Anjalai Ammal Mahalingam Engineering College

Department of Computer Science And Engineering

Completed the project named as

CALCULATING FAMILY EXPENSES USING SERVICE NOW

Team ID : NM2025TMID08169

Team Size : 4

Team Leader :SURYANANTHAN S

Team member 1 : MOHAMED ANWARDEEN M
Team member 2 : THILAK K

Team member 3 : VIGNESH C

Abstract

The ***Calculating Family Expenses using ServiceNow*** project focuses on developing a smart, efficient, and user-friendly system to manage household finances. The primary purpose of the project is to provide families with a centralized platform to record, categorize, and monitor their daily expenses. The system incorporates key features such as budget setting, real-time expense tracking, data visualization, and detailed financial reporting. By leveraging ServiceNow's automation and integration capabilities, the project ensures seamless performance, scalability, and accessibility. The findings demonstrate that using ServiceNow for expense management enhances financial awareness, simplifies budget control, and supports informed financial decision-making within the family unit.

Introduction

Effective financial management is a crucial aspect of maintaining a stable and organized family life. However, many families face challenges in tracking daily expenses, managing budgets, and analyzing spending patterns efficiently. To address these challenges, the *Calculating Family Expenses using ServiceNow* project aims to develop a comprehensive and automated expense management system that simplifies financial tracking and decision-making.

ServiceNow was chosen as the development platform due to its powerful workflow automation, robust database management, and customizable user interface. Its cloud-based architecture ensures real-time data accessibility, scalability, and seamless integration with other tools. Additionally, ServiceNow's ability to automate routine tasks and generate detailed reports makes it an ideal platform for implementing an efficient expense tracking system.

The main objectives of this project are:

1. To design a centralized platform for recording and categorizing family expenses.
2. To enable users to set and monitor budgets effectively.
3. To provide real-time expense tracking and reporting capabilities.
4. To promote better financial awareness and decision-making within families.

By achieving these objectives, the project seeks to empower families with a practical tool for maintaining financial discipline and ensuring long-term financial well-being.

Problem Statement

Managing family expenses is often a time-consuming and error-prone process, especially when handled manually or through multiple disconnected tools such as spreadsheets or mobile notes. These traditional methods lack automation, centralized data storage, and real-time visibility, making it difficult for families to track spending patterns accurately or maintain budget discipline. As a result, financial decisions are often made without a clear understanding of income distribution, expense categories, or long-term financial goals.

Furthermore, the absence of an integrated system leads to inefficiencies such as duplicate data entry, inconsistent records, and limited reporting capabilities. Families need a reliable, automated solution that can simplify financial management, ensure data accuracy, and provide insightful reports to guide spending behavior.

The *Calculating Family Expenses using ServiceNow* project aims to solve these challenges by creating a unified platform that automates expense tracking, enhances visibility into financial activities, and empowers users to make informed, data-driven financial decisions.

Methodology / System Design

Design Approach

The *Calculating Family Expenses using ServiceNow* project follows a user-centric, automated, and scalable design philosophy, emphasizing simplicity, data accuracy, and accessibility.

Key Principles:

- **User-Centered:** Intuitive interface for all users.
- **Automation:** Minimizes manual entry with workflows and notifications.
- **Accuracy & Transparency:** Ensures reliable financial data and clear spending insights.
- **Scalability:** Supports families of varying sizes and complex financial records.
- **Integration:** Connects seamlessly with other modules and tools.

ServiceNow Modules Used:

- **ServiceNow Studio:** Application development and customization.
- **Service Catalog:** Structured expense submission.
- **Flow Designer:** Automates calculations, reports, and alerts.
- **Tables & Forms:** Stores and updates expense details.
- **Reports & Dashboards:** Visualizes spending trends.
- **Notification Engine:** Sends budget alerts and reminders.
- **Access Control (ACL):** Ensures data security.

Workflow Overview:
Users submit expenses via the Service Catalog. Data is stored in custom tables, totals are updated automatically, alerts are triggered for budget limits, and reports/dashboards provide insights for informed financial decisions.

System Architecture

The *Calculating Family Expenses using ServiceNow* system is designed with a layered and modular architecture to ensure efficiency, scalability, and seamless integration. It consists of three main layers:

- **Presentation Layer:**
  This layer provides the user interface through ServiceNow forms, dashboards, and Service Catalog items. It allows users to easily enter expenses, view reports, and monitor budgets through an intuitive and accessible interface.

- **Application Layer:**
  The core logic of the system resides here. Using **Flow Designer**, **Business Rules**, and **Workflows**, the system automates processes such as expense calculations, budget validation, and notifications when spending exceeds limits. This layer ensures smooth operation and minimal manual intervention.
- **Data Layer:**
  All financial data, including expense details, categories, budgets, and user profiles, are stored in **custom ServiceNow tables**. The platform's built-in database management ensures data consistency, accuracy, and security.

Additional components such as **Reporting and Dashboards** provide visual insights into spending trends, while **Access Control (ACLs)** maintains data privacy and user-based permissions. The system can also integrate with tools like **Excel** or **Performance Analytics** for advanced data analysis.

User Interface (UI) and User Experience (UX)

The *Calculating Family Expenses using ServiceNow* system is designed to provide a simple, intuitive, and efficient user experience.

Forms and Layouts:

- Expense entry is performed through **custom Service Catalog forms** with clearly labeled fields such as expense name, category, amount, date, and payment method.
- Forms are structured for easy navigation, minimizing data entry errors and ensuring consistency.
- Dashboards provide a **visual summary** of expenses, budgets, and spending trends using charts, graphs, and tables for quick comprehension.

User Flows:

- Users log in to the system and submit expenses via the Service Catalog.
- The system automatically categorizes and records entries in custom tables.
- Notifications are triggered if budgets are exceeded, ensuring timely awareness.
- Users can view reports and dashboards to analyze spending patterns and make informed financial decisions.

UX Design Principles:

- **Simplicity:** Minimal steps to complete tasks.
- **Clarity:** Clear labeling and visual hierarchy for forms and dashboards.
- **Responsiveness:** Quick system feedback for entries and updates.
- **Accessibility:** Designed for users with varying technical skills.

This UI/UX approach ensures that users can efficiently manage, monitor, and analyze family expenses with minimal effort and maximum clarity.

Implementation Details:

Platform Setup

The *Calculating Family Expenses using ServiceNow* system requires an initial setup to ensure proper functionality, user access, and data management.

1. Instance Configuration:

- A dedicated **ServiceNow instance** is provisioned for the project.
- Default modules are activated, and custom applications for expense management are created in **ServiceNow Studio**.
- System settings such as **time zones, currency, and notification preferences** are configured to align with family needs.

