ISDS 7510: SQL Project – Group #3

Lundyn Harrelson, Julian Navarre, Rachel Packer, Erin Roach

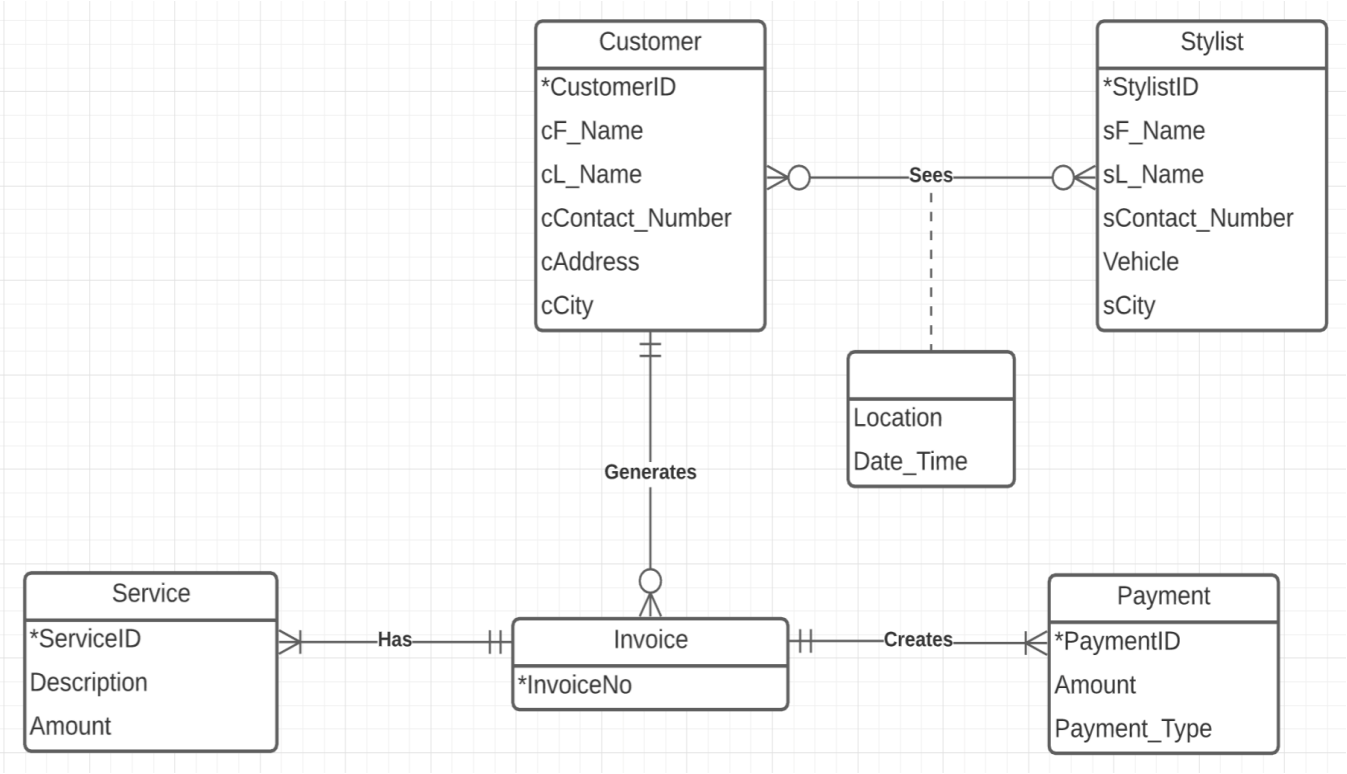
Business Statement

Purple & Gold Consulting specializes in enhancing businesses' tech operations. Recently, we gained a new client, Snip's. Snip's is a virtual salon/barber client who has tasked us with creating an application platform to connect with clients so that stylists may now provide mobile service(s) to customers. Snip's is a mobile salon/barber service provider, which contracts licensed stylists who may bring their service(s) to the client, instead of the traditional business model of enforcing the client to travel to them to receive such service(s). To begin the deployment of Snip's new application, we must first create a database for the application to hold and retrieve information from their entity relationship diagram. Below, we have outlined the final database model for Snip's.

