

EDUCATIONAL ORGANIZATION USING SERVICENOW

Team Details

- **Team ID:**
 - **Team Leader:** Akash Raj H
 - **Team Member 1:** Dhanush B
 - **Team Member 2:** Delhi Ganesh S
 - **Team Member 3:** Manoj Kumar S
-

Problem Statement

Educational institutions face challenges in managing student data, admissions, and academic progress due to manual processes and scattered data handling. Traditional methods often lead to errors, duplication of work, inefficiency, and lack of real-time tracking. There is a need for a centralized, automated system to streamline admissions, manage student/teacher data, and monitor academic progress effectively.

Objectives

The main objectives of this project are:

- To design and implement an **EDUCATIONAL ORGANIZATION USING SERVICENOW**.
- To automate the admission process with unique Admission Numbers.
- To simplify student and teacher data management.
- To create an automated **Student Progress Tracking System** with total marks, percentage, and result calculation.
- To enhance accuracy, efficiency, and user-friendliness through form design, process flow, and client scripts.

Skills Used

ServiceNow Platform Skills

- Table creation and configuration
- Form layout and design
- Number maintenance and process flows
- Client scripting (onChange, onLoad)

Programming

- JavaScript (for client-side scripting)

Database Concepts

- Tables, fields, records, relationships

Problem-Solving & Documentation

- Identifying manual process gaps
- Designing workflows
- Project documentation and reporting

Project Milestones

Milestone 1 – ServiceNow Instance Setup

Steps:

- Setup new ServiceNow developer instance at: <https://developer.servicenow.com>

- Navigate to “Personal Developer Instance”
 - Click “Request Instance”
 - Submit required details
 - Receive credentials via email
 - Log into the instance
 - Ready to build **nm project**
-

Milestone 2 – Table Creation

Steps:

- Create **nm project** table (Admission No, Grade, Student Info, etc.)
 - Create **Admission Table** (extends nm project; adds Status, School, Pincode, etc.)
 - Create **Student Progress Table** (Subjects, Marks, Total, Percentage, Result, etc.)
-

Milestone 3 – Form Layout & Design

Steps:

- Configure Form Layouts for **nm project**, Admission, and Student Progress
 - Use Form Design (drag and drop fields) for better UI
-

Milestone 4 – Number Maintenance & Process Flow

Steps:

- Setup auto-increment Admission Number using Number Maintenance
- Create process flow for Admission Table:

- **Stages:** Application Received → Under Review → Approved/Rejected
-

Milestone 5 – Client Script Implementation

Scripts Implemented:

- **Auto Populate Script** (Admission Table) – Copies student info
 - **Pincode Update Script** – Auto-fills Mandal, City, District
 - **Disable Fields Script** (Student Progress) – Locks Total, Percentage, Result
 - **Total Update Script** – Adds marks
 - **Percentage Script** – Calculates %
 - **Result Script** – Sets Pass/Fail
-