2. User and Group Creation:

- User accounts are created for each family member or authorized user.
- **Groups** are defined to manage roles and permissions, such as Admin, Member, and Viewer.
- **Access Control Lists (ACLs)** are applied to ensure data privacy, restricting sensitive financial information to authorized users only.

3. Initial Data Setup:

- Custom tables for expenses, categories, and budgets are created.
- Predefined expense categories and sample budget limits are added to facilitate immediate use.

This setup ensures a secure, organized, and fully functional environment for managing family expenses on the ServiceNow platform.

Development and Customization

The development of the *Calculating Family Expenses using ServiceNow* system involved creating custom tables, fields, scripts, and UI policies to meet the specific requirements of family expense management.

1. Custom Tables and Fields:

- **Expense Table:** Stores details such as expense name, category, amount, date, and payment method.
- **Category Table:** Defines expense categories like groceries, utilities, and entertainment.
- **Budget Table:** Tracks monthly or yearly budgets per category and overall family spending.
- Custom fields were added to capture additional information such as payment type, notes, or recurring expenses.

2. Scripts and Business Rules:

- **Business Rules** automatically calculate total expenses and update category-wise totals.
- **Script Includes** handle complex calculations and reusable logic.
- **Client Scripts** validate user input on forms to prevent errors such as negative amounts or invalid dates.

3. UI Policies and Form Customization:

- **UI Policies** enforce mandatory fields, conditional visibility, and read-only attributes for certain fields.
- Forms were customized for ease of navigation, grouping related fields, and enhancing clarity.
- Dashboards and reports were designed to provide real-time insights and visual summaries of expenses versus budgets.

4. Workflow Automation:

- Automated workflows trigger notifications when budget thresholds are exceeded.
- Monthly summary reports are generated automatically and made available on user dashboards.

This development and customization process ensures that the system is tailored to family financial management, providing accuracy, automation, and a user-friendly interface.

Workflow Implementation

The *Calculating Family Expenses using ServiceNow* system uses workflows to automate and streamline expense tracking, notifications, and reporting. Although originally designed for incidents or service requests, ServiceNow workflows were customized to manage financial activities efficiently.

1. Workflow Design:

- **Expense Submission Workflow:** When a user submits a new expense via the Service Catalog, the workflow validates the data, categorizes the expense, and records it in the custom expense table.
- **Budget Monitoring Workflow:** Automatically compares the recorded expense against category and overall budgets. If limits are exceeded, notifications are triggered.
- **Monthly Summary Workflow:** Aggregates all expense data at the end of the month and generates visual reports on dashboards for user review.

2. Workflow Automation Tools:

- **Flow Designer:** Used to create automated sequences for approval, notifications, and reporting.
- **Business Rules:** Ensure calculations such as totals, remaining budgets, and recurring expenses are updated automatically.

- **Notifications and Alerts:** Configured to send emails or system notifications for budget overruns or new reports.

3. Workflow Management:

- Workflows are modular and can be modified to add new rules, categories, or notifications.
- Admin users can monitor workflow execution and track logs to ensure proper functioning and data accuracy.

By implementing these workflows, the system minimizes manual effort, reduces errors, and ensures timely financial insights for families.

Conclusion

The *Calculating Family Expenses using ServiceNow* project successfully provides a centralized and automated platform for tracking, managing, and analyzing family expenses. By leveraging ServiceNow's capabilities, the system simplifies expense entry, categorization, and reporting while ensuring data accuracy and security. Automated workflows, notifications, and dashboards empower users to make informed financial decisions, monitor budgets effectively, and improve overall family financial management. The project demonstrates how technology can streamline household expense tracking and enhance financial awareness.

Future Scope

- **Mobile Accessibility:** Develop a mobile-friendly interface or app to allow expense tracking on-the-go.
- **Banking Integration:** Connect with bank accounts or digital wallets for automatic expense import.
- **Advanced Analytics:** Incorporate predictive analytics to forecast future spending trends and suggest budget optimizations.
- **Multi-Family Support:** Extend the system to manage multiple households or shared expenses among family members.
- **Custom Alerts:** Enable personalized notifications based on user-defined rules or spending patterns.
- **AI Integration:** Use AI for automated expense categorization and anomaly detection to identify unusual spending.

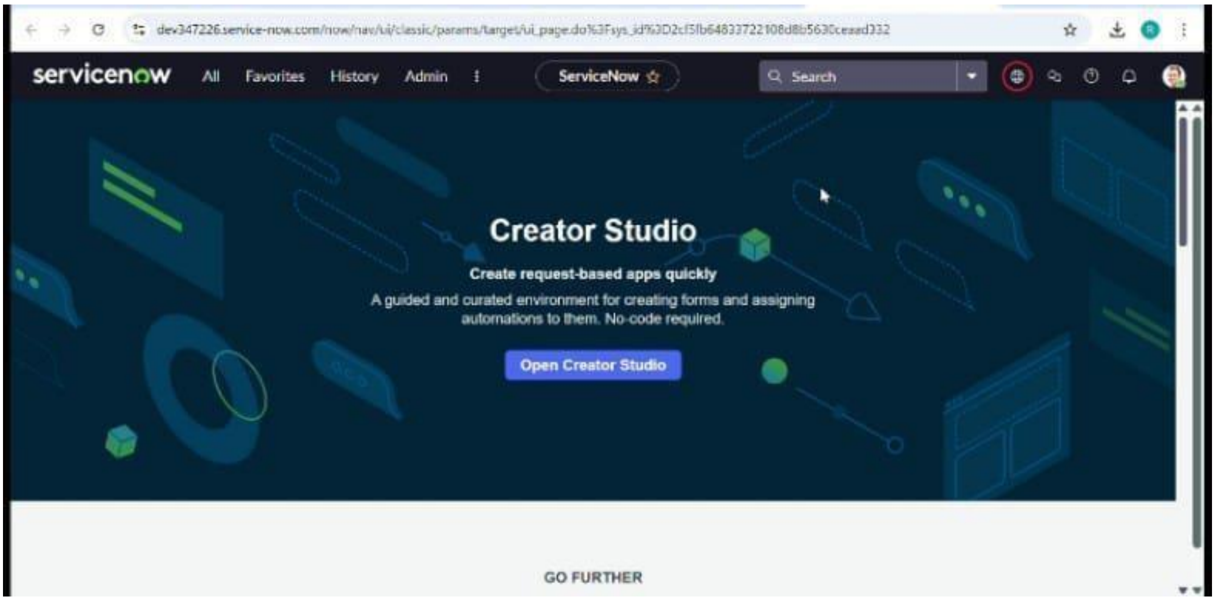
These enhancements can further improve the system's usability, automation, and financial insights, making it a more powerful tool for family expense management.

