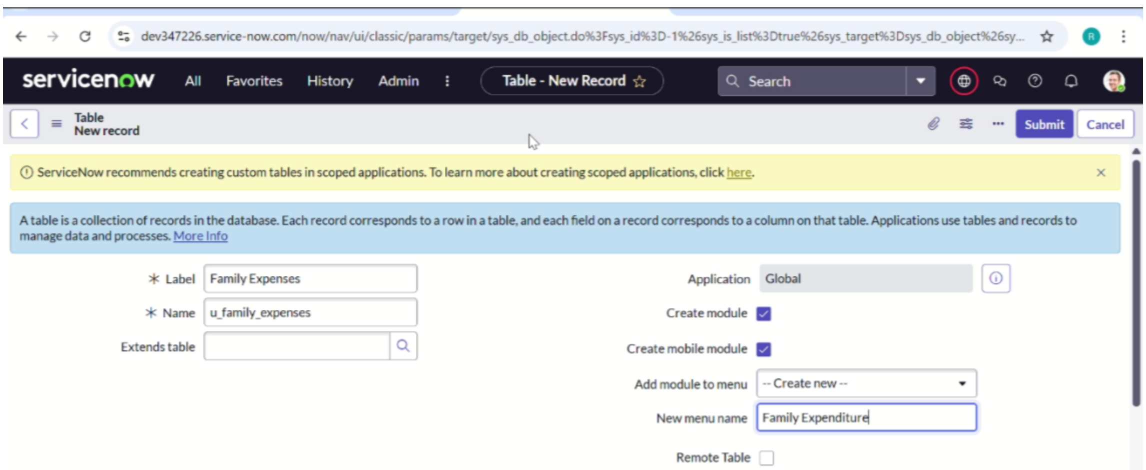
1. Go to All >>In the filter search for Local Update set > click on New.
2. Enter the Details as:
   Name : Family Expenses
3. Then click on Submit and Make current.



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.



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

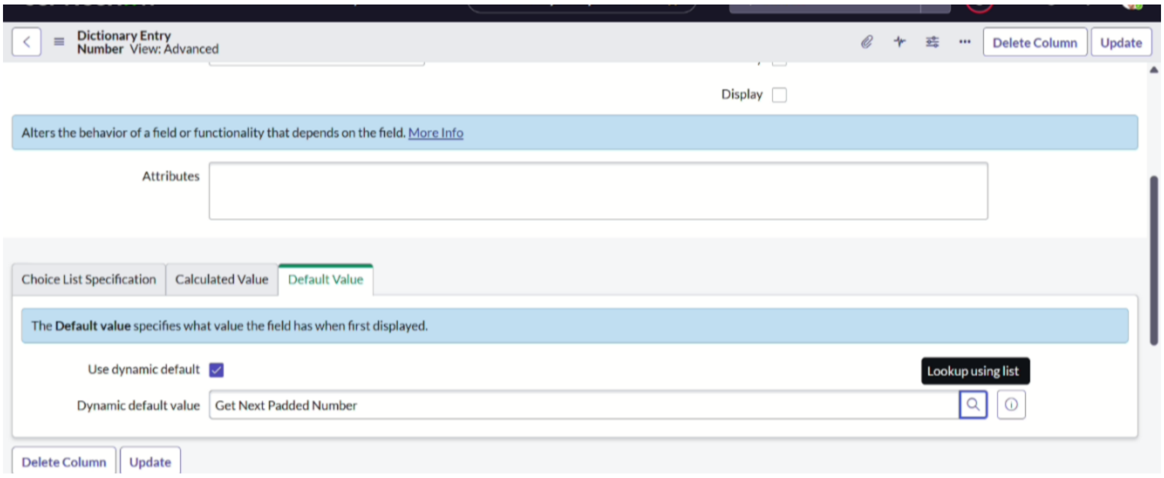
3. Double click on insert a new row again.
4. Give the details as:

   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

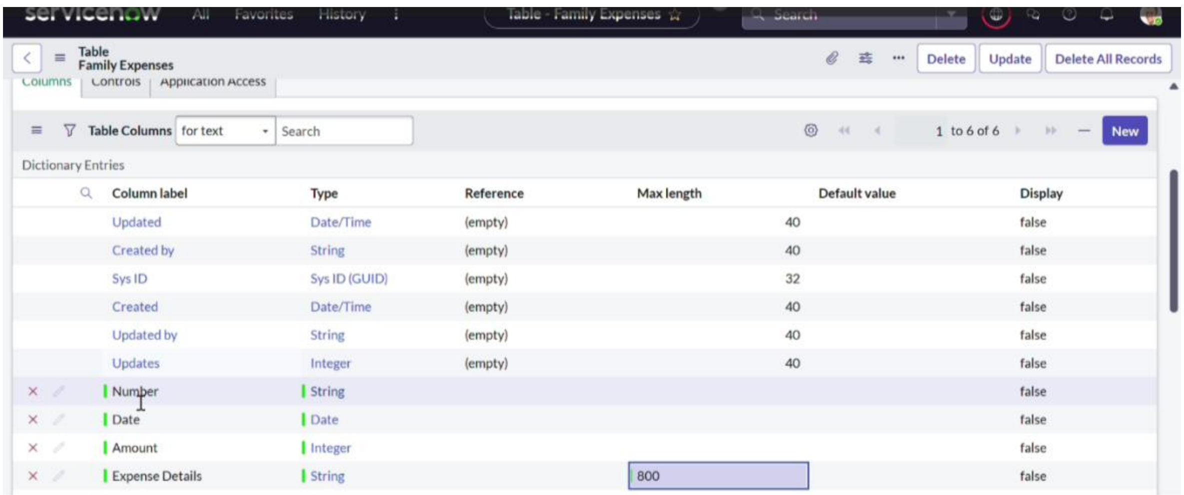
8. Give the details as:

   Column label : Expense Details

   Type : String

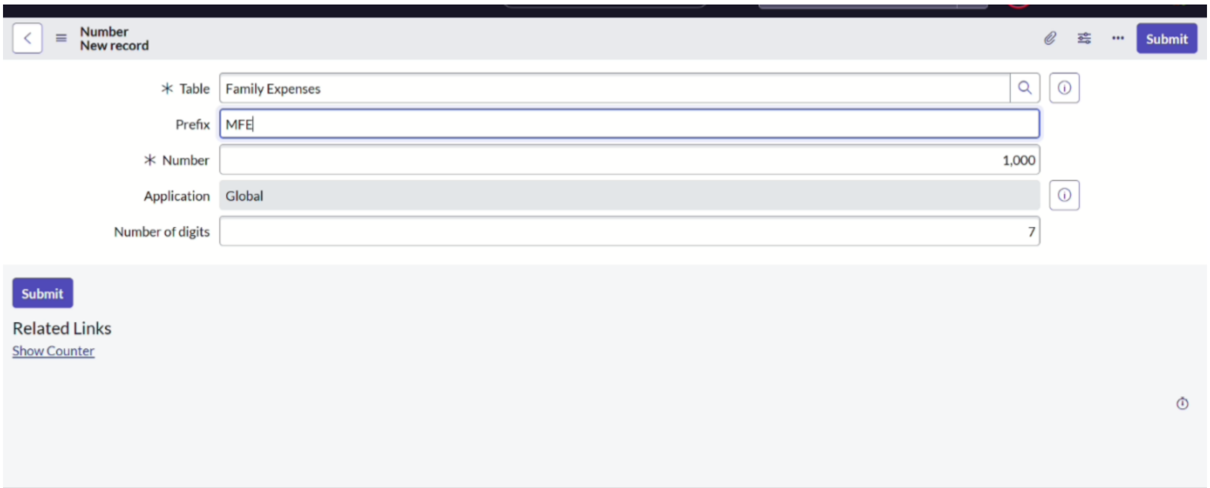
   Max length : 800

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



Activity 3: Making Number Field an Auto-Number

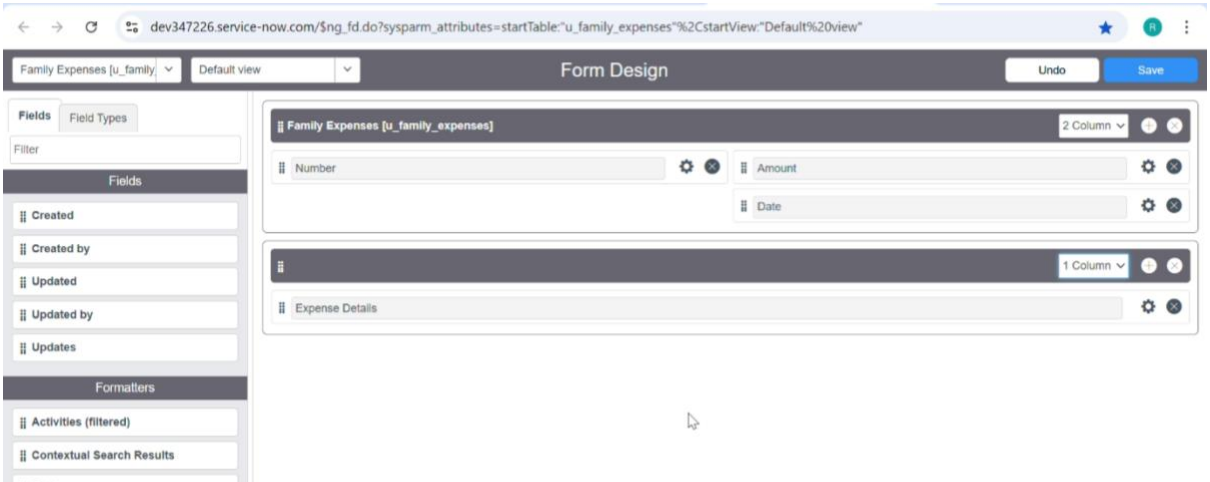
1. Double click on the Number Field/Column.
2. Go down and double click on Advanced view
3. In Default Value:
   Use dynamic default : check the box
   Dynamic default value : Get Next Padded Number
4. Click on Update.
5. Go to All >>In the filter search for Number Maintenance >> select Number Maintenance
6. Click on New.
7. Enter the below Details:
   Table : Family Expenses
   Prefix : MFE
8. Click on Submit



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.

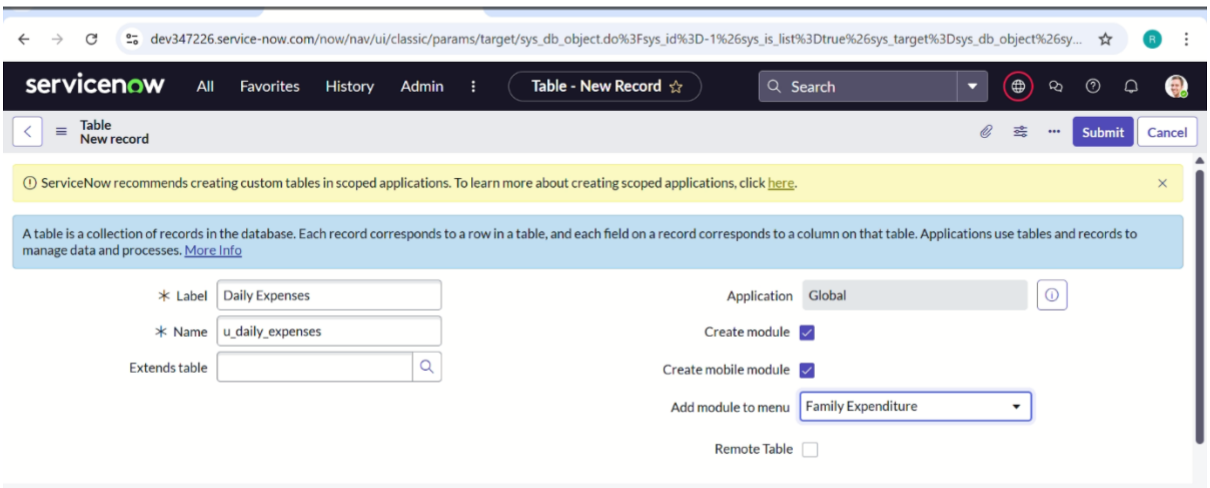


Milestone 4: Table(Daily Expenses)

Activity 1: Creation ofTable(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.



