

Project Design

Team ID	LTVIP2026TMIDS36010
Project Name	Educational Organization Using ServiceNow

Proposed Solution:

S. No	Parameters	Description
1.	Problem Statement	The Educational Management System is a comprehensive digital platform developed to streamline and automate various administrative tasks within educational institutions. It enables efficient and secure management of both student and teacher data, ensuring accuracy and easy accessibility.
2.	Solution Description	This enables real-time tracking and service automation, allowing educational institutions to respond quickly to requests and manage resources more efficiently.
3.	Uniqueness	The uniqueness of the Educational Organization using ServiceNow project lies in its ability to ensure a secure digital transformation, significantly reducing the administrative burden while enhancing accuracy, operational efficiency, and transparency in academic processes.
4.	Social Impact	The Educational Organization using ServiceNow project creates a meaningful social impact by ensuring equal and timely access to academic resources and services for both students and staff, thereby promoting inclusivity and strengthening support across diverse educational communities.
5.	Business Model	Business model focused on enhancing institutional efficiency and user experience through digital transformation. The model leverages the capabilities of the ServiceNow platform to deliver smart, automated solutions for core educational services.
6.	Scalability Of the Solution	This project is built on a scalable architecture that can easily adapt to the growing needs of institutions of all sizes—from small colleges to large universities or education groups with multiple campuses.

Milestone-1: Setting up Instance Purpose:

The ServiceNow instance is used to digitize and streamline key administrative processes in an educational organization. It helps manage student admissions, attendance, faculty data, and service requests through automated workflows. This improves efficiency, data accuracy, and collaboration while reducing manual work and paperwork.

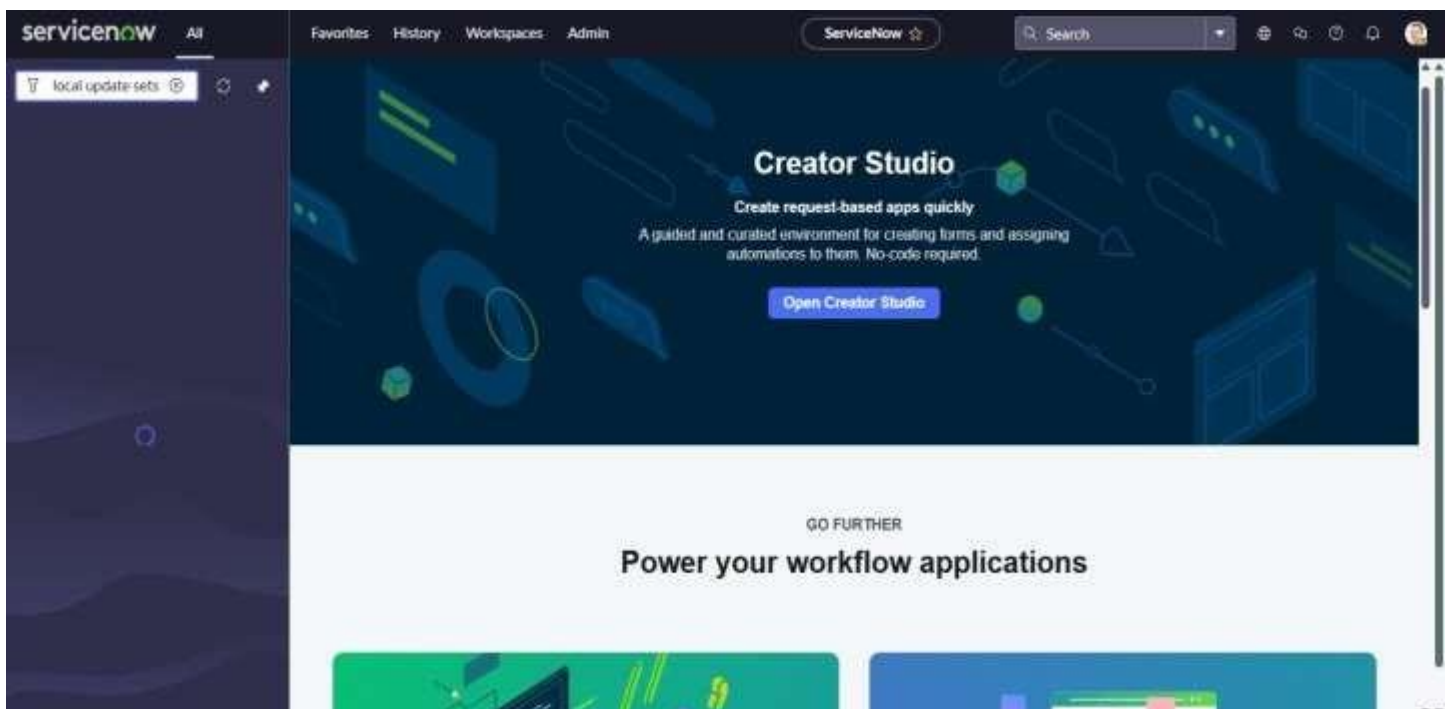
Uses :

The ServiceNow instance is used to automate student admissions, track attendance, manage faculty and student records, handle service requests, and monitor academic progress.