Setup Steps

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.


dev347226.service-now.com/now/nav/ui/classic/params/target/ui_page.do%3Fsys_id%3D2cf5fb64833722108d8b5630ceaad332
servicenow
All
Favorites
History
Admin
ServiceNow ☆
Search
Open Creator Studio
GO FURTHER

Creator Studio

Create request-based apps quickly

A guided and curated environment for creating forms and assigning automations to them. No-code required.

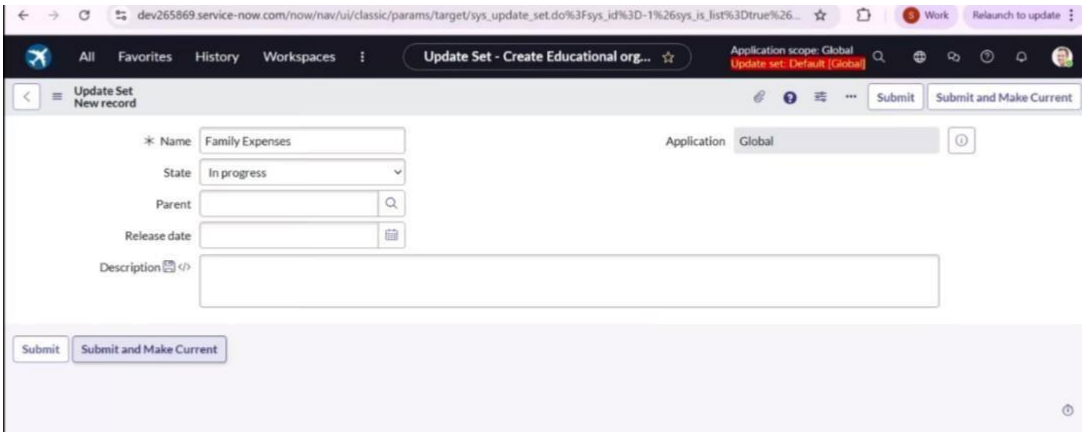
Milestone 2:New Update Set

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.

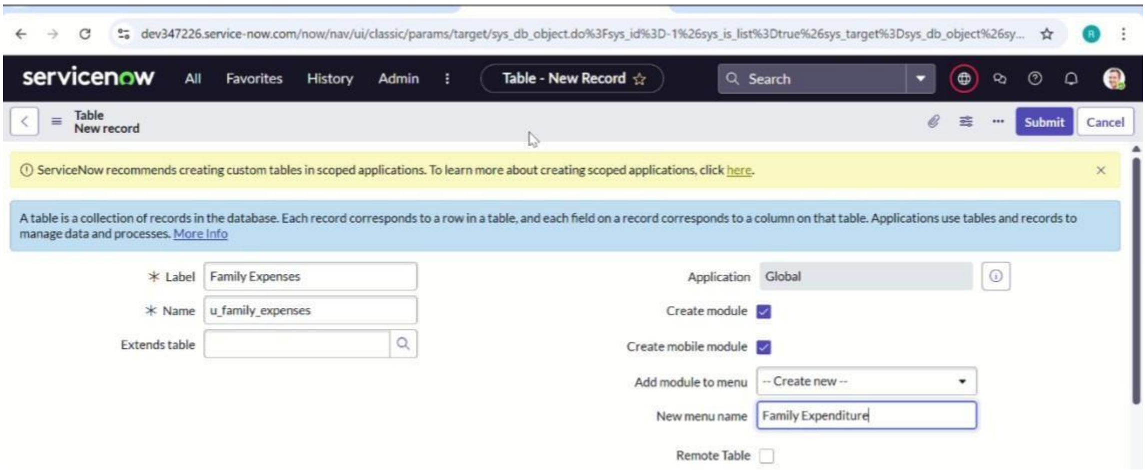

dev265869.service-now.com/now/nav/ui/classic/params/target/sys_update_set.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26...
Work
Relaunch to update
All
Favorites
History
Workspaces
Update Set - Create Educational org...
Application scope: Global
Update set: Default [Global]
Update Set
New record
Submit
Submit and Make Current

* Name	Family Expenses	Application	Global
State	In progress		
Parent			
Release date			
Description			

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.


dev347226.service-now.com/now/nav/ui/classic/params/target/sys_db_object.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26sys_target%3Dsys_db_object%26sy...
servicenow
All
Favorites
History
Admin
Table - New Record
Search
Table
New record
Submit
Cancel

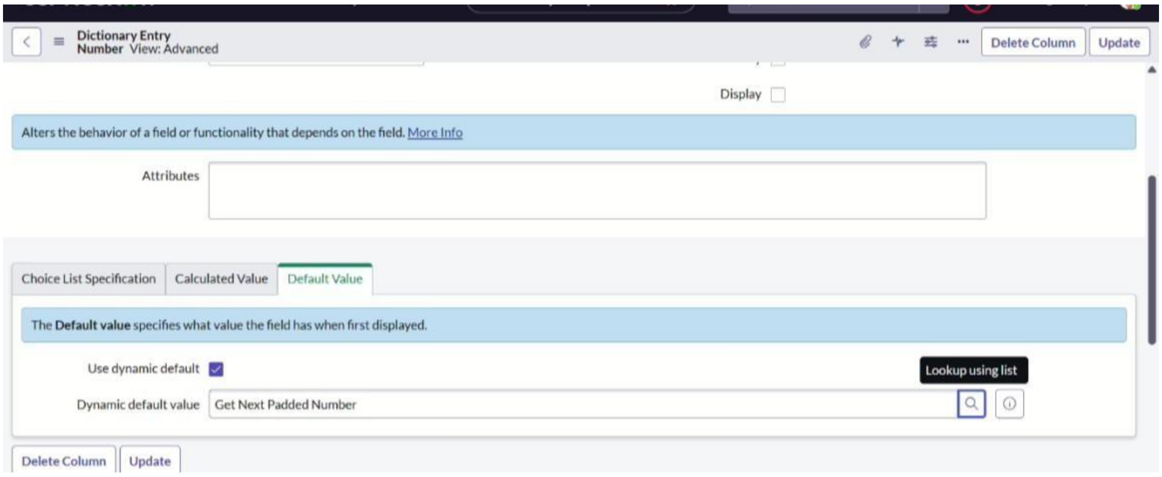
ServiceNow recommends creating custom tables in scoped applications. To learn more about creating scoped applications, click here.

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	Family Expenses	Application	Global
* Name	u_family_expenses	Create module	☑
Extends table		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



Dictionary Entry
Number View: Advanced

Display □

Alters the behavior of a field or functionality that depends on the field. More Info

Attributes

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

Use dynamic default ☑

Dynamic default value: Get Next Padded Number

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.

