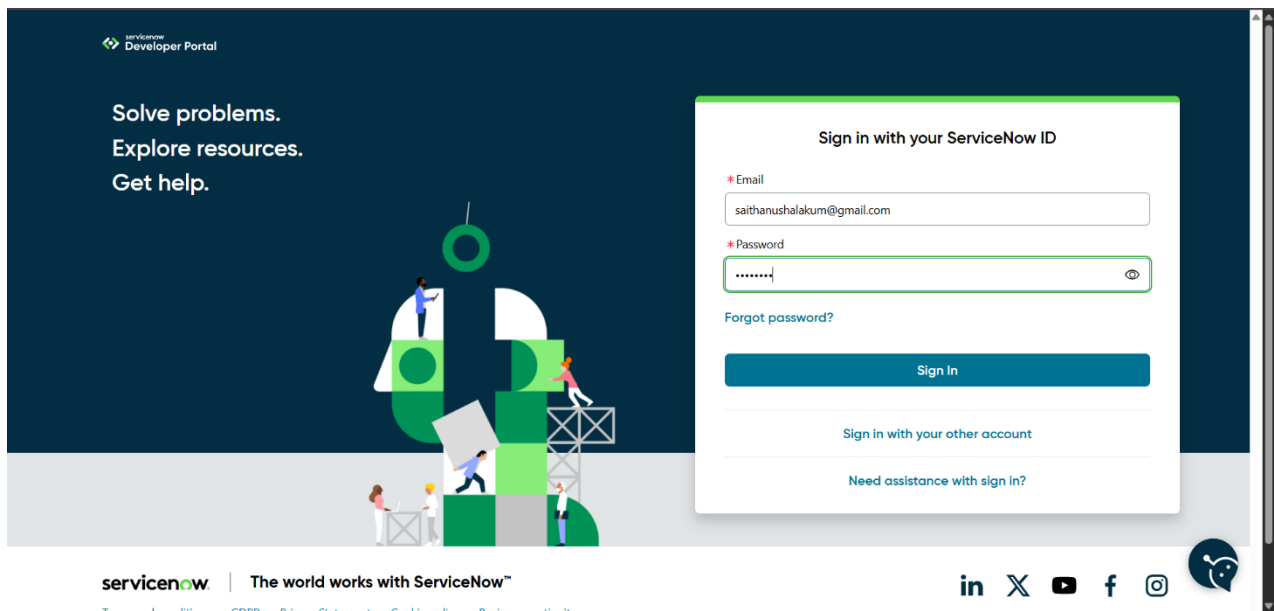


Calculating Family Expenses Using ServiceNow

TEAM ID	LTVIP2025TMID31080
PROJECT NAME	CALCULATING FAMILY EXPENSES USING SERVICENOW
FACULTY MENTOR(S) NAME	

STEP-1: Setting Up ServiceNow Instance

- Sign In to the Developer Account

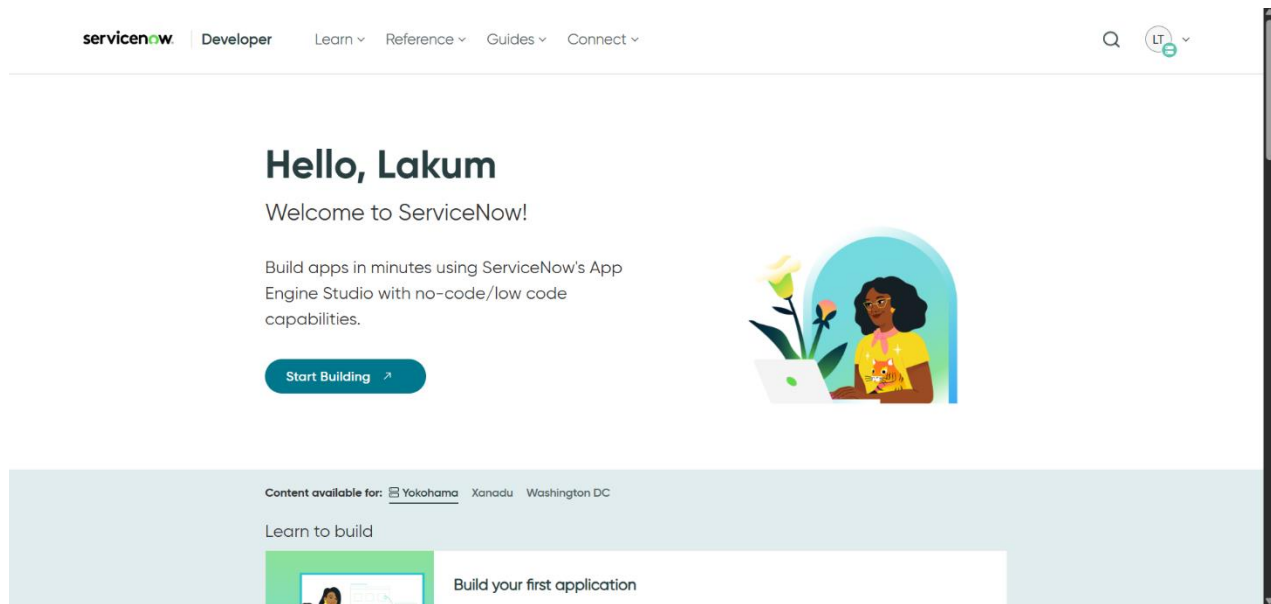


- **Request a Personal Developer Instance (PDI)**
 1. Click on the profile icon at the top-right corner.
 2. Select "Manage Instance" or go directly to the link.
 3. Click "Request Instance".
 4. Choose the latest version (default selection is recommended).
 5. Click "Request" again.

- **Access your Instance**

6. You'll get an instance URL like: <https://devXXXX.service-now.com>
7. Click "Open Instance" to launch it.
8. Your admin credentials (username & password) will be shown-copy and save them securely.

- **Log in to Your Instance**



Step-2: Creation of New Update Set

Key Objective: Track, group, and manage configuration changes or customization made in a ServiceNow instance, so they can be migrated or deployed to another instance (like from development to test or production) in a controlled and organized way.

ServiceNow Developers x ServiceNow x

dev220838.service-now.com/now/nav/ui/classic/params/target/ui_page.do%3Fsys_id%3D865601cd830622101d826260cead3bd

servicenow All Favorites History Workspaces ServiceNow Search

local update set

FAVORITES
No Results

ALL RESULTS
System Update Sets
Local Update Sets

Creator Studio

Create request-based apps quickly

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

Open Creator Studio

GO FURTHER

Power your workflow applications

Activate Windows
Go to Settings to activate Windows.

Type here to search

Very humid 08:45 25-06-2025 ENG

ServiceNow Developers x Create Family Expenses 2 | Upd x

dev220838.service-now.com/now/nav/ui/classic/params/target/sys_update_set.do%3Fsys_id%3D-1%26sys_js_list%3Dtrue%26sys_target%3Dsys_update_set%26sysparm_chec...

servicenow All Favorites History Update Set - Create Family Expense... Search

Update Set
New record

Submit Submit and Make Current

* Name Application

State

Parent

Release date

Description

Submit Submit and Make Current

Type here to search

The art of crafting Ja... 08:47 25-06-2025 ENG

Step-3: Creation of Table:

- **Creation of Family Expenses Table:**

Key Objective:

1. To record individual expenses (e.g., food, bills, travel) for each family member.
2. To categorize and organize expenses by type, date, or person.
