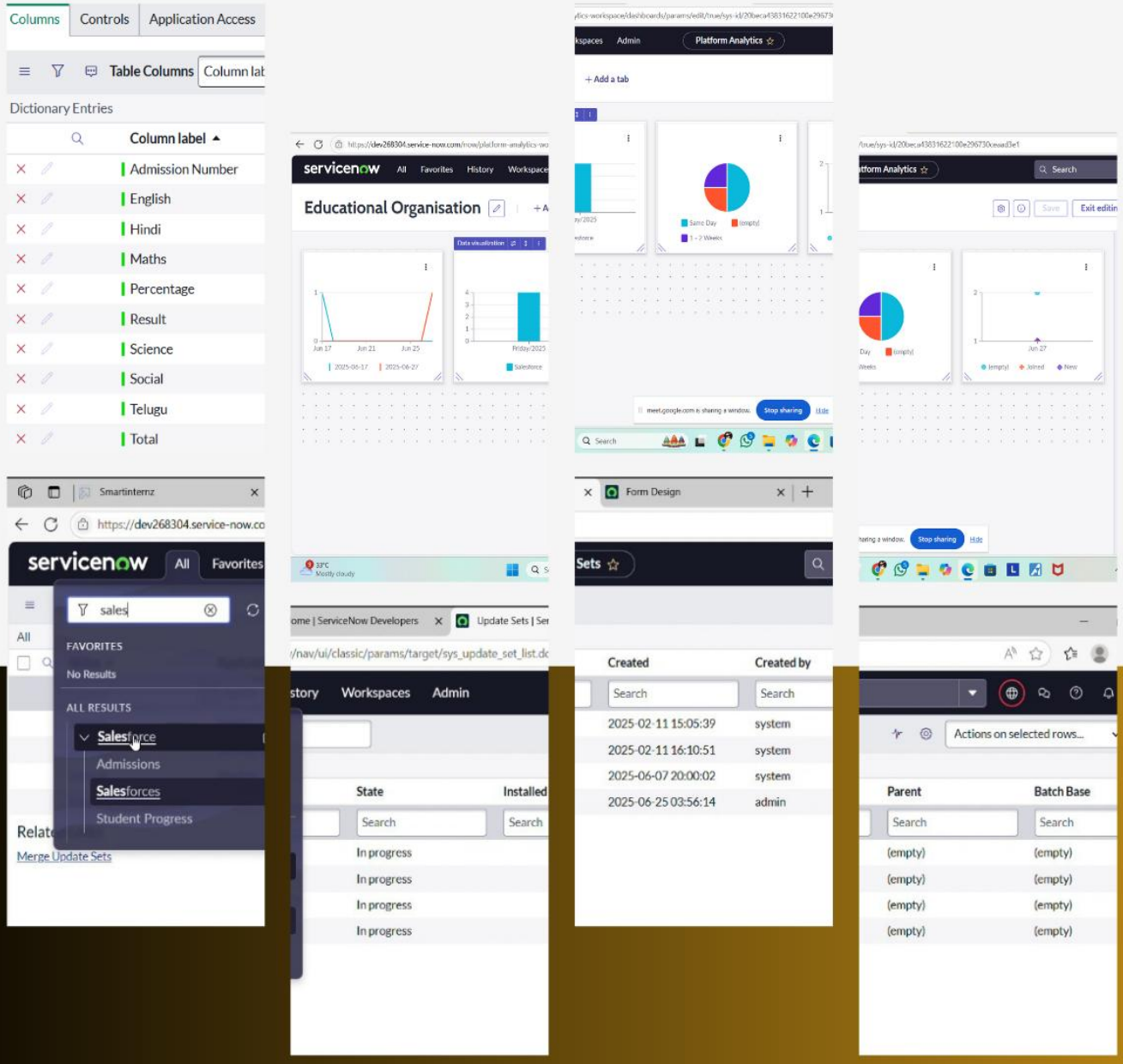


PROJECT REPORT

Educational Organisation Using ServiceNow

Team Members:

- **A. HARSHA VALLI**
- **SAYYAD AAFRIN**
- **G. PRASANNA**
- **CH. VENKAT TEJA**

TEAM ID: LTVIP2025TMID20393

1. INTRODUCTION

Project Title: Educational Organisation Using ServiceNow

Team Id: LTVIP2025TMID20393

Team Members and Roles:

- **Sayyad Aafrin – Initial Phase Developer**
  Led the early development stages,form design, and foundational setup.
- **A. Harsha Valli – Tester & Dashboard Developer**
  Handled system testing and contributed to dashboard development and notification logic.
- **Ch Venkat Teja – Documentation & Phase-wise Documentation**
  Managed structured documentation throughout each development phase.

1.1 **Prasanna – Documentation & Final Report Documentation**
Compiled and formatted the final report and maintained documentation consistency.

1.1 Project Overview:

The "Educational Organisation Using ServiceNow" project is a cloud-based digital solution designed to modernize and automate the core academic and administrative processes of educational institutions. Built on the ServiceNow platform, the system integrates functionalities such as student registration, admission workflows, academic progress tracking, and real-time notifications into a single, user-friendly interface.

The platform supports role-based access for students, teachers, administrators, and parents, ensuring that each stakeholder interacts with the system in a secure and relevant manner. By leveraging ServiceNow's no-code and low-code capabilities, the project aims to reduce manual effort, eliminate data silos, and enhance decision-making through dashboards and analytics.

1.2 Purpose:

The primary goal of this project is to address the inefficiencies and fragmentation commonly found in traditional educational management systems. Manual data entry, lack of real-time updates, and poor communication channels often hinder the smooth functioning of academic institutions.

This project seeks to:

- Digitize the student lifecycle from registration to academic tracking.
- Automate workflows for admissions and notifications.

- Provide real-time dashboards for administrative insights.
- Ensure data security and integrity through role-based access and encryption.
- Enhance user experience with a centralized, intuitive interface.

2. IDEATION PHASE

2.1 Problem Statement:

Educational institutions often struggle with fragmented workflows, manual data entry, and inefficient communication. These issues lead to delays in student registration, inconsistent academic tracking, and limited visibility for administrators. The lack of a centralized system results in poor data management and stakeholder dissatisfaction.