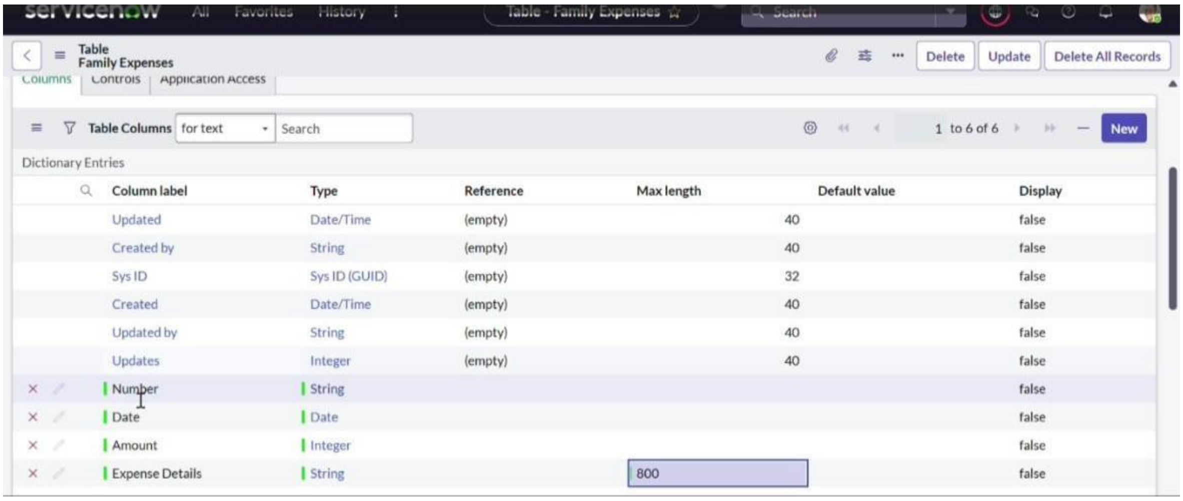


Table
Family Expenses

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
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
Number	String				false
Date	Date				false
Amount	Integer				false
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:

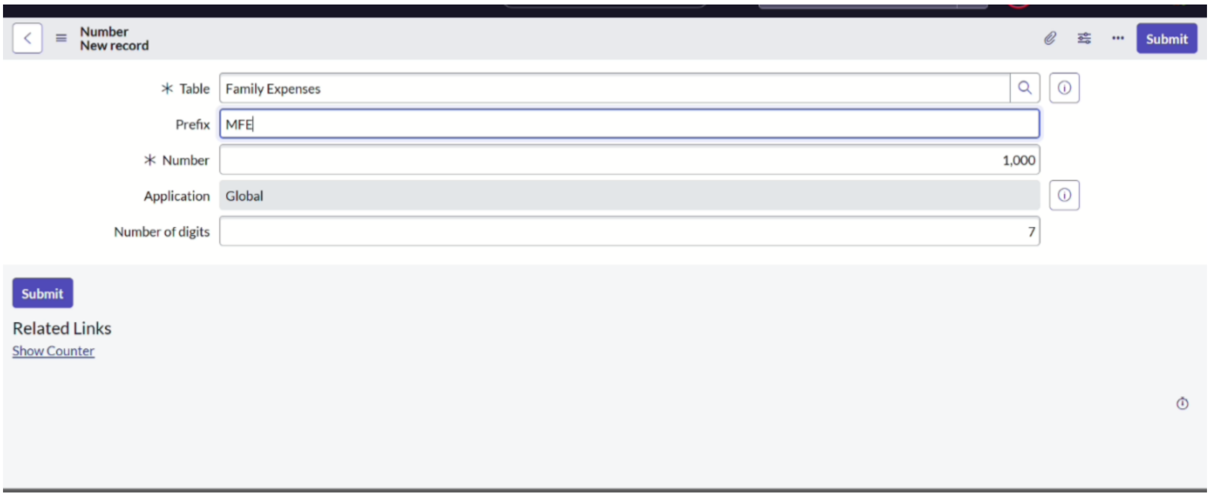
   Use dynamic default : check the box
   Dynamic default value : Get Next Padded Number

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



Number
New record

* Table	Family Expenses
Prefix	MFE
* Number	1,000
Application	Global
Number of digits	7

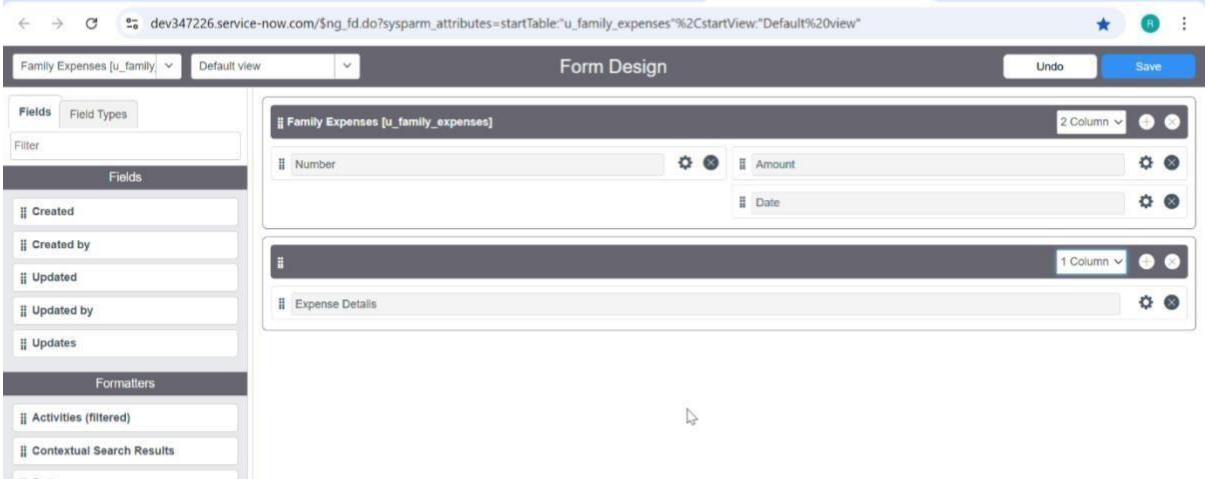
Submit

Related Links
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.



dev347226.service-now.com/$ng_fd.do?sysparm_attributes=startTable:"u_family_expenses"%2CstartView:"Default%20view"