3. To enable calculation of total and individual spending using scripts or reports.
4. To provide a digital system for managing household finances within the ServiceNow platform

The screenshot shows the ServiceNow 'Table - New Record' form. The 'Label' field is set to 'Family Expenses' and the 'Name' field is set to 'u_st_family_expenses'. A context menu is open over the form, showing options like 'Save', 'Track in Update Sets', 'Configure', 'Export', 'Create Favorite', 'Copy URL', 'Copy sys_id', and 'Reload form'. The 'Application' is set to 'Global'. The 'Create module' checkbox is checked. The 'mobile module' checkbox is also checked. The 'module to menu' dropdown is set to '-- Create new --'. The 'New menu name' field is set to 'Family Expenses'. The 'Remote Table' checkbox is checked. Below the form, the 'Columns' tab is selected, showing a table with columns: Column label, Type, Reference, Max length, Default value, and Display. The table is currently empty, with a row for 'Insert a new row...'. The bottom of the screen shows the Windows taskbar with the date 25-06-2025 and time 08:54.

- **Creation of Columns (Fields):**

Key Objective: Creating column fields in the Family Expenses table serves the purpose of capturing detailed and organized data for each expense entry. These fields ensure that the data is structured, easy to query, and meaningful for analysis.

● **Making Number Field An Auto-Number:**

ServiceNow Developers | Family Expenses | Table | ServiceNow | Update Sets | ServiceNow

dev220836.service-now.com/now/nav/ui/classic/params/target/sys_db_object.do%3Fsys_id%3D3f25896d8352a6501d826260cead3c3%26sysparm_record_target%3Dsys_db_...

servicenow All Favorites History Admin Table - Family Expenses Search

Table Family Expenses Delete Update Delete All Records

Columns Controls Application Access

Table Columns Default value Search

Dictionary Entries

Column label	Type
Updated	Date/Time
Date	Date
Created by	String
Expense Details	String
Created	Date/Time
Amount	Integer
Number	String
Sys ID	Sys ID (GUID)
Updates	Integer
Updated by	String

Max length Default value Display

40	false
40	false
40	false
800	false
40	false
40	false
40	false
32	false
40	false
40	false

1 to 10 of 10 New

Save
Analyze Access
Show File Properties
Move to Application...
Show Latest Update
Show Dictionary Record
Configure
Export
View
Create Favorite
Copy URL
Copy sys_id
Show XML
History
Reload form

Activate Windows
Go to Settings to activate Windows.

