ServiceNow Smartinternz Project Report​

**Project Title**: **Educational Organisation using ServiceNow**

**Team Id: LTVIP2025TMID60763**

Team Size:4

Team Leader: Akkarapaku Harshavardhan

Team Member: Ampolu Sudheer kumar

​

Team Member: Arava Vara Lakshmi

Team Member: Asma Shaik

**ABSTARCT**

1 .Introduction

2.Ideation Phase

3.Requirement Analysis

4.Project Design Phase

5.Project Development

6.Result

* Final Result Screenshots
* Project Execution Video Recordings
* Final Conclusion

**1 .Introduction**

**Project Overview:**

In today's fast-paced digital world, educational institutions are expected to manage a wide range of activities efficiently—from student admissions and attendance to faculty coordination and examination planning. Many colleges and schools still rely on manual or outdated systems, which leads to delays, errors, and poor communication.

To overcome these challenges, our team is developing a project titled "Educational Organization Using ServiceNow", which leverages the powerful capabilities of the ServiceNow platform. ServiceNow is widely used in the corporate world for streamlining workflows, and we are applying its features to an educational setting.

This project focuses on building a centralized platform that automates key administrative tasks within an educational institution. It aims to simplify and digitize processes such as student registrations, attendance tracking, faculty management, scheduling, and internal communication. By using forms, workflows, and ServiceNow's intuitive interface, the system enhances transparency, improves accuracy, and saves time for both staff and students.

**Purpose of the Project**

The main purpose of this project is to improve the efficiency and effectiveness of administrative processes in educational institutions by using the ServiceNow platform.

Our goal is to provide a digital solution that:

* Reduces the manual workload for faculty and staff
* Makes student services more accessible and faster
* Ensures data is properly stored, organized, and retrievable
* Enhances the overall communication and coordination between departments

*By implementing this system, educational institutions can focus more on delivering quality education, while routine tasks like attendance, event planning, and data management are handled smoothly by the platform. Ultimately, this project is a step toward modernizing education management using reliable technology.*

**2.Ideation Phase**

**Problem Statement**

Managing a large educational institution comes with many challenges, especially when most of the tasks are handled manually. Students, teachers, and administrators face delays when applying for certificates, submitting requests, or reporting issues. There's often confusion about whom to contact and how long a process will take.

Our goal is to fix these problems by using ServiceNow to create a centralized Educational Management System. This platform will handle everything from student admissions and faculty requests to progress tracking and IT support. It will make the entire process smoother, faster, and more user-friendly.

**Goals and Objectives:**

* Create one platform for students, teachers, and staff to manage all services
* Simplify and speed up the admission process
* Enable users to easily track the status of their requests
* Automate approval processes using workflows
* Improve transparency and communication across departments
* Reduce manual effort and time wasted on repetitive tasks

**Target Users**

1. Students:

* Apply for certificates, admission, or raise complaints
* Want a system that is quick, clear, and trackable

1. Teachers/Faculty:

* Submit resource or maintenance requests
* Need timely approvals and updates on their issues

3. Admin Staff:

* Manage incoming requests efficiently
* Assign tasks and monitor progress in real-time

**Key Ideas from Our Team**

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Student Service Portal | A portal for students to apply for certificates, hostels, and more |
| Faculty Request System | Teachers can submit resource, leave, or issue-related requests |
| Admission Module | Simplify the new student admission and verification process |
| Request Tracker | Users get live updates on the progress of their applications |
| Admin Dashboard | Staff can view, assign, and complete tasks from one place |

**ServiceNow Tools We Plan to Use**

* **Service Catalog:** To display services for students and staff
* **Flow Designer:** To automate request approval and assignment
* **Knowledge Base:** To answer frequently asked questions
* **Reports & Dashboards:** For real-time tracking and insights

*The Ideation Phase gave us a clear picture of the problems faced by educational institutions. With this understanding, we planned a solution using ServiceNow to make all essential services digital and accessible from one place. This system will reduce time, manual errors, and make the lives of students, faculty, and staff much easier.*

**3 .Requirement Analysis**

**Objective**

The goal of the requirement analysis phase is to clearly understand what our users (students, teachers, and admin staff) need from the Educational Management System. This system is meant to simplify day-to-day processes like admissions, managing student and teacher data, handling requests, and tracking progress.

By collecting and analyzing requirements, we ensure that the solution we build in ServiceNow meets real-world needs and improves institutional efficiency.

**Methods Used for Requirement Gathering**

**Team Brainstorming** – Discussed possible features and problems with teammates

**User Feedback** – Noted common complaints and requests from students and staff

**Process Observation** – Looked at how tasks like admissions and support are done manually in colleges

**Functional Requirements**

*These are the main features the system must provide:*

1. Student Services Module

* Submit requests for certificates, hostel, and complaints

View request status in real-time

Access academic records and track attendance

2. Teacher/Faculty Module

Manage attendance and upload grades

Request classroom resources or report issues

Apply for leave or teaching duty changes

3. Admission Management

Online student registration and document upload

Automated verification and approval process

Admission status tracking for students

**Non-Functional Requirements**

These define how the system should behave, not what it does.

**Usability:** Simple, clean UI that's easy to use for students and staff

**Reliability:** Should work 24x7 without unexpected breakdowns

**Performance:** Fast load times, even when many users access it

**Scalability:** Must support future expansion for new features or more users

**Security:** Protect sensitive user data and only allow authorized access

**Accessibility**: Works on mobile phones, tablets, and desktops

These are the people who are either using or managing the system:

* Students
* Faculty Members
* Administrative and IT Staff
* Project Development Team

*At the end of this phase, we have a complete understanding*

*This Requirement Analysis Phase helped us convert user problems and expectations into structured system features. We now have a strong foundation for the system we’ll be designing in ServiceNow. With these clearly defined requirements, our next step is to begin the Design Phase and map out the workflows, forms, and automation.*

**4.Project Design Phase**

In this project, we are building a digital solution for managing key activities in an educational institution using the ServiceNow platform. The goal is to create a user-friendly system where staff can handle student admissions, maintain academic records, and manage important data efficiently.

This Design Phase helped us shape the structure of our system through the use of tables, forms, and simple user flows. We focused on creating a practical and clear design that can be easily used by non-technical users like teachers, administrators, and office staff.

**Design Overview**

The system is designed using the features of the ServiceNow platform, especially focusing on custom tables and forms.

*The project contains three main modules:*

1. Salesforce Table (Basic Student Info)
2. Admission Table (Full Admission Details)
3. Student Progress Table (Academic Performance Tracking)

Each of these modules is built with its own form and table to organize and store important information.

**1. Salesforce Table**

The Salesforce Table is created to store basic student data that might later be used for reporting or sharing with other platforms like Salesforce CRM.

**➤ Fields included:**

* Admission Number (Auto-generated)
* Admin Date
* Student Name
* Father Name
* Mother Name
* Father Cell Number
* Mother Cell Number
* Grade

*This form focuses on basic student identity and contact details.*

*The form is designed with a clean layout, placing identity fields on the left and contact details on the right. It is useful for syncing or exporting data.*

**2. Admission Table**

The Admission Table stores full details about a student when they join the institution. This is the main form used by admission staff.

**➤ Fields included:**

* Admission Number (Auto-generated Unique ID)
* Student Name
* Father Name / Mother Name
* Admin Date
* Grade and Fee
* Purpose of Join
* Admission Status
* Father Cell / Mother Cell
* House No., Area, Mandal
* District, City, Pincode
* School Area and School
* Comments (for extra notes)

*This form ensures that all essential information is collected at the time of admission. With dropdowns and reference fields, we reduce errors during data entry.*

*The form is arranged in two columns for clarity and faster input. Important fields like student name and contact information are place at the top. Additional details like address and comments are added in lower sections*.

**3. Student Progress Table**

The Student Progress Table is used to record students’ marks in various subjects and automatically calculate results.

**➤ Fields included:**

* Admission Number (linked to Admission Table)
* Marks for Hindi, English, Telugu, Maths, Social
* Scheme
* Total
* Average
* Percentage
* Result (Pass/Fail)

*Teachers can enter the student’s marks, and the system calculates*

*the rest. This saves time and avoids manual calculation errors.*

*The form begins with the Admission Number to identify the student. Marks are filled in next, and calculated fields like total, average, and percentage are automatically shown (non-editable). This ensures the correctness of the result.*

**User Flow Summary**

1. Admissions Staff fills the Salesforce and Admission forms while enrolling a student.
2. Teachers update the Student Progress form after exams.
3. Admins view and manage all records using list and form views in ServiceNow.
4. All records are linked via the Admission Number, which acts as a unique ID across all tables.