Form Design

Family Expenses [u_family | Default view

Fields: Created, Created by, Updated, Updated by, Updates

Formatters: Activities (filtered), Contextual Search Results

Family Expenses [u_family_expenses] — 2 Column: Number | Amount, Date

1 Column: Expense Details

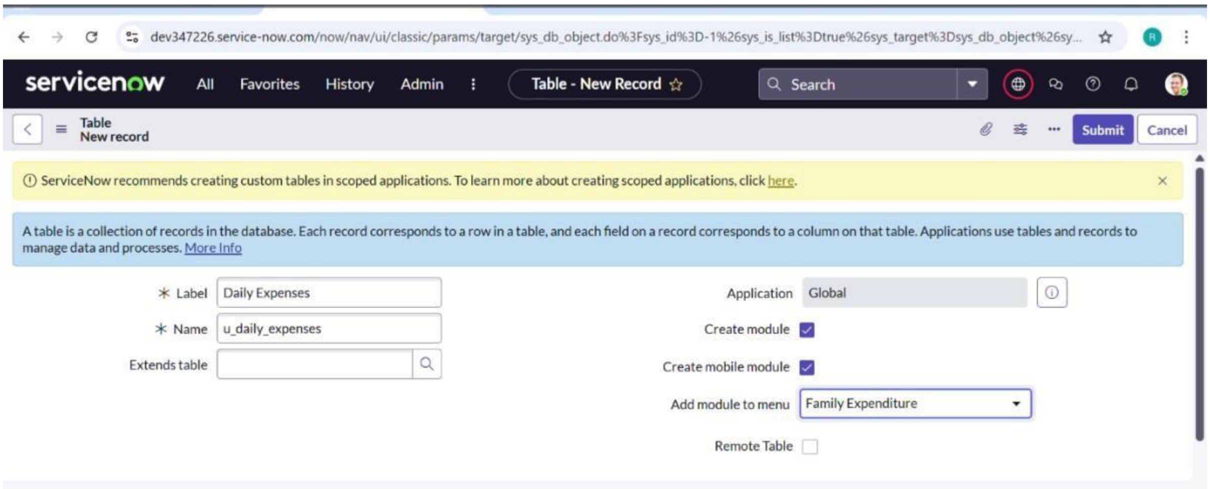
Milestone 4: Table(Daily Expenses)

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.



dev347226.service-now.com/now/nav/ui/classic/params/target/sys_db_object.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26sys_target%3Dsys_db_object%26sy...
servicenow
All
Favorites
History
Admin
Table - New Record ☆
Search
Table
New record
Submit
Cancel

ⓘ ServiceNow recommends creating custom tables in scoped applications. To learn more about creating scoped applications, click here.

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	Daily Expenses	Application	Global
* Name	u_daily_expenses	Create module	☑
Extends table		Create mobile module	☑
		Add module to menu	Family Expenditure
		Remote Table	□

Activity 2: Creation of Columns(Fields)

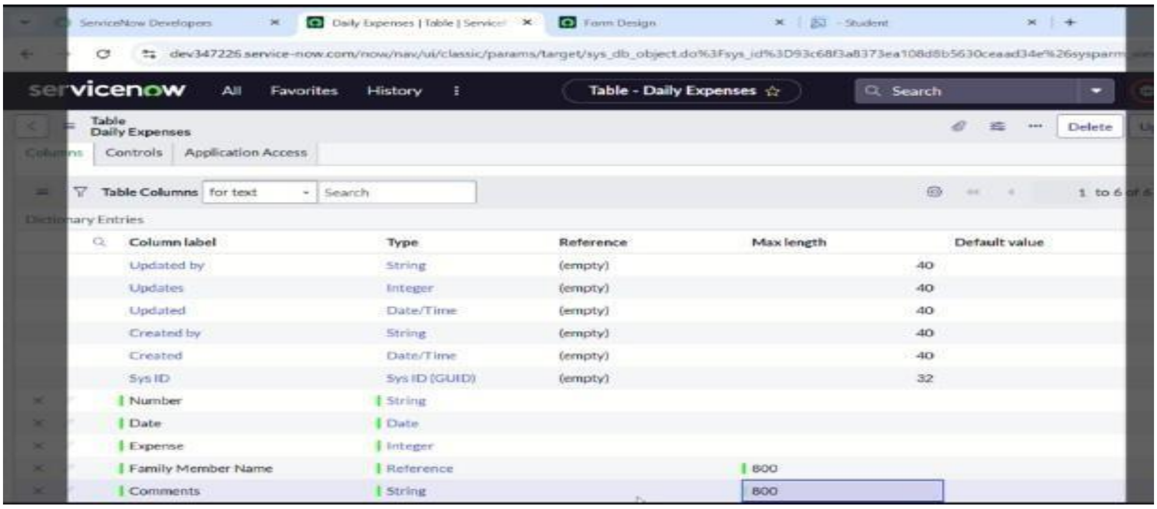
1. Near Columns Double click near insert a new row.
2. Give the details as:
   Column label : Number
   Type : String
3. Double click on insert a new row again
4. Give the details as:
   Column label : Date
   Type : Date
5. Double click on insert a new row again
6. Give the details as:
   Column label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



ServiceNow Developers
Daily Expenses | Table | Service
Form Design
Student
dev347226.service-now.com/now/nav/ui/classic/params/target/sys_db_object.do%3Fsys_id%3D93c68f3a8373ea108d8b5630ceaad34e%26sysparm
servicenow
All
Favorites
History
Table - Daily Expenses
Search
Delete
Columns
Controls
Application Access
for text
Search
1 to 6

