

PERFORMANCE TESTING PHASE

DATE	07 NOVEMBER 2025
TEAM ID	NM2025TMID09183
PROJECT NAME	Calculating Family Expenses using ServiceNow

5.1 Testing Objective:

The objective of testing is to ensure that the Family Expense Tracker in ServiceNow works correctly, calculates totals accurately, and updates records in real-time without errors.

5.2 Types of Testing Performed:

Testing Type	Description
Unit Testing	Tested each table and field individually
Functional Testing	Verified all features work as expected
Integration Testing	Checked link between Daily Expense & Family Expense tables
Validation Testing	Ensured mandatory fields & correct data entry
Performance Testing	Verified system updated totals instantly
User Acceptance Testing (UAT)	Ensured the solution is user-friendly & meets objectives

5.3 Test Cases:

Test Case ID	Test Description	Input	Expected Output	Result
TC-01	Add daily expense record	Dad → 1000	Record saved	Passed
TC-02	Add another daily expense	Dad → 2000	Total = 3000	Passed
TC-03	Add expense for another member	Mom → 500	Record saved	Passed
TC-04	Check total for Mom	Mom → 500	Correct total	Passed
TC-05	Check auto-generated number	Save record	Auto-ID generated	Passed
TC-06	Invalid amount entry	Blank amount	Error shown	Passed
TC-07	Submit without date	Blank date	Error shown	Passed

5.4 Performance Metrics:

Parameter	Result
Record Save Time	Less than a second

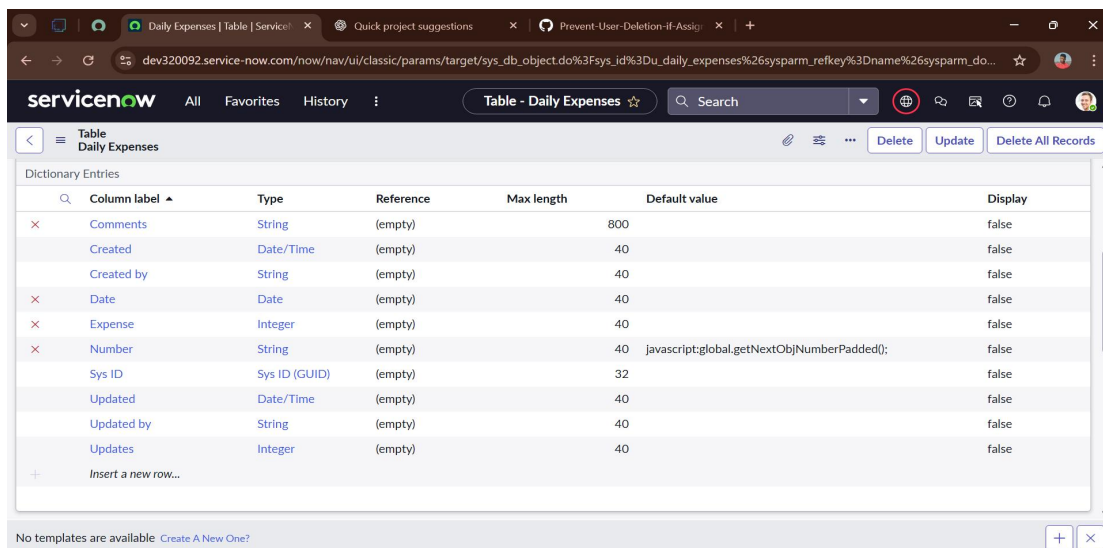
Total Calculation Time	Instant update
Page Load Speed	Smooth
System Load Handling	Stable for multiple entries

5.5 Observations & Findings:

- Daily expenses correctly update Family Total
- Business Rule triggers worked properly
- Auto-Numbering works
- Reference field links correctly
- No performance lag
- Data remains consistent after multiple entries

