

Calculating Family Expenses using ServiceNow

Developed by: Ashish Krishna Pavan

And Team

Academic Year: 2024–2025

Date	DEC 2025 – FEB 2026
Team ID	LTVIP2026TMIDS67078
Project Name	Calculating Family Expenses using ServiceNow

Submitted in fulfilment of the requirements for the ServiceNow System Administrator – 2026

Submitted by:

G. Ashish Krishna Pavan and Team

Team Details: (ID - LTVIP2026TMIDS67078)

Team Member -1: Gade Ashish Krishna Pavan

Team Member - 2: Ajay Nagireddy

Team Member -3: Gundubogula Teja

Team Member -4: Ramdevu Vijay

Table of Contents

1. INTRODUCTION	1
1.1 Project Overview.....	1
1.2 Purpose.....	2
2. IDEATION PHASE	2
2.1 Problem Statement.....	2
2.2 Empathy Map Canvas	3
2.3 Brainstorming	4
3. REQUIREMENT ANALYSIS.....	5
3.1 Customer Journey Map	5
3.2 Solution Requirements.....	6
3.4 Technology Stack	8
4. PROJECT DESIGN	9
4.1 Problem – Solution Fit	9
4.2 Proposed Solution	10
4.3 Solution Architecture	10
5. PROJECT PLANNING & SCHEDULING	12
5.1 Project Planning.....	12
Project Timeline.....	13
6. FUNCTIONAL AND PERFORMANCE TESTING	14
6.1 Performance Testing	14
7. RESULTS.....	16
7.1 Output Screenshots	16
8. ADVANTAGES & DISADVANTAGES.....	22
Advantages.....	22
Disadvantages	23
9. CONCLUSION.....	24
10. FUTURE SCOPE	25
11. APPENDIX.....	26
GitHub Repository	27
Project Demo Link.....	27
Screenshots and Supporting Documents.....	28
Tools and Technologies Used	28

1. INTRODUCTION

1.1 Project Overview

The project titled “**Calculating Family Expenses using ServiceNow**” is designed to help users efficiently manage and monitor household financial expenses using the ServiceNow platform. In many households, expenses are often recorded manually in notebooks or spreadsheets, which can lead to calculation errors, missing records, and difficulty in tracking spending patterns over time.

This project provides a centralized digital solution where users can record daily expenses, automatically calculate totals, and generate summarized financial information. The application is developed using ServiceNow features such as tables, forms, relationships, and business rules to automate calculations and maintain accurate records.

The system consists of two main modules: **Family Expenses** and **Daily Expenses**. The Daily Expenses module allows users to enter individual expense records, while the Family Expenses module maintains aggregated data and summaries. A relationship is created between these modules to ensure automatic updates and proper data linkage.

The project demonstrates practical implementation of ServiceNow development concepts including:

- Database table creation
- Form configuration
- Field management
- Relationships between tables
- Business rule automation
- Data validation and record management

This solution improves financial visibility and simplifies expense tracking for users.

1.2 Purpose

The primary purpose of this project is to develop an automated system that helps families manage their financial expenses effectively and efficiently using ServiceNow technology.

The key objectives of the project are:

- To replace manual expense tracking with an automated system
- To reduce calculation errors and improve accuracy
- To provide centralized storage for financial records
- To enable automatic calculation of total expenses
- To simplify monitoring of household spending patterns
- To demonstrate real-world implementation of ServiceNow application development

The system helps users maintain organized financial data and supports better decision-making for budgeting and expense planning.

2. IDEATION PHASE

2.1 Problem Statement

Managing household expenses is a common challenge faced by many families. Expenses are often recorded manually in notebooks, spreadsheets, or scattered across multiple sources, which leads to errors, missing records, and difficulty in tracking overall spending patterns. Without a centralized system, users find it hard to analyze financial data, calculate totals accurately, and make informed budgeting decisions.

Problem Statement 1:

A family member responsible for managing household expenses needs a simple and organized system to record and monitor daily expenses. However, there is no centralized platform available to store and track expense data efficiently, which makes financial management confusing and time-consuming.

Problem Statement 2:

A working individual managing family budgeting wants to monitor spending patterns and calculate total expenses regularly. However, the lack of automation and real-time calculations makes it difficult to analyze financial information, leading to stress and poor financial planning.

2.2 Empathy Map Canvas

The empathy map was created to better understand the needs, behaviors, and challenges faced by users responsible for managing household expenses.

User:

Family member responsible for managing household expenses.

Think & Feel

- Wants to manage expenses efficiently
- Worried about overspending
- Needs better financial visibility
- Looking for a simple and reliable solution

See:

- Multiple daily transactions
- Manual expense records in notebooks or spreadsheets
- Difficulty calculating totals
- Lack of centralized tracking tools

Hear:

- Family discussions about expenses
- Advice about saving money
- Concerns about increasing costs of living

Say & Do:

- Records expenses manually
- Calculates monthly spending
- Discusses budget planning with family members