Table
Daily Expenses

Table Columns

Dictionary Entries

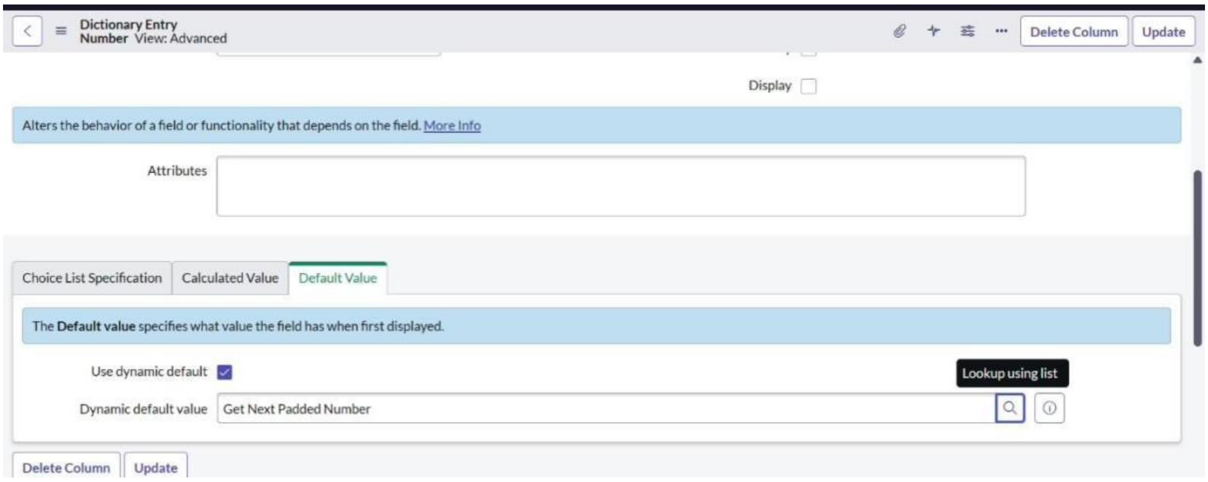
Column label	Type	Reference	Max length	Default value
Updated by	String	(empty)	40	
Updates	Integer	(empty)	40	
Updated	Date/Time	(empty)	40	
Created by	String	(empty)	40	
Created	Date/Time	(empty)	40	
Sys ID	Sys ID (GUID)	(empty)	32	
Number	String			
Date	Date			
Expense	Integer			
Family Member Name	Reference		800	
Comments	String		800	

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.



Dictionary Entry
Number View: Advanced

Alters the behavior of a field or functionality that depends on the field.

Attributes

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

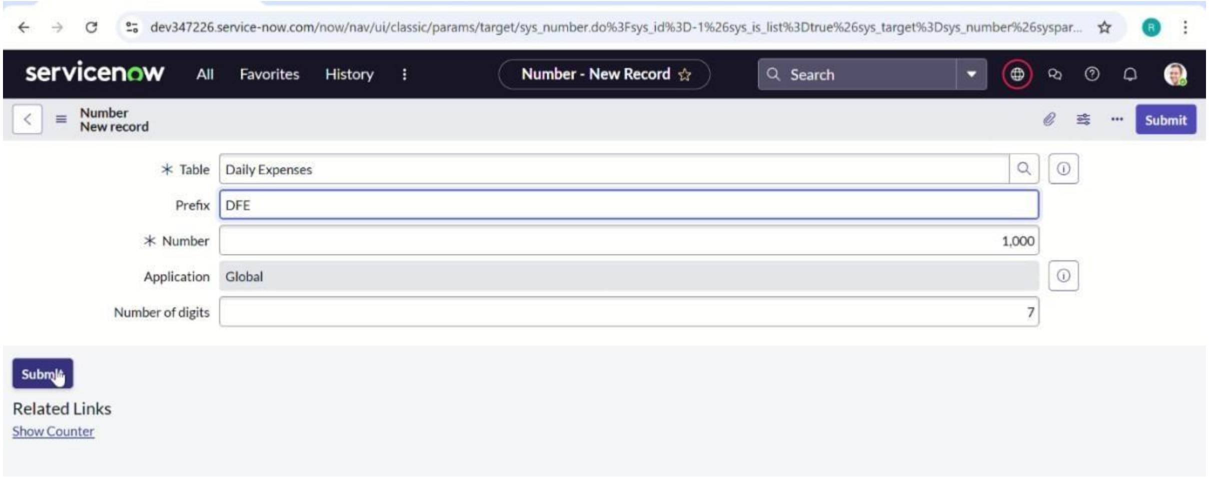
Use dynamic default ☑

Dynamic default value Get Next Padded Number

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.



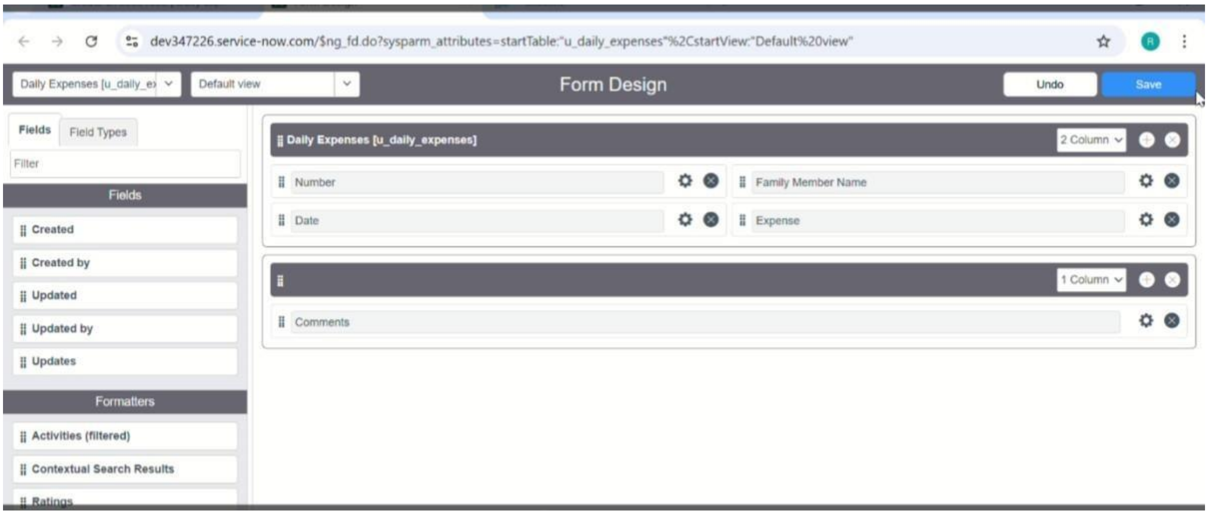
dev347226.service-now.com/now/nav/ui/classic/params/target/sys_number.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26sys_target%3Dsys_number%26syspar...

Number
New record

✱ Table	Daily Expenses
Prefix	DFE
✱ Number	1,000
Application	Global
Number of digits	7

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.