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

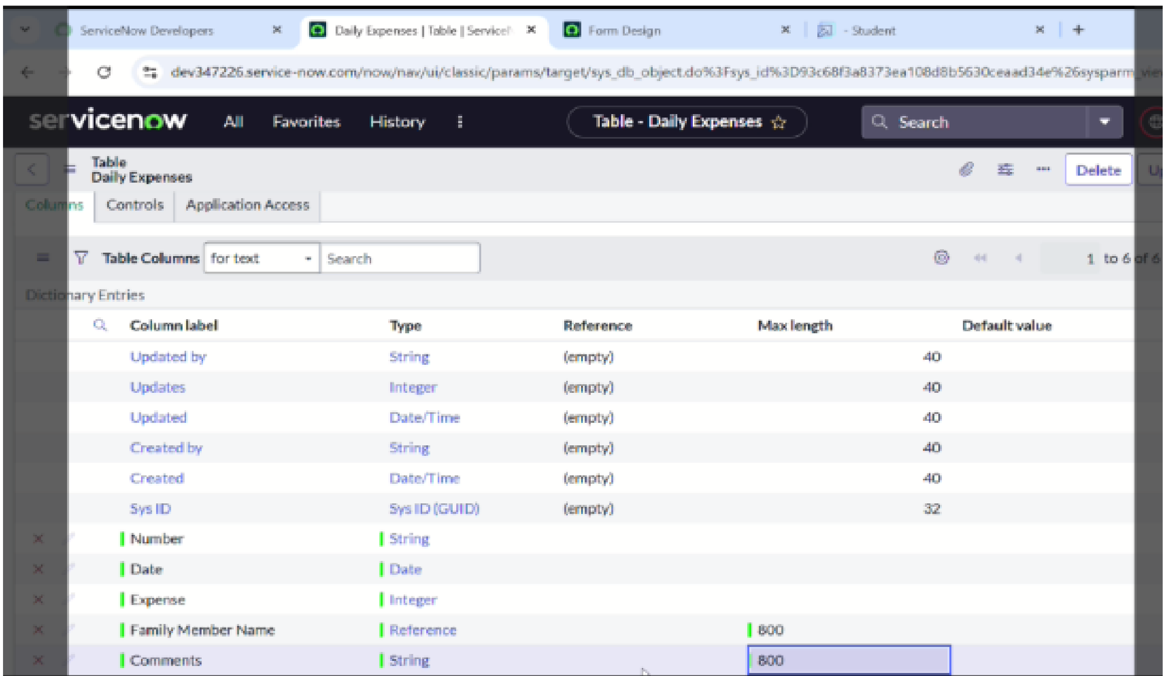
    Type : Reference

    Max length: 800
9. Double click on insert a new row again
10. Give the details as:

    Column label : Comments

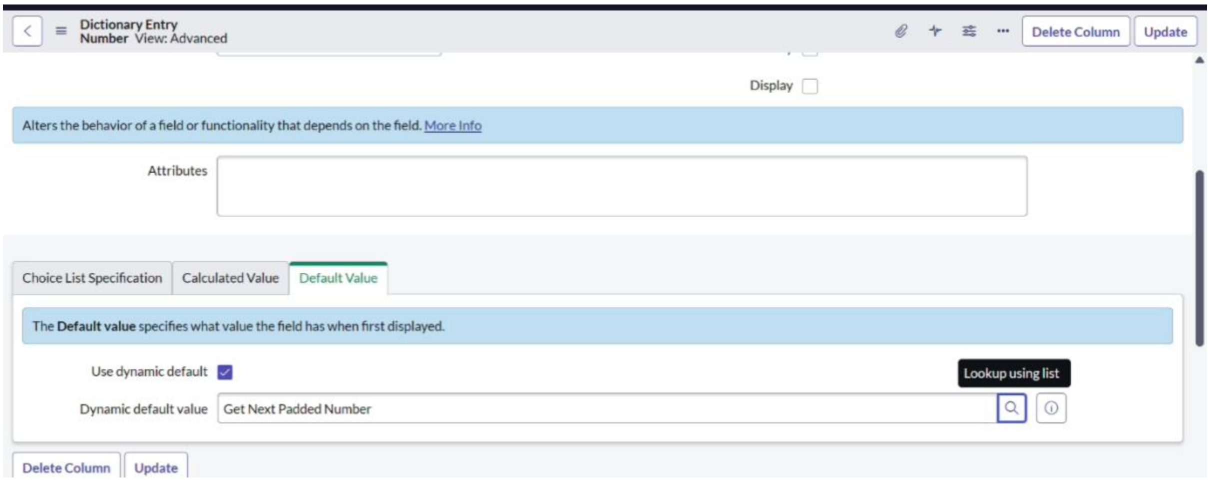
    Type : String

    Max length : 800
11. Go to the Header and right click there>> click on Save.

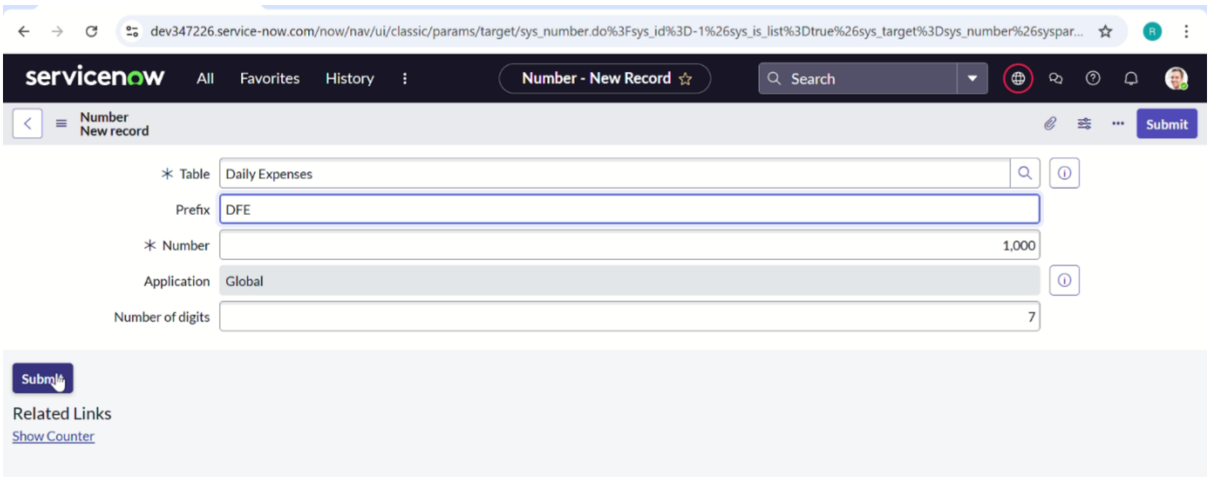


Activity 3: Making Number Field an Auto-Number

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.