5.6 Pictures for references:

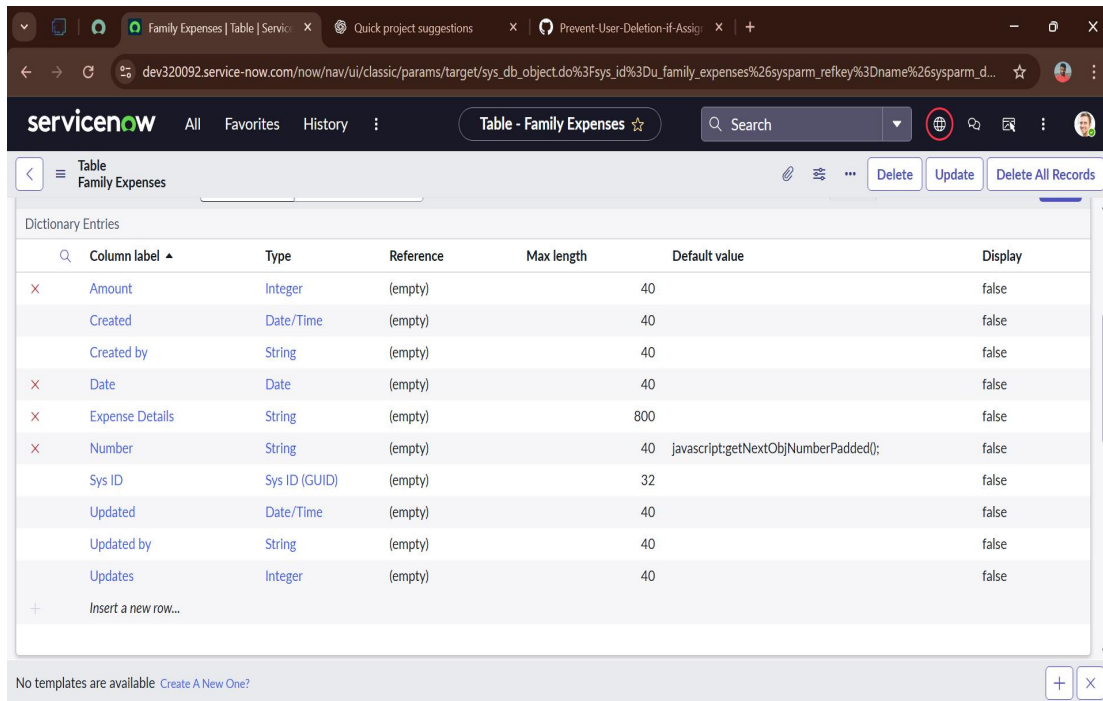
5.6.1 Daily Expense Table:



The screenshot shows the ServiceNow interface for the 'Table - Daily Expenses'. The 'Dictionary Entries' table is displayed with the following columns: Column label, Type, Reference, Max length, Default value, and Display. The table contains 11 rows of data, including fields like Comments, Created, Created by, Date, Expense, Number, Sys ID, Updated, Updated by, and Updates. The 'Number' field has a default value of 'javascript:global.getNextObjNumberPadded();'. The 'Sys ID' field has a default value of 'Sys ID (GUID)'. The 'Updates' field has a default value of '40'. The 'Display' column indicates whether each field is displayed in the form.

Column label	Type	Reference	Max length	Default value	Display
Comments	String	(empty)	800		false
Created	Date/Time	(empty)	40		false
Created by	String	(empty)	40		false
Date	Date	(empty)	40		false
Expense	Integer	(empty)	40		false
Number	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Sys ID	Sys ID (GUID)	(empty)	32		false
Updated	Date/Time	(empty)	40		false
Updated by	String	(empty)	40		false
Updates	Integer	(empty)	40		false