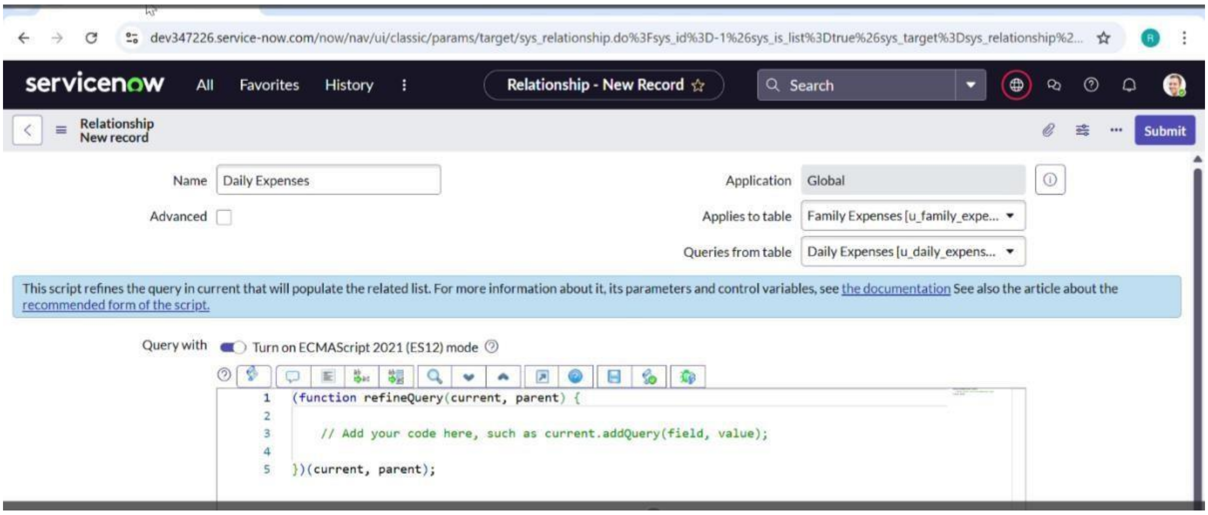
dev347226.service-now.com/$ng_fd.do?sysparm_attributes=startTable:"u_daily_expenses"%2CstartView:"Default%20view"

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.


dev347226.service-now.com/now/nav/ui/classic/params/target/sys_relationship.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26sys_target%3Dsys_relationship%2...
servicenow
All
Favorites
History
Relationship - New Record
Search
Relationship
New record
Submit

Name: Daily Expenses

Advanced: □

Application: Global

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.

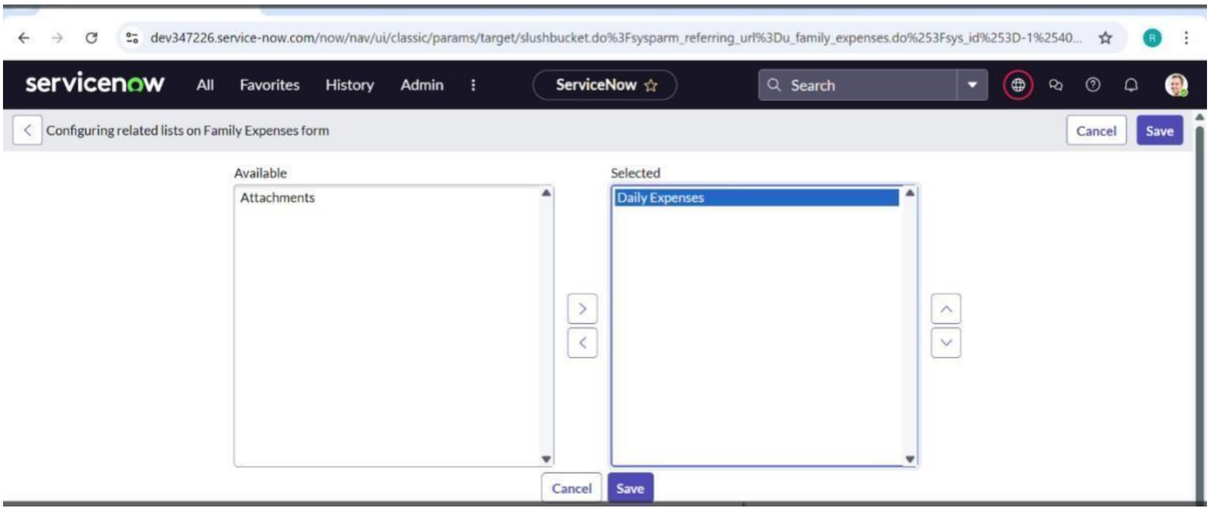
Query with: Turn on ECMAScript 2021 (ES12) mode

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

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

})(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


dev347226.service-now.com/now/nav/ui/classic/params/target/slushbucket.do%3Fsysparm_referring_url%3Du_family_expenses.do%253Fsys_id%253D-1%2540...
servicenow
All
Favorites
History
Admin
ServiceNow
Search
Cancel
Save
Cancel
Save