**Design Purpose**

The design ensures:

* Easy and clean data entry
* Automatic generation of Admission Numbers
* Dropdowns and reference fields for accuracy
* Proper linking of student records across modules
* Readable and well-organized forms for different roles

*The design of our Educational Organization project in ServiceNow focuses on simplicity, functionality, and clarity. Each form is created with the user in mind, making it easy for staff to complete tasks efficiently. The use of three separate but connected modules helps in organizing student data effectively. This phase has laid a strong foundation for the next steps of development and implementation.*

**5.Project Development**

After completing the project design phase, we moved into the **Development Phase** of our project. This is where our ideas and designs were brought to life inside the ServiceNow platform. We began creating actual tables, forms, fields, and modules, and ensured that the application worked as intended.

This phase helped us transform a basic concept into a working digital solution for managing educational activities such as student admissions, academic performance tracking, and record storage.

**Development Activities**

During the development phase, we used the **ServiceNow Developer Instance** to build the application. Our main tasks included creating custom tables, designing forms, adding fields, setting up relationships between tables, and building navigation modules.

***Here’s a breakdown of the development work we completed:***

**Salesforce Table Development**

* We created a new custom table named **Salesforce**.
* Added fields like:
  + **Admin Number** (Auto-generated ID)
  + **Student Name**
  + **Father Name / Mother Name**
  + **Admin Date**
  + **Grade**
  + **Parent Contact Numbers**
  + *This form stores basic student information and can be used for future integrations or reporting.*

**Admission Table Development**

* Designed a comprehensive **Admission Table** to handle full student enrollment details.
* Fields added include:
  + **Admission Number** (Auto-generated)
  + **Student Name**, **Father Name**, **Mother Name**
  + **Grade**, **Fee**, **Admin Date**
  + **Purpose of Join**, **Admission Status**
  + **Address Fields** (House No., Area, Mandal, City, District, Pincode)
  + **Contact Numbers**
  + **School**, **School Area**
  + **Comments**
  + *Dropdowns were used for fields like Grade and Admission Status to reduce errors and improve consistency.*

**Student Progress Table Development**

* Built the **Student Progress Table** to record academic marks and generate results.
* Fields added:
  + **Admission Number** (Reference to Admission Table)
  + **Subject Marks:** Hindi, English, Telugu, Maths, Social
  + **Scheme**
  + **Calculated Fields:** Total, Average, Percentage, Result
  + *We ensured that the calculated fields are read-only and auto-updated based on subject marks. The Admission Number links progress records to the main student profile.*

**Module Creation**

* Created a main application menu called **“Education Management”**.
* Under it, created modules for:
  + **Salesforce Records**
  + **Admission Management**
  + **Student Progress Tracking**
  + These modules help users quickly navigate to forms and lists related to each section of the application

**Form Layout & User Interface**

* Designed all forms in a **clean and user-friendly layout**.
* Grouped fields in logical order (e.g., personal details at top, address below).
* Used a **two-column layout** to avoid long scrolling forms.
* Enabled reference fields to connect tables (e.g., linking progress to admission).
* *All forms were tested for correct field alignment, readability, and flow.*

**Challenges Faced**

* Adjusting form layout to make it clean and user-friendly
* Managing dropdown values and data consistency
* Linking reference fields properly between tables
* Handling automatic field calculations

Each challenge was tackled step-by-step through practice, team discussions, and ServiceNow platform help.

**Key Learnings**

* Learned how to create custom applications in ServiceNow
* Understood how tables, forms, and reference fields work
* Improved teamwork and communication during development
* Gained experience in UI layout, field types, and data handling

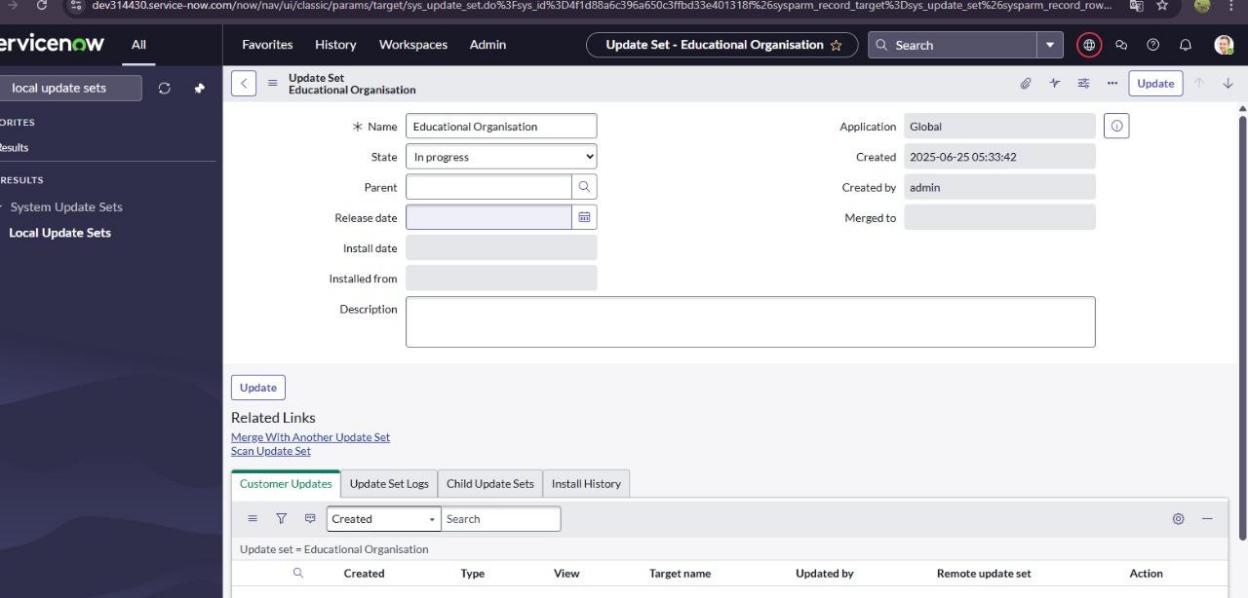
*The development phase was the most exciting and important part of our project. We took our ideas and turned them into a working solution using ServiceNow’s features. With well-designed tables and user-friendly forms, we now have a system that helps educational institutions manage admissions and student progress smoothly.*

*This phase not only helped us apply technical skills but also improved our ability to work as a team and solve real-world problems through software.*