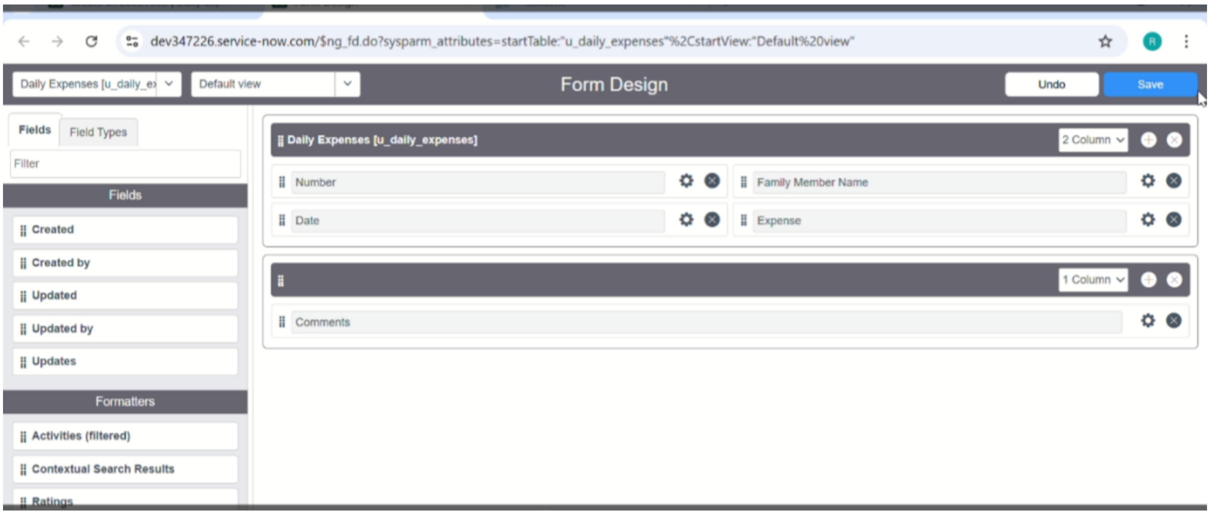
5. Go to All >>In the filter search for Number Maintenance >> select Number Maintenance
6. Click on New.
7. Enter the below Details:
   Table : Daily Expenses
   Prefix : DFE
8. Click on Submit.



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
4. Customize or Drag Drop the form as per your requirement.

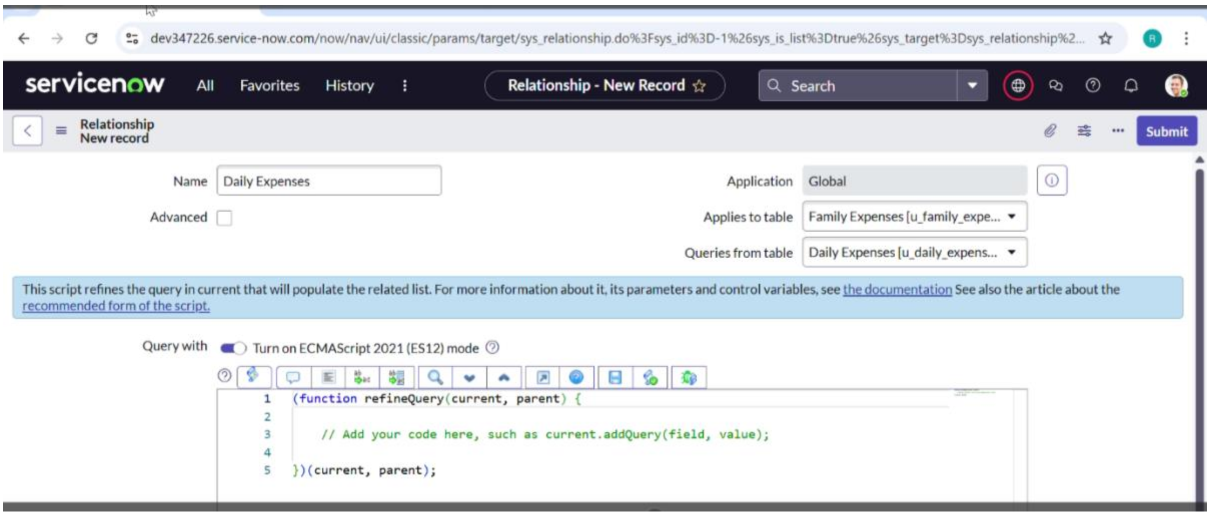
5. Make Number Read-Only Field by clicking on the gear icon and checking Read-Only
6. Make Date, Family Member Name Mandatory Field by clicking on the gear icon and checking Mandatory
7. Click on Save.