Configuring related lists on Family Expenses form

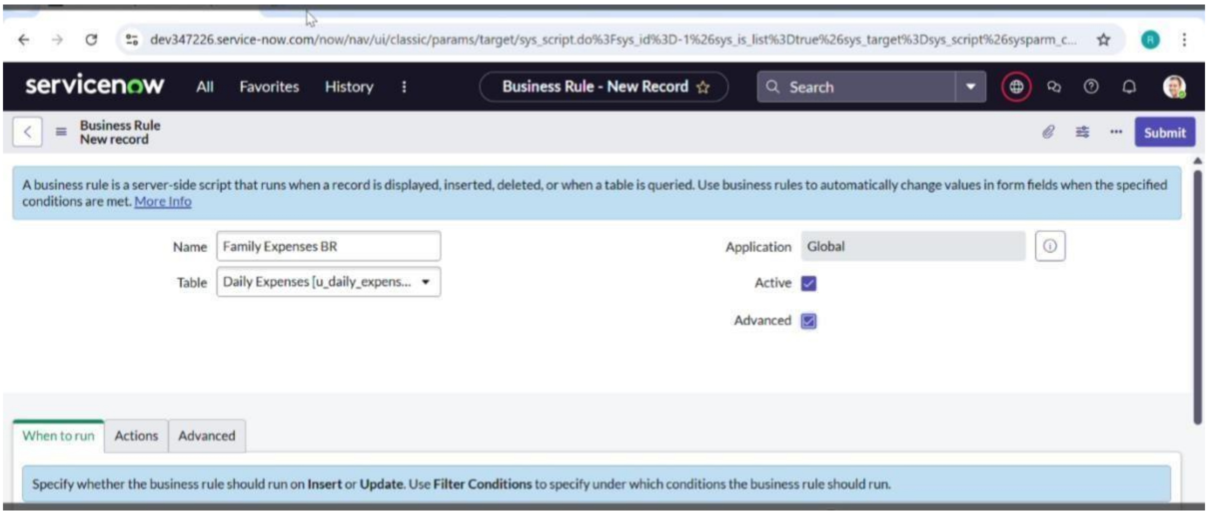
Available: Attachments

Selected: Daily Expenses

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

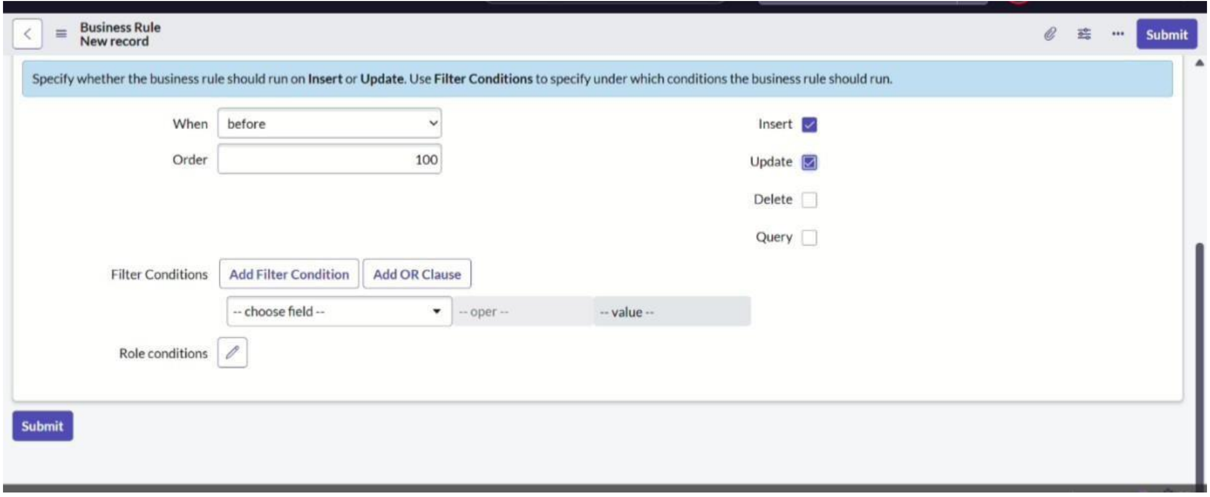

dev347226.service-now.com/now/nav/ui/classic/params/target/sys_script.do%3Fsys_id%3D-1%26sys_is_list%3Dtrue%26sys_target%3Dsys_script%26sysparm_c...
servicenow
All
Favorites
History
Business Rule - New Record
Search
Business Rule
New record
Submit
More Info
When to run
Actions
Advanced

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
Table: Daily Expenses [u_daily_expens...
Application: Global
Active: ☑
Advanced: ☑

Specify whether the business rule should run on **Insert** or **Update**. Use **Filter Conditions** to specify under which conditions the business rule should run.

4. In when to run Check Insert and Update



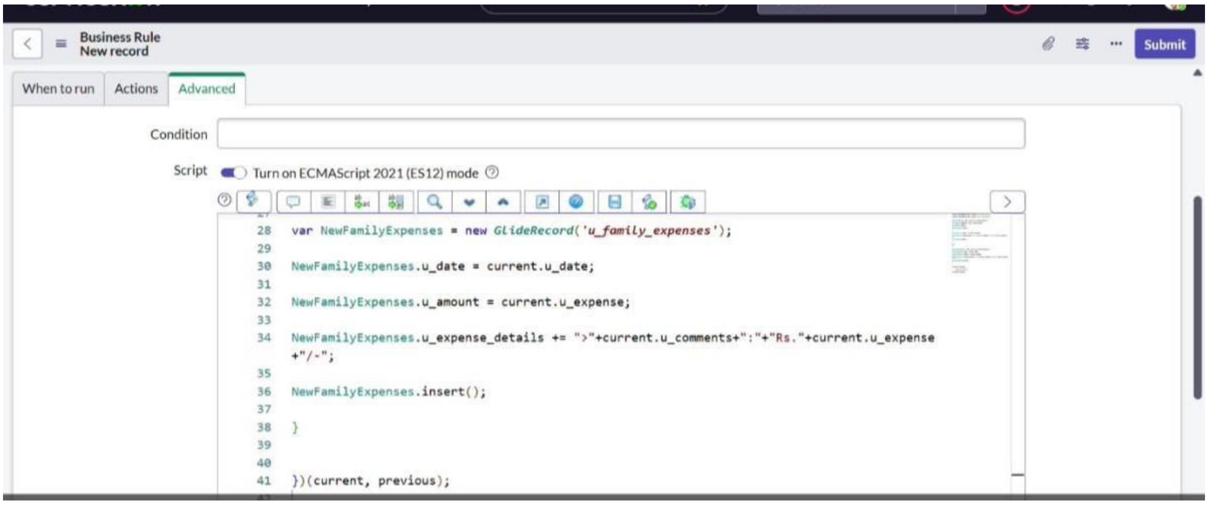
Specify whether the business rule should run on **Insert** or **Update**. Use **Filter Conditions** to specify under which conditions the business rule should run.

When: before
Order: 100
Insert: ☑
Update: ☑
Delete: □
Query: □
Filter Conditions
Role conditions

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.



```
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);
```

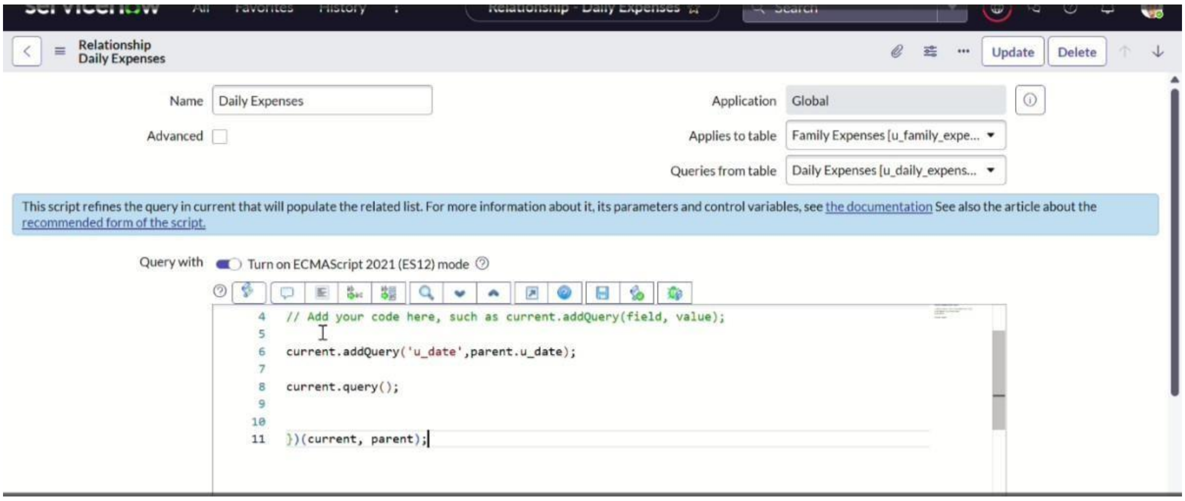
Milestone 8: Relationship

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.



Name: Daily Expenses

Advanced □

Application: Global

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

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

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

current.query();


})(current, parent);
```