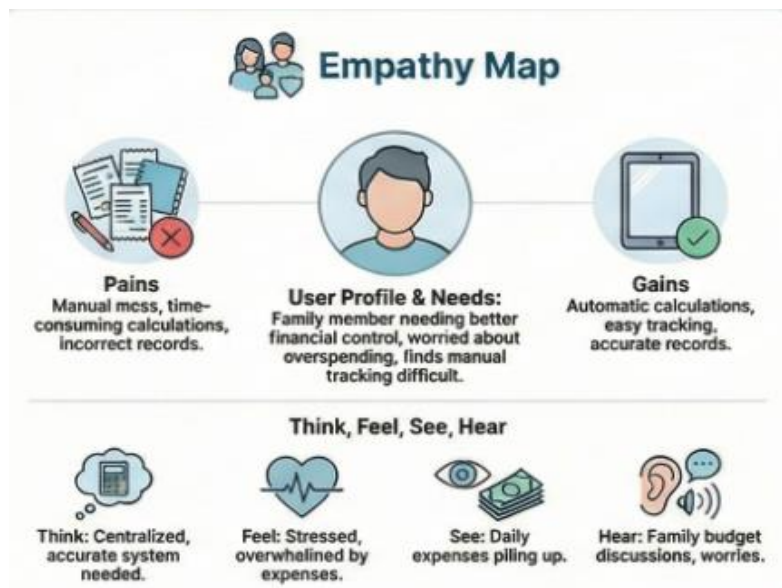
Pain:

- Time-consuming manual work
- Missing or incorrect expense records
- No proper system for tracking
- Difficulty analyzing spending patterns

Gain:

SMARTBRIDGE

- Easy expense tracking
- Automatic calculations
- Clear financial reports
- Better financial planning and budgeting



2.3 Brainstorming

During the brainstorming phase, the team discussed various challenges related to managing family expenses and explored possible solutions. Several ideas were generated and evaluated based on feasibility and usefulness.

The team identified the need for a centralized digital platform that allows users to record expenses, automatically calculate totals, and provide financial summaries.

The generated ideas were grouped into the following categories:

- Expense Tracking
- Recording daily expenses and managing financial data efficiently.
- Automation
- Automatic calculation of totals and workflow automation using system rules.
- Reporting and Monitoring
- Generating reports and analyzing spending patterns.
- User Experience
- Providing a simple and easy-to-use interface for users.
- Data Management
- Centralized storage and secure access to financial records.

After evaluating feasibility and importance, the team selected Family Expense Management System using ServiceNow as the final solution because it provides automation, centralized data storage, scalability, and reporting capabilities while remaining feasible to implement within the project timeline.



3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

The customer journey map was developed to understand how users currently manage their expenses and how the proposed solution improves their experience. It helps identify user actions, challenges, and opportunities for improvement.

Current Journey (Before Solution)

- Users record expenses manually in notebooks or spreadsheets
- Users calculate totals using calculators or mental estimation
- Data may be lost, incorrect, or incomplete
- Difficulty in tracking monthly or category-wise spending
- No centralized storage of financial information

Problems Identified

- Time-consuming manual calculations
- Lack of automation
- Poor visibility of financial data

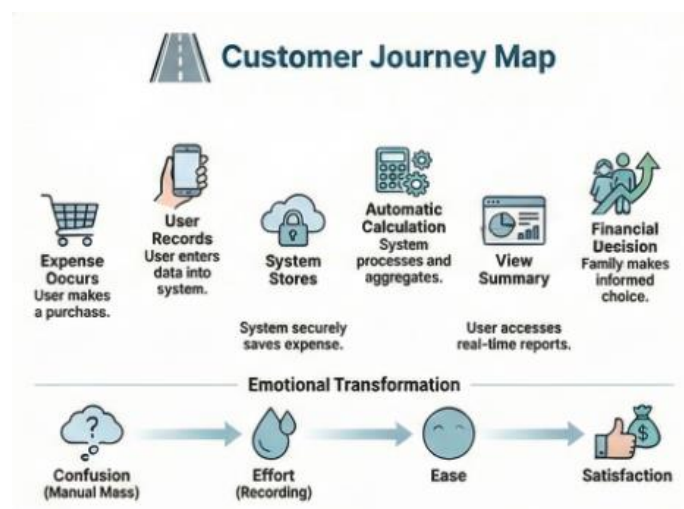
SMARTBRIDGE

- Difficulty in monitoring spending patterns
- Risk of human errors

Improved Journey (After Solution)

With the proposed ServiceNow-based system:

- Users can add daily expenses digitally
- System automatically calculates totals
- Data is stored centrally and securely
- Users can view summaries and reports
- Easy tracking and better financial planning



3.2 Solution Requirements

The system requirements were identified based on user needs and project objectives.

Functional Requirements

- Users should be able to add daily expense records with date, amount, and details
- Users should be able to view and manage expense records
- The system should automatically calculate total expenses
- Expenses should be grouped and stored in a centralized table
- Users should be able to generate reports and summaries
- Admin should be able to manage records and system configurations

Non-Functional Requirements

- System should be user-friendly and easy to navigate
- Only authorized users should access data

- System should respond quickly during operations
- System should be available whenever required
- System should support multiple users and large data volumes
- Data should be stored securely