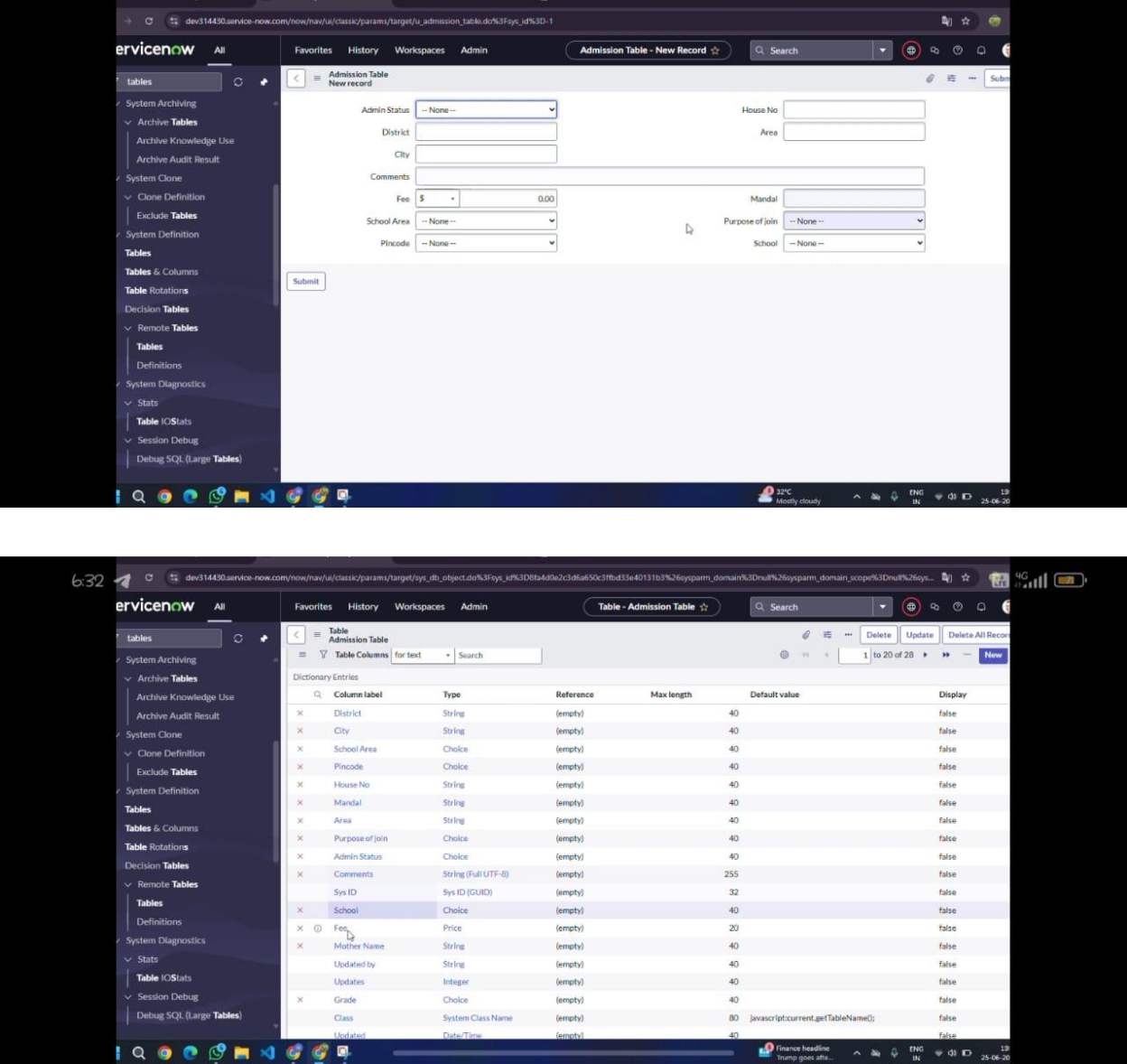
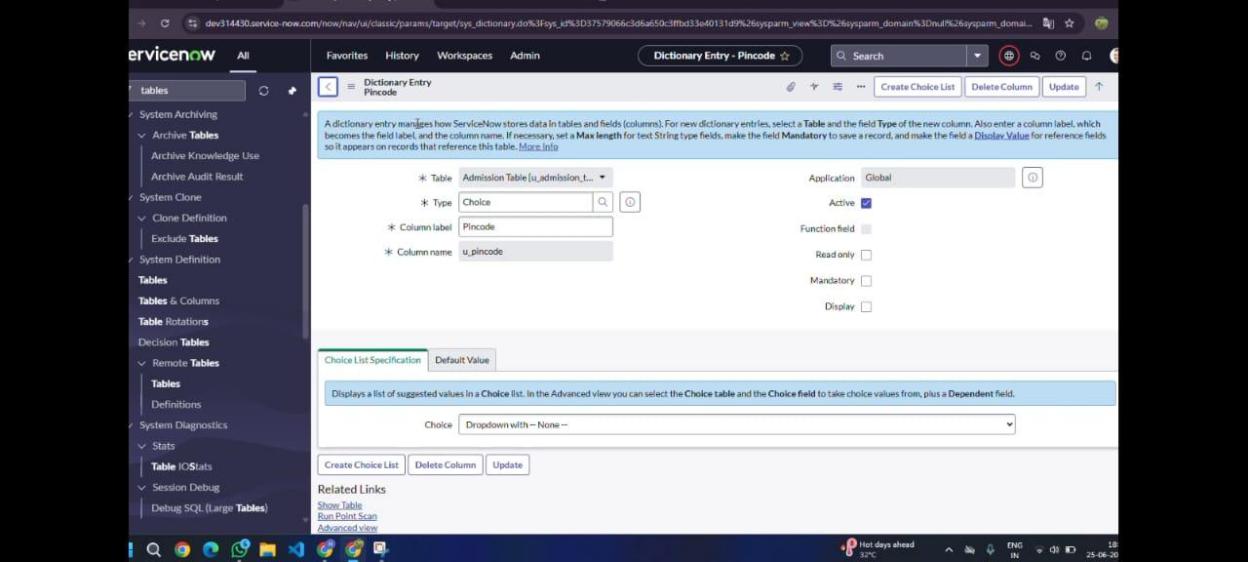
**6.Result:**