Activity-1: ServiceNow instance

Steps :

1. Sign in to your ServiceNow instance with the given credentials.
2. Go to the ServiceNow Developer site: developer.servicenow.com.
3. Sign up and log in to your developer account.
4. Navigate to Personal Developer Instance.
5. Click Request Instance and submit the required details. 6. Wait for an email with your instance access link.



Milestone-2: Update Set

Purpose :

Update set is used to track, manage, and migrate customizations like forms, tables, and workflows related to student admissions, attendance, and academic records. It ensures safe deployment, error-free updates, and helps maintain consistency across development and production environments.

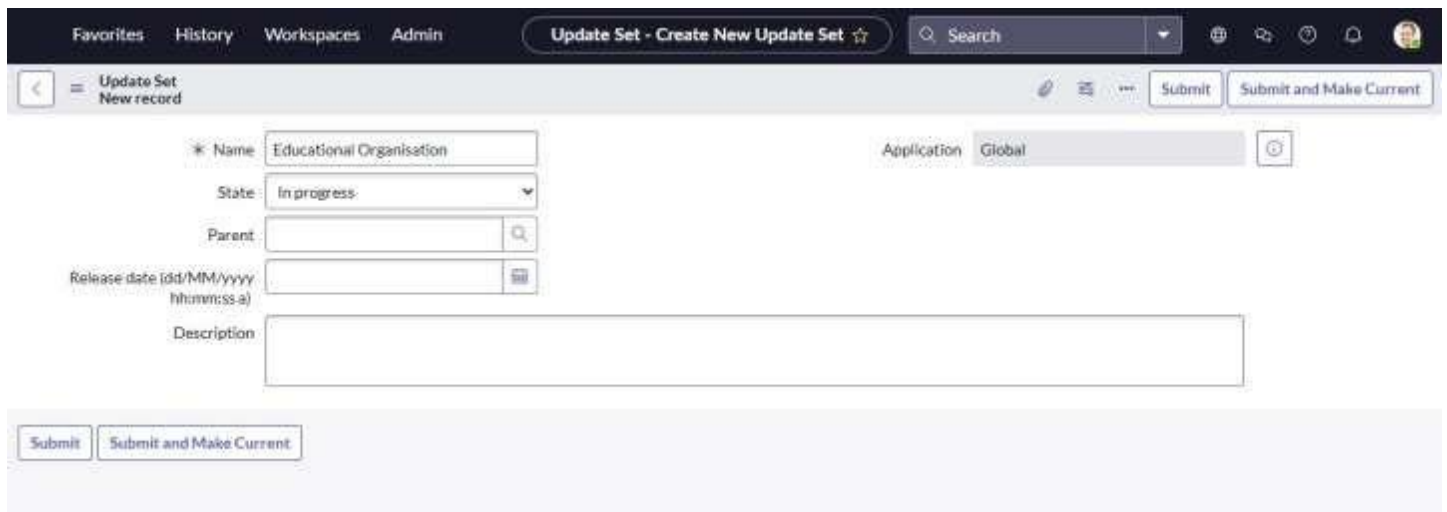
Uses :

Tracks modifications made to applications (like student admission or attendance modules), allowing rollback if needed. Once an update set is created for features like course registration or faculty evaluation, it can be reused or cloned for other departments or campuses.

Activity-1: Create update set

Steps :

1. Click on All >> Local update sets
2. Click on new



The screenshot shows the 'Update Set - Create New Update Set' form. The top navigation bar includes 'Favorites', 'History', 'Workspaces', 'Admin', and a search bar. The main form area has a title bar 'Update Set - Create New Update Set' with a star icon. Below the title bar, there are buttons for 'Submit' and 'Submit and Make Current'. The form fields include: 'Name' (text input with value 'Educational Organisation'), 'State' (dropdown menu with value 'In progress'), 'Parent' (text input with a search icon), 'Release date (dd/MM/yyyy hh:mm:ss a)' (datetime picker), and 'Description' (text area). At the bottom, there are buttons for 'Submit' and 'Submit and Make Current'.

3. Enter the Details Name: Educational Organisation
4. Click on Submit and make Current.

Milestone-3: Table

Purpose :

Tables are used to store and organize data such as student records, faculty details, attendance, admissions, and academic progress. They help in centralizing information, enabling easy access, retrieval, and management of educational data within the ServiceNow platform.

Uses :

Store and organize structured data ,Allow easy retrieval and filtering , Enable data relationships across systems , Support report generation , Help in process automation , Maintain data accuracy and integrity.

Activity-1: Creating Salesforce Table.

1. All >> Tables.
2. Click on new
3. Enter the Label Salesforce double Click on Name it will Automatically generate Api name.
4. Create columns as given below, Double Click on Column label and Enter the Column labels and click on the tick mark <<Give Type as given .