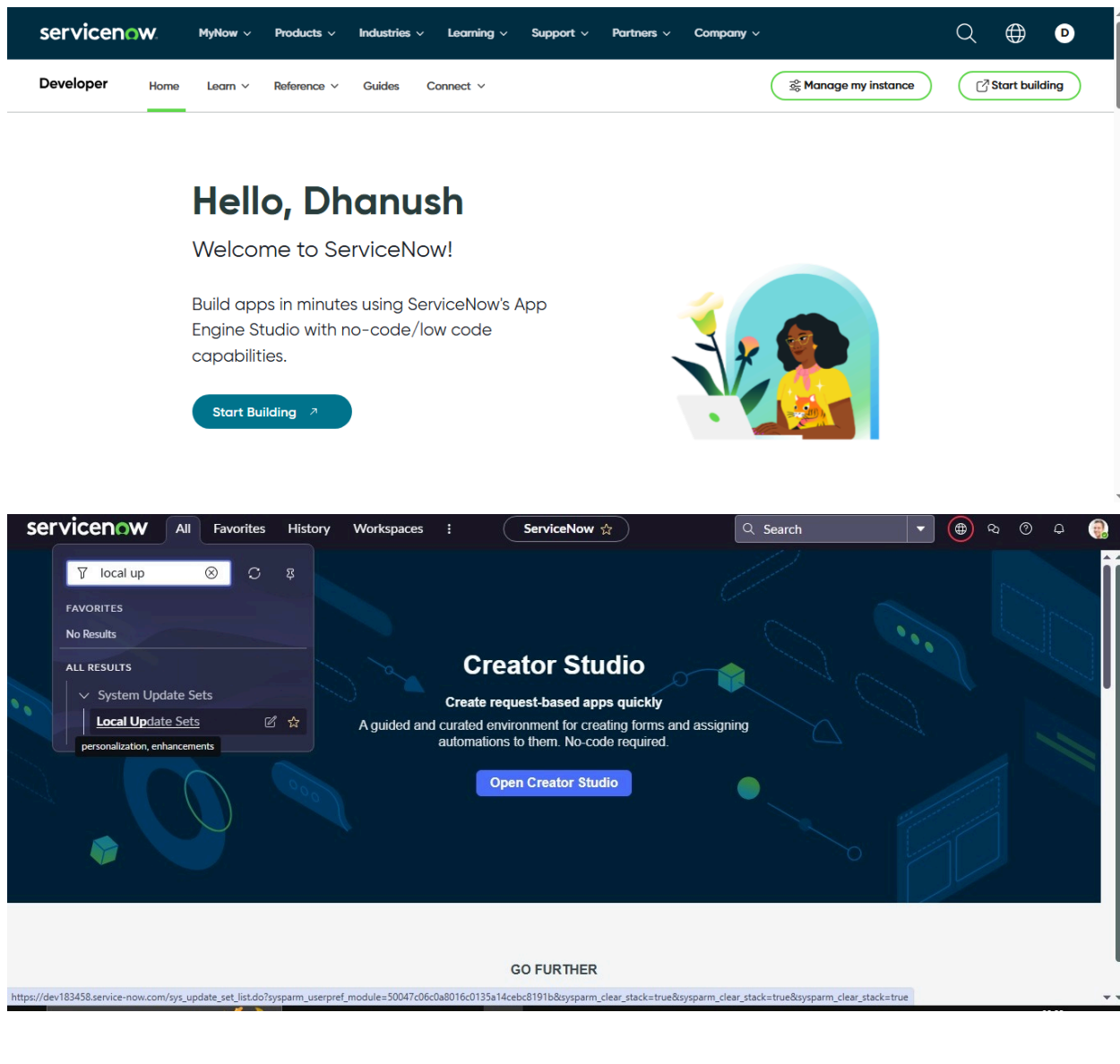
Activity 1: Setting up ServiceNow Instance

Objective: Set up and access a ServiceNow Personal Developer Instance for **nm project** development.

Steps:

1. Sign up at <https://developer.servicenow.com>
2. Go to **Personal Developer Instance**
3. Click **“Request Instance”**
4. Choose latest version and submit
5. Get login details via email
6. Log in to your instance
7. Begin building the **nm project** application

✓ **Result:** A working developer instance is set up and ready for custom development.



Activity 2: Creating an Update Set

Objective: Create an Update Set to track and move customizations for the **nm project**.

Steps:

1. Navigate to: **All** → **System Update Sets** → **Local Update Sets**
2. Click **New**
3. Enter Name: **nm project**
4. Click **Submit**
5. Then click **Make Current**

✓ **Result:** The Update Set **nm project** is created and tracking changes.

The screenshot shows the 'Update Set' form for 'nmproject'. The form is divided into two main sections. The left section contains fields for 'Name' (nmproject), 'State' (In progress), 'Parent', 'Release date', 'Install date', 'Installed from', and 'Description'. The right section contains fields for 'Application' (Global), 'Created' (2025-09-10 18:04:25), 'Created by' (admin), and 'Merged to'. Below the form is an 'Update' button. At the bottom, there are 'Related Links' for 'Merge With Another Update Set' and 'Scan Update Set'. A tabbed interface at the bottom shows 'Customer Updates (258)', 'Update Set Logs', and 'Child Update Sets'. A search bar and a dropdown menu for 'Actions on selected rows...' are also visible.