**Output Screenshots**

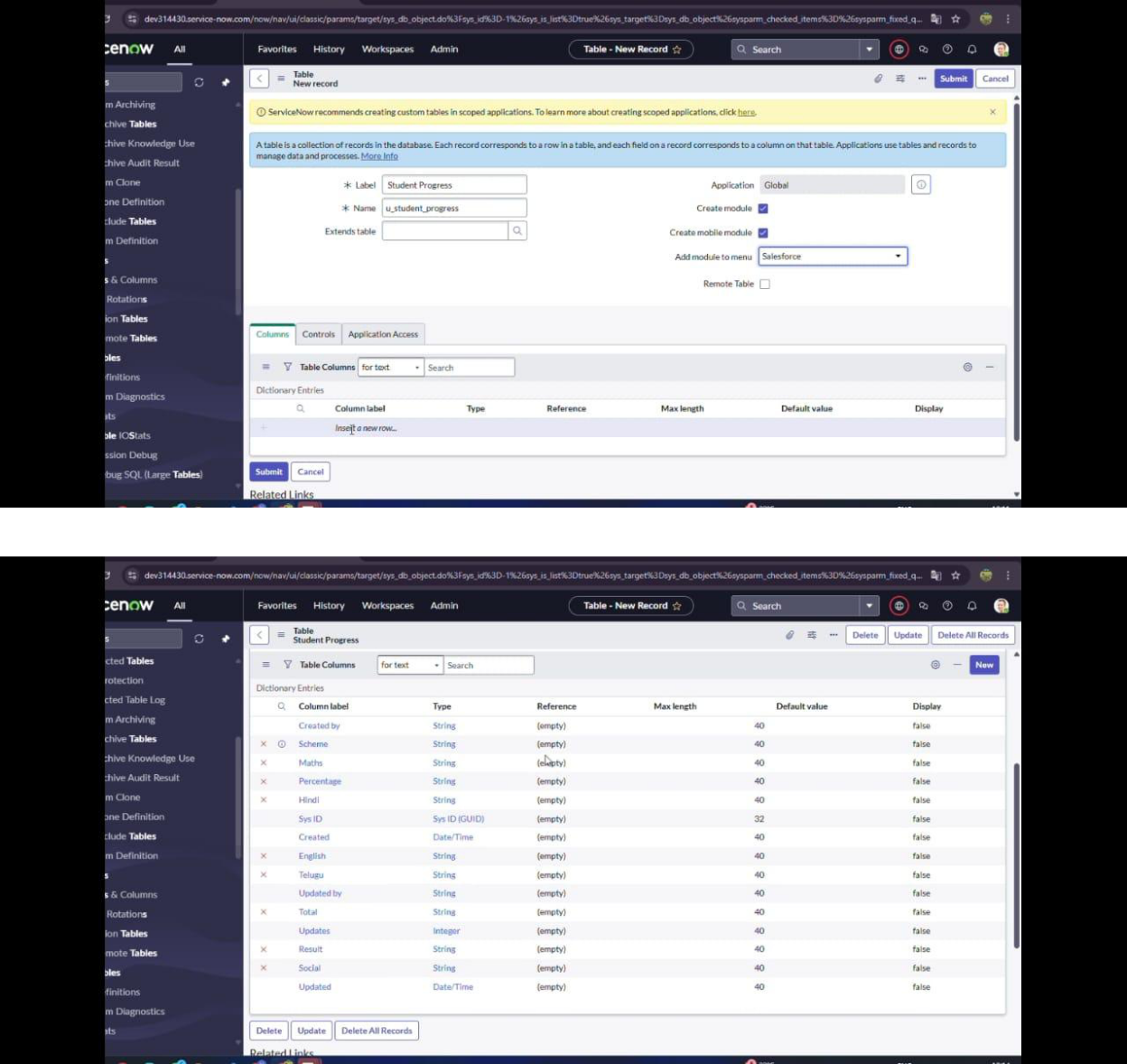
* CREATING UPDATE SET



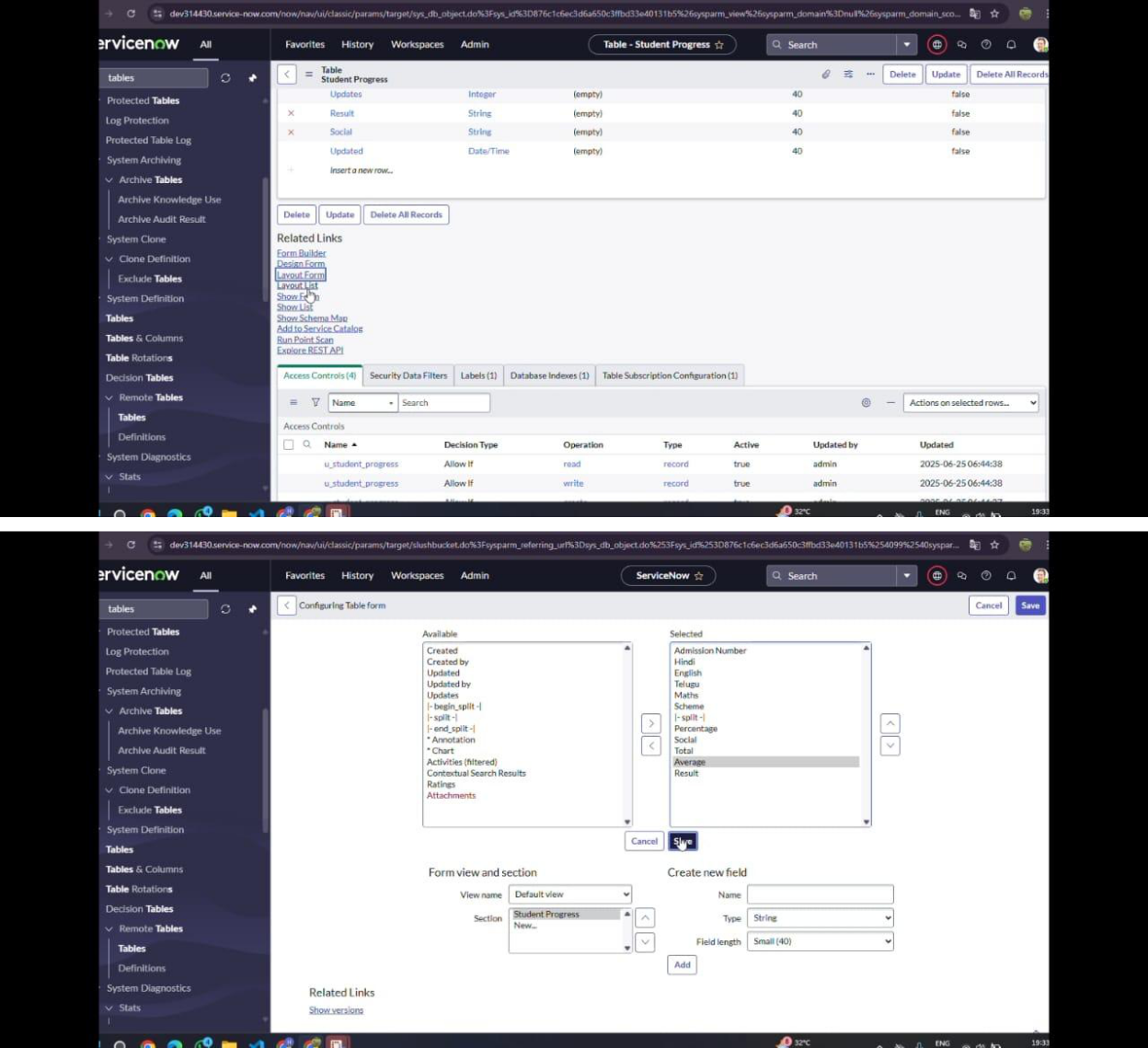
* CREATING ADMISSION TABLE:​



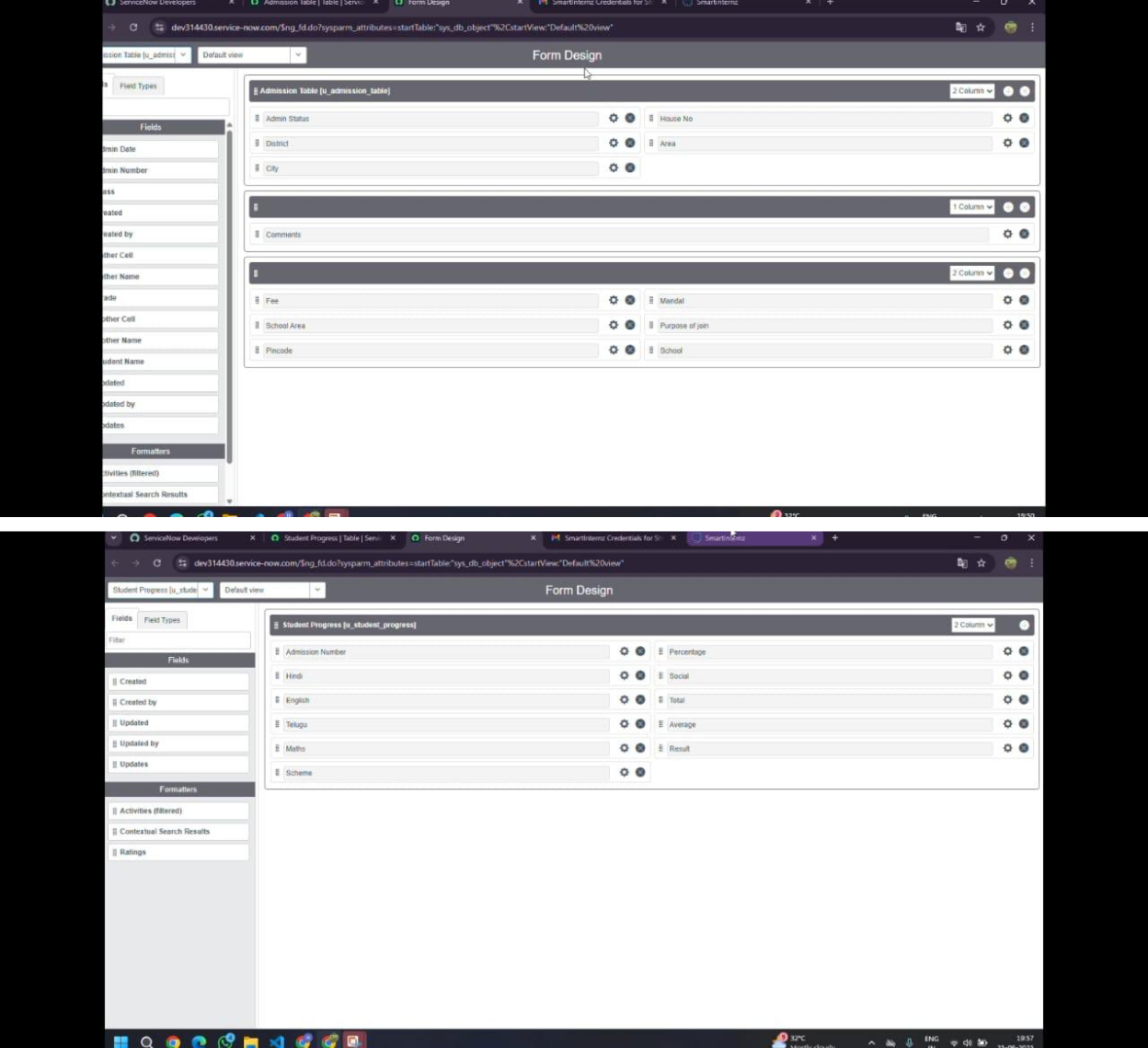
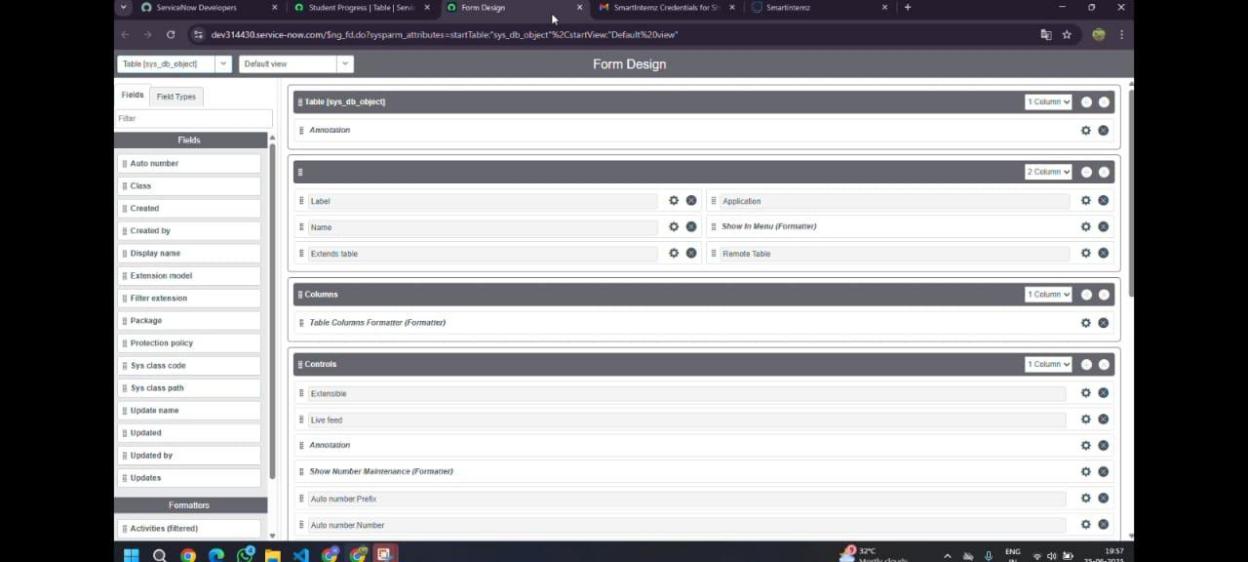
* CREATING SALESFORCE TABLE:​