Milestone 5: Creation of Relationship

Activity 1: Creation of Relationship between Family Expenses and Daily Expenses tables

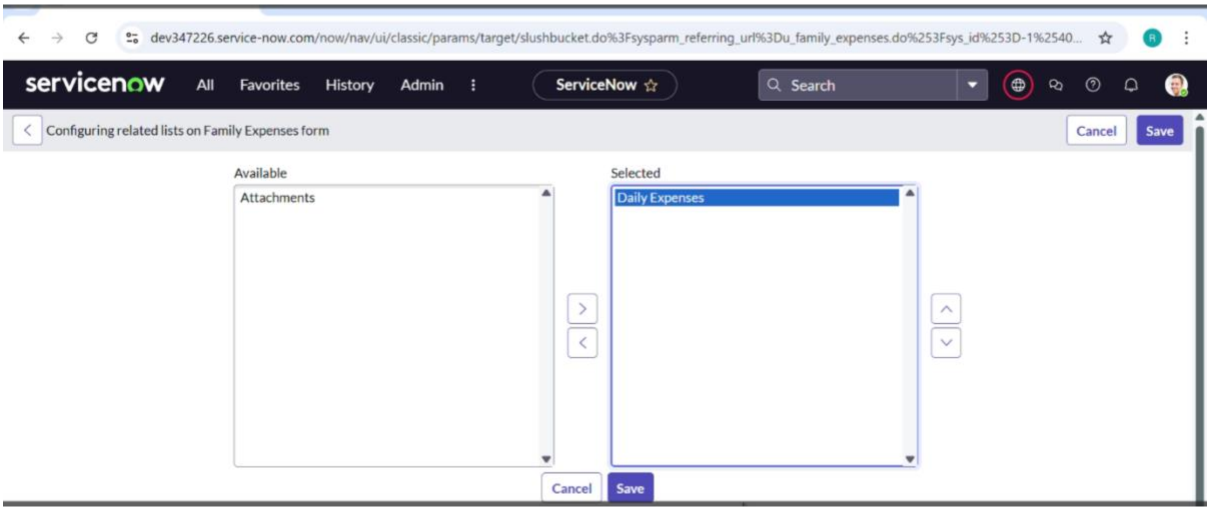
1. Go to All >> In the filter search for Relationships >> Open Relationships
2. Click on New.
3. Enter the details:
   Name : Daily Expenses
   Applies to table : Select Family Expenses
   Daily Expenses : Select Daily Expenses
4. Click Save.



