

Performance and Testing

| | |
|---------------|--|
| Date | 02 NOV 2025 |
| Team ID | NM2025TMID08125 |
| Project Name | Calculating Family Expenses Using ServiceNow |
| Maximum Marks | 4 Marks |

Model Performance Testing

1. Instance Setup Validation

The screenshot shows the ServiceNow developer instance setup page. At the top, there's a navigation bar with links like MyNow, Products, Industries, Learning, Support, Partners, and Company. Below that is a secondary navigation bar with Developer, Home, Learn, Reference, Guides, and Connect. On the left, a sidebar displays 'Your PDI: dev189381' with a status of 'Online'. It also lists 'App Engine Studio' and 'Creator Studio' under the 'Status' section. On the right, there's a 'Your learnings' section with a book icon, a brief description, and two buttons: 'Explore Learning' and 'Browse'.

| Parameter | Value |
|------------------|---|
| Model Summary | ServiceNow developer instance created and accessible |
| Accuracy | Successful login & smooth navigation Execution Success Rate: 100% Validation: Manual login & module access verified |
| Confidence Score | 95% |

2. Update Set Configuration Test

The screenshot shows the ServiceNow interface for updating a set named 'Family Expenses'. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Admin', and a search bar. The main title is 'Update Set - Family Expenses'. The form fields include:

- * Name: Family Expenses
- State: In progress
- Parent: (empty)
- Release date: (empty)
- Install date: (empty)
- Installed from: (empty)
- Description: (empty)
- Application: Global
- Created: 2025-10-30 08:06:41
- Created by: admin
- Merged to: (empty)

A blue 'Update' button is at the bottom left.

| Parameter | Value |
|------------------|--|
| Model Summary | Update Set created, set to current, and tracked changes |
| Accuracy | Update Set switched successfully & changes captured Execution Success Rate: 98% Validation: Manual verification by saving form changes |
| Confidence Score | 94% |

3. Table Creation Test - Family Expenses

The screenshot shows the ServiceNow interface for creating a new table named 'Family Expenses'. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Admin', and a search bar. The main title is 'Table - Family Expenses'. A message at the top states: '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 to store data and processes.' with a 'More Info' link. The form fields include:

- * Label: Family Expenses
- * Name: u_family_expenses
- Application: Global
- Remote Table: (empty)

Below the form, there are tabs for 'Columns', 'Controls', and 'Application Access'. Under 'Columns', there is a table with columns: 'Column Label', 'Type', 'Reference', 'Max length', and 'Default value'. A search bar and a toolbar are also present.

| Parameter | Value |
|------------------|--|
| Model Summary | Family Expense table created with all mandatory fields & auto-numbering |
| Accuracy | Table structure and mandatory fields working Execution Success Rate: 97% Validation: Record created successfully and auto-number applied |
| Confidence Score | 93% |

4. Table Creation Test - Daily Expenses

The screenshot shows the ServiceNow application interface. At the top, the navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and a search bar labeled 'Table - Daily Expenses ★'. Below the navigation is a breadcrumb trail: 'Table Daily Expenses'. A tooltip explains what a table is. The main area shows the table creation form with the following details:

- * Label: Daily Expenses
- * Name: u_daily_expenses
- Application: Global
- Remote Table: (empty)

Below the form is a table titled 'Table Columns' with columns: Column Label, Type, Reference, Max length, and Default value. The first row has 'for text' selected in the Type dropdown.

| Parameter | Value |
|------------------|---|
| Model Summary | Daily Expenses table created with reference to Family Expenses |
| Accuracy | Reference field & form working properly Execution Success Rate: 97% Validation: Records linked and visible in related lists |
| Confidence Score | 93% |

5. Relationship Test

The screenshot shows the ServiceNow interface for configuring a relationship named "Daily Expenses". The top navigation bar includes "servicenow", "All", "Favorites", "History", "Admin", and a search bar. The main title is "Relationship - Daily Expenses". Below the title, there are fields for "Name" (set to "Daily Expenses"), "Application" (set to "Global"), and "Advanced" (unchecked). It also shows "Applies to table" (set to "Family Expenses [u_family_ex]") and "Queries from table" (set to "Daily Expenses [u_daily_expe]"). A note at the bottom states: "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 [script](#)". A code editor window displays the following ECMAScript 2021 (ES12) mode script:

```

Query with  Turn on ECMAScript 2021 (ES12) mode ?
①             
1 (function refineQuery(current, parent) {
2
3
4 // Add your code here, such as current.addQuery(field, value);
5
6 current.addQuery('u_date',parent.u_date);
7

```

| Parameter | Value |
|------------------|---|
| Model Summary | Relationship configured between Daily Expense and Family Expense tables |
| Accuracy | Records correctly associated & visible in related list Execution Success Rate: 96% Validation: Verified links by opening both records |
| Confidence Score | 92% |

6. Business Rule Execution Test

The screenshot shows the ServiceNow interface for configuring a business rule named "New record". The top navigation bar includes "servicenow", "All", "Favorites", "History", "Admin", and a search bar. The main title is "Business Rule - New Record". Below the title, there are fields for "Name" (empty), "Application" (set to "Global"), "Table" (set to "-- None --"), "Active" (checked), and "Advanced" (unchecked). A note at the bottom 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 met. [More Info](#)". A "When to run" tab is selected, showing the instruction: "Specify whether the business rule should run on Insert or Update. Use Filter Conditions to specify under which conditions the business rule should run."

| Parameter | Value |
|------------------|--|
| Model Summary | Business rule created to auto-update total family expense when daily expense is added. |
| Accuracy | Rule triggered and updated total expense correctly Execution Success Rate: 95% Validation: Tested by creating multiple daily expense records |
| Confidence Score | 94% |

7. Data Entry & Result Verification

| Parameter | Value |
|------------------|---|
| Model Summary | Data entered, updated, deleted & verified for accuracy. |
| Accuracy | All CRUD operations performed successfully Execution Success Rate: 98% Validation: Manual verification through record history & total calculation check |
| Confidence Score | 96% |