Type here to search 25°C Haze 08:58 25-06-2025

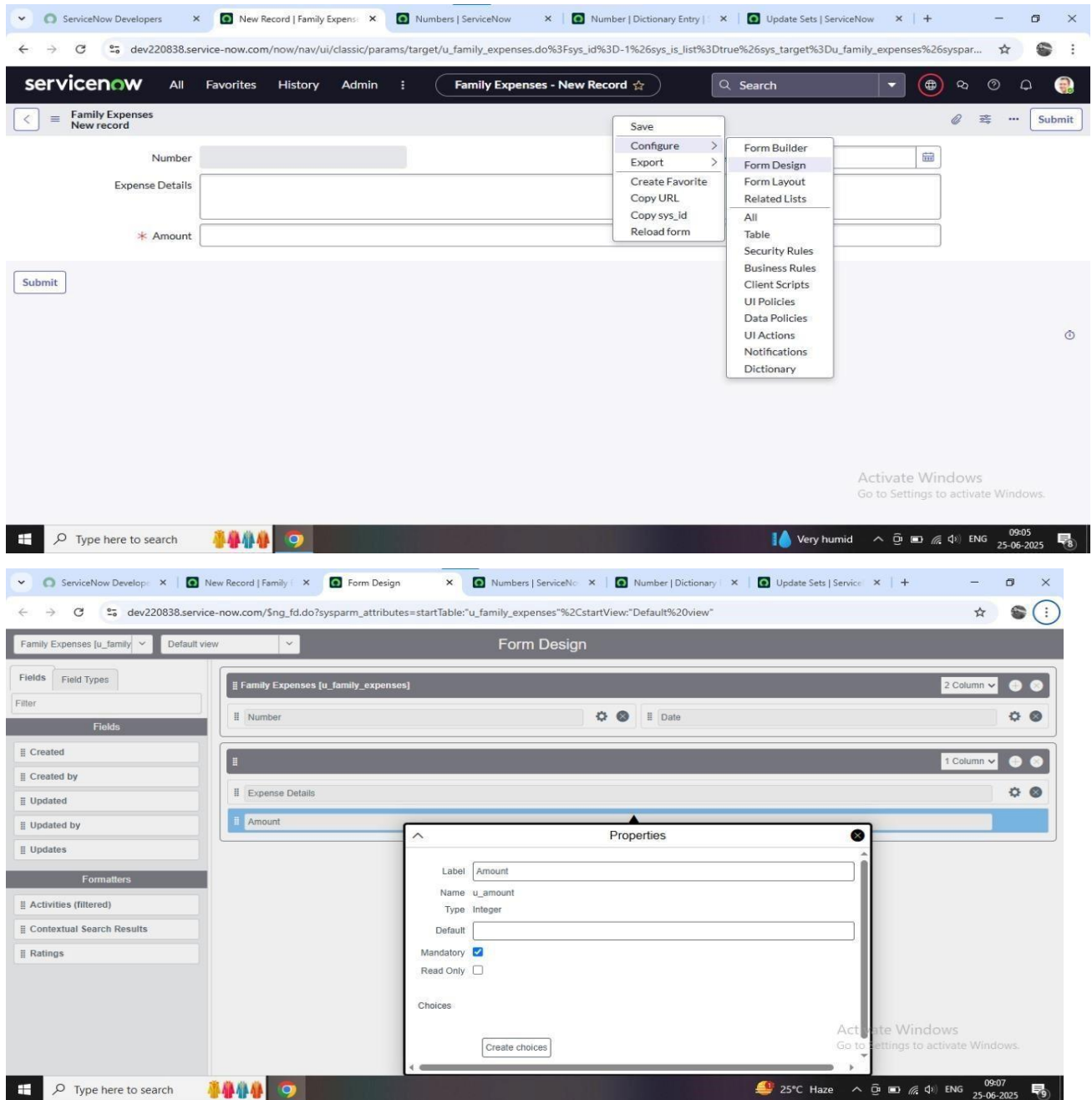
Key Objective: Automatically generate a unique, sequential identifier for each expense record, ensuring easy tracking, reference, and organization of entries without manual input.

This screenshot shows the 'Dictionary Entry - Number' configuration page in ServiceNow. The 'Default Value' tab is active, showing the 'Use dynamic default' checkbox checked. The 'Dynamic default value' field contains 'Get Next Padded Number'. Below this, there are buttons for 'Delete Column' and 'Update'. The 'Access Controls' tab is also visible, showing 'Name' and 'Search' fields. The bottom of the page shows the Windows taskbar with the time 09:01 on 25-06-2025.