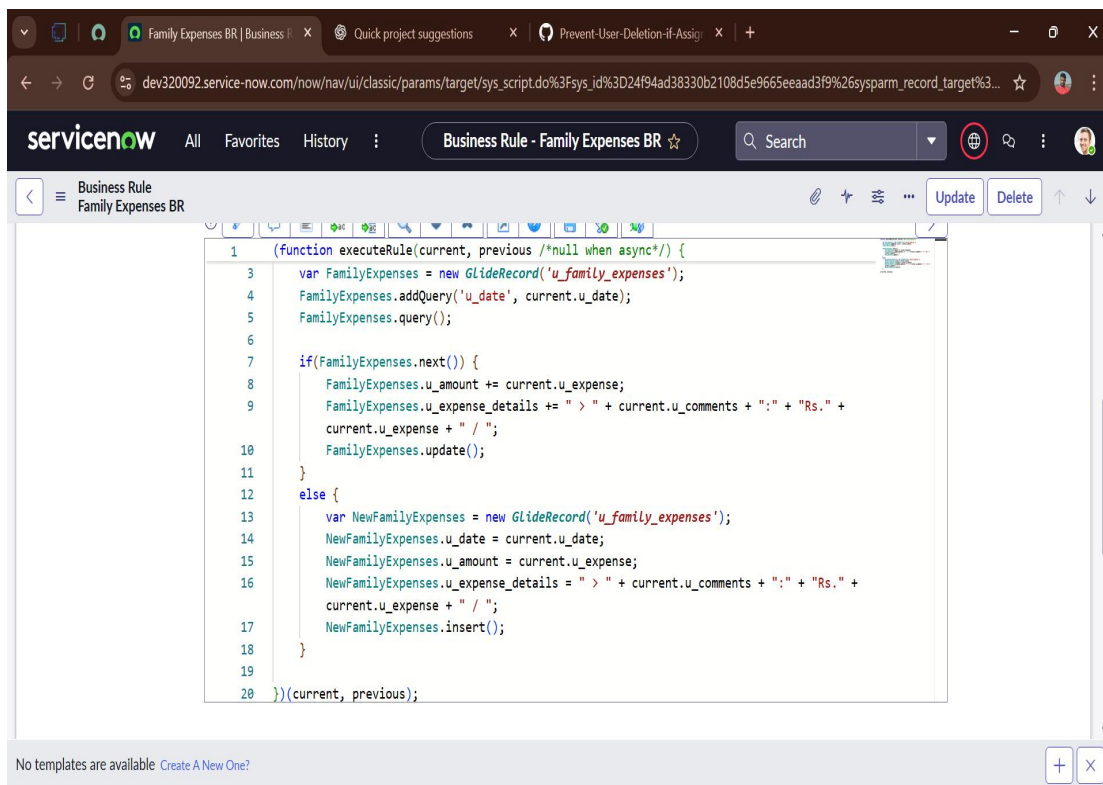
5.6.2 Family Expense Table:



The screenshot shows the ServiceNow Table Editor for the 'Family Expenses' table. The interface includes a top navigation bar with the ServiceNow logo, a search bar, and a table name dropdown. Below the navigation bar, there are buttons for 'Delete', 'Update', and 'Delete All Records'. The main content area displays a table of 'Dictionary Entries' with columns for 'Column label', 'Type', 'Reference', 'Max length', 'Default value', and 'Display'. The table lists various fields such as 'Amount', 'Created', 'Created by', 'Date', 'Expense Details', 'Number', 'Sys ID', 'Updated', 'Updated by', and 'Updates'. Each entry has a corresponding type (e.g., Integer, Date/Time, String, Sys ID (GUID)) and a reference (empty for most). The 'Default value' column shows 'javascript:getNextObjNumberPadded()' for the 'Number' field. At the bottom, there is a message 'No templates are available' and a 'Create A New One?' link.

Column label	Type	Reference	Max length	Default value	Display
Amount	Integer	(empty)	40		false
Created	Date/Time	(empty)	40		false
Created by	String	(empty)	40		false
Date	Date	(empty)	40		false
Expense Details	String	(empty)	800		false
Number	String	(empty)	40	javascript:getNextObjNumberPadded();	false
Sys ID	Sys ID (GUID)	(empty)	32		false
Updated	Date/Time	(empty)	40		false
Updated by	String	(empty)	40		false
Updates	Integer	(empty)	40		false