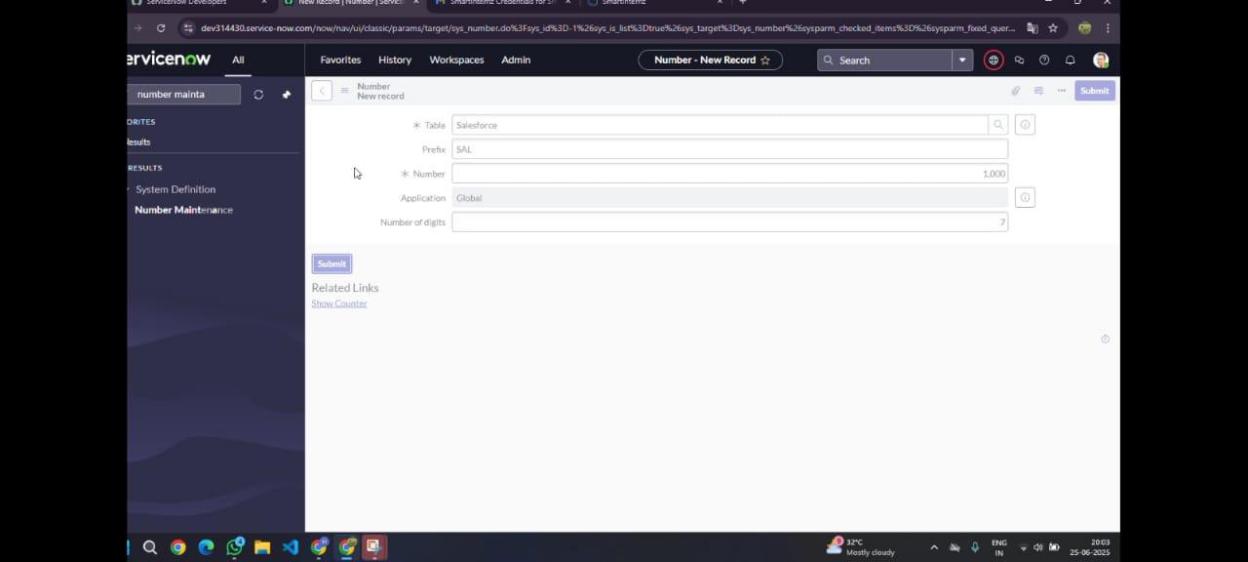
* + .LAYOUT FORM:​



* + - FORM DESIGN:​



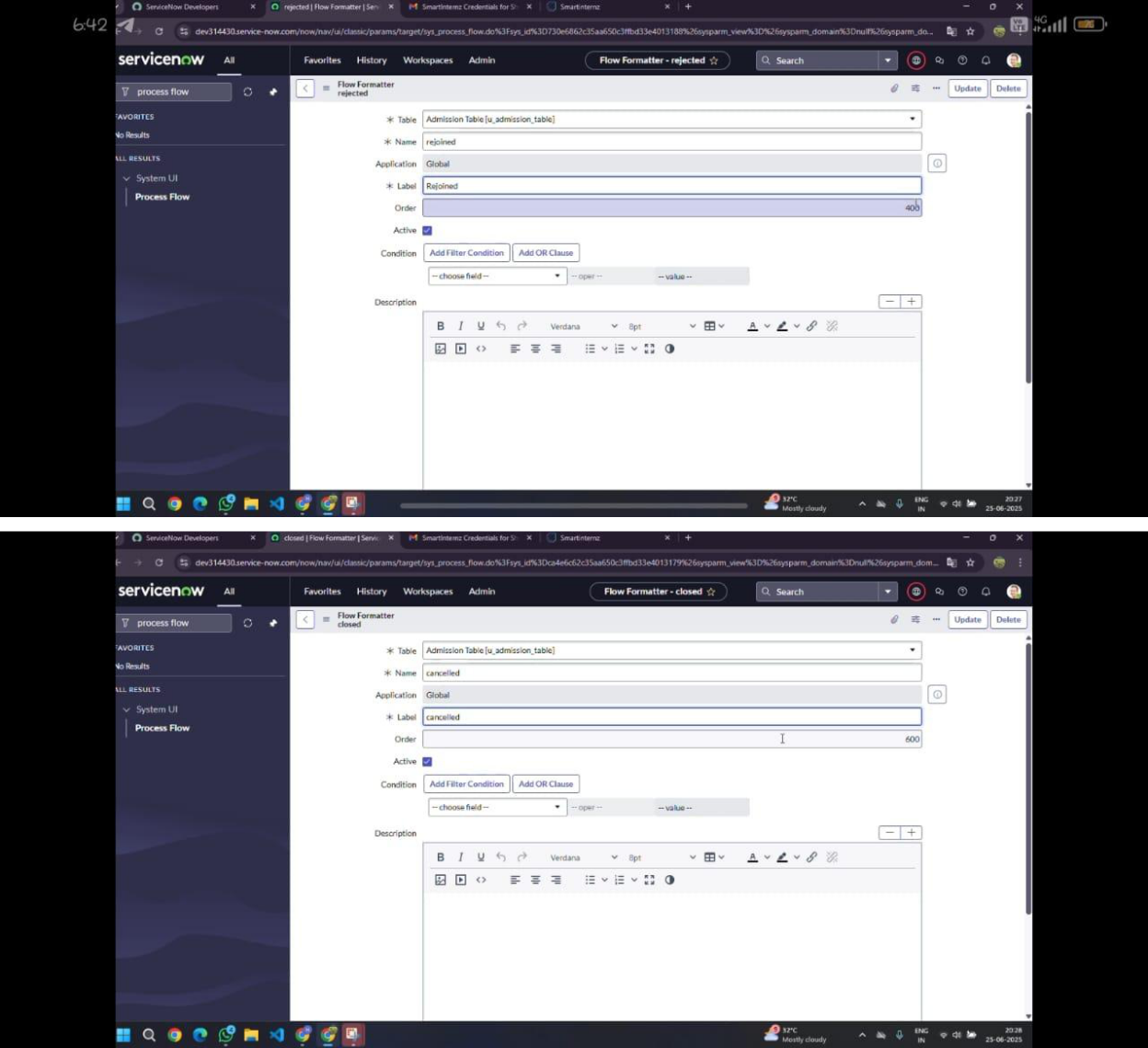
* + CREATING NUMBER MAINTENANCE​



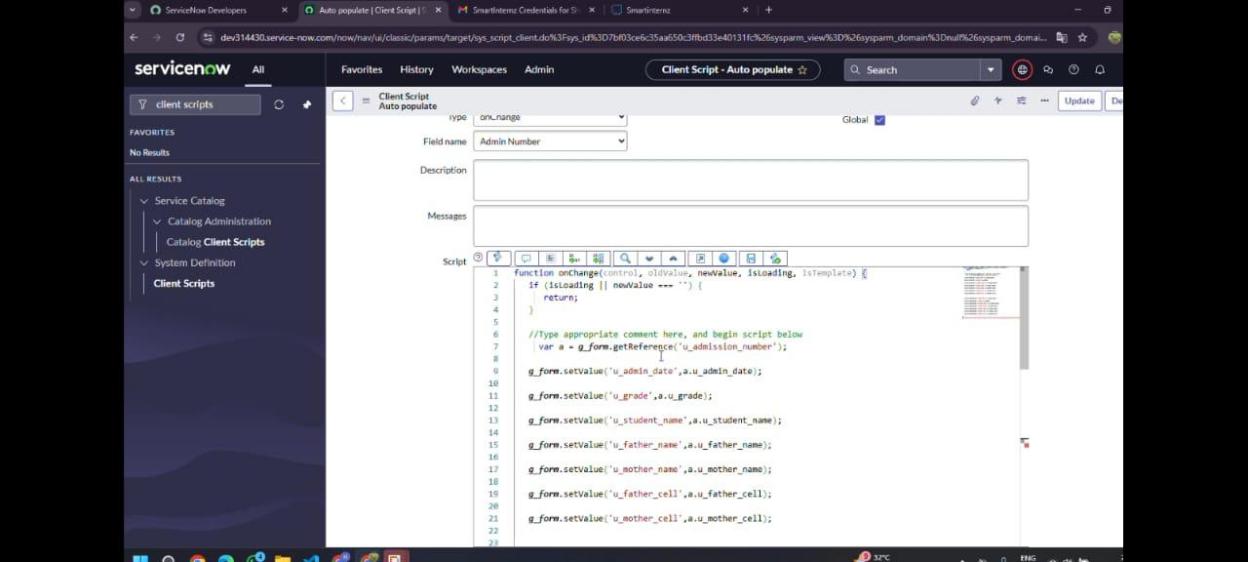
​

​

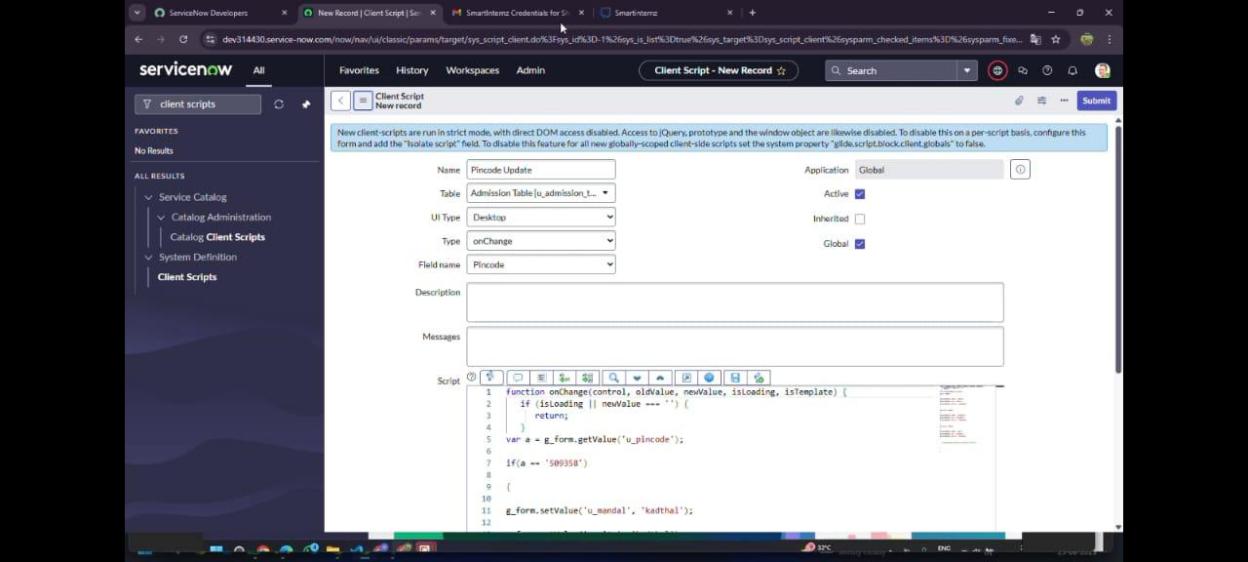
