



**ANJALAI AMMAL MAHALINGAM ENGINEERING COLLEGE**

**DEPARTMENT OF COMPUTER SCIENCE AND  
ENGINEERING**

**EDUCATIONAL ORGANIZATIONAL USING SERVICENOW**

**Team ID : NM2025TMID04882**

**Team Size : 4**

**Team Leader : Harini J**

**Team member : Iswarya K**

**Team member : Kaviya K**

**Team member : Lakshaya S A**

## **PROBLEM STATEMENT:**

Educational institutions often struggle with managing large volumes of student and staff information through manual or disconnected systems. Tasks such as admissions, student record management, and progress tracking become time-consuming and prone to errors. The lack of a unified digital platform leads to inefficiency, communication gaps, and delays in decision-making. This project addresses these challenges by developing an **Educational Management System** on the **ServiceNow platform** to automate administrative workflows, centralize data, and improve institutional efficiency.

## **OBJECTIVES:**

To design and implement a centralized system for managing student and teacher information using ServiceNow.

1. To automate key processes such as admissions, attendance, and progress monitoring.
2. To enable real-time access and updates of academic and administrative data.
3. To improve coordination among departments and reduce manual workloads.
4. To generate reports and analytics for better academic and administrative decision-making.
5. To enhance transparency and user experience for students, faculty, and administrators.

**Skills and Technologies Used:**

**Platform:** ServiceNow

**Skills:**

- ServiceNow Administration and Configuration
- Form and Table Design
- Workflow and Flow Designer
- Reporting and Dashboard Creation

**Task Initiation:**

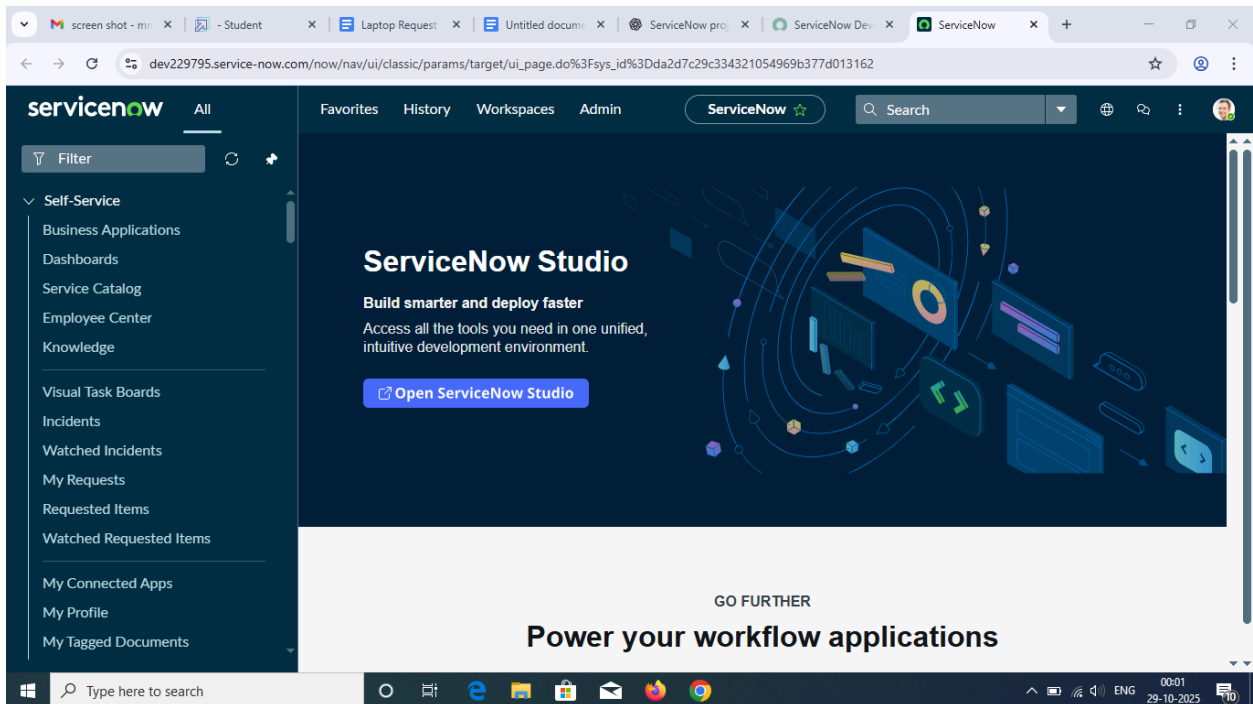
The **Educational Organization Using ServiceNow** project begins with the initiation phase, where the project scope, goals, and resources are clearly defined. The purpose of this phase is to establish a clear understanding of what the project aims to achieve and to prepare the foundation for successful implementation.