2.2 Empathy Map Canvas:

To understand the needs of users (students, teachers, admins, and parents), the team created an empathy map:

- **Students** want easy registration, access to marks, and timely updates.
- **Teachers** need a simple way to enter and track academic progress.
- **Admins** require dashboards for admissions and performance insights.
- **Parents** expect transparency and notifications about their child's progress

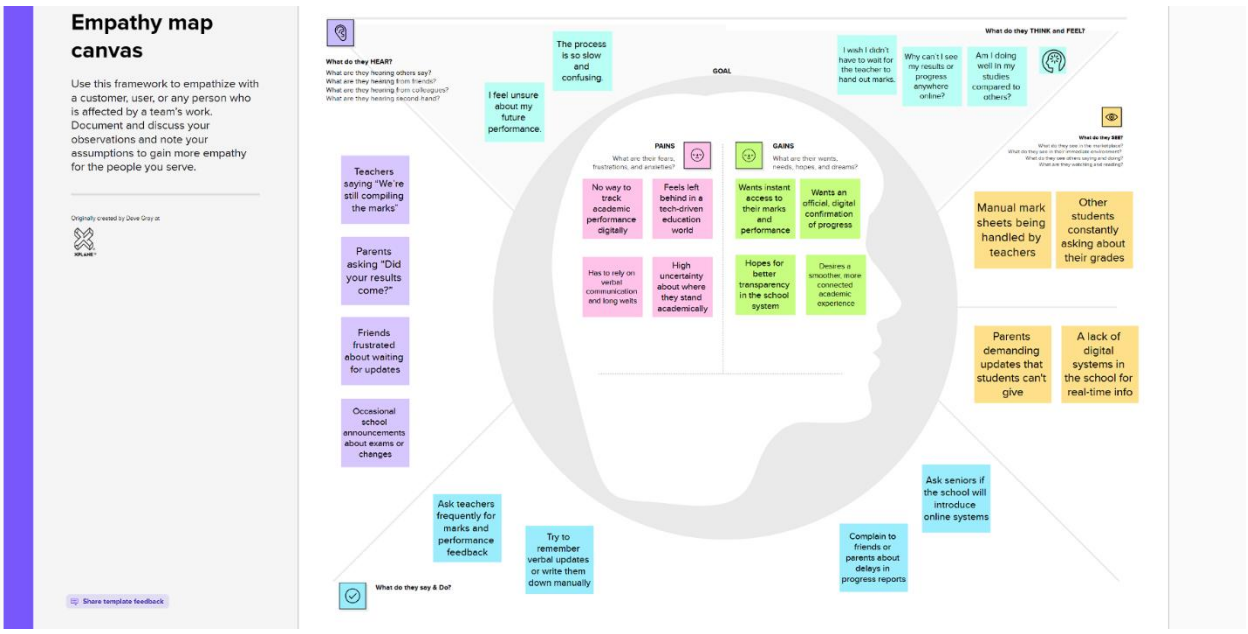


FIG. Empathy Map

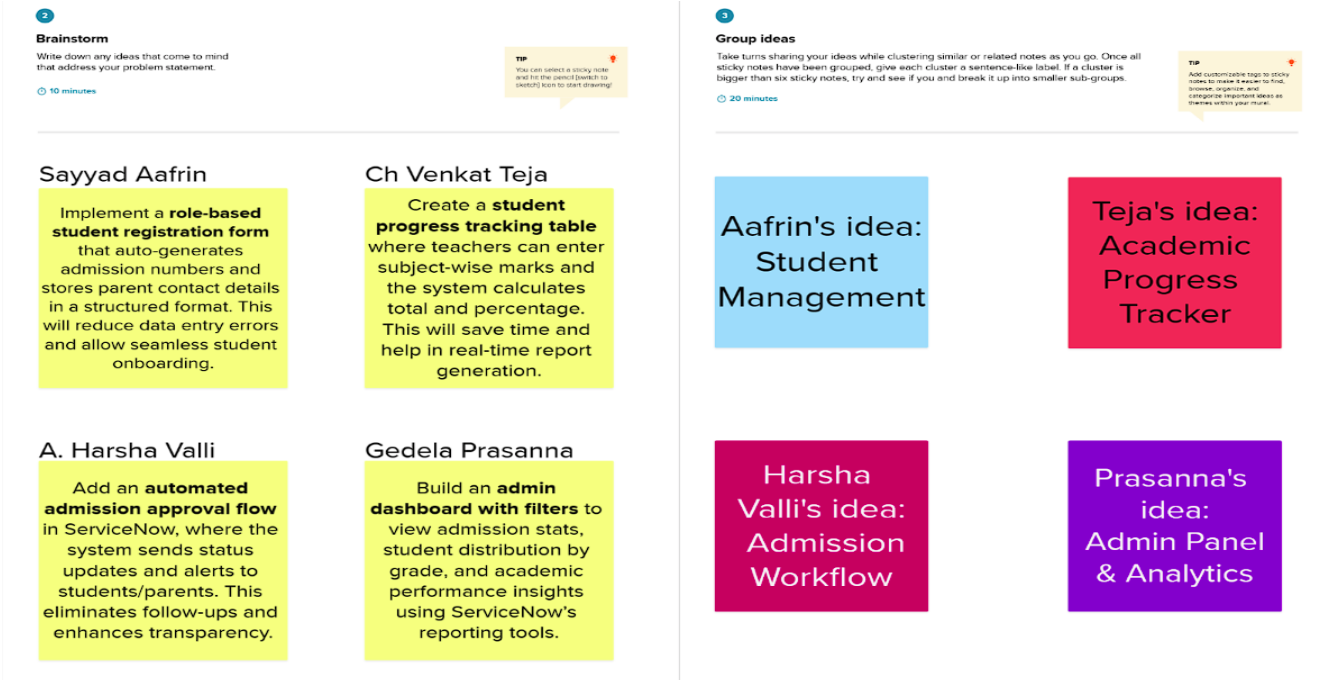
This helped identify pain points and expectations, guiding the solution design.

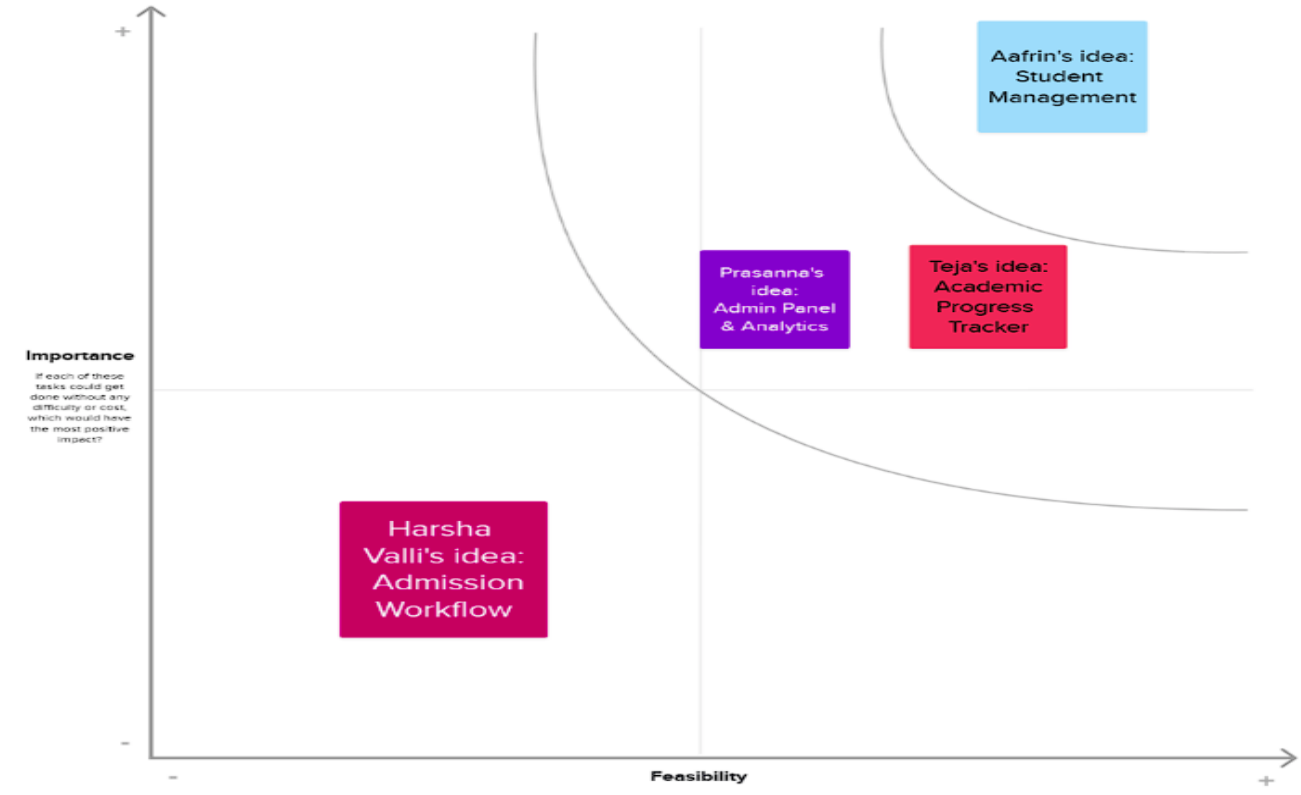
2.3 Brainstorming:

The team conducted collaborative brainstorming sessions to explore potential solutions. Ideas were grouped and prioritized based on feasibility and impact. Out-of-the-box thinking was encouraged, and the final concept selected was:

> ***A ServiceNow-based educational management system*** *that automates registration, admission workflows, academic tracking, and notifications.*

This idea was chosen for its scalability, security, and alignment with institutional needs.





3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map:

The customer journey outlines how different users interact with the system:

- **Students** begin by registering through a portal, submitting personal and academic details.
- **Admins** review admission forms, verify data, and update status.
- **Teachers** input academic marks and track student progress.
- **Students and Parents** receive notifications and view performance dashboards.
- **Admins** use dashboards to analyze trends and make decisions.

This journey ensures that each stakeholder has a clear, role-specific path through the system.

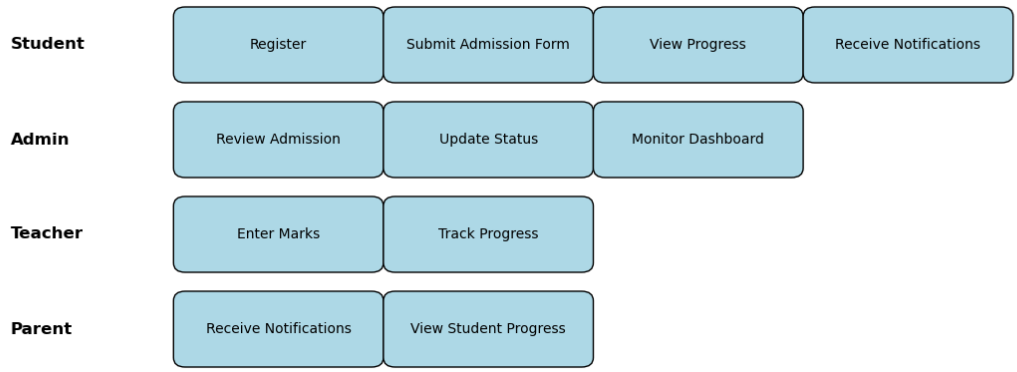


FIG.Customer Journey Map

3.2 Solution Requirements:

The system must fulfill the following functional and non-functional requirements:

Functional Requirements:

- Student registration and profile creation
- Admission form submission and status tracking
- Academic progress entry and result generation
- Role-based access for Admin, Teacher, Student, and Parent
- Notification system for updates and alerts

Non-Functional Requirements:

- Secure data handling with ACLs and encryption
- Scalable architecture for future expansion
- High availability and performance
- Intuitive user interface for all roles

3.3 Data Flow Diagram (DFD):

The DFD illustrates how data moves through the system:

- **External Entities:** Student, Parent, Teacher, Administrator
- **Processes:** Registration, Admission Workflow, Academic Tracking, Dashboard Reporting, Notification System
- **Data Stores:** Student Profile Table, Admission Records, Academic Marks Table

Flow Summary:

- Student submits registration → Data stored in profile table
- Admin processes admission → Status updated → Notification sent
- Teacher enters marks → Calculations performed → Stored in marks table
- Admin views dashboard → Filters data → Makes decisions

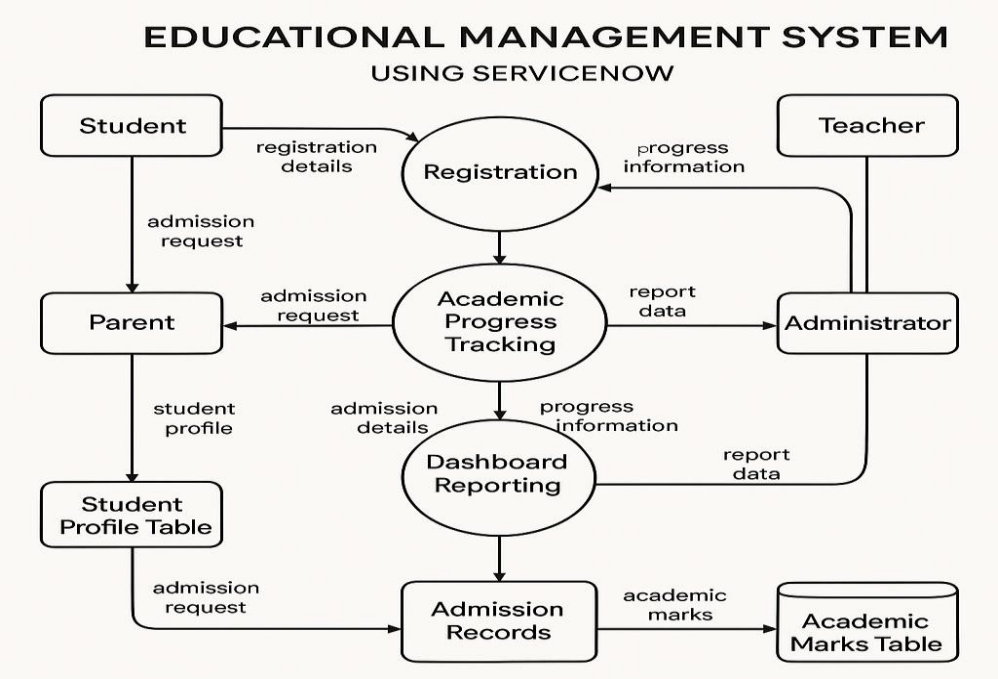


FIG. Data Flow Diagram

3.4 Technology Stack:

The system is built using the following technologies:

Component	Technology Used	Description
UI	ServiceNow UI Pages, JavaScript, HTML	Web portal interface for students, teachers, and admins.
Application Logic	Flow Designer, Client Scripts	Handles workflows, academic calculations, and automation.
Notifications	ServiceNow Notification Module	Sends email/SMS alerts for status updates.
Database	ServiceNow Tables (CMDB, Custom Tables)	Stores structured data like profiles, marks, and admissions.
Cloud Hosting	ServiceNow SaaS	Provides scalable and secure cloud infrastructure.