Column Label	Type	Default Value	Sequence	API Name
Admin Date	Date	(empty)	40	false
Admin Number	String	(empty)	40	<code>javascript:getNextObjNumberPadded();</code>
Father Cell	String	(empty)	40	false
Father Name	String	(empty)	40	false
Grade	Choice	(empty)	40	false
Mother Cell	String	(empty)	40	false
Mother Name	String	(empty)	40	false
Student Name	String	(empty)	40	false

5. Click on controls >> Enable Extensible.
6. Click on “Admin Number” column, In Related Links Click on Advanced View >> Default View (Enable Use dynamic default) >> select Get Next Padded Number in Dynamic default value >> Update .



Choice List Specification | Calculated Value | **Default Value**

The Default value specifies what value the field has when first displayed.

Use dynamic default ☒

Dynamic default value: [Search] [Info]

7. Click on “Grade” Column >> Click on Choices and give Label, Value and Sequence as given below.

Dictionary Entry - Grade

Access Controls: Choices Attributes Labels (1)

Label Search

Choices

	Label	Value	Language	Sequence	Inactive	Updated
1	Prept	Prept	en	1	false	24/06 10:33:19 AM
2	Nursery	Nursery	en	2	false	24/06 10:33:49 AM
3	UKG	UKG	en	3	false	24/06 10:34:27 AM
4	I	1st	en	4	false	24/06 10:35:03 AM
5	II	2nd	en	5	false	24/06 10:35:33 AM
6	III	3rd	en	6	false	24/06 10:35:55 AM
7	IV	4th	en	7	false	24/06 10:36:14 AM
8	V	5th	en	8	false	24/06 10:36:36 AM
9	VI	6th	en	9	false	24/06 10:37:01 AM
10	VII	7th	en	10	false	24/06 10:37:25 AM
11	VIII	8th	en	11	false	24/06 10:38:04 AM
12	IX	9th	en	12	false	24/06 10:38:37 AM
13	X	10th	en	13	false	24/06 10:38:59 AM

Insert a new row...

Activity-2: Creating Admission Table

1. Create an Admission Table with Columns given.
2. Select Extends Table >> Salesforce and also Select Add module to menu >> Salesforce.
3. Create Fields as shown
4. Create choice for Admin Status as

Dictionary Entry - Admin Status

Choice: Dropdown with -- None --

Create Choice List Delete Column Update

Related Links

Show Table Run Point Scan Advanced view

Access Controls: Choices Attributes Labels (1)

Label Search

Choices

	Label	Value	Language	Sequence	Inactive	Updated
1	New	New	en	1	false	24/06 11:26:16 AM
2	Join In progress	In progress	en	2	false	24/06 11:27:07 AM
3	Joined	Joined	en	3	false	24/06 11:27:42 AM
4	Rejected	Rejected	en	4	false	24/06 11:30:13 AM
5	Closed	Closed	en	5	false	24/06 11:30:13 AM
6	Rejoined	Rejoined	en	6	false	24/06 11:32:14 AM
7	Cancelled	Cancelled	en	7	false	24/06 11:32:50 AM

Insert a new row...

5. Create choice for Pincode as

Dictionary Entry - Pincode

Displays a list of suggested values in a Choice list. In the Advanced view you can select the Choice table and the Choice field to take choice values from, plus a Dependent field.

Choice: — None —

Delete Column Update

Related Links:
[Show Table](#)
[Run Point Scan](#)
[Advanced view](#)

Access Controls: Choices (3) Attributes Labels (11)

Label Search

Actions on selected rows... New

Label	Value	Language	Sequence	Inactive	Updated
509055	509355	en	1	false	24/06 11:34:58 AM
500079	500079	en	2	false	24/06 11:37:57 AM
500081	500081	en	3	false	24/06 11:40:07 AM

Insert a new row...

1 to 3 of 3

6. Create choice for purpose of join

Dictionary Entry - Purpose of join

Displays a list of suggested values in a Choice list. In the Advanced view you can select the Choice table and the Choice field to take choice values from, plus a Dependent field.

Choice: — None —

Delete Column Update

Related Links:
[Show Table](#)
[Run Point Scan](#)
[Advanced view](#)

Access Controls: Choices (1) Attributes Labels (11)

Label Search

Actions on selected rows... New

Label	Value	Language	Sequence	Inactive	Updated
Tuition	Tuition	en	1	false	24/06 11:45:13 AM
Coaching	Coaching	en	2	false	24/06 11:50:57 AM
Teacher	Teacher	en	3	false	24/06 11:51:21 AM

Insert a new row...