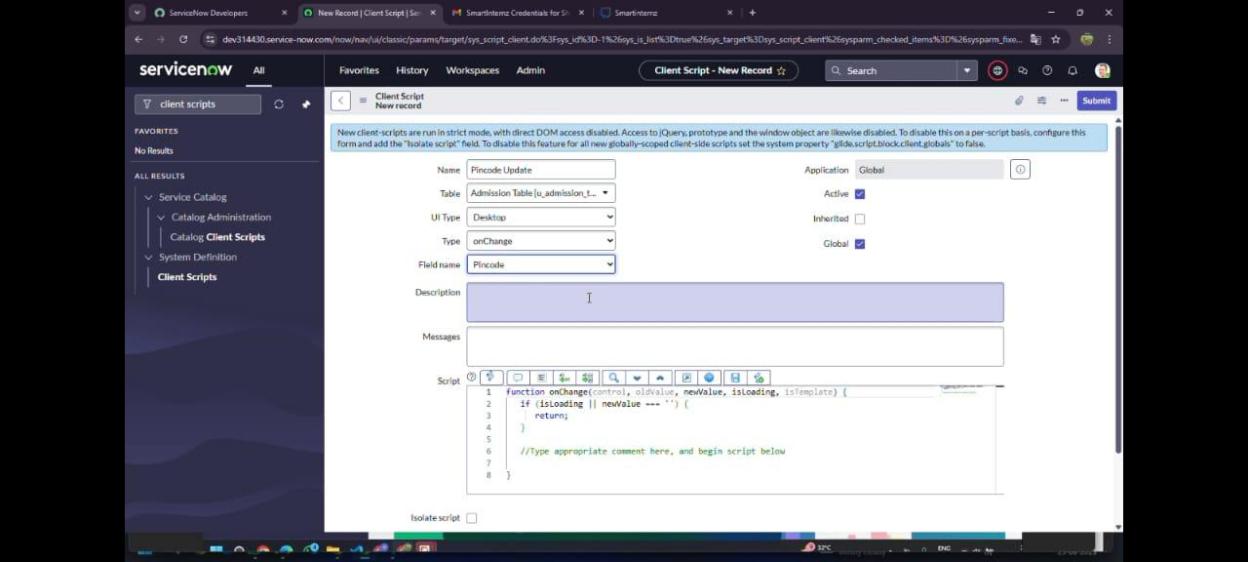
* + PROCESS FLOW:​



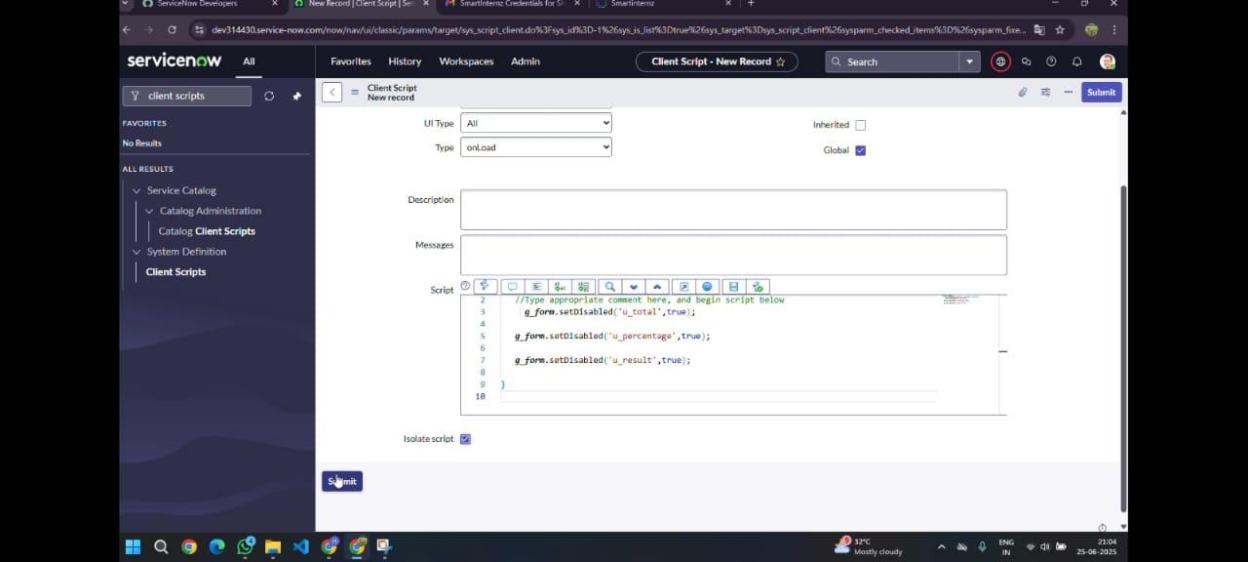
* + CLIENT SCRIPT (AUTO POPULATE)



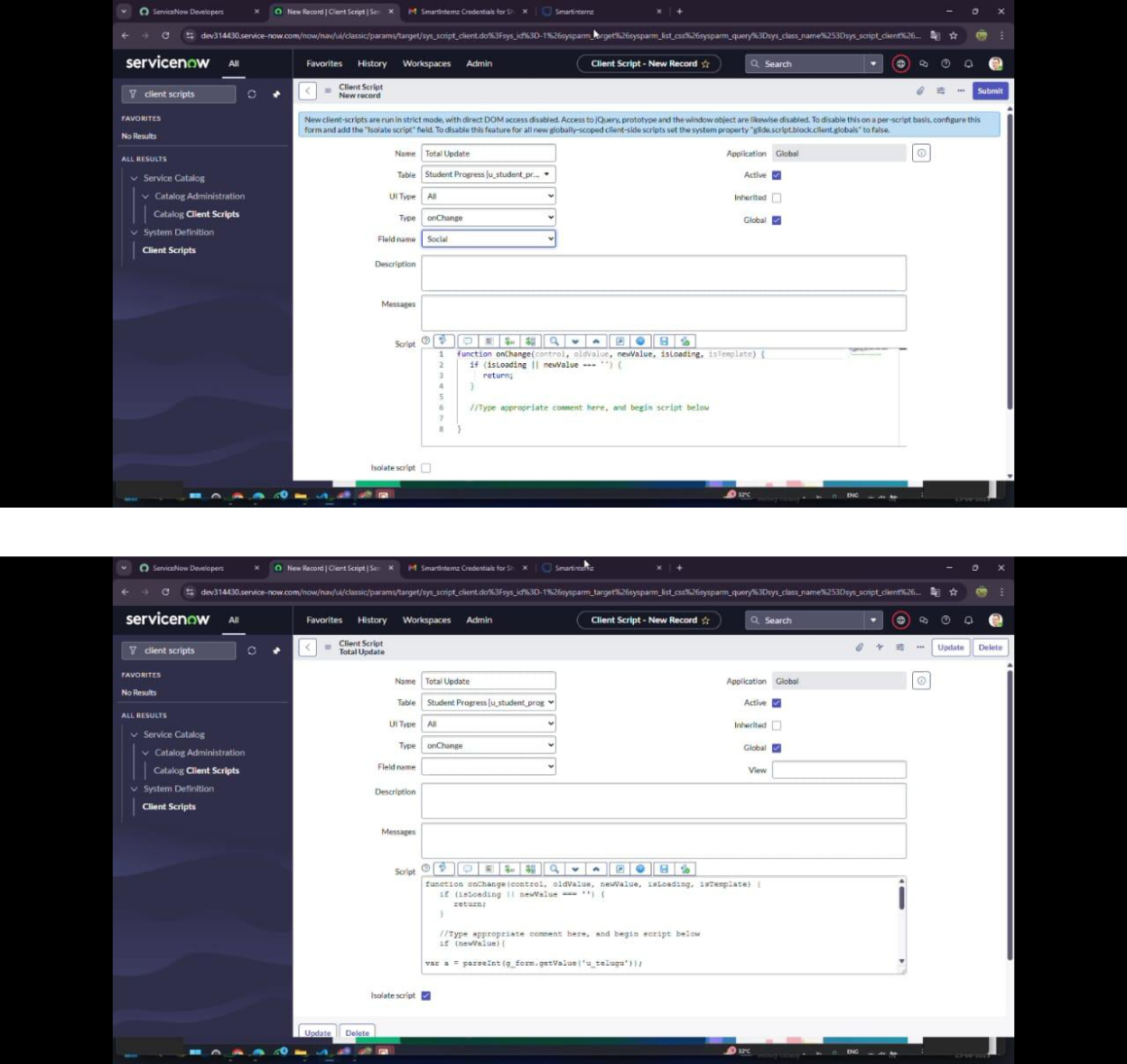
* + CLIENT SCRIPT (PINCODE UPDATE)