Characteristics:

- **Security:** Role-based ACLs, HTTPS, OAuth
- **Scalability:** Modular backend on ServiceNow
- **Performance:** Real-time updates, optimized workflows
- **Availability:** High via ServiceNow cloud infrastructure

4. PROJECT DESIGN

4.1 Problem-Solution Fit:

he proposed solution directly addresses the core challenges faced by educational institutions—manual data handling, fragmented workflows, and lack of real-time insights. By implementing a centralized system on the ServiceNow platform, the project ensures that each stakeholder can interact with the system efficiently and securely.

The solution fits the problem by:

- Automating student registration and admission workflows
- Enabling teachers to track academic progress with minimal effort
- Providing administrators with dashboards for data-driven decisions
- Ensuring timely communication through notifications

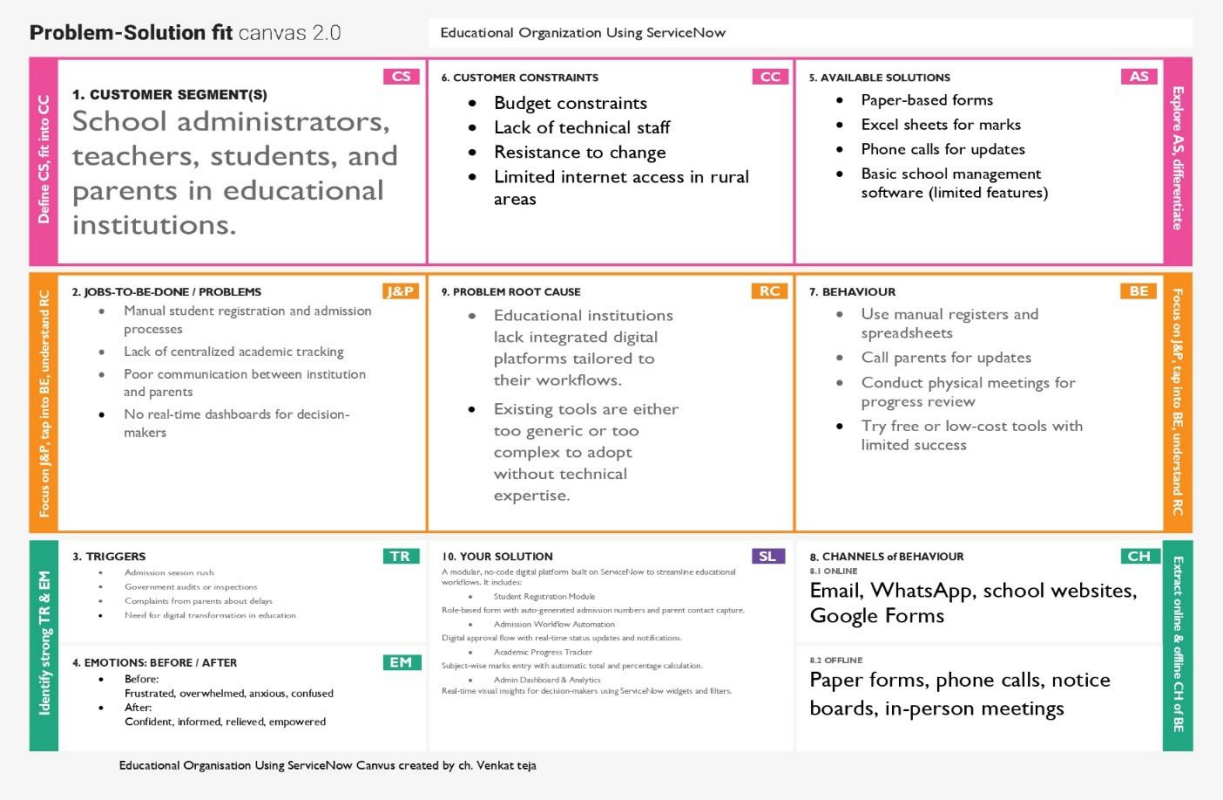


FIG. Problem-Solution Fit Canvas

4.2 Proposed Solution:

The system is designed as a modular, role-based platform hosted on ServiceNow. It includes:

- **Student Portal**: For registration, viewing progress, and receiving updates
- **Admin Dashboard**: For managing admissions, monitoring performance, and accessing analytics

- **Teacher Interface**: For entering marks and tracking student progress
- **Notification System**: For sending alerts via email/SMS

Each module is built using ServiceNow's no-code tools, ensuring rapid development and easy maintenance.

4.3 Solution Architecture:

The architecture is structured into four layers:

1. User Interface Layer

- Service Portal access for students, parents, teachers, and administrators
- Role-based access control for secure and relevant data visibility

2. Application Logic Layer

- Flow Designer for automating workflows and notifications
- Client Scripts for academic calculations (e.g., total marks, percentage)
- Business Rules for data validation and automation

3. Data Layer

- Custom tables for storing student profiles, academic records, and admission data
- Defined relationships between student and academic entities
- Secure data access using Access Control Lists (ACLs)

4. Reporting & Analytics Layer

- Dashboards for administrative insights
- Performance Analytics for trend analysis and visual data representation

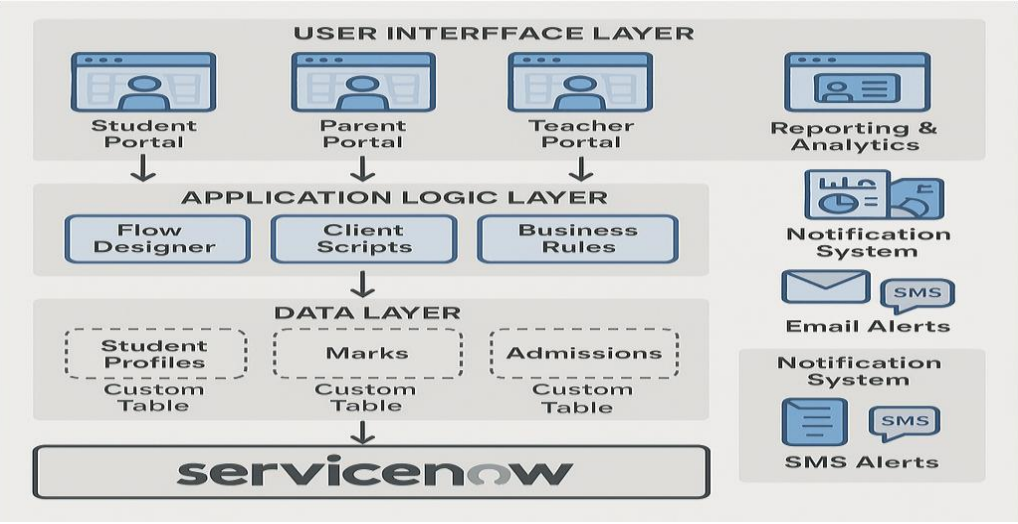


FIG. Architecture Diagram

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning:

The project followed an Agile development methodology, structured into four sprints. Each sprint focused on specific functional requirements (epics) and user stories, with clearly defined priorities and story points.