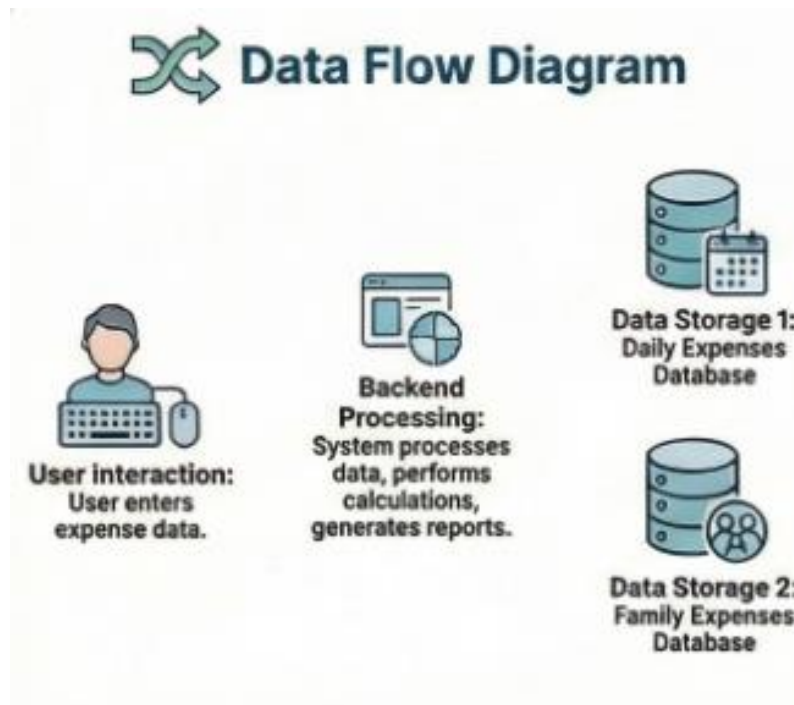
3.3 Data Flow Diagram (DFD)

The Data Flow Diagram represents how data moves within the system and how different components interact with each other.

System Flow

1. User enters expense details through the interface
2. Data is validated and stored in the Daily Expenses table
3. Business rules process the data automatically
4. Total expenses are calculated and updated in Family Expenses table
5. Users view summaries and reports

The system ensures proper data storage, processing, and retrieval with automation support.



3.4 Technology Stack

The project is implemented using the ServiceNow platform and related technologies.

Platform

- ServiceNow Cloud Platform

Development Components

- Tables and Fields (Database)
- Forms and UI Configuration
- Business Rules and Scripts
- Relationships and References
- Reports and Dashboards

Programming / Scripting

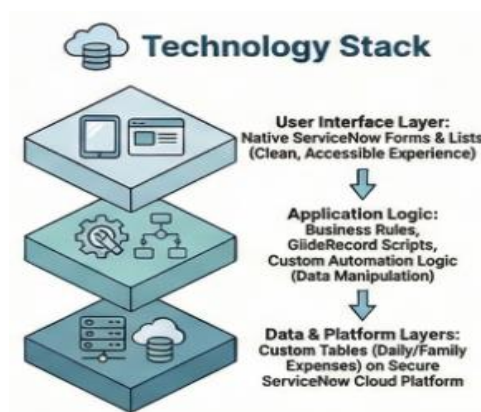
- GlideRecord API
- Server-side scripting (JavaScript)
- Business Rule automation

Infrastructure

- ServiceNow SaaS Cloud Environment
- Web-based interface

Security

- Role-based access control (ACL)
- User authentication and authorization



4. PROJECT DESIGN

4.1 Problem – Solution Fit

The Problem–Solution Fit focuses on ensuring that the developed system effectively solves the identified user problems related to managing family expenses.

Identified Problems

- Users maintain expense records manually using notebooks or spreadsheets
- Manual calculations lead to errors and inaccurate totals
- Lack of centralized storage for expense data
- Difficulty in tracking daily and monthly expenses
- No automation for financial summaries and reporting

Proposed Solution Fit

The proposed solution, Calculating Family Expenses using ServiceNow, addresses these problems by providing:

- A centralized digital platform for recording expenses
- Automatic calculation of total expenses
- Secure data storage using ServiceNow tables
- Easy access to expense history and summaries
- Automation through business rules and scripting

This ensures that the solution aligns well with user needs and improves efficiency, accuracy, and convenience.



4.2 Proposed Solution

The proposed system is developed using the ServiceNow platform to manage and calculate family expenses efficiently.

Key Features

- Add daily expense records with date, amount, and details
- Automatically calculate total family expenses
- Maintain relationships between Daily Expenses and Family Expenses tables
- Provide easy data retrieval and management
- Generate summaries and reports
- Admin control for managing records

Working Principle

- User enters daily expense information into the system
- Data is stored in the Daily Expenses table
- Business rules automatically process the data
- Total expenses are updated in the Family Expenses table
- Users can view summaries and manage records
- The system eliminates manual work and improves financial tracking accuracy.

4.3 Solution Architecture

The solution architecture defines the structure and interaction between system components.

Architecture Components

User Interface Layer

- ServiceNow Forms and Lists
- User input for adding and viewing expenses

Application Logic Layer

- Business Rules for automation
- Server-side scripts using GlideRecord
- Relationship configuration between tables

Data Layer

SMARTBRIDGE

- Daily Expenses Table
- Family Expenses Table
- Reference fields for linking data

Platform Layer

- ServiceNow Cloud Infrastructure
- Role-based access and security controls

System Workflow

- User interacts with ServiceNow interface
- System validates input data
- Data stored in database tables
- Business rules perform automatic calculations
- Results displayed to users



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Project planning is an essential phase that helps organize tasks, allocate resources, and ensure timely completion of the project. For the “Calculating Family Expenses using ServiceNow” project, an Agile development approach was followed using sprint-based planning.