**Features:**

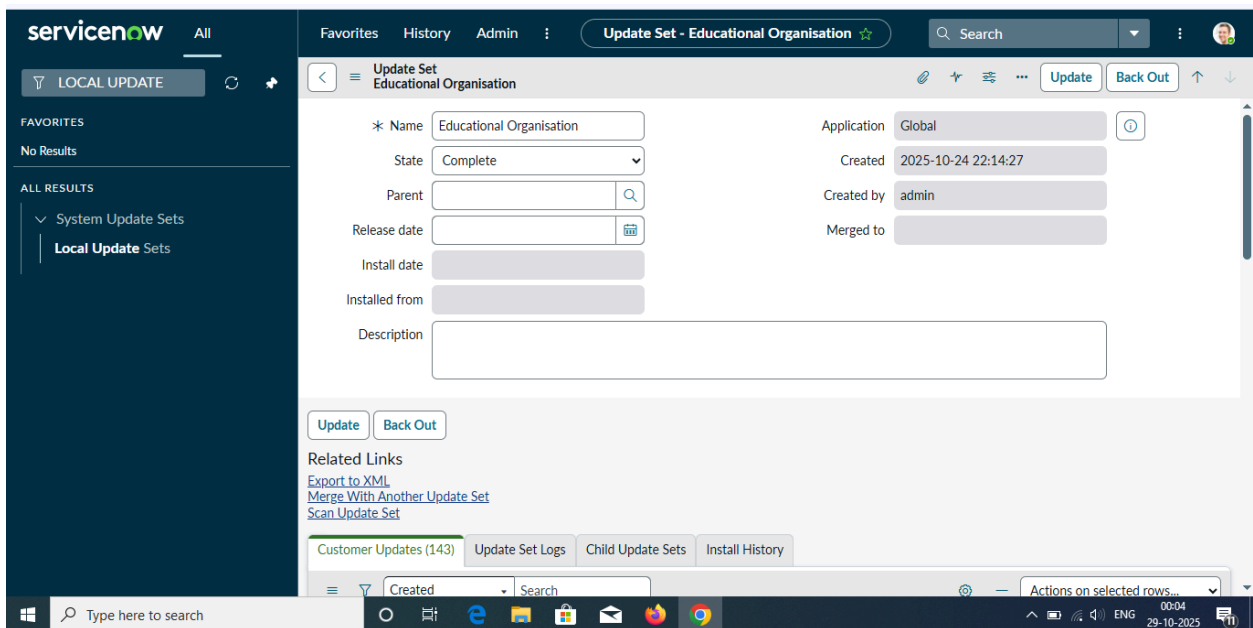
1. Centralized management of student and teacher information on a single ServiceNow platform.
2. Automated workflows for admissions, attendance, and progress tracking.
3. Real-time data access with reporting and dashboard analytics.
4. User-friendly interface for administrators, faculty, and students to improve efficiency.

## STEPS:

### STEP 1: Setting up ServiceNow Instance



### STEP 2: CREATING AN UPDATE SET



## STEP 3: CREATING A TABLE

### SALESFORCE:

The screenshot shows the ServiceNow interface for configuring a table named 'Salesforce'. The left sidebar contains a navigation menu with categories like 'Platform Analytics Administration', 'Data Sources', 'Protected Tables', 'Log Protection', 'Protected Table Log', 'System Archiving', 'Archive Tables', 'System Clone', 'Clone Definition', 'Exclude Tables', 'System Definition', 'Tables', 'Tables & Columns', 'Table Rotations', 'Decision Tables', 'Remote Tables', 'Definitions', and 'System Diagnostics'. The main content area is titled 'Table - Salesforce' and includes a description: '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 and records to manage data and processes. [More Info](#)'. Below this, there are fields for 'Label' (Salesforce), 'Name' (u\_salesforce), 'Application' (Global), and 'Remote Table'. The 'Columns' tab is selected, showing a list of 'Table Columns' with a search bar. The 'Dictionary Entries' table lists various columns with their types, references, max lengths, default values, and display settings.