1 to 1 of 1

7. Create choice for school as

Activity-3: Creating Student Progress Table

1. Create a Student Progress Table with Columns given.
2. Select Add module to menu >> Salesforce.
3. Create Fields as show

X	Admission Number	Reference	Salesforce	32	false
X	English	String	(empty)	40	false
X	Hindi	String	(empty)	40	false
X	Maths	String	(empty)	40	false
X	Percentage	String	(empty)	40	false
X	Result	String	(empty)	40	false
X	Science	String	(empty)	40	false
X	Social	String	(empty)	40	false
X	Telugu	String	(empty)	40	false
X	Total	String	(empty)	40	false
...	Insert a new row...				

Milestone-4: Form Layout

Purpose:

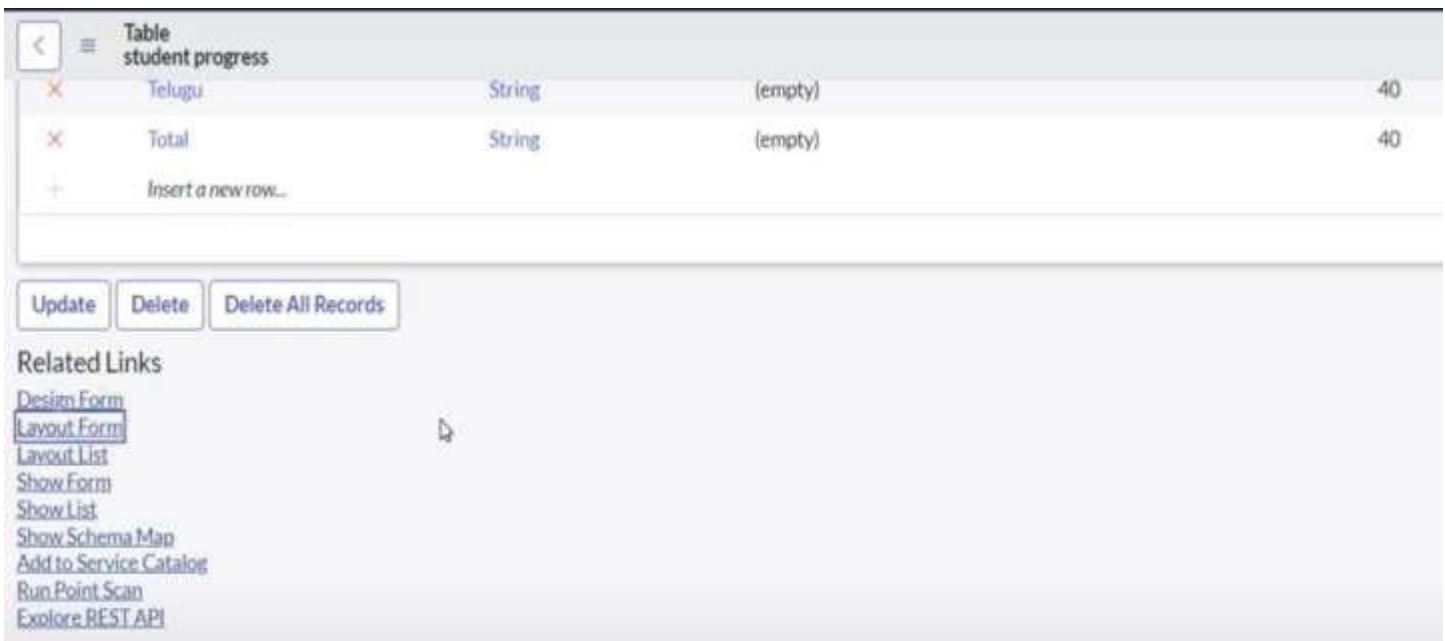
Form Layout is used to organize and display fields in a structured way on forms. It helps educational institutions separate student and faculty data, show important details clearly, and improve data entry and user experience.

Uses :

Organizes form fields for better clarity , Separates student, faculty, and admin data sections , Highlights mandatory fields for accurate data entry , Improves user experience and efficiency, Supports customized views for different users.

Activity-1: Table form for Student Progress Table

1. In the Student Progress Table Page , Click on Layout form.



2. Click on Admission Number [+].
3. Select below Admission Number fields in Available side and send it to selected side as below >> save.

Milestone-5: Form Design

Purpose :

Form Design is used to customize the appearance and structure of forms to suit the institution's needs. It helps in displaying relevant fields, improving data accuracy, and creating user-friendly interfaces for managing student, faculty, and academic records.

Activity-1: Form design for salesforce table

1. All >> System Definition >> Tables .
2. In Label Search for Salesforce and open .
3. Right Click on top Toggle >> Configure >> Form Design.
4. In drop down select Salesforce(u_salesforce)
5. Drag and drop the fields to the left side as below.

The screenshot shows the 'Form Design' interface for a table named 'salemone [v, caleforce]'. On the left, there is a sidebar with a 'Fields' section containing a list of fields: 'Class', 'Created', 'Created by', 'Updated', 'Updated by', 'Updates', 'Activities (Related)', 'Consultant Search Results', and 'Ratings'. The main area displays a table structure with the following columns: 'Admission Number', 'Admission Date', 'Grade', 'Student Name', 'Father Name', 'Mother Name', 'Father Cell', and 'Mother Cell'. Each column has a corresponding field type icon and a 'Get Data' button.