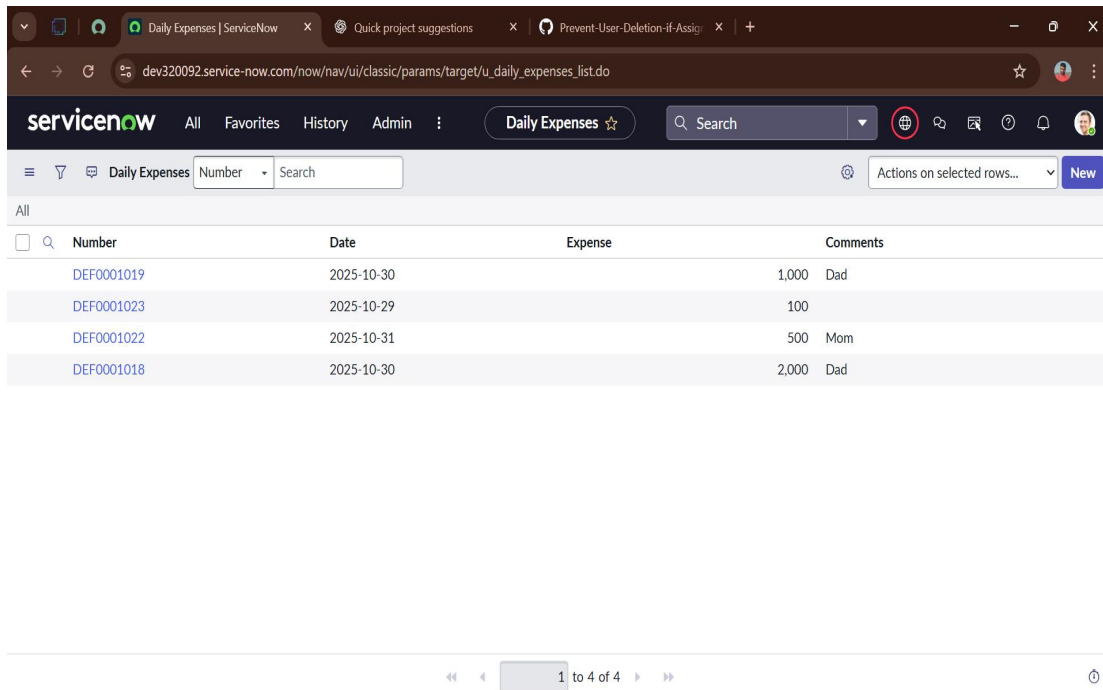
5.6.3 Business Rules Script:



The screenshot shows the ServiceNow Business Rule Editor for the 'Family Expenses BR' business rule. The interface includes a top navigation bar with the ServiceNow logo, a search bar, and a business rule name dropdown. Below the navigation bar, there are buttons for 'Update' and 'Delete'. The main content area displays a JavaScript script for the business rule. The script defines a function 'executeRule' that takes 'current' and 'previous' parameters. It uses the 'GlideRecord' class to query and update records in the 'Family Expenses' table. The script checks if the 'u_date' field is present and updates the 'u_amount' and 'u_expense_details' fields accordingly. At the bottom, there is a message 'No templates are available' and a 'Create A New One?' link.