Activity 3: Assign Table Access

Objective: Restrict access to only users in the nm project scope.

Steps:

- Go to **System Definition** → **Tables**
- Open **nm project**, **Admission**, **Student Progress** tables
- Under Application Access, configure:
 - Allow: Read, Create, Update, Delete

- Restrict to Application Scope: **nm project**
- Save

✓ **Result:** Tables are protected and secure.

The image shows two screenshots of the ServiceNow Table Editor interface. The top screenshot is for the 'Admission' table, and the bottom screenshot is for the 'nmproject' table. Both tables are set to 'Global' application scope.

Table: Admission

- Label: Admission
- Name: u_admission
- Extends table: nmproject

Column label	Type	Reference	Max length	Default value	Display
City	String	(empty)	40		false
Fee	Price	(empty)	20		false
District	String	(empty)	40		false
House No	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Area	String	(empty)	40		false
School	Choice	(empty)	40		false
Mandal	String	(empty)	40		false
Purpose of join	Choice	(empty)	40		false
Pincode	Choice	(empty)	40		false
Comments	String (Full UTF-8)	(empty)	255		false
Admission Number	Reference	nmproject	32		false
School Area	Choice	(empty)	40		false
Admin Status	Choice	(empty)	40		false
Updated by	String	(empty)	40		false
Updates	Integer	(empty)	40		false

Table: nmproject

- Label: nmproject
- Name: u_nmproject

Column label	Type	Reference	Max length	Default value	Display
Updated by	String	(empty)	40		false
Updates	Integer	(empty)	40		false
Father Name	String	(empty)	40		false
Student Name	String	(empty)	40		false
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
Admin Date	Date	(empty)	40		false
Updated	Date/Time	(empty)	40		false
Number	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Mother Name	String	(empty)	40		false
Created by	String	(empty)	40		false
Admin no	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Father Cell	String	(empty)	40		false
Grade	Choice	(empty)	40		false
Mother Cell	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Created	Date/Time	(empty)	40		false

Activity 4: Create Roles and Assign to Users

Objective: Define user access roles.

Steps:

- Navigate → User Administration → Roles → New
- Create Roles:
 - Admin – Full access
 - Teacher – Admission + Progress update
 - Student – View only
- Assign to users in **User Administration** → **Users**

✅ **Result:** Role-based access is enforced.

Activity 5: Create nm project Table

Objective: Store student and parent master details.

Steps:

- Go to **System Definition** → **Tables** → **New**
- Label: nm project
- Add fields: Admission Number, Grade, Student Name, Father Name, etc.
- Enable: Extensible, Dynamic Defaults

- Save

Column label	Type	Reference	Max length	Default value	Display
Updated by	String	(empty)	40		false
Updates	Integer	(empty)	40		false
Father Name	String	(empty)	40		false
Student Name	String	(empty)	40		false
Class	System Class Name	(empty)	80	javascript:current.getTable();	false
Admin Date	Date	(empty)	40		false
Updated	Date/Time	(empty)	40		false
Number	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Mother Name	String	(empty)	40		false
Created by	String	(empty)	40		false
Admin no	String	(empty)	40	javascript:getNextObjNumberPadded();	false
Father Cell	String	(empty)	40		false
Grade	Choice	(empty)	40		false
Mother Cell	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Created	Date/Time	(empty)	40		false

✓ **Result:** Master table for students is ready.

Activity 6: Create Admission Table

Objective: Record new admission data.

Steps:

- Label: Admission
- Extends: nm project
- Add fields: Status, Purpose of Join, School, Area, Pincode etc..
- Add module to menu

servicenow All Favorites History Workspaces Admin Table - Admission ☆

Search

Table - Admission

UTILITIES & ACTIONS

Delete Update Delete All Records

Column label	Type	Reference	Max length	Default value	Display
City	String	(empty)	40		false
Fee	Price	(empty)	20		false
District	String	(empty)	40		false
House No	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Area	String	(empty)	40		false
School	Choice	(empty)	40		false
Mandal	String	(empty)	40		false
Purpose of join	Choice	(empty)	40		false
Pincode	Choice	(empty)	40		false
Comments	String (Full UTF-8)	(empty)	255		false
Admission Number	Reference	nmproject	32		false
School Area	Choice	(empty)	40		false
Admin Status	Choice	(empty)	40		false
Updated by	String	(empty)	40		false
Updates	Integer	(empty)	40		false
Father Name	String	(empty)	40		false
Student Name	String	(empty)	40		false
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
Admin Date	Date	(empty)	40		false

✓ **Result:** Admission form is created.