Activity-2: Form Design for Admission Table

1. Follow the same steps as Activity1, Configure the fields as below and Save.

The screenshot shows the 'Form Design' interface for a table named 'Admission [v, admission]'. On the left, there is a sidebar with a 'Fields' section containing a list of fields: 'Class', 'Created', 'Created by', 'Updated', 'Updated by', 'Updates', 'Activities (Related)', 'Consultant Search Results', and 'Ratings'. The main area displays a table structure with the following columns: 'Admission Number', 'Admission Date', 'Grade', 'Student Name', 'Father Name', 'Mother Name', 'Father Cell', and 'Mother Cell'. Each column has a corresponding field type icon and a 'Get Data' button.

Activity-3: Form Design for Student progress Table

1. Follow the same steps as Activity1, Configure the fields as below and Save.

The screenshot shows a 'Form Design' window for 'Student Properties'. The interface includes a left sidebar with a tree view containing 'Fields' and 'Formulators'. The main area is divided into three sections: 'New Section', 'Form', and 'Formulator Properties'. The 'Form' section contains a table with columns for 'Field Name', 'Field Type', and 'Field Value'. The 'Formulator Properties' section contains a table with columns for 'Property Name', 'Property Type', and 'Property Value'.

Field Name	Field Type	Field Value
1. Admission Number	Text	
2. Admission Number	Text	
3. Admission Number	Text	
4. Admission Number	Text	
5. Admission Number	Text	
6. Admission Number	Text	
7. Admission Number	Text	
8. Admission Number	Text	
9. Admission Number	Text	
10. Admission Number	Text	
11. Admission Number	Text	
12. Admission Number	Text	
13. Admission Number	Text	
14. Admission Number	Text	
15. Admission Number	Text	
16. Admission Number	Text	
17. Admission Number	Text	
18. Admission Number	Text	
19. Admission Number	Text	
20. Admission Number	Text	
21. Admission Number	Text	
22. Admission Number	Text	
23. Admission Number	Text	
24. Admission Number	Text	
25. Admission Number	Text	
26. Admission Number	Text	
27. Admission Number	Text	
28. Admission Number	Text	
29. Admission Number	Text	
30. Admission Number	Text	
31. Admission Number	Text	
32. Admission Number	Text	
33. Admission Number	Text	
34. Admission Number	Text	
35. Admission Number	Text	
36. Admission Number	Text	
37. Admission Number	Text	
38. Admission Number	Text	
39. Admission Number	Text	
40. Admission Number	Text	
41. Admission Number	Text	
42. Admission Number	Text	
43. Admission Number	Text	
44. Admission Number	Text	
45. Admission Number	Text	
46. Admission Number	Text	
47. Admission Number	Text	
48. Admission Number	Text	
49. Admission Number	Text	
50. Admission Number	Text	
51. Admission Number	Text	
52. Admission Number	Text	
53. Admission Number	Text	
54. Admission Number	Text	
55. Admission Number	Text	
56. Admission Number	Text	
57. Admission Number	Text	
58. Admission Number	Text	
59. Admission Number	Text	
60. Admission Number	Text	
61. Admission Number	Text	
62. Admission Number	Text	
63. Admission Number	Text	
64. Admission Number	Text	
65. Admission Number	Text	
66. Admission Number	Text	
67. Admission Number	Text	
68. Admission Number	Text	
69. Admission Number	Text	
70. Admission Number	Text	
71. Admission Number	Text	
72. Admission Number	Text	
73. Admission Number	Text	
74. Admission Number	Text	
75. Admission Number	Text	
76. Admission Number	Text	
77. Admission Number	Text	
78. Admission Number	Text	
79. Admission Number	Text	
80. Admission Number	Text	
81. Admission Number	Text	
82. Admission Number	Text	
83. Admission Number	Text	
84. Admission Number	Text	
85. Admission Number	Text	
86. Admission Number	Text	
87. Admission Number	Text	
88. Admission Number	Text	
89. Admission Number	Text	
90. Admission Number	Text	
91. Admission Number	Text	
92. Admission Number	Text	
93. Admission Number	Text	
94. Admission Number	Text	
95. Admission Number	Text	
96. Admission Number	Text	
97. Admission Number	Text	
98. Admission Number	Text	
99. Admission Number	Text	
100. Admission Number	Text	

Milestone-6: Number Maintenance Purpose:

Number Maintenance is used to automatically generate unique identification numbers for records like student IDs, admission forms, and attendance entries. It ensures consistency, avoids duplication, and helps in easily tracking and managing records.