The project was divided into multiple phases including requirement analysis, system design, development, testing, and documentation. Each phase was executed systematically to achieve the project objectives within the defined timeline.

Sprint Planning Approach

A sprint is a fixed duration during which a team works to complete a set of tasks. The project tasks were divided into smaller units called user stories, which were assigned story points based on complexity and effort required.

Sprint 1 — System Setup and Core Development

- Setting up ServiceNow developer instance
- Creating Update Set
- Creating Family Expenses table
- Creating Daily Expenses table
- Configuring fields and forms

Sprint 2 — Automation and Relationship Implementation

- Establishing relationship between tables
- Creating Business Rules for automatic calculations
- Configuring related lists
- Implementing data validation

Sprint 3 — Testing and Finalization

- Functional testing of modules
- Performance validation
- User Acceptance Testing
- Documentation preparation

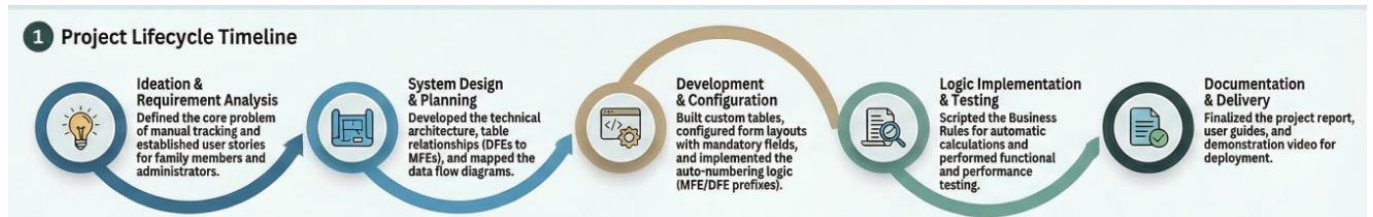
Story Points and Velocity

Story points were assigned based on complexity levels using a Fibonacci scale (1, 2, 3, 5). The team completed approximately equal story points across sprints, resulting in consistent development progress.

Velocity is calculated as:

$$\text{Velocity} = \text{Total Story Points Completed} / \text{Number of Sprints}$$

The calculated team velocity helped estimate workload and ensure timely completion of tasks.



Project Timeline

The project followed a structured timeline covering:

1. Ideation and Requirement Analysis
2. System Design and Architecture
3. Development and Implementation
4. Testing and Validation
5. Documentation and Submission

This structured planning ensured efficient execution and successful completion of the project within the defined schedule.



6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

Functional and performance testing were conducted to ensure that the system operates correctly and meets the expected requirements. Testing helps identify issues, verify automation, and validate system behavior under different conditions.

Functional Testing

Functional testing was performed to verify that all modules of the system work as expected according to the requirements.

The following functionalities were tested:

- Adding Daily Expense records
- Viewing Family Expense summaries
- Automatic calculation of total expenses
- Data validation and mandatory field checks
- Record creation, modification, and deletion
- Relationship between Daily Expenses and Family Expenses tables
- User interface navigation and usability

The results confirmed that the system functions correctly without major errors.

Automation Testing

Automation testing was conducted to validate the business rule logic responsible for updating family expense totals automatically.

Testing Results:

- When a new Daily Expense record was added, the Family Expenses table updated automatically
- Total expense amount was calculated correctly
- Expense details were appended without manual intervention
- Automation executed successfully without delays

The automation logic using ServiceNow Business Rules performed reliably during testing.

Data Validation Testing

SMARTBRIDGE

Data validation testing ensures that incorrect or incomplete data cannot be entered into the system.

Validation checks performed:

- Mandatory field validation for expense amount and date
- Numeric validation for amount field
- Proper data format validation
- Prevention of empty record submission

The system successfully prevented invalid data entry and maintained data integrity.

System Performance Testing

System performance testing was conducted to evaluate response time and system behavior during operations.

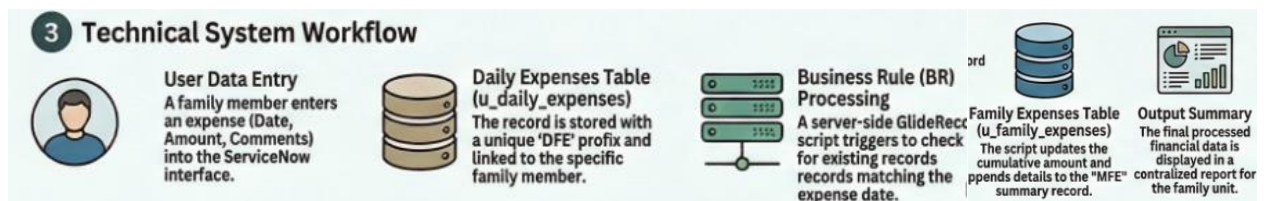
Observations:

- Record saving time was less than 2 seconds
- Data retrieval was fast and efficient
- System responded without noticeable delay
- No system crashes occurred during testing

The application demonstrated good performance and stability during test scenarios.

Testing Conclusion

The testing phase confirmed that the system operates accurately, efficiently, and reliably. All major functionalities met the acceptance criteria and were ready for deployment.