This screenshot shows the 'Number - New Record' configuration page in ServiceNow. The 'Table' field is set to 'Family Expenses'. The 'Prefix' field is set to 'MFE'. The 'Number' field is set to '1,000'. The 'Application' field is set to 'Global'. The 'Number of digits' field is set to '7'. Below these fields, there is a 'Submit' button. The bottom of the page shows the Windows taskbar with the time 09:03 on 25-06-2025.

- **Configure The Form:**

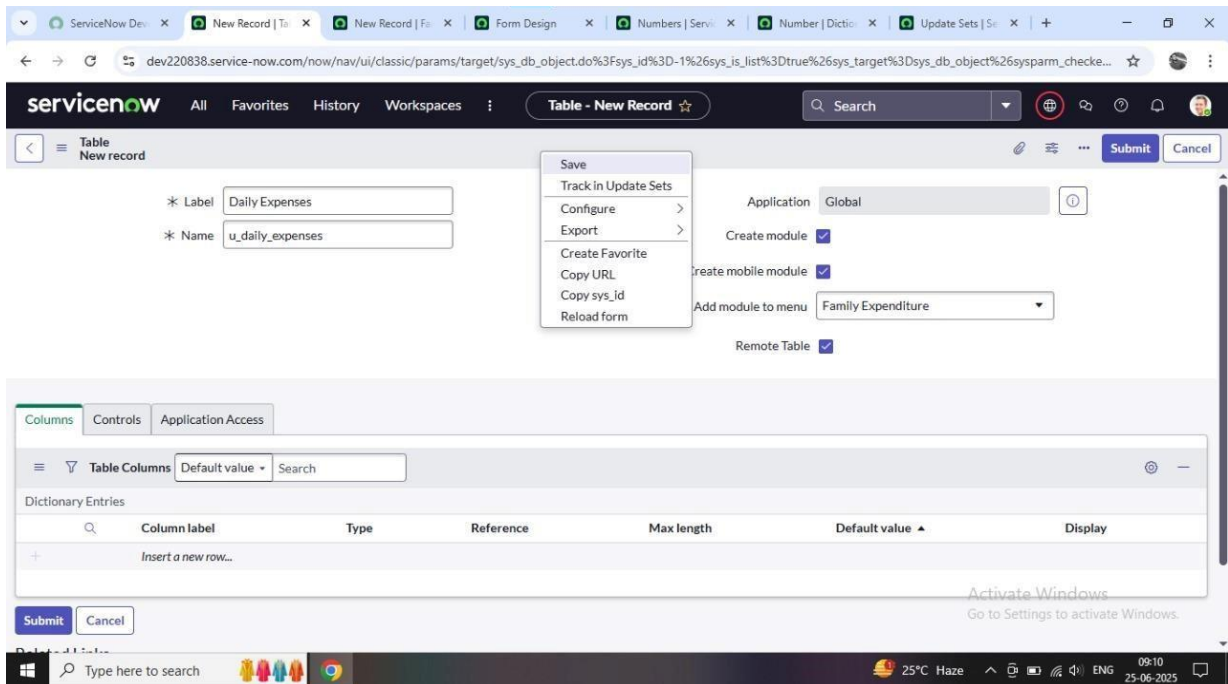
Key Objective: Customize the user interface to display only the relevant fields in a clear, organized, and user-friendly layout-making it easier to enter, view and manage expense records efficiently.



Step-4: Creation of Table (Daily Expenses)