Sprint Velocity & Burndown Chart:

Sprint	Total Points	Duration	Start Date	End Date	Points Completed
Sprint-1	6	3 Days	18 June 2025	20 June 2025	6
Sprint-2	6	3 Days	21 June 2025	23 June 2025	6
Sprint-3	4	3 Days	24 June 2025	26 June 2025	4
Sprint-4	2	2 Days	27 June 2025	28 June 2025	2

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing:

The system underwent **User Acceptance Testing (UAT)** from **26 June 2025 to 28 June 2025** on the ServiceNow Developer Instance. The goal was to validate core functionalities and ensure role-based access control across different user types.

Testing Scope:

- Student registration and profile creation
- Admission form submission and status tracking
- Academic progress entry and result generation
- Role-based access for Admin, Teacher, and Student

Test Cases Summary:

Test Case ID	Scenario	Steps	Expected Result	Actual Result	Status
TC-001	Student Registration	Fill registration form and submit	Data saved and visible to admin	Successfully saved and listed	Pass
TC-002	Admission Data Entry by Admin	Admin logs in, adds	Data saved, status	Works as expected	Pass

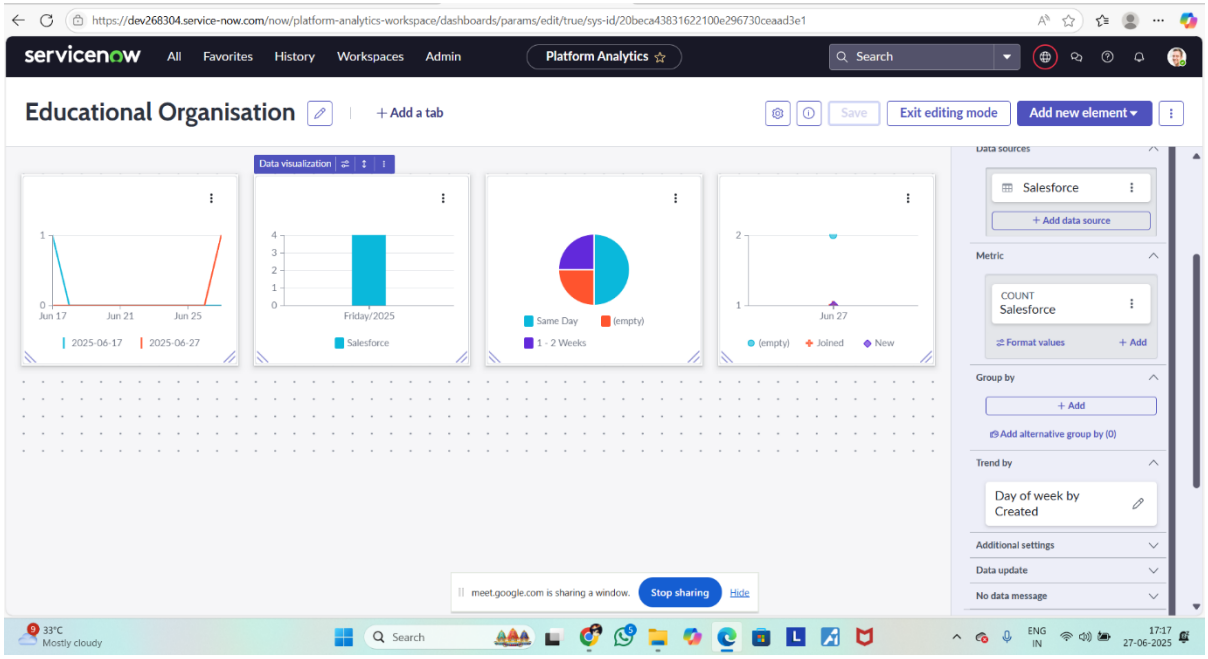
		student details	marked as 'Pending'		
TC-003	Teacher Updates Progress	Teacher enters marks and submits	Total and percentage auto-calculated and saved	Percentage calculated accurately	Pass
TC-004	Student Views Progress	Student logs in and views progress module	Marks, percentage, and result visible	Progress visible with correct data	Pass
TC-005	Access Control Validation	Student tries to access admin table	Access denied	Access restricted as expected	Pass

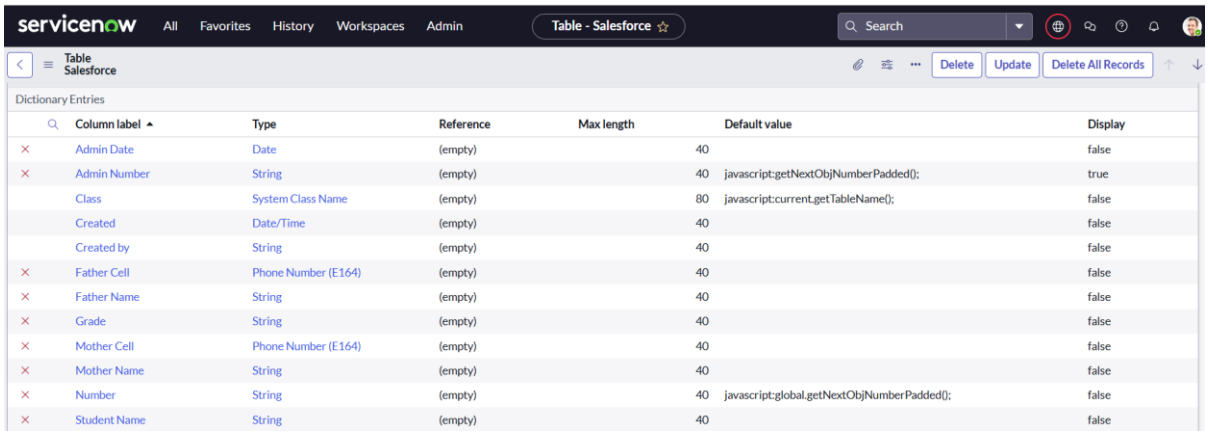
Bug Tracking:

Bug ID	Description	Steps to Reproduce	Severity	Status	Feedback
BG-001	Teacher unable to edit old marks	Login as teacher → Open submitted progress → Edit	Medium	Closed	Consider allowing limited edit rights
BG-002	Notification not triggering	Submit admission → Wait for email alert	Low	In Progress	Recheck flow configuration or remove feature