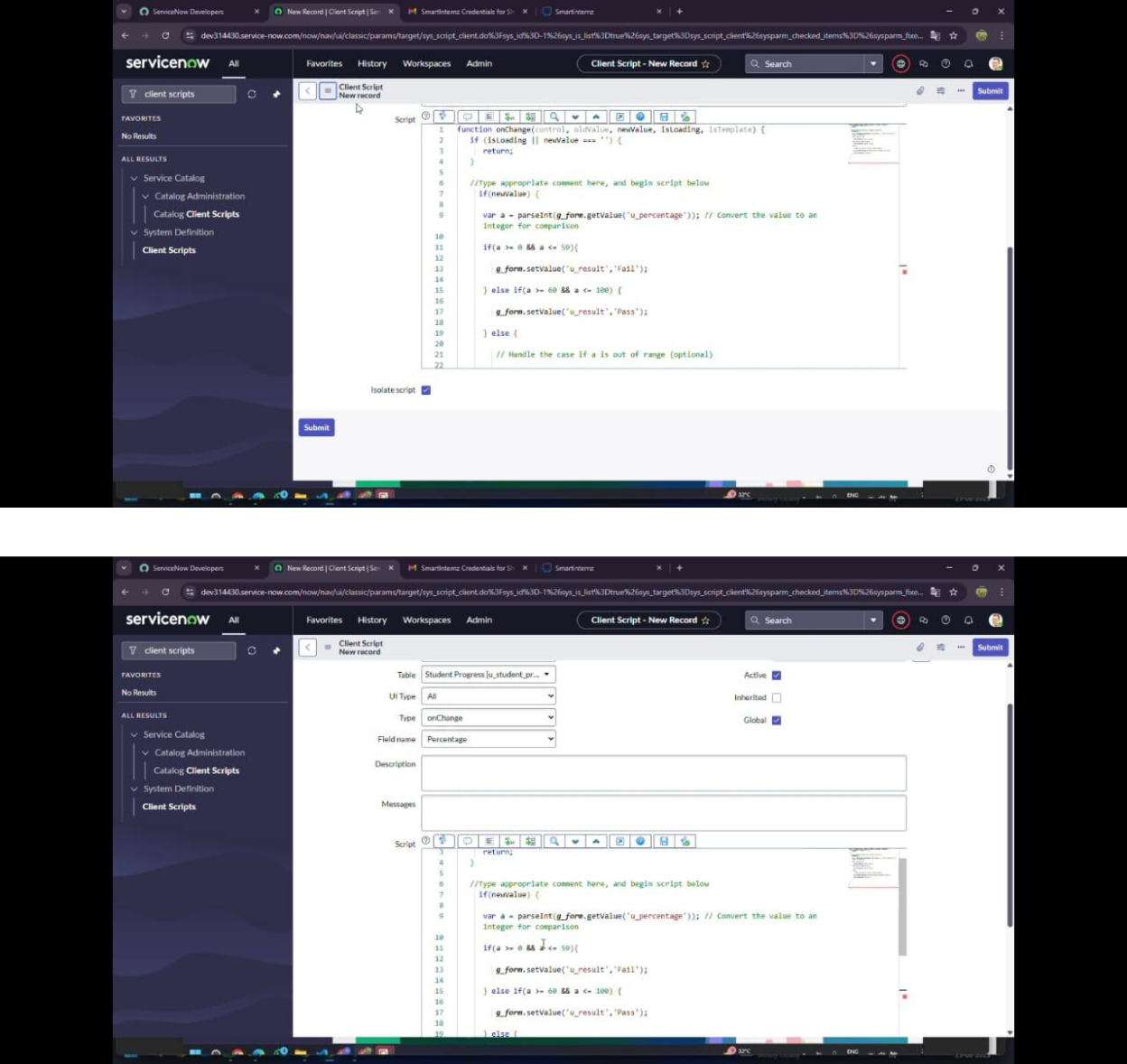
* + CLIENT SCRIPT (DISABLE FIELDS)



CLIENT SCRIPT (TOTAL UPDATE)

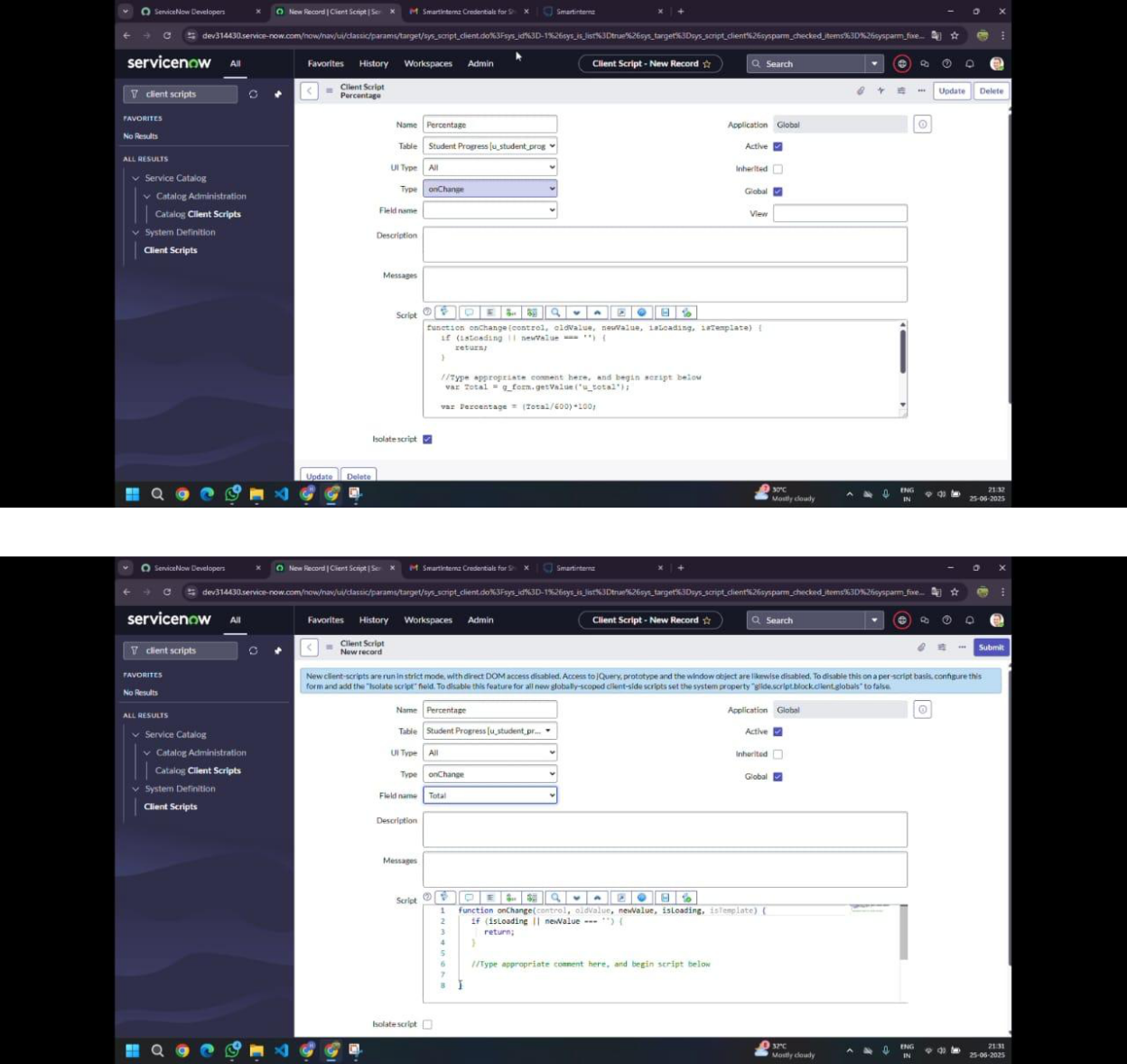


* + CLIENT SCRIPT (RESULT):