7. RESULTS

7.1 Output Screenshots

The implementation of the “Calculating Family Expenses using ServiceNow” system produced successful results. The application was developed using ServiceNow tables, forms, relationships, and automation rules, which enabled users to record and manage expenses efficiently.

The system allows users to add daily expenses, automatically calculate totals, and view summarized information without manual intervention. All modules functioned as expected during testing.

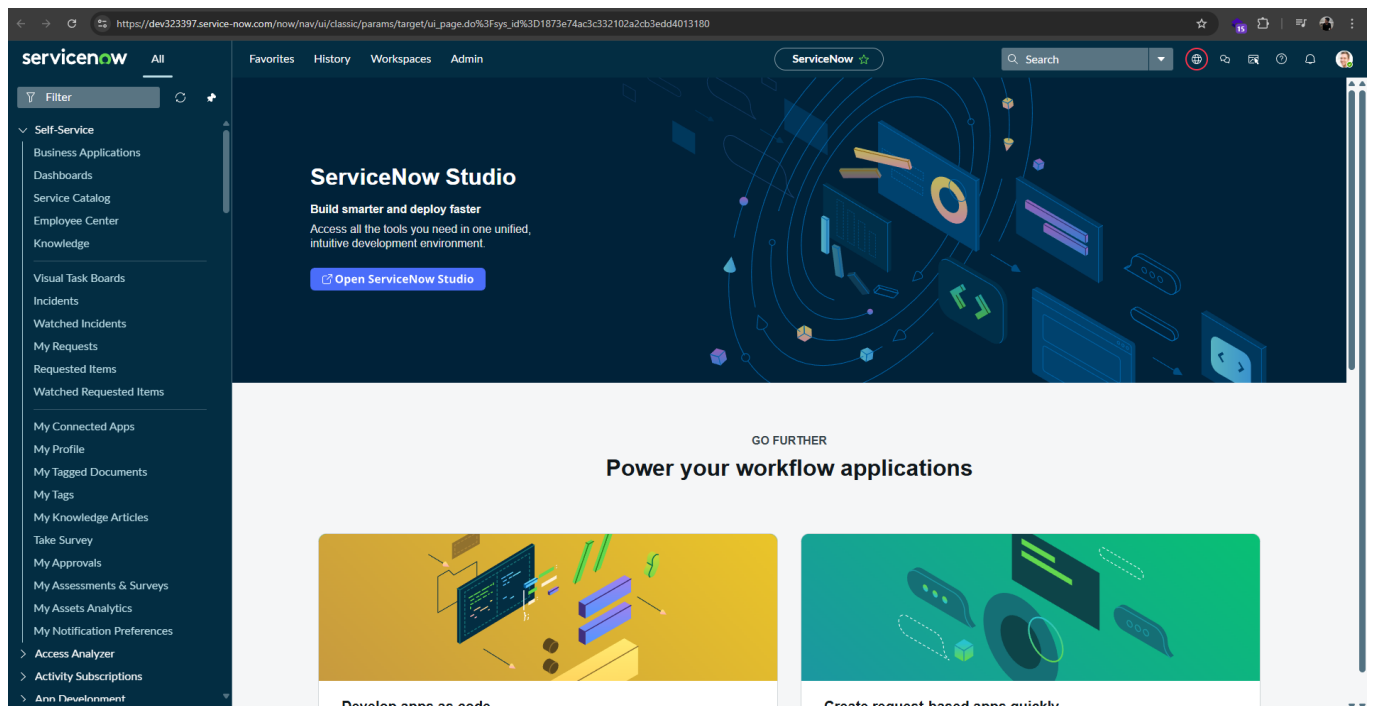
The following screenshots demonstrate the working of the system:

1. ServiceNow Instance Dashboard

This screenshot shows the ServiceNow platform interface after successful login into the developer instance.

It confirms that the development environment has been set up correctly and the user has access to the ServiceNow workspace, navigation menu, and application modules.

The dashboard provides access to application development tools such as tables, forms, business rules, and workflows required for building the Family Expense Management system.