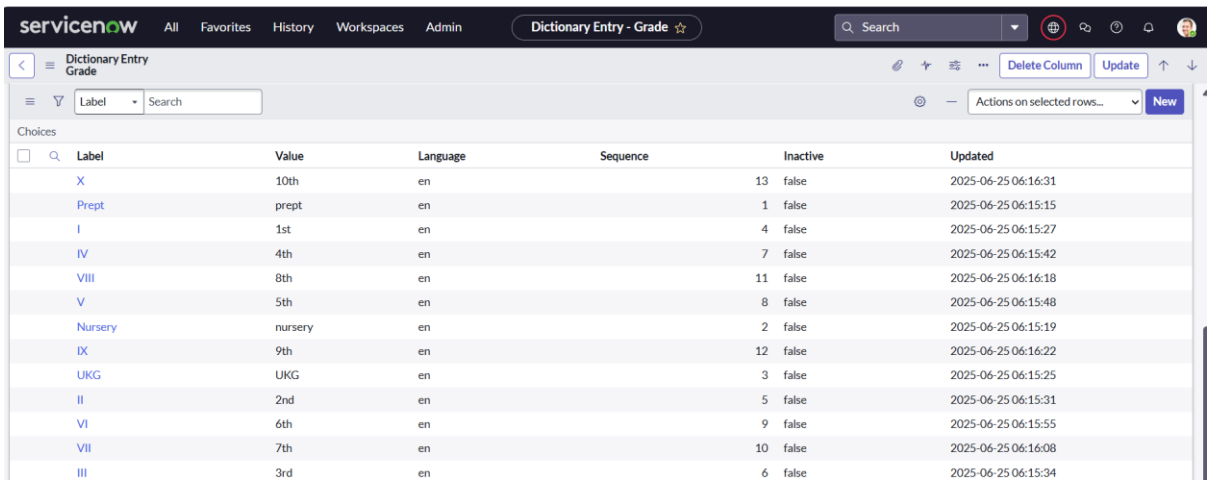
7. RESULTS

7.1 Output Screenshots:

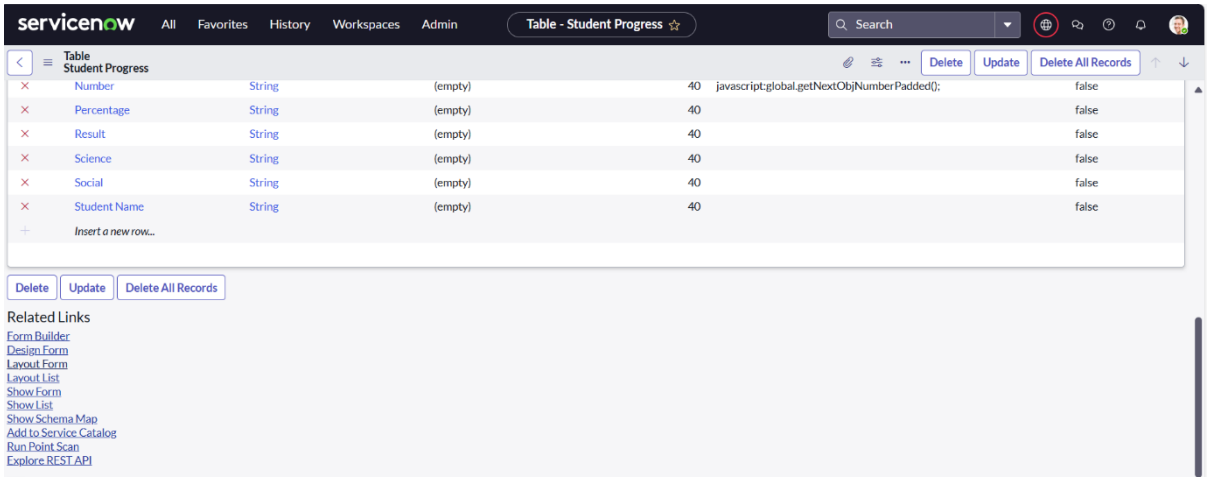


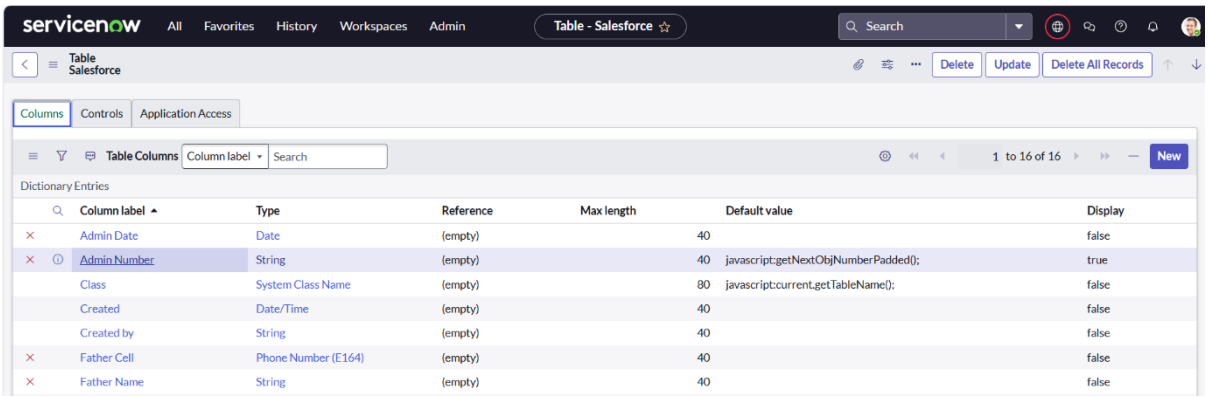


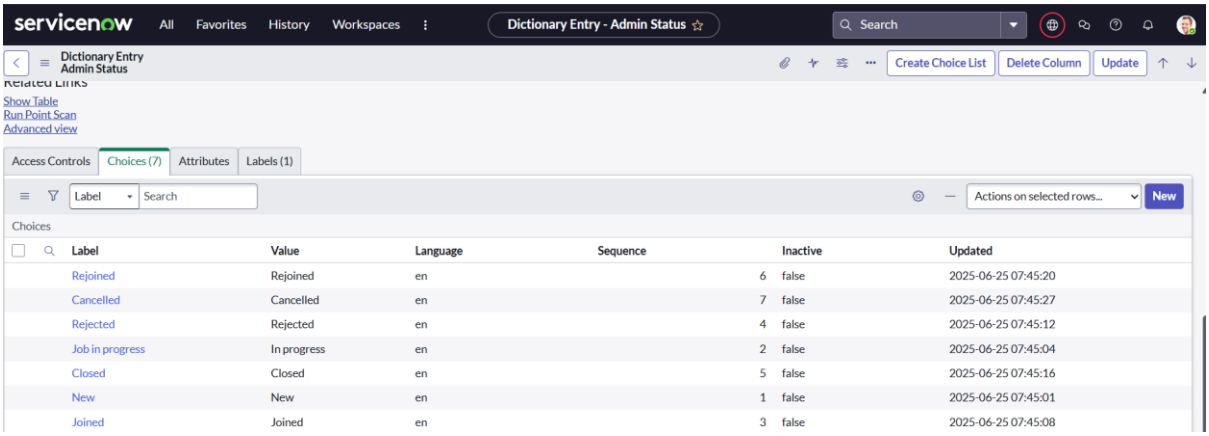
Column label	Type	Reference	Max length	Default value	Display
Admin Date	Date	(empty)	40		false
Admin Number	String	(empty)	40	javascript:getNextObjNumberPadded();	true
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
Created	Date/Time	(empty)	40		false
Created by	String	(empty)	40		false
Father Cell	Phone Number (E164)	(empty)	40		false
Father Name	String	(empty)	40		false
Grade	String	(empty)	40		false
Mother Cell	Phone Number (E164)	(empty)	40		false
Mother Name	String	(empty)	40		false
Number	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Student Name	String	(empty)	40		false