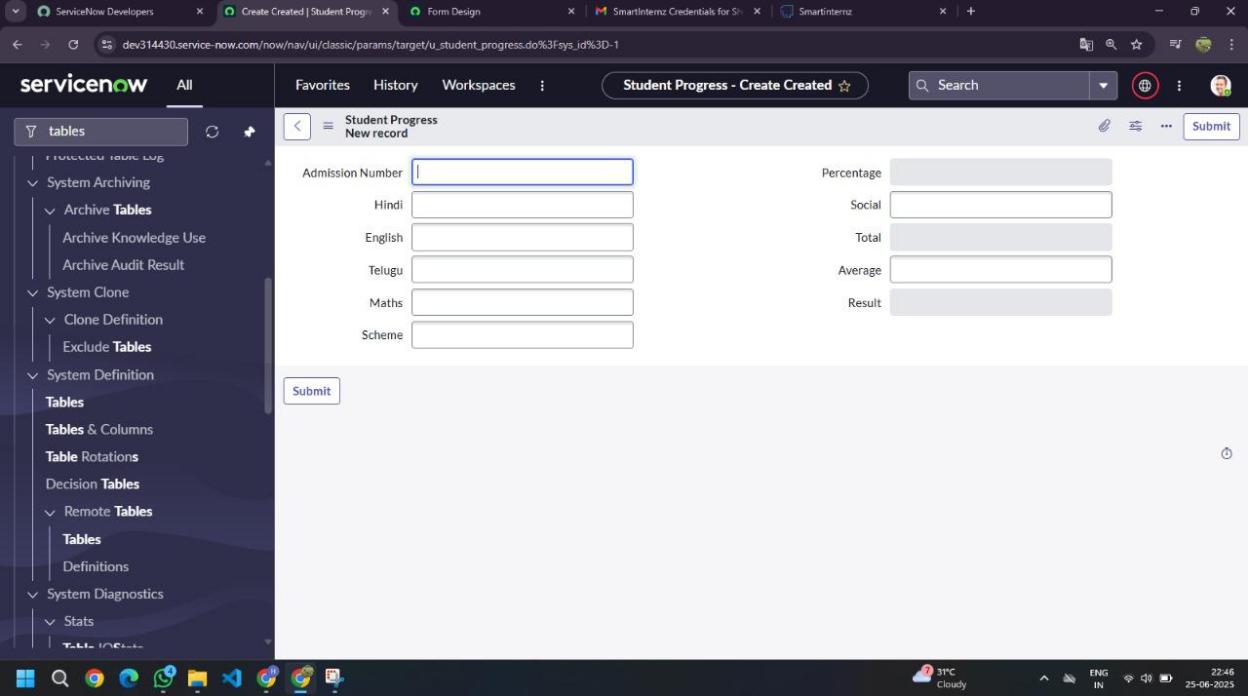
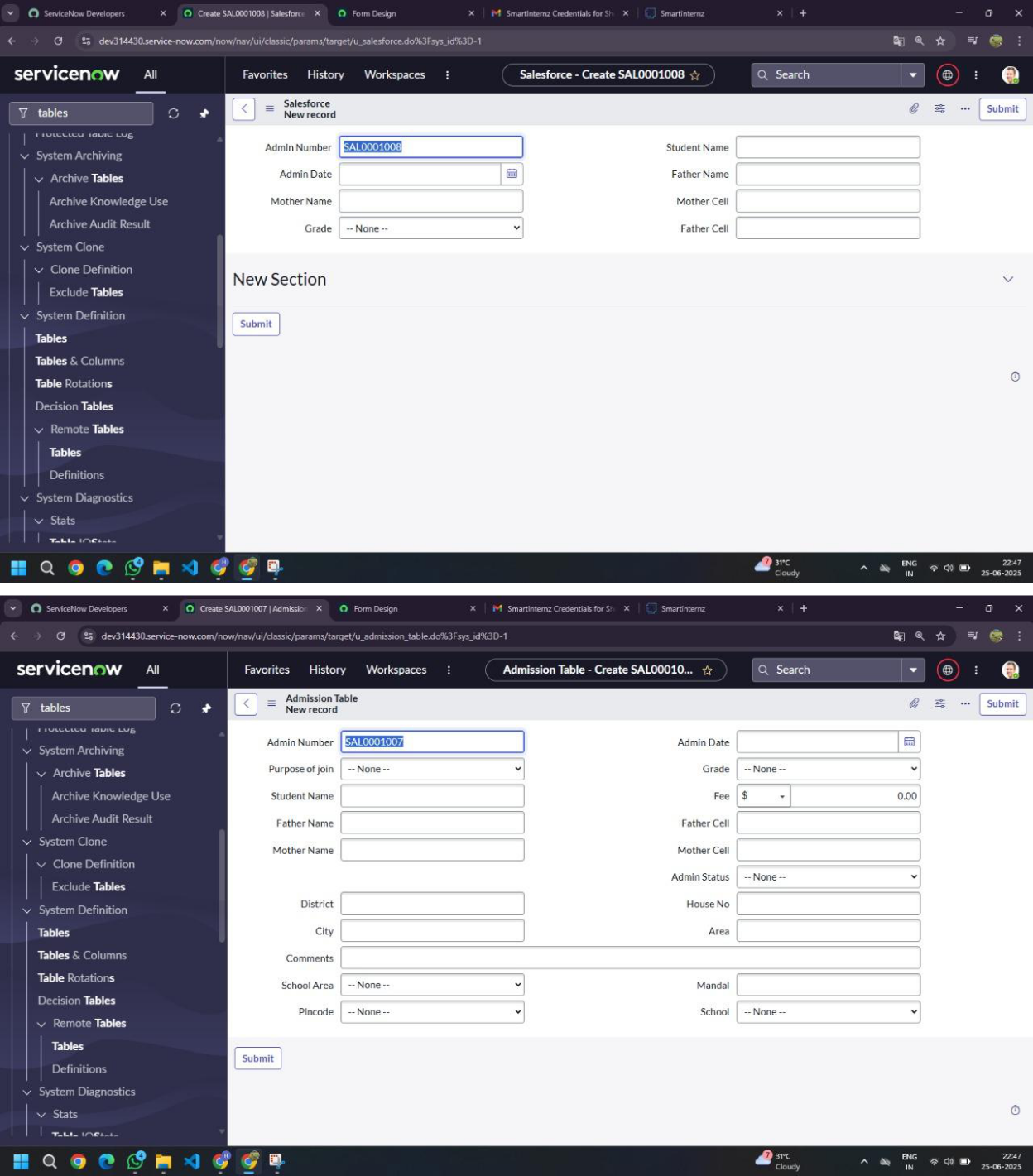


* + CLIENT SCRIPT (PERCENTAGE)

​



**Final Result Screenshots:**



**Project Execution Video Recordings:**

* Click here to view the complete screen recording of the project:(35 mins video)

<https://drive.google.com/file/d/1JICLP1AiFQnkR1291muzH8FTeKRE2iaC/view?usp=drivesdk>

* Click here to watch the demo video which include only the result of the project:(2 mins video)

<https://drive.google.com/file/d/1kXDtG2KuXM_4wvu0nxYsUAz8cB5OcFx0/view?usp=drive_link>

**Final Conclusion**

When we first began this project, we only had a simple idea — to make things easier for educational institutions. There was no complete clarity, no strong roadmap, and even some doubts about whether we could really build something meaningful using the ServiceNow platform. But slowly, as we started planning, designing, developing, and testing, we saw that we were creating something real, something impactful.

Our project, **“Educational Organization using ServiceNow”** has not just been about creating tables and forms. It has been about solving a real problem faced by many educational institutions — the burden of manual work, mismanaged student data, scattered academic records, and outdated admission systems.

Through this project, we developed a complete system where:

* Student admissions are managed smoothly with clean data entry.
* Academic progress is tracked and auto-calculated without errors.
* Essential student records are available in a structured and easy-to-use interface.

**More than just a technical exercise, this project taught us:**

* How digital systems can solve real-life problems.
* How platforms like ServiceNow, when used properly, can bring transformation even to non-IT sectors like education.

*We also learned that building something is not just about reaching the final screen or form — it's about understanding the user’s needs and delivering what actually helps them.*