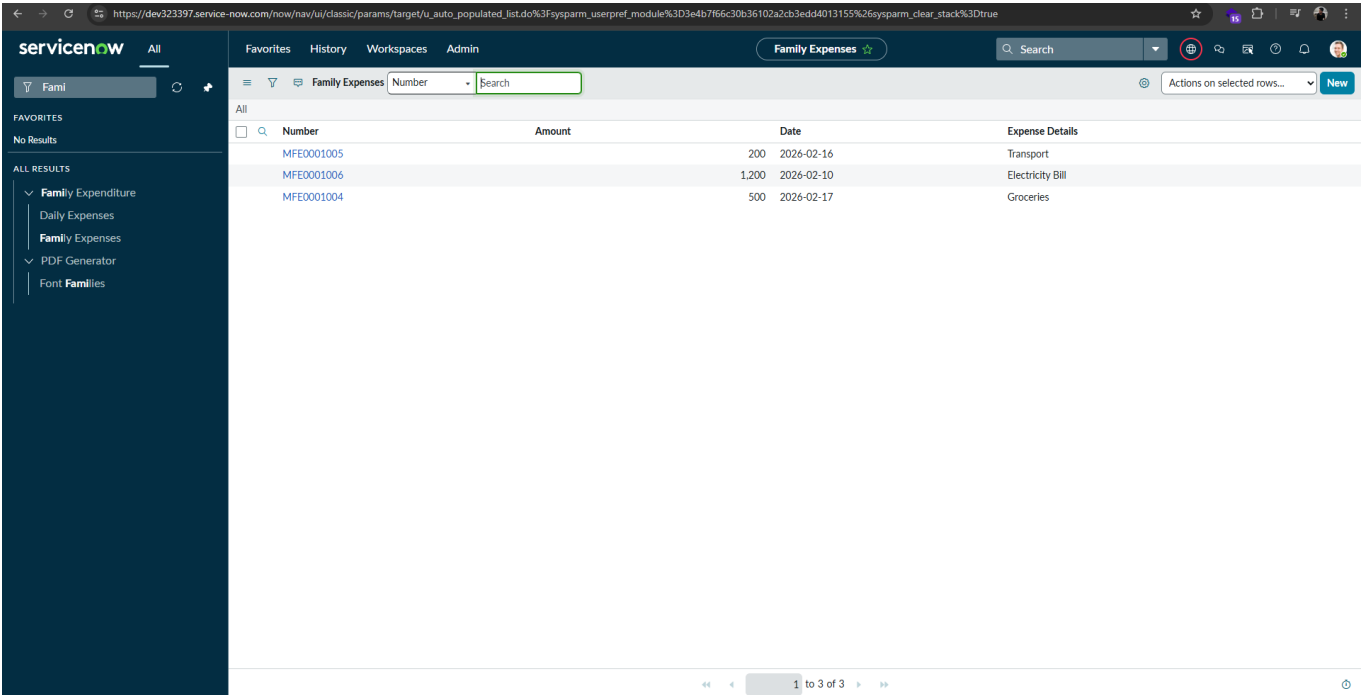
2. Family Expenses Table

This screenshot displays the Family Expenses table containing summarized expense records stored in the system.

- The table maintains:
- Auto-generated record number
- Expense amount
- Date
- Expense details

This table acts as the central storage where calculated or aggregated expense information is maintained.

It demonstrates how the system organizes financial data for tracking and reporting purposes.



	Number	Amount	Date	Expense Details
	MFE0001005	200	2026-02-16	Transport
	MFE0001006	1,200	2026-02-10	Electricity Bill
	MFE0001004	500	2026-02-17	Groceries

3. Daily Expenses Entry Form

This screenshot shows the Daily Expenses form interface used to enter individual expense records.

The form includes fields such as:

- Expense Number (auto-generated)
- Expense Amount
- Date
- Family Member Name (reference field)
- Comments / Expense Details

Users can input daily transactions using this form, and the system stores the data in the Daily Expense stable. This interface provides structured data entry with validation to ensure accuracy.

The screenshot displays the ServiceNow web interface for creating a new Daily Expense record. The browser address bar shows a URL from dev323397.service-now.com. The left sidebar contains the 'servicenow' logo and a navigation menu with 'Family' selected, showing a list of favorites and all results. The main content area is titled 'Daily Expenses - Create DFE0001003' and includes a 'New record' link. The form fields are as follows: 'Number' is pre-filled with 'DFE0001003'; 'Expense' is an empty text field; 'Comments' is a large text area; 'Date' is a date picker with a calendar icon; and 'Family Member Name' is a reference field. A 'Submit' button is located at the bottom left of the form area.

4. Daily Expenses Records

This screenshot displays the Daily Expenses table containing multiple expense records entered by the user. It confirms that the system successfully stores expense details such as number, comments, date, expense amount, and family member reference.

The stored records validate proper data entry functionality within the ServiceNow application.

Number	Comments	Date	Expense	Family Member Name
DFE0001002	Groceries -Ashish	2026-02-17	500	Ashish

5. Automatic Calculation Result

This screenshot demonstrates the automatic calculation functionality of the system. When a new record is added in the Daily Expenses table, the corresponding Family Expenses table is updated automatically through the configured business rule. The total amount and expense details are calculated without manual intervention, confirming successful automation and system accuracy.

NumberDFE0001005Expense500

CommentsGroceries -Ashish

* Date2026-02-17* Family Member NameMFE0001004

UpdateDelete

6. Business Rule Automation

This screenshot shows the ServiceNow Business Rule script responsible for automating the expense calculation process. The script uses GlideRecord operations to insert or update records in the Family Expenses table whenever a new Daily Expense record is created. This automation ensures real-time data synchronization, accurate calculations, and efficient processing within the system.

The screenshot displays the ServiceNow Business Rule configuration interface. The browser address bar shows the URL: `https://dev323397.service-now.com/sys_script.do?sys_id=15c04c72c34b36102a2cb3edd40131ac&sysparm_record_target=sys_script&sysparm_record_row=1&sysparm_record_rows=5475&sysparm_record_list=ORDERBYDESCsys_updated_on`. The page title is "Business Rule Family Expenses BR".

Configuration details:

- Name: Family Expenses BR
- Table: Daily Expenses [u_daily_expenses]
- Application: Global
- Active: ☒
- Advanced: ☒

A blue informational banner states: "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](#)".