Column label	Type	Reference	Max length	Default value	Display
Admin Number	Integer	(empty)	40		true
Class	System Class Name	(empty)	80	javascript:current.getTableNames();	false
Created	Date/Time	(empty)	40		false
Created	Date/Time	(empty)	40	40	false
Created by	String	(empty)	40		false
Created by	String	(empty)	40	40	false
Grade	Choice	(empty)	40		false
Number	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Sys ID	Integer	(empty)	40	32	false
Sys ID	Sys ID (GUID)	(empty)	32		false
Updated	Date/Time	(empty)	40		false
Updated by	String	(empty)	40		false
Updated by	String	(empty)	40	40	false
Updates	Integer	(empty)	40		false
Updates	Integer	(empty)	40	40	false

### ADMISSION:

The screenshot shows the ServiceNow interface for configuring a table named 'Admission'. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Table - Admission' and includes a description: '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 and records to manage data and processes. [More Info](#)'. Below this, there are fields for 'Label' (Admission), 'Name' (u\_admission), 'Application' (Global), 'Remote Table', and 'Extends table' (Salesforce). The 'Columns' tab is selected, showing a list of 'Table Columns' with a search bar. The 'Dictionary Entries' table lists various columns with their types, references, max lengths, default values, and display settings.

Column label	Type	Reference	Max length	Default value	Display
Admin Number	Integer	(empty)	40		true
Admin Status	Choice	(empty)	40	40	false
Area	String	(empty)	40	40	false
City	String	(empty)	40	40	false
Class	System Class Name	(empty)	80	javascript:current.getTableNames();	false

## STUDENT PROGRESS:

The screenshot shows the ServiceNow interface for configuring a table named 'Student Progress'. The left sidebar contains a navigation menu with categories like System Archiving, System Clone, System Definition, and Tables. The main area displays the table configuration details.

**Table Configuration:**

- Label: Student Progress
- Name: u\_student\_progress
- Application: Global
- Remote Table: (empty)

**Columns Tab:**

Table Columns: Column label | Search | 1 to 15 of 15 | New

Column label	Type	Reference	Max length	Default value	Display
Created	Date/Time	(empty)	40		false
Created by	String	(empty)	40		false
English	String	(empty)	40	40	false
Hindi	String	(empty)	40	40	false
Maths	String	(empty)	40	40	false
Percentage	String	(empty)	40	40	false
Result	String	(empty)	40	40	false

## STEP 4: FORM LAYOUT

The screenshot shows the ServiceNow Form Design interface for the 'Student Progress' table. The left sidebar contains a navigation menu with categories like Platform Analytics Administration, Data Sources, Protected Tables, System Archiving, System Clone, System Definition, and Tables. The main area displays the form layout configuration.

**Form Design Configuration:**

- View name: Default view
- Section: Student Progress

**Available Fields:**

- Created
- Created by
- Updated
- Updated by
- Updates
- begin\_split -
- split -
- end\_split -
- \* Annotation
- \* Chart
- Activities (filtered)
- Contextual Search Results
- Ratings
- Attachments

**Selected Fields:**

- begin\_split -
- English
- Hindi
- Result
- Maths
- Total
- split -
- Percentage
- Science
- Social
- Telugu
- end\_split -

**Create new field:**

- Name: (empty)
- Type: String
- Field length: Small (40)

**Related Links:**

- Show versions

## STEP 5: FORM DESIGN

The screenshot shows the ServiceNow Form Designer interface for the 'Admission' table. The left sidebar contains a list of fields and formatters. The main area displays the form layout with fields like Created(u\_created), Updated by(u\_updated\_by), Sys ID, Updates(u\_updates), Admin Status, Purpose of join, Number, School Area, Created by, Grade, Fee, Updated by, School, and Pincode. The bottom status bar shows 'BAN - WI' and 'Game score'.