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

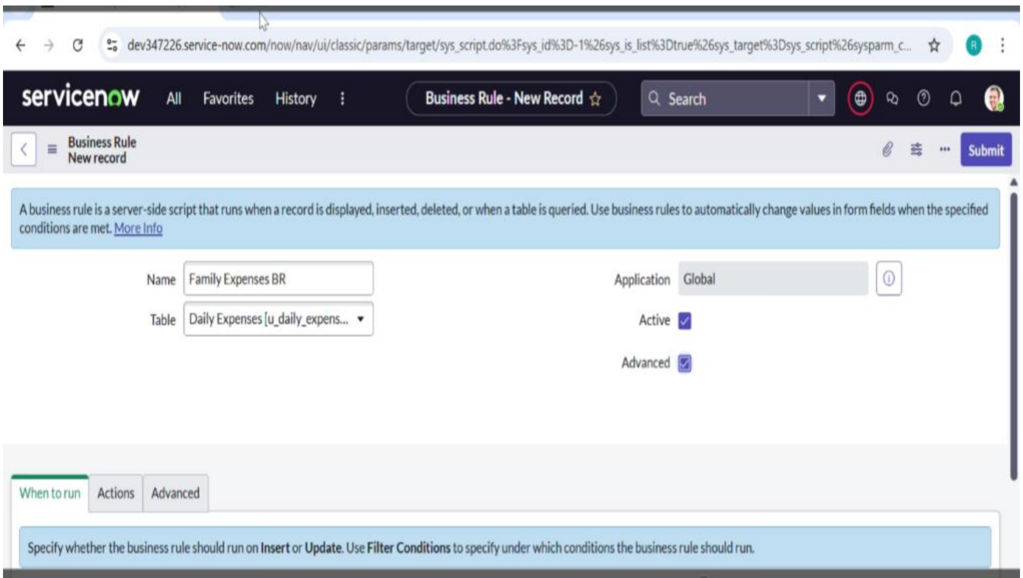


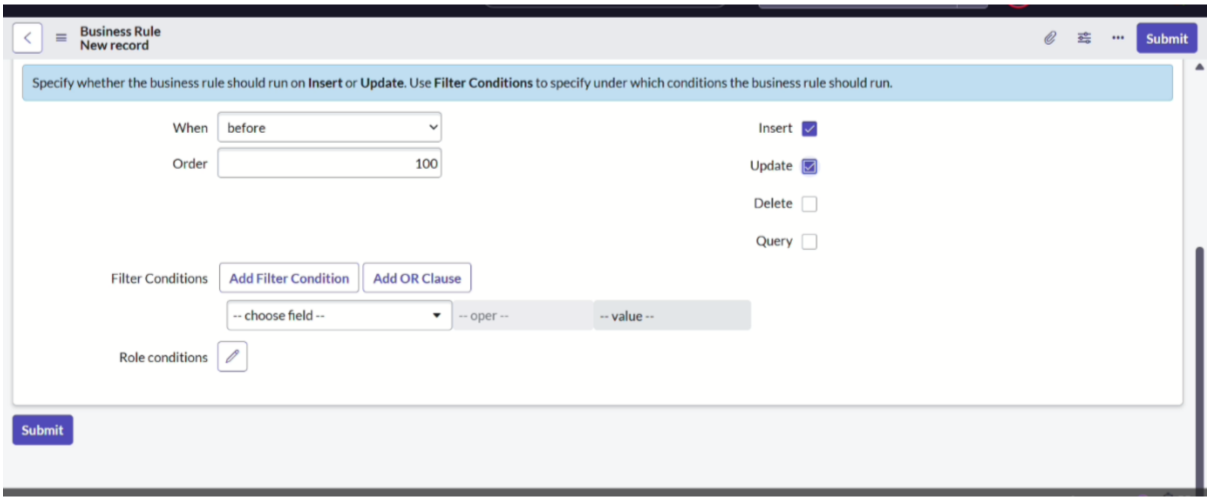
Milestone 7: Business Rules

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



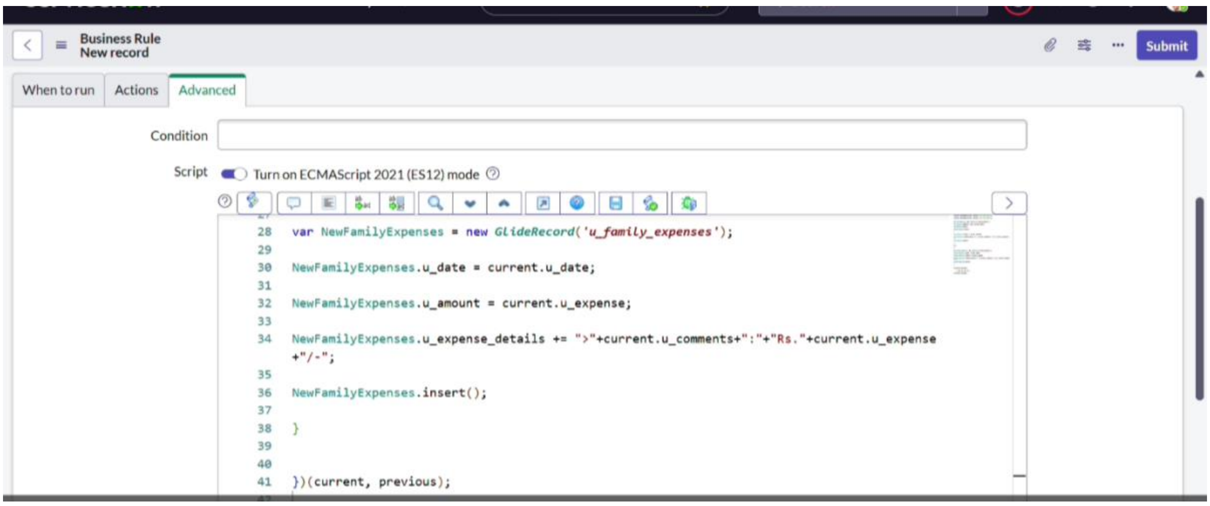


5. In Advance(we write the code): Write the below code >>

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

varFamilyExpenses = 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.

```
(functionrefineQuery(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.