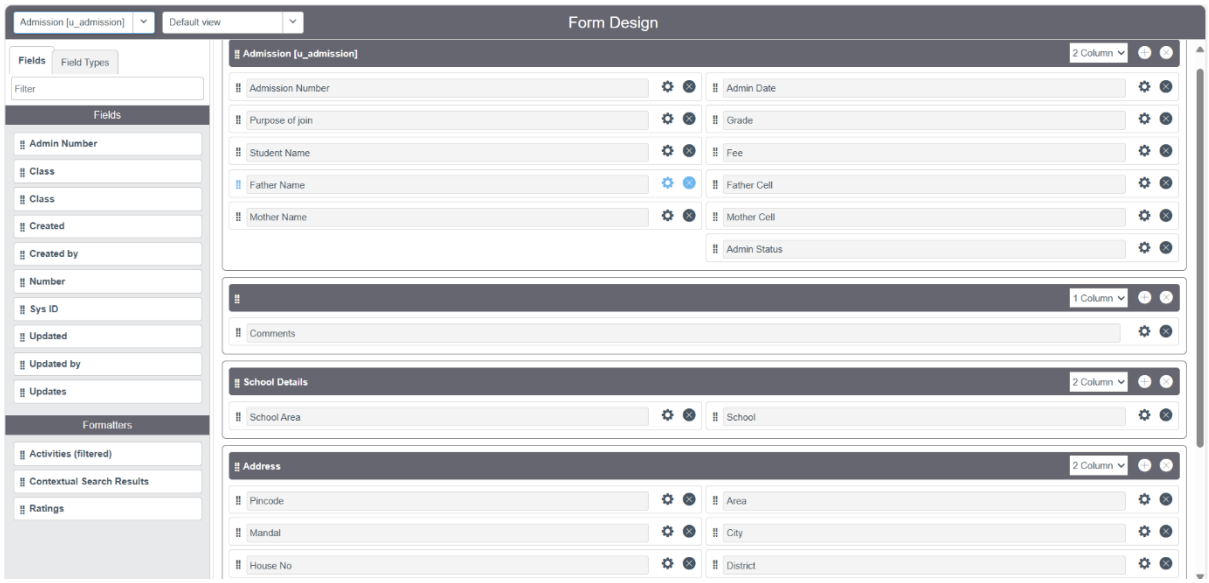


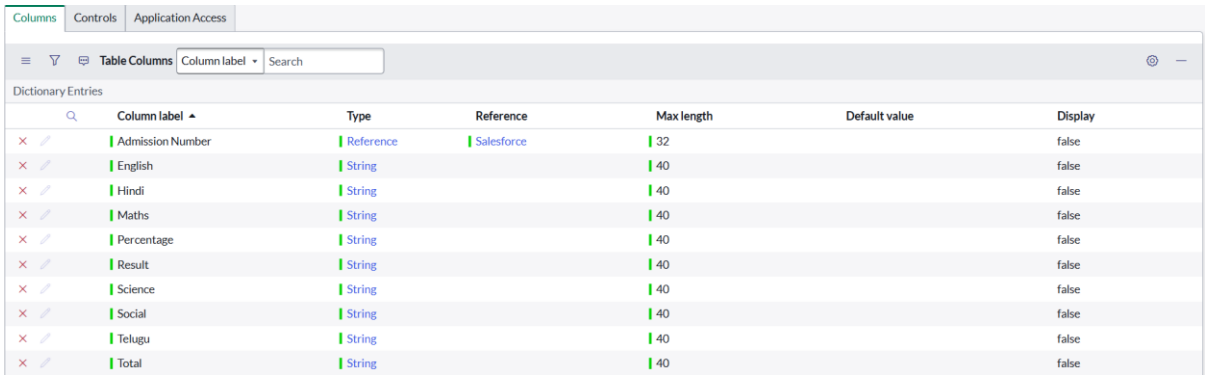
Label	Value	Language	Sequence	Inactive	Updated
X	10th	en	13	false	2025-06-25 06:16:31
Prept	prept	en	1	false	2025-06-25 06:15:15
I	1st	en	4	false	2025-06-25 06:15:27
IV	4th	en	7	false	2025-06-25 06:15:42
VIII	8th	en	11	false	2025-06-25 06:16:18
V	5th	en	8	false	2025-06-25 06:15:48
Nursery	nursery	en	2	false	2025-06-25 06:15:19
IX	9th	en	12	false	2025-06-25 06:16:22
UKG	UKG	en	3	false	2025-06-25 06:15:25
II	2nd	en	5	false	2025-06-25 06:15:31
VI	6th	en	9	false	2025-06-25 06:15:55
VII	7th	en	10	false	2025-06-25 06:16:08
III	3rd	en	6	false	2025-06-25 06:15:34

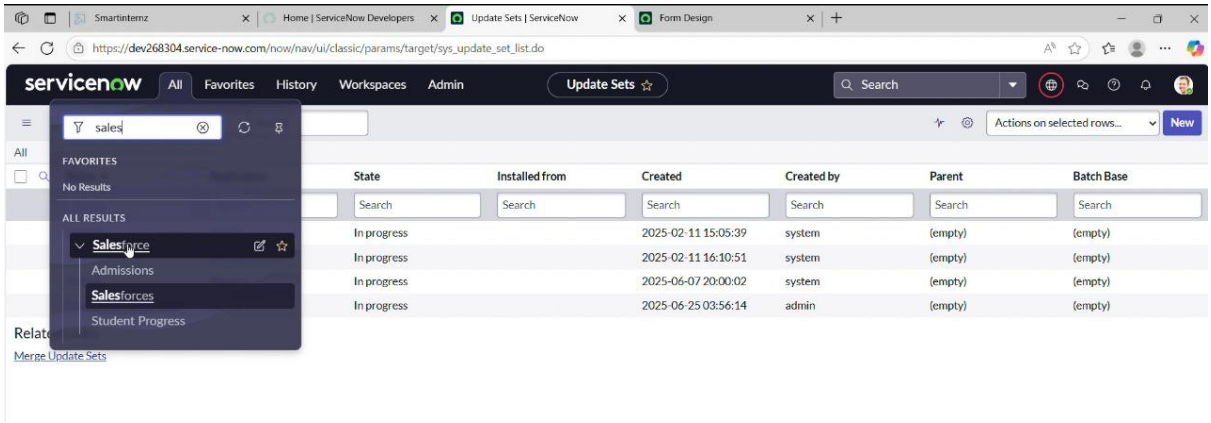

servicenow
All
Favorites
History
Workspaces
Admin
Table - Student Progress
Search
Table
Student Progress
Delete
Update
Delete All Records
Number String (empty) 40 javascript:global.getNextObjNumberPadded(); false
Percentage String (empty) 40 false
Result String (empty) 40 false
Science String (empty) 40 false
Social String (empty) 40 false
Student Name String (empty) 40 false
Insert a new row...
Delete
Update
Delete All Records
Related Links
Form Builder
Design Form
Layout Form
Layout List
Show Form
Show List
Show Schema Map
Add to Service Catalog
Run Point Scan
Explore REST API


