

# RETO FINAL SALESFORCE DEVELOPER I

---

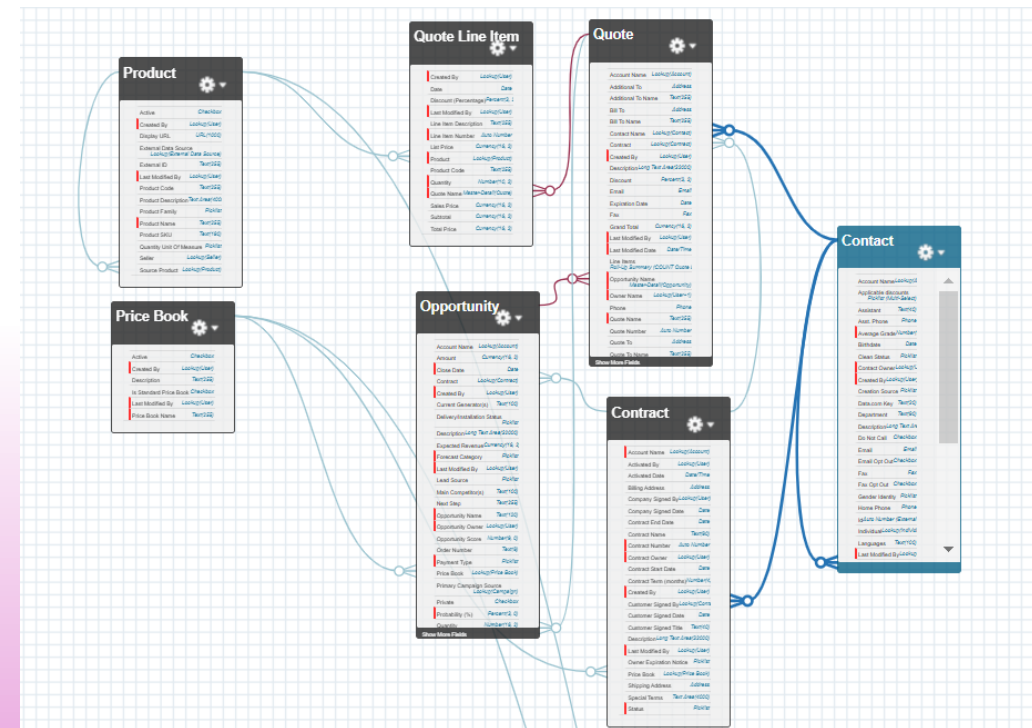
Omar Alonso López  
Valenzuela

# OBJETIVOS DEL RETO

- *¿Cómo puedo diseñar e implementar una solución integral en Salesforce que automatice el proceso de cotización y gestión de descuentos para estudiantes, garantizando eficiencia y precisión en el sistema de cobranza de la universidad?*

# CREACIÓN DE OBJETOS Y CAMPOS PERSONALIZADOS

- Para lograr resolver esta propuesta se utilizaron algunos de los objetos estándar de Salesforce aprovechando las relaciones que tienen objetos como Contacts, Opportunities y Quotes principalmente.
- Además, se implementaron campos personalizados a algunos de los objetos, lo que ayudaría a la implementación lógica de algunos valores solicitados.



# CREACIÓN DE PRICE BOOKS PARA ESTABLECER EL PRECIO DE MATERIA POR CAMPUS

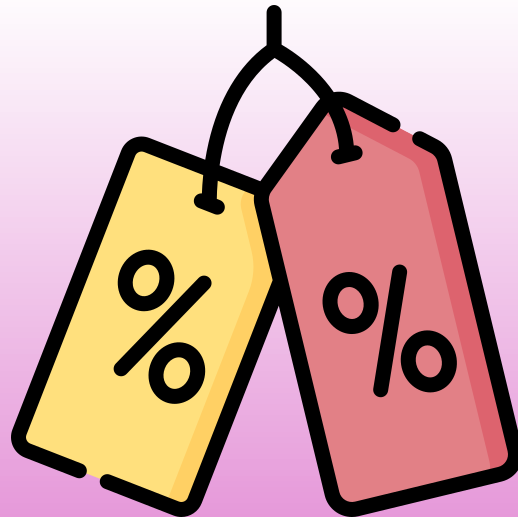
- Para la definición de precios y descuentos se creó primeramente un Price Book para cada estado que tuviera un campus y asignándole su respectivo precio al Producto Materia dentro de cada Price Book, por lo que al crear una oportunidad sólo se podrían seleccionar cuantas materias se ocupen pero del mismo valor dentro del campus.