## STEP 6: NUMBER MAINTAINANCE

### Creating Number Maintenance for Admin Number

servicenow

All

Favorites

History

Workspaces

Admin

Table - Salesforce ☆

Search

Update

Delete

Delete All Records

number

Favorites

No Results

ALL RESULTS

System Definition

Number Maintenance

1 to 15 of 15

New

	Reference	Max length	Default value	Display
Created by	String	(empty)	80	javascript:current.getTableName();
Created	Date/Time	(empty)	40	false
Sys ID	Sys ID (GUID)	(empty)	32	false
Updates	Integer	(empty)	40	false
Updated by	String	(empty)	40	false

Fill the details >> Submit.

servicenow All Favorites History Workspaces Admin Number - SAL

Number SAL

\* Table Salesforce

Prefix SAL

\* Number 1,000

Application Global

Number of digits 7

Update Delete

Related Links  
[Show Counter](#)

## STEP 7: Process Flow

### Creating Process Flow for Admission Table

- All >> Process Flow>> New.
- Fill the Details as given Below

Flow Formatter New

\* Table Admission [u\_admission]

\* Name New

Application Global

\* Label New

Order

Active ☒

Condition [Add Filter Condition](#) [Add "OR" Clause](#)

Admin Status is New

Description

Update Delete

Save  
Insert  
Insert and Stay  
Analyze Access  
Show File Properties  
Move to Application...  
Show Latest Update  
Configure  
Export  
View  
Create Favorite  
Copy URL  
Copy sys\_id  
Show XML  
History  
Reload form

- Right Click on toggle and click on the save .
- Replace the Name and Label as below and click on Insert on stay.

Flow Formatter In progress

\* Table Admission [u\_admission]

\* Name In progress

Application Global

\* Label In progress

Order

Active ☒

Condition [Add Filter Condition](#) [Add "OR" Clause](#)

Admin Status is Join in progress

Description

Update Delete

Save  
Insert  
Insert and Stay  
Analyze Access  
Show File Properties  
Move to Application...  
Show Latest Update  
Configure  
Export  
View  
Create Favorite  
Copy URL  
Copy sys\_id  
Show XML  
History  
Reload form

AND OR



- Replace the Name and Label in order and click on Insert on stay.  
Joined >> Rejected >> Rejoined >> Closed >> Cancelled.
- Order should be New >> InProgress >> Joined >> Rejected >> Rejoined >> Closed >> Cancelled.

## STEP 8: Client Script

### 1. Creating “Auto populate” Client Scripts for Admission Table

- All >> Client Scripts >> New.
- Fill the Details as given.

This form has annotations - click ⓘ to toggle them - ([click here](#) to never show this again)

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

```

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

```

- Write the Code as below, Enable Isolate script and Save.

```

function onChange(control, oldValue, newValue, isLoading, isTemplate) {
  if (isLoading || newValue === '') {
    return;
  }
}

```

### 2. Creating “Pincode Update” Client Scripts for Admission Table

- Fill the Details as given.

Client Script  
Pincode Update

This form has annotations - click ⓘ to toggle them - (click [here](#) to never show this again)

Name: Pincode Update

Table: Admission [u\_admission]

UI Type: Desktop

Type: onChange

Field name: Pincode

Application: Global ⓘ

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script

```

1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2   if (isLoading || newValue === '') {
3     return;
4   }
5   var a = g_form.getValue('u_pincode');
6   if(a == '509358')
7   {
8     g_form.setValue('u_mandal', 'kadthal');
9     g_form.setValue('u_city', 'kadthal');

```

- Write the Code as below, 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

```
}
```

### 3.Creating “Disable Fields” Client Scripts for Student progress Table

- Fill the Details as given.

The screenshot shows a web application configuration form titled 'Client Script' with the subtitle 'Disable Fields'. At the top, there is a blue notification bar that reads: 'This form has annotations - click ⓘ to toggle them - (click [here](#) to never show this again)'. The form contains several sections: 'Name' (Disable Fields), 'Table' (Student Progress [u\_student\_progress]), 'UI Type' (All), and 'Type' (onLoad). To the right, there are checkboxes for 'Active' (checked), 'Inherited' (unchecked), and 'Global' (checked). Below these are empty text areas for 'Description' and 'Messages'. At the bottom, there is a 'Script' section with a code editor. The code in the editor is as follows:

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

- Write the Code as below, 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);
```