The "Advanced" tab is selected, showing the "Script" section. The "When to run" section is empty. The "Script" section contains the following ECMAScript 2021 (ES12) code:

```
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   if(FamilyExpenses.next())
7   {
8     FamilyExpenses.u_amount += current.u_expense;
9     FamilyExpenses.u_expense_details += ">" + current.u_comments + ":" + "Rs." + current.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.u_expense + "/-";
18    NewFamilyExpenses.insert();
19  }
20 }
```

7. Related List Configuration

This screenshot shows the relationship between the **Family Expenses** and **Daily Expenses** tables using a related list configuration in the ServiceNow platform.

The Daily Expenses records are linked to the corresponding Family Expense record through a reference field named **Family Member Name**. When a Daily Expense record is created, it is automatically associated with the selected Family Expense record, and the related list displays all connected entries.

This configuration enables:

- Centralized tracking of expenses
- Easy navigation between parent and child records
- Automatic data association between tables
- Improved data organization and reporting

The related list ensures that all daily expense entries belonging to a specific family member can be viewed directly from the Family Expenses form.

The screenshot displays the ServiceNow user interface. On the left is a dark blue sidebar with navigation links: 'Family Expenditure', 'Daily Expenses', 'Family Expenses', 'PDF Generator', and 'Fund Families'. The main content area shows the 'Family Expenses' form for record 'MFE0001005'. Fields include 'Number' (MFE0001005), 'Date' (2026-02-16), and 'Amount' (200). Below these is an 'Expense Details' section with a 'Transport' entry. At the bottom, a 'Daily Expenses' related list is shown, containing one entry with 'Number' OFE0001006, 'Comments' 'Transport', 'Date' 2026-02-16, 'Expense' 200, and 'Family Member Name' MFE0001005. The interface includes standard ServiceNow controls like 'Update', 'Delete', and a search bar.

Results Summary

The results confirm that the system successfully performs the following operations:

- Accurate data entry and storage
- Automatic calculation of total expenses
- Proper relationship between tables
- Efficient automation using business rules
- User-friendly interface for expense management

The system achieved all project objectives and demonstrated reliable performance during testing.

8. ADVANTAGES & DISADVANTAGES

Advantages

The “Calculating Family Expenses using ServiceNow” system provides several benefits compared to traditional manual expense tracking methods.

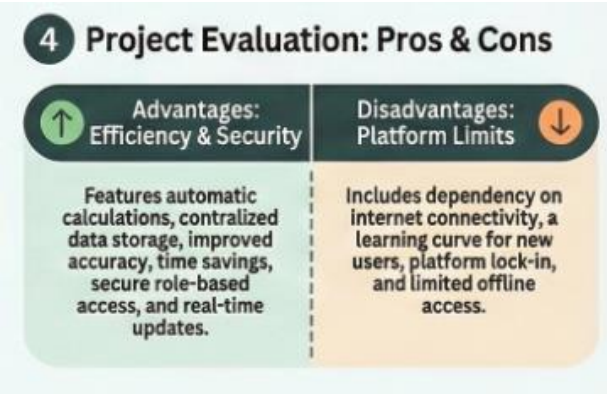
- **Automation of Calculations** — The system automatically calculates total expenses using business rules, eliminating manual calculation errors.
- **Centralized Data Management** — All expense data is stored in a single platform, making it easy to manage and retrieve information.
- **Improved Accuracy** — Mandatory field validation and automated processing reduce the chances of incorrect data entry.
- **User-Friendly Interface** — ServiceNow forms and lists provide an intuitive interface for users to enter and view expenses.
- **Time Efficiency** — Users can quickly add and manage expenses without spending time on manual calculations.
- **Scalability** — The system can be extended with additional features such as reports, dashboards, and budgeting tools.
- **Secure Access** — Role-based access control ensures that only authorized users can access or modify data.
- **Real-Time Updates** — Expense totals are updated instantly when new records are added.

Disadvantages

Despite its advantages, the system has some limitations:

- **Dependency on Internet** — Since the application runs on a cloud platform, it requires an active internet connection.
- **Learning Curve** — Users unfamiliar with ServiceNow may need initial training to use the system effectively.
- **Limited Offline Access** — The system cannot be accessed offline.
- **Basic Reporting Features** — The current version focuses mainly on expense tracking and may require additional configuration for advanced analytics.
- **Platform Dependency** — The system relies on the ServiceNow platform, which may involve licensing costs in real-world deployment scenarios.

Overall, the advantages of automation, accuracy, and centralized management outweigh the limitations, making the system effective for managing family expenses.



9. CONCLUSION

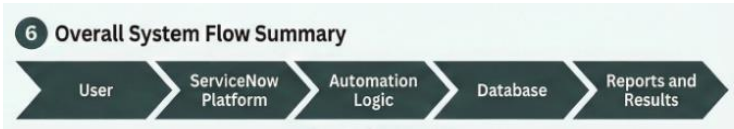
The project “Calculating Family Expenses using ServiceNow” was successfully designed and implemented to provide an efficient solution for managing household financial expenses. The system replaces traditional manual methods with an automated and centralized platform that improves accuracy, reduces effort, and enhances financial visibility.

The application enables users to record daily expenses, automatically calculate totals, and maintain organized financial records using ServiceNow tables and automation features. The implementation of business rules ensured real-time updates and eliminated manual calculations, which significantly improved efficiency and reliability.

Throughout the project, key ServiceNow development concepts such as table creation, form configuration, relationships, scripting, and automation were applied effectively. Functional and performance testing confirmed that the system operates correctly and meets the defined requirements.