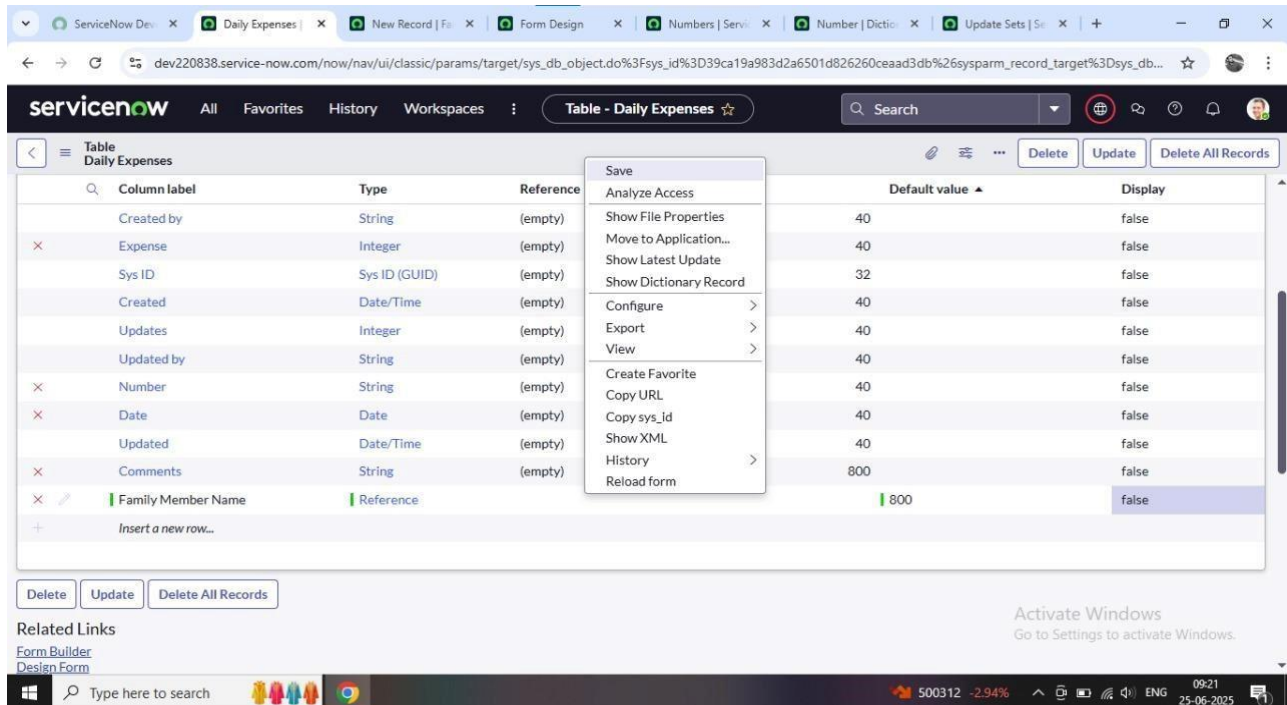
- **Creation of Daily Expenses Table:**

Key Objective: Record and manage everyday spending data in a structured format, enabling users to track daily financial activity, monitor patterns, and generate useful reports for budgeting or analysis.



● Creation of Column (Fields):

Key Objective: Structure and organize daily financial data in a way that supports accurate entry, easy tracking, and efficient reporting of daily spending activities.



- **Making Number Field An Auto-Number:**

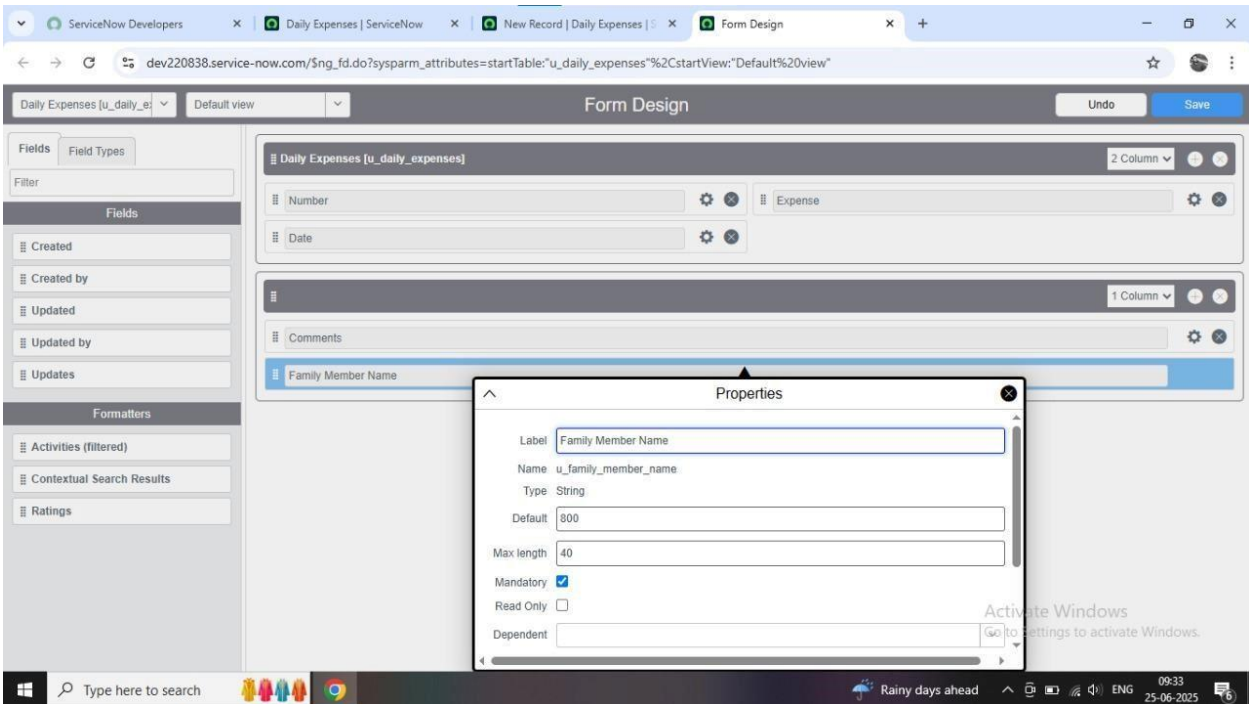
Key Objective: Automatically generate a unique and sequential identifier for each daily expense record, ensuring easy tracking, organization, and reference of individual entries without manual effort.