```
}
```

### 4.Creating “Total Update” Client Scripts for Student progress Table

- Fill the Details as given.

The screenshot shows the ServiceNow interface for configuring a Client Script. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. The main header is 'Client Script - Total Up...'. Below this, a message states 'You are editing a record in the Global application (cancel)'. The configuration form includes the following fields:

- Name:** Total Update
- Table:** Student Progress [u\_student\_progress]
- UI Type:** All
- Type:** onChange
- Field name:** Social
- Application:** Global
- Active:** ☒
- Inherited:** ☐
- Global:** ☒
- Description:** (Empty text area)
- Messages:** (Empty text area)
- 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   if (newValue){
8     var a = parseInt(g_form.getValue('u_telugu'));
9     var b = parseInt(g_form.getValue('u_hindi'));
10    var c = parseInt(g_form.getValue('u_english'));
11    var d = parseInt(g_form.getValue('u_maths'));
12    var e = parseInt(g_form.getValue('u_science'));
13    var f = parseInt(g_form.getValue('u_social'));
14    var Total = parseInt(a+b+c+d+e+f);
15    g_form.setValue('u_total', Total);
16  }
17 }

```
- Isolate script:** ☒

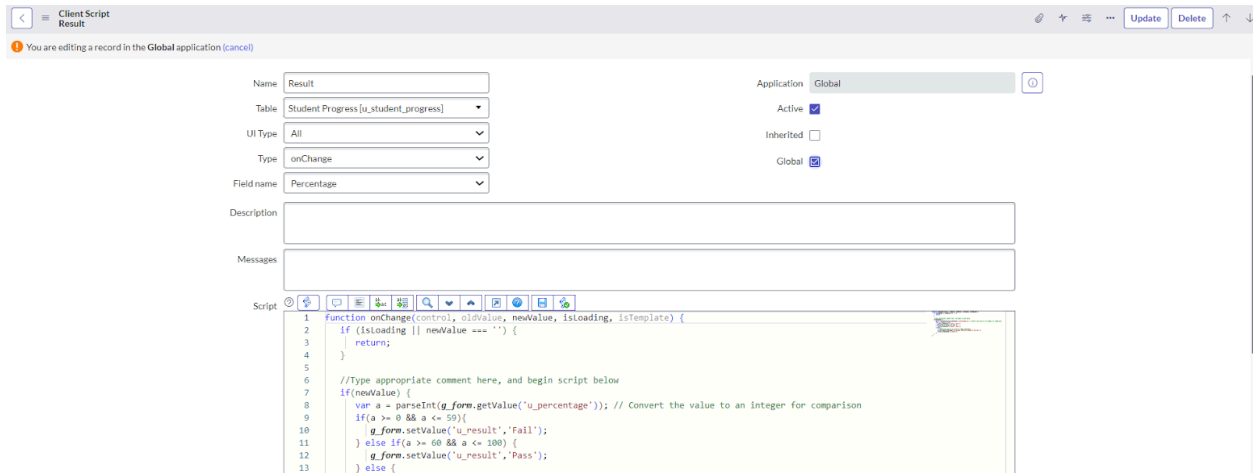
- Write the Code as below, Enable Isolate script and Save.

```
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);
}
}
```

## Creating “Result” Client Scripts for Student progress Table

- Fill the Details as given.



- Write the Code as below, 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!');
      g_form.clearValue('u_result');
    }
  }
}

```

## 5.Creating “Percentage” Client Scripts for Student progress Table

- Fill the Details as given.

The screenshot shows the ServiceNow Client Script configuration interface. The top bar indicates 'Client Script Percentage' and 'Global application'. A warning message states 'You are editing a record in the Global application (cancel)'. The configuration fields are as follows:

- Name: Percentage
- Table: Student Progress [u\_student\_progress]
- UI Type: All
- Type: onChange
- Field name: Total
- Application: Global
- Active: ☒
- Inherited: ☐
- Global: ☒

Description and Messages fields are empty. The Script editor contains the following code:

```
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   var Total = g_form.getValue('u_total');  
8   var Percentage = (Total/600)*100;  
9   g_form.setValue('u_percentage',Percentage+'%');  
10 }
```

The 'Isolate script' checkbox is checked. At the bottom, there are 'Update' and 'Delete' buttons.

- Write the Code as below, 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+'%');  
}
```

Result:

The ServiceNow-based Educational Management System successfully demonstrates the automation of student admission, academic progress, and information management. Data is centralized, workflows are simplified, and administrators can easily generate reports and monitor academic progress.