Uses :

Generates unique IDs for students, staff, and records, Prevents duplication of entries, Helps in tracking records efficiently, Maintains consistency across forms and module, Supports automated record creation for processes like admissions and attendance

Activity-1: Number Maintenance for Admin Number

1. All >> Number Maintenance >> New
2. Fill the details >> Submit

Number - SAL

Table: Salesforce

Prefix: SAL

Number: 1,000

Application: Global

Number of digits: 7

Update Delete

Related Links

Show Counter

Milestone-7 : Process Flow

Activity-1 : Create flow

1. All >> Process Flow>> New
2. Fill the Details as given Below
3. Right Click on toggle and click on the save .
4. Replace the Name and Label as below and click on Insert on stay.
5. Replace the Name and Label in order and click on Insert on stay. Joined >> Rejected >> Rejoined >> Closed >> Cancelled.
6. Order should be New >> InProgress >> Joined >> Rejected >> Rejoined >> Closed >> Cancelled.

Milestone-8: Client Scripts

Purpose :

Client Scripts are used to run custom code on the client side (browser) to enhance form behavior. They help in validating data, auto-filling fields, and improving user interaction in real-time without needing a server request.

Uses :

Validate form data before submission, Show/hide fields dynamically , Auto-fill or modify fields based on user input , Improve user experience with real-time responses , Reduce server load by handling logic on the client side

Activity-1: Auto populate Client Scripts for Admission Table

1. All >> Client Scripts >> New.
2. Fill the Details as given
3. Write the Code as given, Enable Isolate script and Save.

```
Function onChange(control, oldValue, newValue, isLoading, isTemplate) {
```

```
If (isLoading || newValue === '') {
```

```
Return;
```

```
}
```

```
//Type appropriate comment here, and begin script below
```

```
Var a = g_form.getReference('u_admission_number');
```

```
G_form.setValue('u_admin_date',a.u_admin_date);
```

```
G_form.setValue('u_grade',a.u_grade);
```

```
G_form.setValue('u_student_name',a.u_student_name);
```

```
G_form.setValue('u_father_name',a.u_father_name);
```

```
G_form.setValue('u_mother_name',a.u_mother_name);
```

```
G_form.setValue('u_father_cell',a.u_father_cell);
```

```
G_form.setValue('u_mother_cell',a.u_mother_cell);
```

```

G_form.setDisabled('u_admin_date',a.u_admin_date);

G_form.setDisabled('u_grade',a.u_grade);

G_form.setDisabled('u_student_name',a.u_student_name);

G_form.setDisabled('u_father_name',a.u_father_name);

G_form.setDisabled('u_mother_name',a.u_mother_name);

G_form.setDisabled('u_father_cell',a.u_father_cell);

G_form.setDisabled('u_mother_cell',a.u_mother_cell);

}

```

Client Script - Auto populate

Search

Update Delete

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: Auto populate

Table: Admission [u_admission]

UI Type: Mobile / Service Portal

Type: onChange

Field name: Admin Number

Application: Global

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script:

```

14
15 }
16
17
18 //Type appropriate comment here, and begin script below
19
20 var a = g_form.getReference('u_admission_number');
21
22 g_form.setValue('u_admin_date',a.u_admin_date);
23
24 g_form.setValue('u_grade',a.u_grade);
25

```

Activity-2: Pincode Update Client Scripts for Admission Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