The screenshot shows the 'Dictionary Entry - Number' configuration page in ServiceNow. The 'Default Value' tab is selected, showing the 'Use dynamic default' checkbox checked and the 'Default value' field set to 'Get Next Padded Number'. The 'Attributes' field is empty. The 'Access Controls' tab is also visible at the bottom.

The screenshot shows the 'Number - New Record' configuration page in ServiceNow. The 'Table' is set to 'Daily Expenses', the 'Prefix' is 'DFE', the 'Number' is set to '1,000', the 'Application' is 'Global', and the 'Number of digits' is set to '7'. The 'Submit' button is visible at the bottom left.

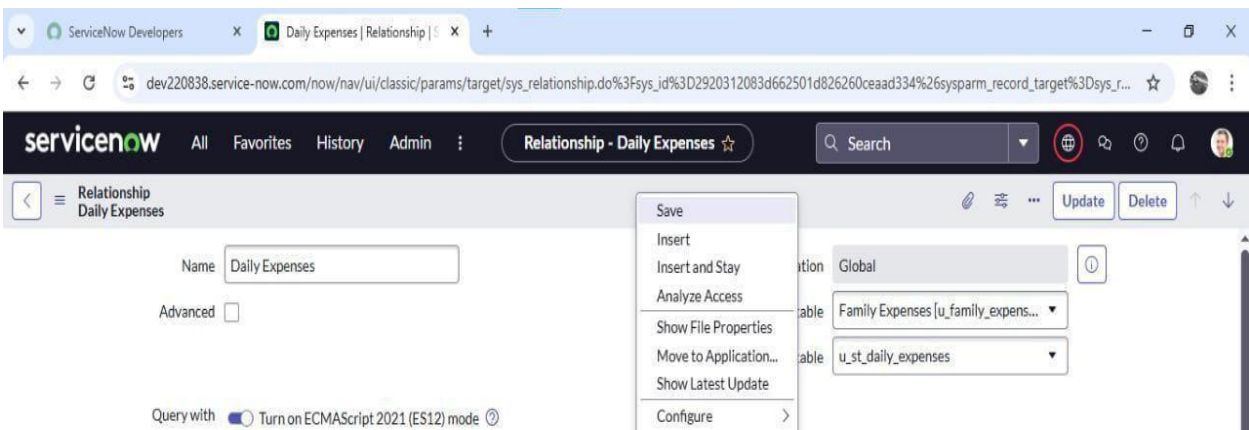
- **Configure the Form:**

Key Objective: Design a user-friendly, well-organized interface that makes it easy to input, view, and manage daily expense records accurately and efficiently.