Sede	Costos por materia
Guanajuato	\$17,000.00
Nuevo León	\$18,000.00
Jalisco	\$13,000.00
Querétaro	\$15,000.00

Price Books					New
Recently Viewed					
5 items • Updated a few seconds ago					Search this list...
	Price Book Name	Description	Last Modified Date	Active	
1	<input type="checkbox"/> Queretaro		17/11/2024, 15:02	<input checked="" type="checkbox"/>	▼
2	<input type="checkbox"/> Jalisco		17/11/2024, 15:02	<input checked="" type="checkbox"/>	▼
3	<input type="checkbox"/> Nuevo Leon		17/11/2024, 15:01	<input checked="" type="checkbox"/>	▼
4	<input type="checkbox"/> Guanajuato	Costo de materias para alumnos de Guanajuato	17/11/2024, 14:36	<input checked="" type="checkbox"/>	▼
5	<input type="checkbox"/> Standard Price Book		12/11/2024, 17:42	<input type="checkbox"/>	▼

# CODIFICACIÓN DE LÓGICA PARA DESCUENTOS Y BECAS

- Se desarrollaron dos triggers, primero uno que aplicara los descuentos por cantidad de materias, esto para que no se aplicara el descuento a la primer materia inscrita y solo a las subsecuentes en caso de ser más de una. El segundo trigger ejecutaba los descuentos por las becas asignadas a cada alumno, con la condición de que este descuento estaría limitado al 60%.



# ENVÍO DE COTIZACIÓN POR CORREO

- Para que al crearse una cotización se enviara automáticamente al contacto a su correo electrónico se creó una plantilla en Visualforce y un trigger adicional que ejecutaba el envío de la plantilla al crearse la cotización.

```
1 <apex:page standardController="Quote" renderAs="pdf" showHeader="false">
2   <h1>Resumen de Cotización</h1>
3   <p><strong>Nombre del estudiante:</strong> {!Quote.Contact.Name}</p>
4   <p><strong>Matrícula:</strong> {!Quote.Contact.Id}</p>
5   <p><strong>Forma de pago:</strong> {!Quote.Opportunity.Payment_Type__c}</p>
6   <p><strong>Total antes de descuentos:</strong> {!Quote.Subtotal}</p>
7   <p><strong>Descuentos aplicados:</strong> {!Quote.Contact.Applicable_discounts__c}</p>
8   <p><strong>Total después de descuentos:</strong> {!Quote.TotalPrice}</p>
9
10  <apex:repeat value="{!Quote.QuoteLineItems}" var="lineItem">
11    <p><strong>Producto:</strong> {!lineItem.Product2.Name}</p>
12    <p><strong>Precio Unitario:</strong> {!lineItem.UnitPrice}</p>
13    <p><strong>Cantidad:</strong> {!lineItem.Quantity}</p>
14    <p><strong>Descuento:</strong> {!lineItem.Discount}</p>
15    <p><strong>Total:</strong> {!lineItem.TotalPrice}</p>
16    <br/>
17  </apex:repeat>
18
19  <h2>Calendario de Pagos</h2>
20
21  <apex:outputPanel rendered="{!Quote.Opportunity.Payment_Type__c == 'Monthly payments'}">
22    <p>Inscripción: 10 de julio</p>
23    <p><strong>Mensualidades:</strong></p>
24    <ul>
25      <!-- Cálculo de cada mensualidad -->
26      <apex:variable var="totalAfterDiscount" value="{!Quote.TotalPrice}" />
27      <apex:variable var="monthlyPayment" value="{!totalAfterDiscount / 6}" />
28      <li>10 de agosto: {!TEXT(monthlyPayment)} </li>
29      <li>10 de septiembre: {!TEXT(monthlyPayment)} </li>
30      <li>10 de octubre: {!TEXT(monthlyPayment)} </li>
31      <li>10 de noviembre: {!TEXT(monthlyPayment)} </li>
32      <li>10 de diciembre: {!TEXT(monthlyPayment)} </li>
33    </ul>
34  </apex:outputPanel>
35
36  <apex:outputPanel rendered="{!Quote.Opportunity.Payment_Type__c == 'Cash payment'}">
37    <p>Pago único: 10 de julio: {!TEXT(Quote.TotalPrice)}</p>
38  </apex:outputPanel>
39 </apex:page>
```

# REFLEXIONES

- Creo que lo más complicado del reto fue comprender la arquitectura y el modelado de los datos que se pueden utilizar en la herramienta, pero cuando se tienen dominados, los objetos estándar pueden ser de muy buena utilidad para ahorrar tiempo y recursos en la programación y creación de estos datos.
- A pesar del corto tiempo del que disponía creo que pude aprovechar la situación para entender aún más la herramienta y sus datos, y que a pesar de que no logré el resultado esperado sé que con un equipo que me apoye en el desarrollo y en el modelado de los datos podemos realizar un trabajo más efectivo ahora que sé que la parte de desarrollo y codificación fue lo que mejor logré en este proyecto.