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

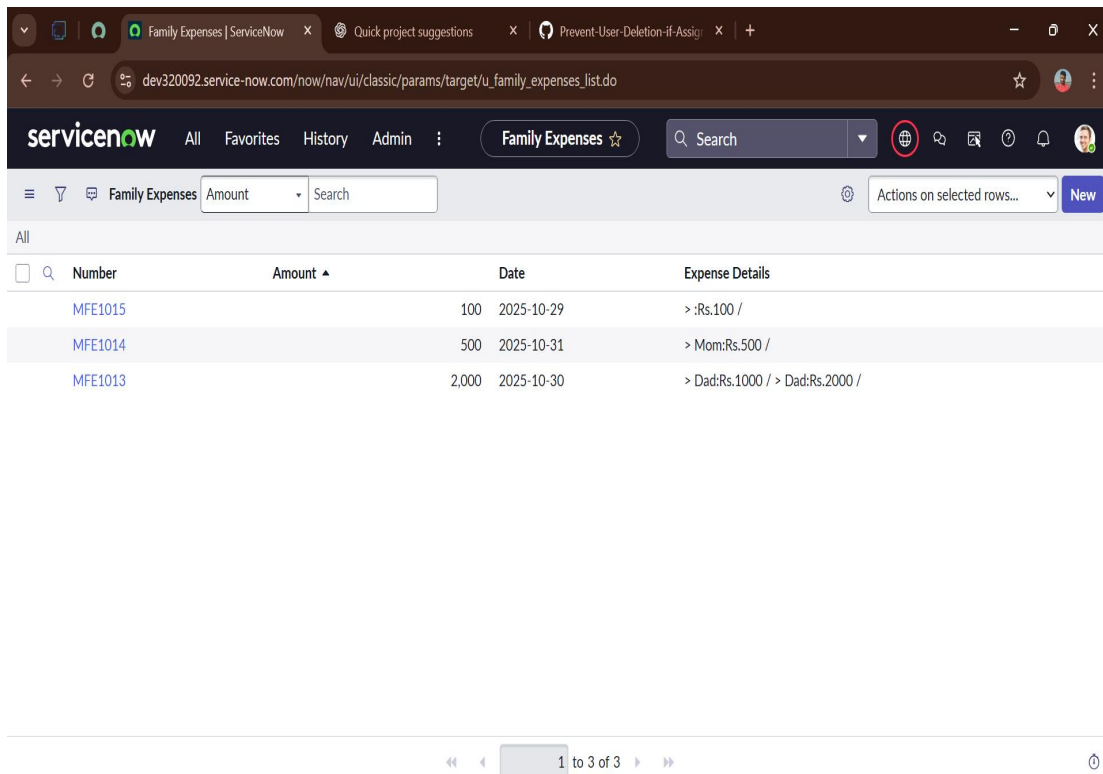
5.6.4 Input in Daily Expenses



The screenshot shows the ServiceNow interface for the 'Daily Expenses' list. The browser address bar displays 'dev320092.service-now.com/now/nav/ui/classic/params/target/u_daily_expenses_list.do'. The ServiceNow header includes the logo, navigation tabs (All, Favorites, History, Admin), a search bar, and a 'Daily Expenses' button. Below the header, there is a filter bar with 'Daily Expenses' and a 'Number' dropdown. The table below has columns: Number, Date, Expense, and Comments. It contains four rows of data. At the bottom, there is a pagination bar showing '1 to 4 of 4'.

Number	Date	Expense	Comments
DEF0001019	2025-10-30	1,000	Dad
DEF0001023	2025-10-29	100	
DEF0001022	2025-10-31	500	Mom
DEF0001018	2025-10-30	2,000	Dad

5.6.5 Output in Family Expenses:



The screenshot shows the ServiceNow interface for the 'Family Expenses' list. The browser address bar displays 'dev320092.service-now.com/now/nav/ui/classic/params/target/u_family_expenses_list.do'. The ServiceNow header includes the logo, navigation tabs (All, Favorites, History, Admin), a search bar, and a 'Family Expenses' button. Below the header, there is a filter bar with 'Family Expenses' and an 'Amount' dropdown. The table below has columns: Number, Amount, Date, and Expense Details. It contains three rows of data. At the bottom, there is a pagination bar showing '1 to 3 of 3'.

Number	Amount	Date	Expense Details
MFE1015	100	2025-10-29	> :Rs.100 /
MFE1014	500	2025-10-31	> Mom:Rs.500 /
MFE1013	2,000	2025-10-30	> Dad:Rs.1000 / > Dad:Rs.2000 /

CONCLUSION

The **Family Expense Tracker** successfully simplifies daily expense management using ServiceNow's low-code platform. It provides a secure, organized, and user-friendly solution for tracking household spending. With digital records and visual reports, users gain better financial awareness and control. The system efficiently replaces manual record-keeping with automation and accuracy. This project demonstrates the effectiveness of cloud-based tools for solving everyday financial challenges.