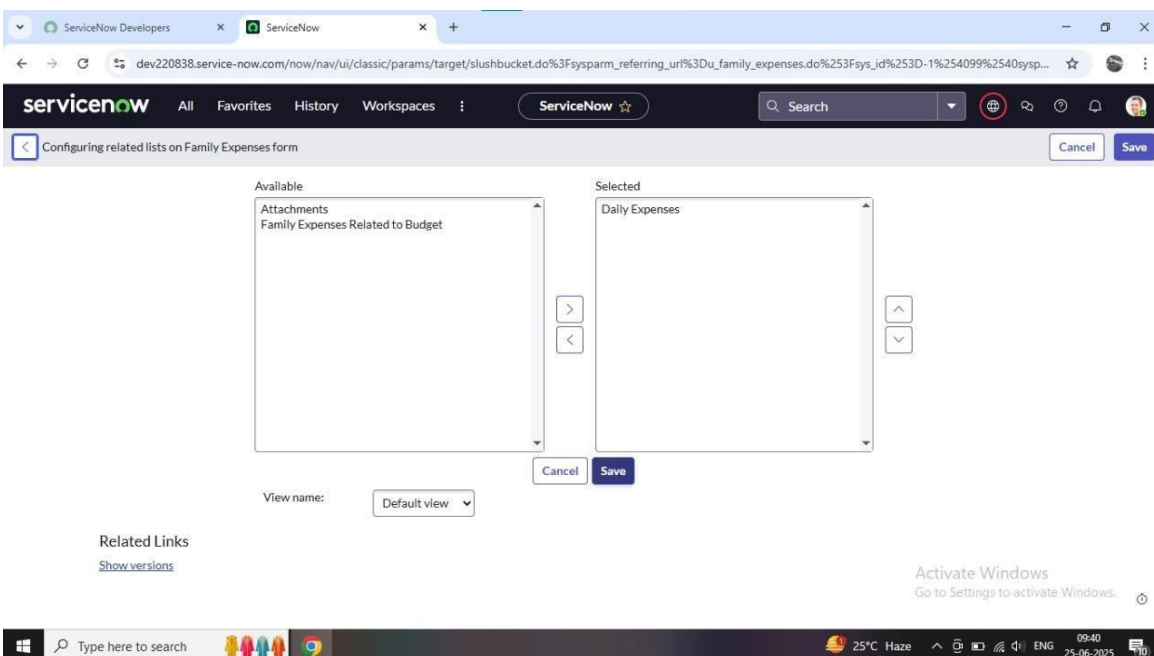
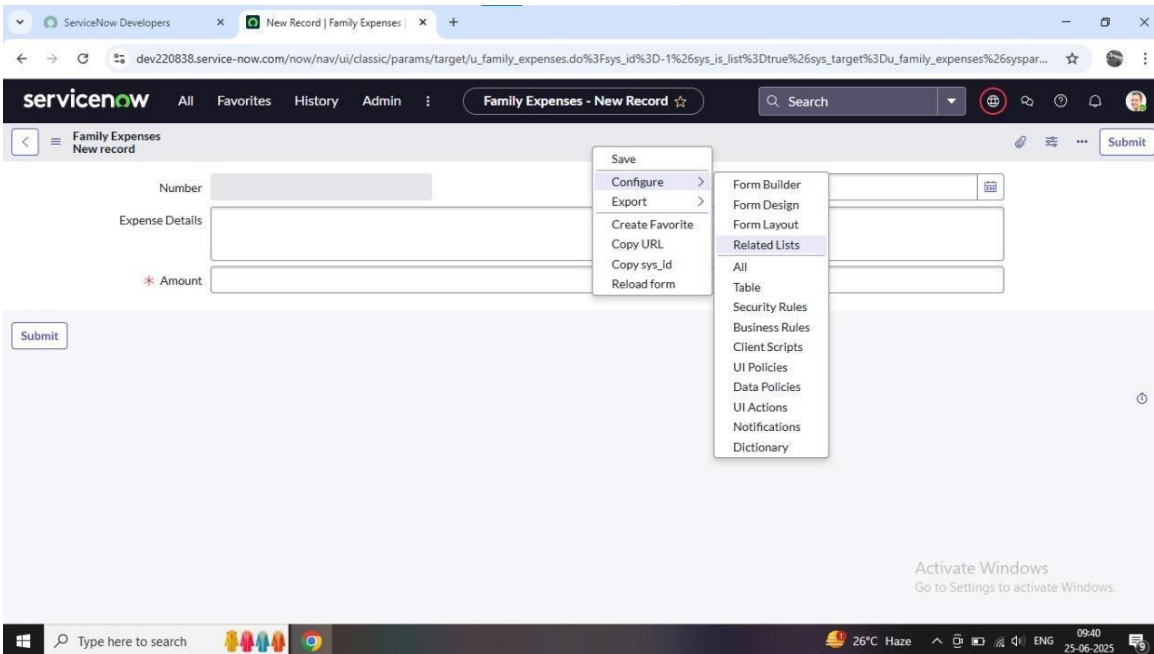
Step-5: Creation of Relationship Between Family Expenses and Daily Expenses Tables

Key Objective: Establish a logical link between each daily expense and the corresponding family member, allowing for better data organization, tracking, and reporting of expenses by individual or household.



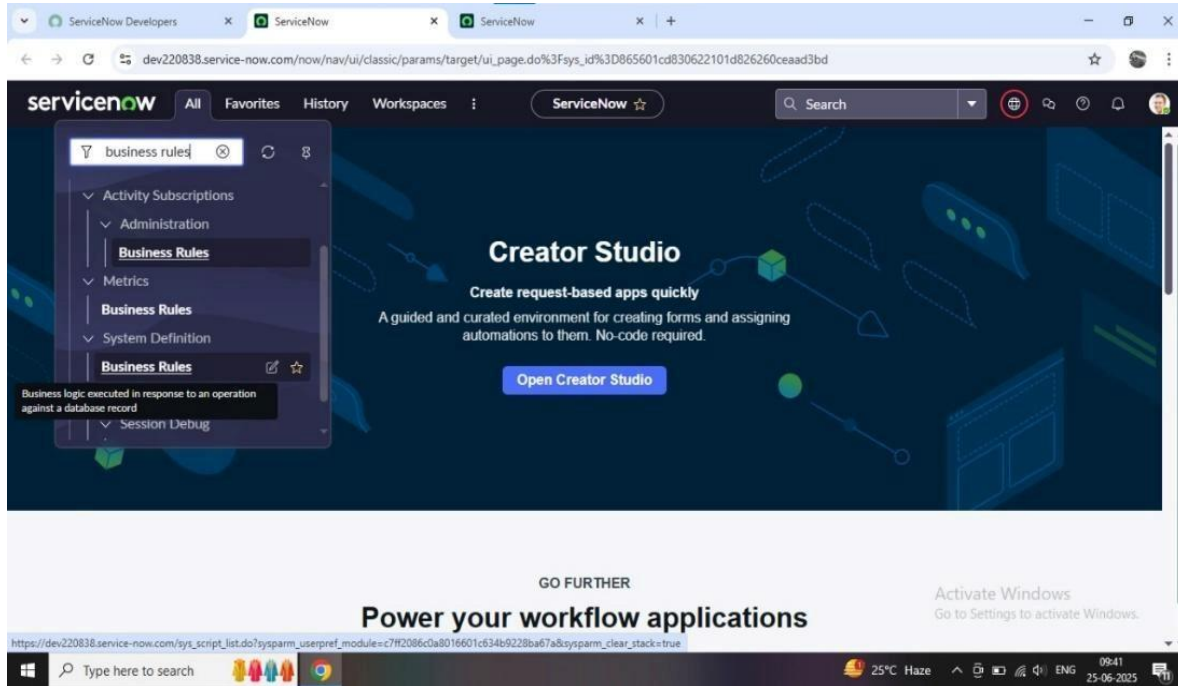
Step-6: Configuring Related List on Family Expenses:

Key Objective: Display all related Daily Expenses record for each family member directly within their family expenses form, enabling users to view and manage associated expense entries in one place.



Step-7: Creation of Business Rules:

Key Objective: Automate backend logic and enforce business processes by executing custom scripts when records are created, updated, deleted, or queried ensuring data consistency, validation, and process automation.



The image displays two screenshots of the ServiceNow Business Rule configuration interface.

Top Screenshot: 'When to run' tab

- Name:** Family Expenses BR
- Table:** Daily Expenses [u_daily_expenses]
- Application:** Global
- Active:** ☒
- Advanced:** ☒
- When to run:**
 - When:** before
 - Order:** 100
- Filter Conditions:** Add Filter Condition, Add OR Clause
- Role conditions:** [icon]
- Actions:**
 - Insert:** ☒
 - Update:** ☒
 - Delete:** ☐
 - Query:** ☐

Bottom Screenshot: 'Script' tab

- Condition:** [empty field]
- Script:** Turn on ECMAScript 2021 (ES12) mode


```

1 (function executeRule(current, previous /*null when async*/) {
2   var FamilyExpenses = new GlideRecord('u_family_expenses');
3   FamilyExpenses.addQuery('u_date', current.u_date);
4   FamilyExpenses.query();
5   if(FamilyExpenses.next())
6   {
7     FamilyExpenses.u_amount += current.u_expense;
8     FamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.
9       u_expense + "/-";
10    FamilyExpenses.update();
11  }
12  else
13  {
14    var NewFamilyExpenses = new GlideRecord('u_family_expenses');
15    NewFamilyExpenses.u_date = current.u_date;
16    NewFamilyExpenses.u_amount = current.u_expense;
17    NewFamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.
18      u_expense + "/-";
19    NewFamilyExpenses.insert();
20  }
21 }

```
- Context Menu:**
 - Save
 - Configure
 - Export
 - Create Favorite
 - Copy URL
 - Copy sys_id
 - Reload form

Step-8: Configure the Relationship:

Key Objective: Logically connect two related tables so data can be linked, referenced, and accessed easily supporting better data structure, consistency, and reporting.

ServiceNow Developers | Daily Expenses | Relationship | New Record | Business Rule | ServiceNow

dev220838.service-now.com/now/nav/ui/classic/params/target/sys_relationship.do%3Fsys_id%3D2920312083d662501d826260cead334%26sysparm_record_target%3Dsys_r...

servicenow All Favorites History Admin Relationship - Daily Expenses Search

Relationship Daily Expenses Update Delete

Name: Daily Expenses Application: Global

Advanced: ☐ Applies to table: Family Expenses [u_family_expens...] Queries from table: Daily Expenses [u_daily_expenses]

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     current.addQuery('u_date', parent.u_date);
5     current.query();
6
7 })(current, parent);
```

Update Delete

Related Links
[Run Point Scan](#)

Activate Windows
Go to Settings to activate Windows.

Type here to search Afternoon rain 09:49 25-06-2025

Step-9: Conclusion:

PROJECT DETAILS

Calculating Family Expenses
Using Service Now

- Setting Up ServiceNow Instance
- Setting Up ServiceNow Instance
- Creation Of New Update Set
- Creation Of Table
- Creation Of Table(Daily Expenses)
- Creation Of Relationship
- Configuring Related List On Family Expenses
- Creation Of Business Rules
- Configure The Relationship
- Conclusion

TASK & PROGRESS

MENTOR REVIEW

Conclusion

In conclusion, implementing the "Calculation of Family Expenses Using ServiceNow" project offers numerous benefits to the household. By leveraging the robust capabilities of ServiceNow, family members can efficiently submit, track, and manage their expenses in a centralized and streamlined manner.