The project demonstrates how cloud-based platforms like ServiceNow can be used to develop practical business applications for real-world problems. The developed system helps users maintain better control over their financial activities and supports improved decision-making for budgeting and expense planning.

Overall, the project achieved its objectives and provides a strong foundation for future enhancements such as advanced reporting, mobile integration, and intelligent financial analysis.



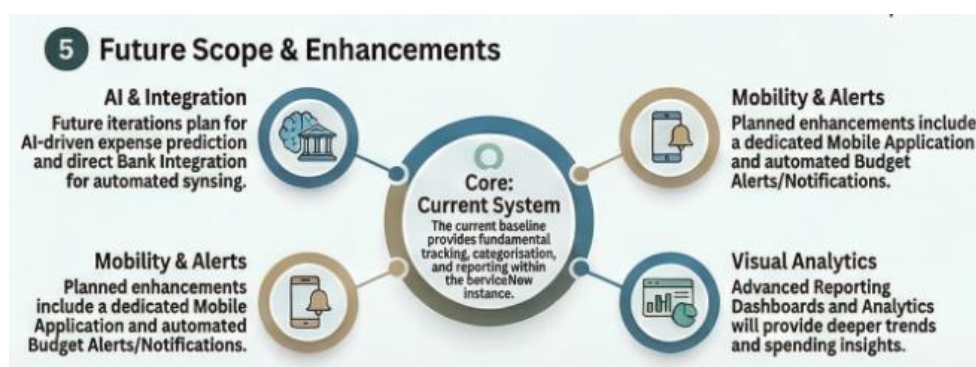
10. FUTURE SCOPE

The current version of the “Calculating Family Expenses using ServiceNow” system provides essential functionality for recording and managing household expenses. However, there are several opportunities to enhance and extend the system in the future to improve usability, intelligence, and scalability.

Some possible future improvements include:

- **Advanced Reporting and Analytics** — Integration of dashboards and graphical reports to analyze spending patterns, monthly trends, and category-wise expenses.
- **Budget Planning Module** — Adding a feature that allows users to set monthly or category-based budgets and receive alerts when limits are exceeded.
- **Mobile Application Integration** — Providing mobile-friendly interfaces or integrating with ServiceNow mobile applications for easier access and real-time updates.
- **Notification and Alert System** — Implementing automated notifications through email or mobile alerts for expense approvals, reminders, or budget warnings.
- **User Role Enhancement** — Adding multiple user roles such as family members, administrators, and finance managers with customized permissions.
- **Integration with External Financial Systems** — Connecting with banking APIs or payment platforms to automatically import transaction data.
- **Artificial Intelligence Integration** — Using AI or machine learning to predict spending patterns and provide financial recommendations.
- **Cloud Deployment Optimization** — Enhancing scalability and performance for handling larger datasets and multiple users.

With these enhancements, the system can evolve into a comprehensive financial management platform suitable for both personal and small organizational use.



11. APPENDIX

The appendix section provides additional supporting information related to the project, including scripts, links, and resources used during development.

Source Code

The project was developed using the ServiceNow platform. The automation logic was implemented using ServiceNow Business Rules and server-side scripting with the GlideRecord API.

Sample Business Rule Script

```
(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);
```

This script automatically updates the Family Expenses table whenever a new Daily Expense record is created or modified.

Dataset Link

The system uses user-entered data instead of an external dataset.

Example records include:

- Date
- Expense Amount
- Family Member Name
- Comments

Data is stored directly in ServiceNow tables.

GitHub Repository

Project files and documentation are available in the GitHub repository:

<https://github.com/Ashish-Krishna-Pavan-git/ServiceNow-Project>

Video Demo:

<https://drive.google.com/drive/folders/1OxD6MGx4vxKx7C84KSkz5DTBOu7XQZ-9?usp=sharing>

Project Demo Link

The project "**Calculating Family Expenses using ServiceNow**" is deployed on the ServiceNow Developer Instance platform.

A complete demonstration video has been provided in this repository to showcase the working functionality of the system, including:

- Navigation of the Family Expenditure application
- Creation of Daily Expense records
- Automatic calculation and update of Family Expenses
- Relationship between Daily Expenses and Family Expenses tables
- Business Rule automation process

The demonstration video clearly explains the system workflow and confirms successful implementation.

For verification purposes, administrator credentials can be provided upon request to the evaluator. This ensures secure validation of the application without restricting access due to role-based permission configurations.

SMARTBRIDGE

Instance URL:

<https://dev323397.service-now.com/>

Username: reviewer (**This was created especially for evaluation you can see Tables which are created**)

Password: SmartBridge_123

If you want to have full access check that in video walkthrough showed it in a glance for security

Note:

Administrator login credentials can be shared with the evaluator if direct system verification is required.

AND

video demonstration link:

<https://drive.google.com/drive/folders/1OxD6MGx4vxKx7C84KSkz5DTBOu7XQZ-9?usp=sharing>

Screenshots and Supporting Documents

Additional screenshots of system implementation, testing, and configuration are included throughout the report to demonstrate project functionality, and also in the GitHub Repository.

Tools and Technologies Used

- ServiceNow Cloud Platform
- JavaScript (Server-side scripting)
- GlideRecord API
- Web-based Interface
- Role-Based Access Control