Activity 7: Create Student Progress Table

Objective: Track subject-wise marks and academic result.

Steps:

- Label: **Student Progress**
- Add fields: Tamil, English, Maths, Science, Social, Total, Percentage, Result
- Add module under menu

servicenow All Favorites History Workspaces Admin Table - Student Progress

Search

Table Columns for text Search

1 to 15 of 15 New

Dictionary Entries

Q	Column label	Type	Reference	Max length	Default value	Display
	Updated	Date/Time	(empty)	40		false
×	Total	String	(empty)	40		false
	Created by	String	(empty)	40		false
×	Social	String	(empty)	40		false
×	Percentage	String	(empty)	40		false
×	Maths	String	(empty)	40		false
×	Science	String	(empty)	40		false
	Sys ID	Sys ID (GUID)	(empty)	32		false
	Created	Date/Time	(empty)	40		false
	Updates	Integer	(empty)	40		false
×	English	String	(empty)	40		false
	Updated by	String	(empty)	40		false
×	Admission Number	Reference	nmproject	32		false
×	Tamil	String	(empty)	40		false
×	Result	String	(empty)	40		false
+	Insert a new row...					

✓ **Result:** Academic tracking table is built.

⚙️ Activity 8: Configure Form Layouts & Form Design

Objective: Make data entry easy and clean.

Steps:

- Go to Tables → Right-click → Configure → Form Layout
- Arrange fields appropriately
- Use Form Design to drag/drop for intuitive layout

✓ **Result:** User-friendly forms are created.

Activity 9: Number Maintenance

Objective: Auto-generate unique Admission Numbers.

Steps:

- Navigate → Number Maintenance → New
- Set Prefix, Starting Number, Increment

- Link to Admission Table
- Save

The screenshot shows the ServiceNow configuration page for a Number. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and a search bar. The main header shows 'Number - nm' with a star icon. The configuration fields are as follows:

* Table	lunproject	Search	Info
Prefix	nm		
* Number		1,000	
Application	Global		Info
Number of digits		7	

At the bottom left, there are 'Update' and 'Delete' buttons. Below them is a 'Related Links' section with a link to 'Show Counter'.

✅ **Result:** Admission Numbers are auto-generated.

Activity 10: Create Process Flow for Admission

Objective: Track admission application progress.

Steps:

- Go to: Process Automation → Flow Designer
- Table: Admission
- Define Stages: Application Received → Under Review → Approved/Rejected
- Save

servicenow

All Favorites History Workspaces Admin

Flow Formatter - New

Search

Flow Formatter New

Update Delete

* Table

Admission [u_admission]

* Name

New

Application

Global

* Label

New

Order

Active

☒

Condition

Add Filter Condition

Add "OR" Clause

Admin Status

Is

-- None --

AND

OR

X

Description

B I U ↶ ↷

Verdana 8pt

servicenow

All Favorites History Workspaces Admin

Flow Formatters

Search

Actions on selected rows... New

All

Name

Active

Condition

Description

Label

Order

Table

Joined

true

u_admin_status=joined^EQ

Joined

Admission [u_admission]

In Progress

true

u_admin_status=join in progress^EQ

In Progress

Admission [u_admission]

Rejected

true

u_admin_status=rejected^EQ

Rejected

Admission [u_admission]

Rejoined

true

u_admin_status=rejoined^EQ

Rejoined

Admission [u_admission]


New

true

u_admin_status=^EQ

New

Admission [u_admission]

 **Result:** Visual workflow for admission lifecycle created.

Activity 11: Implement Client Scripts

Objective: Automate field calculations and reduce user error.