servicenow
All
Favorites
History
Workspaces
Admin
Table - Salesforce
Search
Table
Salesforce
Delete
Update
Delete All Records
Columns
Controls
Application Access
Table Columns
Column label
Search
1 to 16 of 16
New
Dictionary Entries
Column label
Type
Reference
Max length
Default value
Display
Admin Date Date (empty) 40 false
Admin Number String (empty) 40 javascript:getNextObjNumberPadded(); true
Class System Class Name (empty) 80 javascript:current.getTableName(); false
Created Date/Time (empty) 40 false
Created by String (empty) 40 false
Father Cell Phone Number (E164) (empty) 40 false
Father Name String (empty) 40 false


servicenow
All
Favorites
History
Workspaces
Dictionary Entry - Admin Status
Search
Dictionary Entry
Admin Status
Create Choice List
Delete Column
Update
Show Table
Run Point Scan
Advanced view
Access Controls
Choices (7)
Attributes
Labels (1)
Label
Search
Actions on selected rows...
New
Choices
Label
Value
Language
Sequence
Inactive
Updated
Rejoined Rejoined en 6 false 2025-06-25 07:45:20
Cancelled Cancelled en 7 false 2025-06-25 07:45:27
Rejected Rejected en 4 false 2025-06-25 07:45:12
Job in progress In progress en 2 false 2025-06-25 07:45:04
Closed Closed en 5 false 2025-06-25 07:45:16
New New en 1 false 2025-06-25 07:45:01
Joined Joined en 3 false 2025-06-25 07:45:08


Admission [u_admission]
Default view
Form Design
Fields
Field Types
Filter
Fields
Admin Number
Class
Class
Created
Created by
Number
Sys ID
Updated
Updated by
Updates
Formatters
Activities (filtered)
Contextual Search Results
Ratings
Admission [u_admission]
2 Column
Admission Number
Admin Date
Purpose of join
Grade
Student Name
Fee
Father Name
Father Cell
Mother Name
Mother Cell
Admin Status
1 Column
Comments
School Details
2 Column
School Area
School
Address
2 Column
Pincode
Area
Mandal
City
House No
District


Columns
Controls
Application Access
Table Columns
Column label
Search
Dictionary Entries
Column label
Type
Reference
Max length
Default value
Display
Admission Number
Reference
Salesforce
32
false
English
String
40
false
Hindi
String
40
false
Maths
String
40
false
Percentage
String
40
false
Result
String
40
false
Science
String
40
false
Social
String
40
false
Telugu
String
40
false
Total
String
40
false


Smartinternz
Home | ServiceNow Developers
Update Sets | ServiceNow
Form Design
https://dev268304.service-now.com/now/nav/ui/classic/params/target/sys_update_set_list.do
servicenow
All
Favorites
History
Workspaces
Admin
Update Sets
Search
sales
FAVORITES
No Results
ALL RESULTS
Salesforce
Admissions
Salesforces
Student Progress
Actions on selected rows...
New
All
State
Installed from
Created
Created by
Parent
Batch Base
Search
In progress
2025-02-11 15:05:39
system
(empty)
(empty)
In progress
2025-02-11 16:10:51
system
(empty)
(empty)
In progress
2025-06-07 20:00:02
system
(empty)
(empty)
In progress
2025-06-25 03:56:14
admin
(empty)
(empty)
Merge Update Sets

8. ADVANTAGES & DISADVANTAGES

Advantages:

- **Automation**: Reduces manual work and human error.
- **Real-Time Analytics**: Dashboards provide instant insights.
- **Security**: Role-based access and encrypted data handling.
- **Scalability**: Built on ServiceNow's cloud infrastructure.
- **User-Friendly**: Intuitive interface for all stakeholders.

Disadvantages:

- **Initial Setup Complexity**: Requires familiarity with ServiceNow.
- **Platform Dependency**: Tied to ServiceNow's ecosystem and licensing.
- **Notification Limitations**: Email/SMS alerts need further configuration.

9. CONCLUSION

The **"Educational Organisation Using ServiceNow"** project successfully demonstrated how cloud-based platforms can transform academic administration. By automating key workflows such as registration, admissions, academic tracking, and notifications, the system enhances operational efficiency, transparency, and user satisfaction.

All core modules were developed, tested, and validated through UAT. Minor bugs were identified and addressed. The project is ready for deployment with scope for future enhancements.

10. FUTURE SCOPE

- **Mobile App Integration**: Developing a mobile version of the platform for easier access by students, teachers, and parents.
- **Advanced Notification System**: Enhancing the email/SMS system with push notifications and real-time alerts.
- **Integration with Government APIs**: Using Aadhar or DigiLocker APIs for identity verification and document validation.
- **AI-Based Analytics**: Implementing predictive analytics to forecast student performance and dropout risks.
- **Multi-Institution Support**: Scaling the system to support multiple schools or colleges under a single administrative dashboard.

11. APPENDIX

Source Code:

1. Auto Populate:

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

   // Auto-populate student details from reference field
   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', true);
   g_form.setDisabled('u_grade', true);
   g_form.setDisabled('u_student_name', true);
   g_form.setDisabled('u_father_name', true);
   g_form.setDisabled('u_mother_name', true);
   g_form.setDisabled('u_father_cell', true);
   g_form.setDisabled('u_mother_cell', true);
}
```

2. Pincode Update:

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

   // Set mandal, city, and district based on pincode
   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');
    }
}
```

3. Disable Fields:

```
function onLoad() {
    // Disable result-related fields on form load
    g_form.setDisabled('u_total', true);
    g_form.setDisabled('u_percentage', true);
    g_form.setDisabled('u_result', true);
}
```

4. Total Update:

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

    // Calculate total marks
    var a = parseInt(g_form.getValue('u_telugu')) || 0;
    var b = parseInt(g_form.getValue('u_hindi')) || 0;
    var c = parseInt(g_form.getValue('u_english')) || 0;
    var d = parseInt(g_form.getValue('u_maths')) || 0;
    var e = parseInt(g_form.getValue('u_science')) || 0;
    var f = parseInt(g_form.getValue('u_social')) || 0;

    var total = a + b + c + d + e + f;
    g_form.setValue('u_total', total);
}
```

5. Percentage Calculation:

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

    // Calculate percentage from total
    var total = parseInt(g_form.getValue('u_total')) || 0;
```

```
    var percentage = (total / 600) * 100;

    g_form.setValue('u_percentage', percentage.toFixed(2) + '%');
}
```

6. Result Evaluation:

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

    // Determine result based on percentage
    var percentageStr = g_form.getValue('u_percentage').replace('%', '');
    var a = parseFloat(percentageStr);

    if (!isNaN(a)) {
        if (a >= 0 && a <= 59) {
            g_form.setValue('u_result', 'Fail');
        } else if (a >= 60 && a <= 100) {
            g_form.setValue('u_result', 'Pass');
        } else {
            g_form.addErrorMessage('Percentage should be between 0 and 100.');
            g_form.clearValue('u_result');
        }
    }
}
```

Dataset Link:

No external datasets used; all data was generated within the ServiceNow instance.

GitHub & Project Demo Link:

GitHub Link: https://github.com/Aafrin-Sayyad/ServiceNow-Admin

Demo video Link: https://youtu.be/645sDSsHY6o