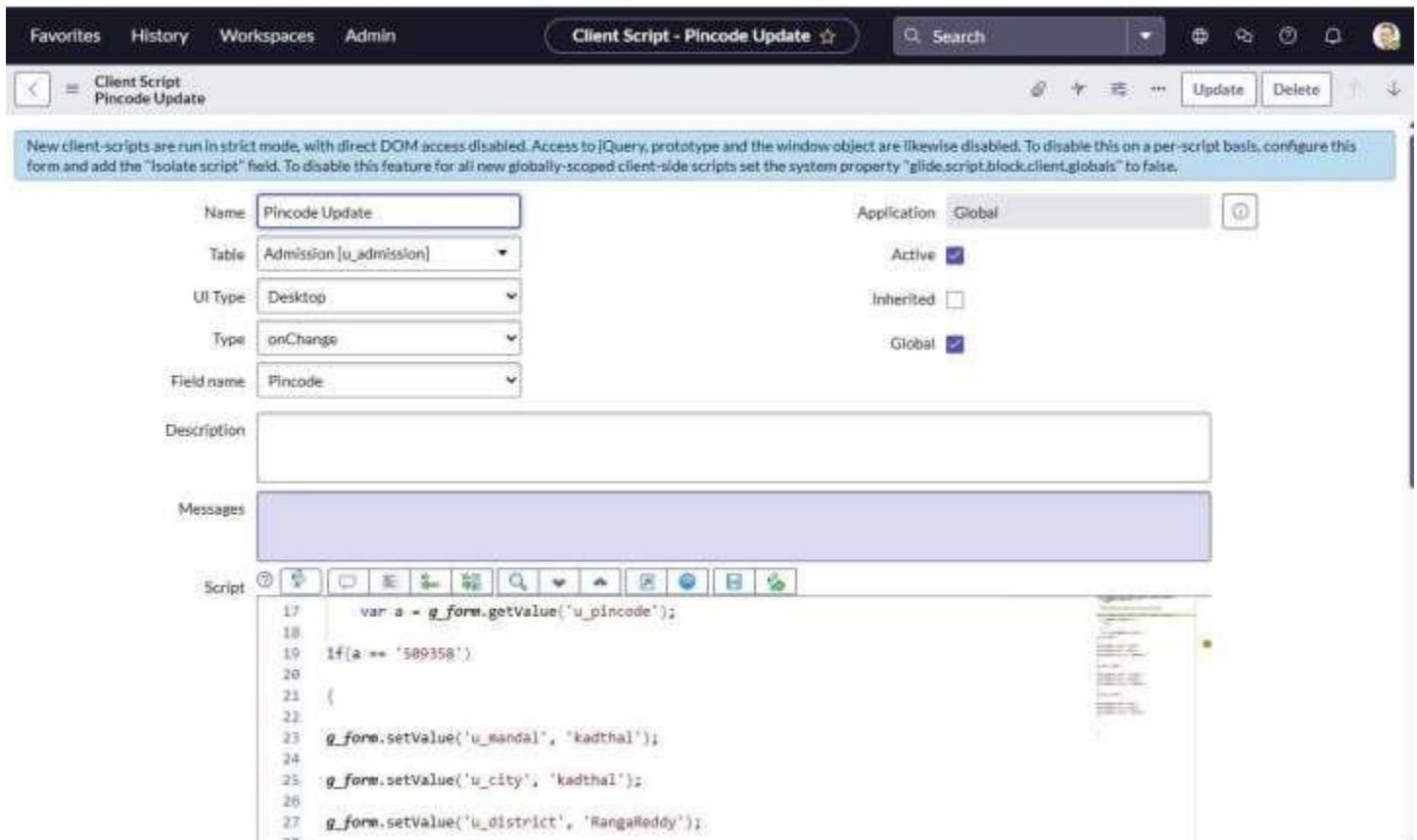
```
Function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
    If (isLoading || newValue === '') {  
        Return ;  
    }  
    Var a = g_form.getValue('u_pincode');  
    If(a == '509358')  
    {  
        G_form.setValue('u_mandal', 'kadthal');  
        G_form.setValue('u_city', 'kadthal')  
        G_form.setValue('u_district', 'RangaReddy');  
    }  
    Else if(a == '500081')  
    {  
        G_form.setValue('u_mandal', 'karmanghat');  
        G_form.setValue('u_city', 'karmanghat');  
        G_form.setValue('u_district', 'RangaReddy');  
    }  
    Else if(a == '500079')  
    {  
        G_form.setValue('u_mandal', 'Abids');  
        G_form.setValue('u_city', 'AsifNagar');
```

```
G_form.setValue('u_district', 'Hyderabad');

}

//Type appropriate comment here, and begin script below

}
```



Activity-3 :Disable Fields Client Scripts for Student progress Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

```
Function onLoad() {
```

```
//Type appropriate comment here, and begin script below
```



```

G_form.setDisabled('u_total',true);

G_form.setDisabled('u_percentage',true);

G_form.setDisabled('u_result',true);

}

```

Client Script - Disable Fields

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:

Table:

UI Type:

Type:

Application:

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script

```

1 function onLoad() {
2     //Type appropriate comment here, and begin script below
3
4 }
5 function onLoad() {
6
7     //Type appropriate comment here, and begin script below
8
9     g_form.setDisabled('u_total',true);
10
11     g_form.setDisabled('u_percentage',true);

```

Activity-4: Total Update Client Scripts for Student progress Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

Function onChange(control, oldValue, newValue, isLoading, isTemplate) {

 If (isLoading || newValue === '') {

 Return;

 }

 //Type appropriate comment here, and begin script below

```

If (newValue){

    Var a = parseInt(g_form.getValue('u_telugu'));

    Var b = parseInt(g_form.getValue('u_hindi'));

    Var c = parseInt(g_form.getValue('u_english'));

    Var d = parseInt(g_form.getValue('u_maths'));

    Var e = parseInt(g_form.getValue('u_science'));

    Var f = parseInt(g_form.getValue('u_social'));

    Var Total = parseInt(a+b+c+d+e+f);

    G_form.setValue('u_total', Total);

}

}

```

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

Table: Student Progress [u_student_pr...

