SERVICENOW ADMINISTRATOR

Project Title:

Educational Organisations Using ServiceNow

Team Members:

- Team Leader: Rushitha Konangi (Designed overall project structure)

- Team Member: C Prathyusha (Created fields and reference relationships)

- Team Member: C Sai Prathyusha Reddy (Worked on result logic)

- Team Member: Guduru Saritha (Prepared documentation)

Project Overview

• Purpose:

The purpose of this project is to manage educational institution activities like student admission, student progress, and record management using ServiceNow.
It helps to store and track student details digitally instead of manual paperwork.

• Goals:

- To automate admission process
- To manage student academic progress
- To calculate total and percentage automatically
- To maintain student database securely

• Features:

- Admission form creation
- Student Progress form
- Auto population of data using Client Script
- Automatic total and percentage calculation
- Result generation (Pass/Fail)
- Status workflow (New → In Progress → Joined → Closed)
- Reports and List view

Architecture

• Platform Used:

ServiceNow (Cloud-based platform)

• Modules Used:

- Tables (Admission, Student Progress)

- Forms

- Client Scripts

- List View

- Workflow Status

- Reference Fields

• Database:

- Custom tables created in ServiceNow

- Admission table stores student admission details

- Student Progress table stores subject marks and result

Data is stored securely in ServiceNow database.

Setup Instructions

• Prerequisites:

- ServiceNow Developer Instance

- Admin access

- Basic knowledge of ServiceNow

• Installation Steps:

1. Login to ServiceNow Developer Instance

2. Create new custom table (Admission)

3. Create another table (Student Progress)

4. Design forms using Form Designer

5. Add fields like:

   ○ Student Name

   ○ Father Name

   ○ Mother Name

   ○ Grade

   ○ Marks fields

6. Create Client Script for:

- Auto population

  - Total calculation

  - Percentage calculation

  - Result generation

7. Test the forms

Folder Structure

Since this is ServiceNow based project:

- Tables

  - u_admission
  - u_student_progress
  - u_Salesforce

- Client Scripts
- Forms
- Workflows
- List Views
- Records

Running the Application

1. Login to ServiceNow

2. Open Admission module

3. Create New Record

4. Fill student details

5. Submit the form

6. Open Student Progress

7. Enter marks

8. Total and Percentage calculated automatically

API Documentation

This project mainly works inside ServiceNow platform.

Operations supported:

- Create Admission record
- Update Admission status
- Create Student Progress record
- Retrieve student details using reference field

Methods:

- Insert
- Update
- Delete

- View

SCRIPTING:

Admission table:
a.Autopopulate:

```
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);
}
```

b.pincode:

```
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');
}
```

Student progress Table:

a.Total update

```
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);
}
}
```

b.Disable fields:

```
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);
}
```

c. Result:

```
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');
    }
  }
}
```

d. percentage:

```
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+'%');
}
```

Future Enhancements

- Add report dashboard

- Add student attendance module

- Add fee payment tracking

- Add parent login portal

- Improve UI design

- Add automatic grade calculation

Authentication

- ServiceNow login authentication
- Role-based access control
- Admin role can create and modify records
- Users have limited access

Security is handled by ServiceNow platform.

User Interface

◆ **Admission Form**

- Admin Number auto-generated
- Student details form
- Grade selection
- Workflow status bar
- Comments section

◆ **Student Progress Form**

- Subject marks entry (Hindi, English, Telugu, Science, Maths, Social)
- Total auto-calculated
- Percentage auto-calculated
- Result automatically shown (Pass/Fail)

◆ **List View**

- Displays all student records

- Shows admission date
- Shows grade
- Shows parent details

Testing

Testing done by:

- Entering different marks

- Checking percentage calculation

- Verifying pass/fail logic

- Checking auto population functionality

- Validating marks between 0 and 100

All features tested successfully.

 Known Issues

- If marks exceed 100, error message is displayed

- Percentage shows long decimal value

- Needs rounding improvement

Results and Outputs:

A.SALESFORCE TABLE

B.ADMISSION TABLE

C.STUDENT PROGRESS TABLE