Snip's Business Rules:

1. Each stylist can see zero to many customers. Each customer can be seen by zero or more stylists. An appointment requires that one stylist visits one customer at one time.
2. Appointments should be created when one stylist sees one customer at one location, date, and time.
3. Each customer can generate zero to many invoices. Each invoice is generated by one and only one customer.
4. An invoice must include at least one service, but many services may be included within one invoice. Additionally, an invoice can be generated from one or many different forms of payment. However, one to many different forms of payment generates one and only one invoice.
5. A service belongs to one and only one invoice.
6. A payment belongs to one and only one invoice.

Entity Relationship Diagram and Relational Data Model



Stylist(**StylistID**, sF_Name, sL_Name, sContact_Number, Vehicle, sCity)

Customer(**CustomerID**, cF_Name, cL_Name, cContact_Number, cAddress, cCity)

Appointment(**AppointmentID**, Location, Date_Time, */StylistID/*, */CustomerID/*)

Invoice(**InvoiceNo**, */CustomerID/*)

Service(**ServiceID**, Description, Amount, */InvoiceNo/*)

Payment(**PaymentID**, Amount, Payment_Type, */InvoiceNo/*)

Data Dictionary

Table:	Stylist				
Columns					
Name	Data Type	Nullable	Primary Key	Foreign Key	Data Sample
StylistID	Varchar	No	X		1
sF_Name	Char	Yes			Iseabal
sL_Name	Char	Yes			Lorey
sContact_Number	Char	Yes			722-292-2456
Vehicle	Char	Yes			WAUEFAFL2BN418624
sCity	Char	Yes			Baton Rouge

Table:	Customer				
Columns					
Name	Data Type	Nullable	Primary Key	Foreign Key	Data Sample
CustomerID	Varchar	No	X		1
cF_Name	Char	Yes			Starr
cL_Name	Char	Yes			Marcum
cContact_Number	Char	Yes			745-155-7909
cAddress	Char	Yes			408 Westend Court
cCity	Char	Yes			Baton Rouge

Table:	Appointment				
Columns					
Name	Data Type	Nullable	Primary Key	Foreign Key	Data Sample
AppointmentID	Varchar	No	X		1
Location	Char	yes			0 Bayside Plaza
Date_Time	Char	yes			11/16/2019 22:24
StylistID	Varchar	yes		X	23
CustomerID	Varchar	yes		X	40

Data Dictionary Continued

Table:	**Invoice**				
Columns					
Name	Data Type	Nullable	Primary Key	Foreign Key	Data Sample
InvoiceNo	Varchar	No	X		1
CustomerID	Varchar	Yes		X	1

Table:	**Payment**				
Columns					
Name	Data Type	Nullable	Primary Key	Foreign Key	Data Sample
PaymentID	Varchar	No	X		1
Amount	Char	Yes			75
Payment_Type	Char	Yes			Cash
InvoiceNo	Varchar	Yes		X	1

Table:	**Service**				
Columns					
Name	Data Type	Nullable	Primary Key	Foreign Key	Data Sample
ServiceId	Varchar	No	X		1
Description	Char	Yes			Cut and Wash
Amount	Char	Yes			50
InvoiceNo	Varchar	Yes		X	1

Query Matrix

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	SUM
1. Multiple Join	1	1	1	1	1	1	1			1	1		1	8
2. Subquery				1										1
3. GROUP BY		1	1	1		1								4
4. GROUP BY with HAVING										1			1	1
5. ORDER BY			1		1	1								3
6. Aggregate		1	1	1	1	1			1					6
7. LIKE									1					1
8. Date												1		0
9. IN/NOT IN										1			1	1
10. IS NULL							1	1						2