UI Type: All

Type: onChange

Field name: Social

Application: Global

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script:

```

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

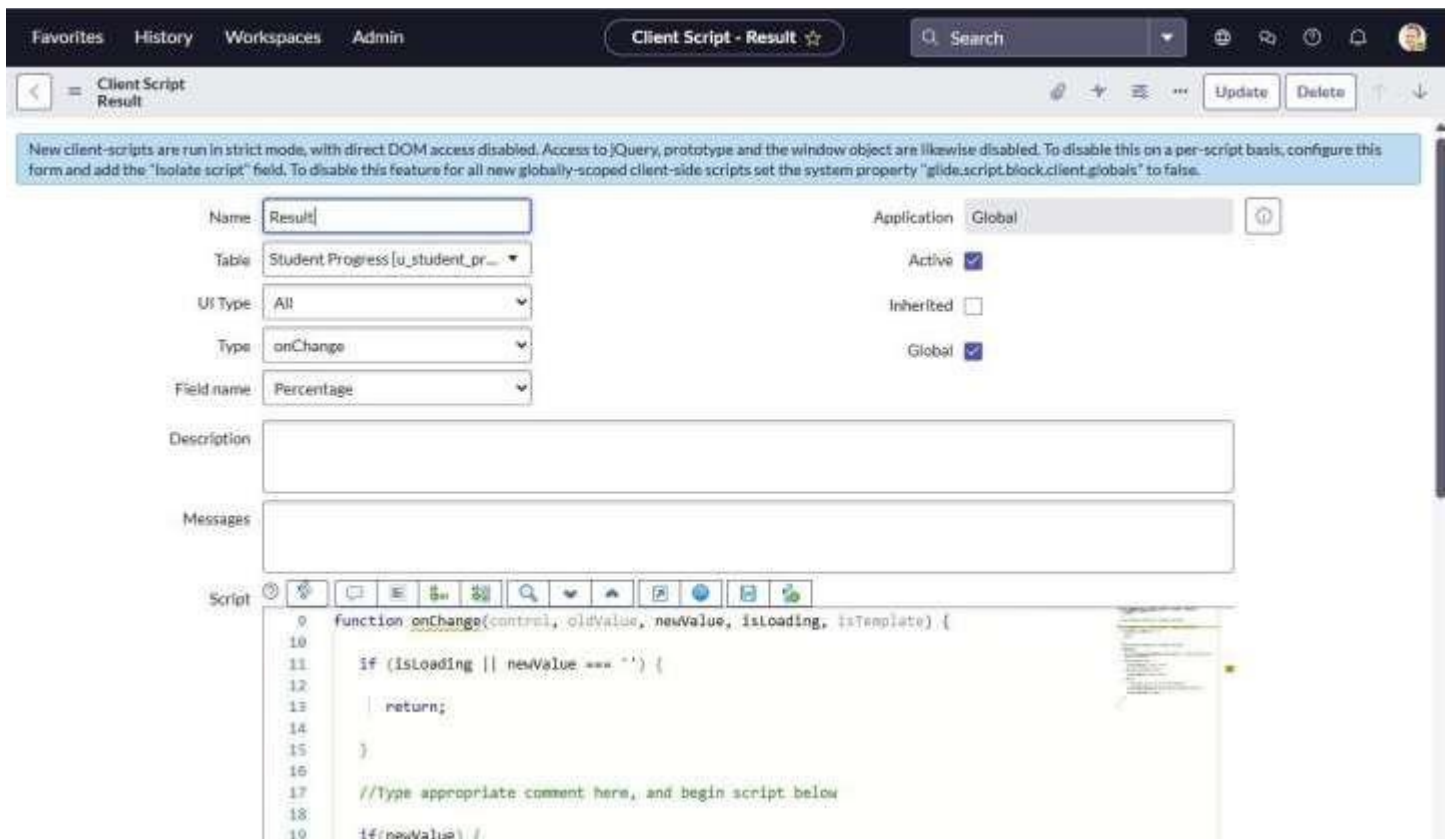
```

Activity-5: Result Client Scripts for Student progress

Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
  
    if (isLoading || newValue === '') {  
  
        return;  
  
    }  
  
    //Type appropriate comment here, and begin script below  
    if(newValue) {  
        var a = parseInt(g_form.getValue('u_percentage')); // Convert the value to an integer  
for comparison  
        if(a >= 0 && a <= 59){  
            g_form.setValue('u_result','Fail');  
        }  
        else if(a >= 60 && a <= 100) {  
            g_form.setValue('u_result','Pass');  
        }  
        else {  
            // Handle the case if a is out of range (optional)  
            g_form.addErrorMessage('Percentage should be between 0 and 100.');
```



Activity-6: Percentage Client Scripts for Student progress Table

1. Fill the Details as given.
2. Write the Code as given, Enable Isolate script and Save.

Function onChange(control, oldValue, newValue, isLoading, isTemplate) {

 If (isLoading || newValue === '') {

 Return;

 }

 //Type appropriate comment here, and begin script below

 Var Total = g_form.getValue('u_total');

 Var Percentage = (Total/600)*100;

```
G_form.setValue('u_percentage',Percentage+'%');
```

```
}
```

Client Script - Percentage

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: Percentage

Table: Student Progress [u_student_pr...

UI Type: All

Type: onChange

Field name: Total

Application: Global

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script

```
17
18 //Type appropriate comment here, and begin script below
19
20 var Total = g_form.getValue('u_total');
21
22 var Percentage = (Total/500)*100;
23
24 g_form.setValue('u_percentage',Percentage+'%');
25
26 }
27
28
```