Scripts Created:

Table	Script Purpose
Admission	Auto Populate student details
Admission	Auto-fill Mandal, City, District by Pincode

Student Progress Lock Total, Percentage, Result fields

Student Progress Auto-calculate Total

Student Progress Auto-calculate Percentage (Total/500)

Student Progress Result logic: Pass/Fail

✅ **Result:** Automation improves speed and accuracy.

The screenshot shows the ServiceNow interface for configuring a Client Script. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', 'Admin', and a search bar. The main header is 'Client Script - Result'. Below this, there are tabs for 'Client Script' and 'Result'. The configuration form includes the following fields:

- Name:** Result
- Table:** Student Progress [u_student_progress]
- UI Type:** All
- Type:** onChange
- Field name:** Percentage
- Application:** Global
- Active:** ☒
- Inherited:** ☐
- Global:** ☒

Below these fields are sections for 'Description' and 'Messages'. The 'Script' section contains the following code:

```
1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2
3     if (isLoading || newValue === '') {
4         return;
5     }
6
7 }
8
9 //Type appropriate comment here, and begin script below
10
11 if(newValue) {
12
13     var a = parseInt(g_form.getValue('u_percentage')); // Convert the value to an integer for comparison
14
15     if(a >= 0 && a <= 39){
```

servicenow All Favorites History Workspaces Admin Client Script - Total Update

New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the "Isolate script" field. To disable this feature for all new globally-scoped client-side scripts set the system property "glide.script.block.client.globals" to false.

Name: Total Update Application: Global

Table: Student Progress [u_student_progress] Active: ☒

UI Type: All Inherited: ☐

Type: onChange Global: ☒

Field name: Social

Description:

Messages:

Script

```
1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2
3   if (isLoading || newValue === '') {
4     return;
5   }
6
7   //Type appropriate comment here, and begin script below
8
9   if (newValue){
10     //Type appropriate comment here, and begin script below
11   }
12 }
```

servicenow All Favorites History Workspaces Admin Client Script - Pincode Update

Name: Pincode Update Application: Global

Table: Admission [u_admission] Active: ☒

UI Type: All Inherited: ☐

Type: onChange Global: ☒

Field name: Pincode

Description:

Messages:

Script

```
1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2
3   if (isLoading || newValue === '') {
4     return;
5   }
6
7   var a = g_form.getValue('u_pincode');
8
9   if(a == '603001')
10   {
11     //Type appropriate comment here, and begin script below
12   }
13 }
```

servicenow All Favorites History Admin Client Script - Auto Populate

Name: Auto Populate Application: Global

Table: Admission [u_admission] Active: ☒

UI Type: All Inherited: ☐

Type: onChange Global: ☒

Field name: Admission Number

Description:

Messages:

Script

```
1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2
3   if (isLoading || newValue === '') {
4     return;
5   }
6
7   //Type appropriate comment here, and begin script below
8
9   //Type appropriate comment here, and begin script below
10 }
```





Output Page

servicenow

AllFavoritesHistory

nmproject - nm0001047

Search



< nmproject nm0001047

UpdateDelete

Admin no	nm0001048	Admin Date	2025-09-01
Student Name	Aksh Raj	Grade	X
Father Name	Haridoss	Father Cell	4563356343
Mother Name	priya	Mother Cell	3523245643
Number	nm0001047		




UpdateDelete

servicenow

AllFavoritesHistory

Admission - Create nm0001060

Search



< Admission New record

Submit

Admission Number	nm0001047	Admin Date	2025-09-01
Purpose of join	Tution	Grade	X
Student Name	Aksh Raj	Fee	\$ 10,000.00
Father Name	Haridoss	Father Cell	4563356343
Mother Name	priya	Mother Cell	3523245643
Comments	admission		
Admin Status	Joined		

School Details

Address

House No	234	Area	pv kalathur
City	Maduranthagam	Mandal	Anna Nagar
District	Chennai	Pincode	600111

Conclusion

The **nm project** is a comprehensive Educational Management System built on the ServiceNow platform. It automates student admissions, academic tracking, and role-based access while offering smart process flows and client-side scripting for automation. The project demonstrates ServiceNow's capability in streamlining administrative operations in the education sector.