

Spotify Database System

CIS 3400

EMWA

Group 11

Stephen Chan (schan3436@gmail.com)

Steven Du (Steven.Du@baruchmail.cuny.edu)

Jackie Mok (jmok3634@gmail.com)

Andrew Ivanov (Andrew.Ivanov@baruchmail.cuny.edu)

Denny Huang (huangdenny35@gmail.com)

Project Proposal

- A narrative description of the business used for the project or application being created. This should also include a description of the problem or opportunity being addressed.

In Spotify, many accounts are created with different Spotify plans, followed up by multiple playlists and songs linked to those accounts. One of the few solutions in preventing issues from arising would be to create a Spotify database system. This database system will prevent such problems from happening, such as keeping track of which plans are accounts subscribed to, how many accounts are subscribed it, what type of music each account listens to, whose credit card to charge from, etc.

Opportunities that this database system can provide is the organization and tracking of payment information, customers, accounts, their subscribed plans, and the music they listen to. With this data, Spotify can easily figure out what a specific account is linked to.

- Identification of the information needs - what information would help solve the problem or allow one to take advantage of the opportunity.

A customer would have to sign up for an account by providing general information and also payment information. Then they will be given the opportunity to choose which Spotify plan to subscribe to so it will be linked to that account and charge it from the payment information given. Afterwards, whatever playlist and song the customer's associated account adds, it'll be stored with a unique ID.

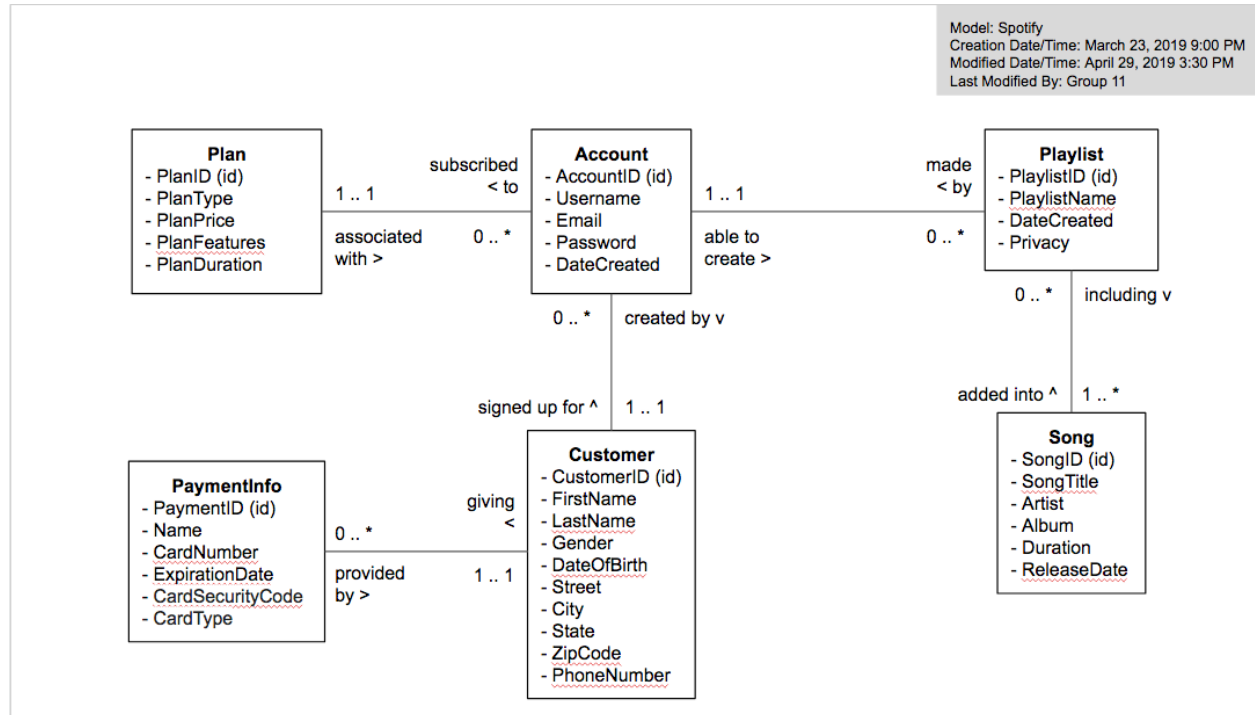
- Initial list of entities (tables) that have been identified. This should come naturally from the above discussions.

- Plan
- Account
- Customer
- Payment Information
- Playlist
- Song

- Distribution of duties for the project. List the names of each group member and what their primary role will be (e.g., systems analyst, application developer, documentation writer).

- Application Developer: Stephen Chan, Steven Du
- System Analyst: Denny Huang, Andrew Ivanov
 - Documentation: Jackie Mok

Entity-Relationship Model Diagram



Relationships

One **PaymentInfo** must be *provided by* one and only one **Customer**.
One **Customer** may be *giving* one or more **PaymentInfo**.

One **Plan** may be *associated with* one or more **Accounts**.
One **Account** must be *subscribed to* one and only one **Plan**.

One **Customer** may be *signed up* for one or more **Accounts**.
One **Account** must be *created by* one and only one **Customer**.

One **Account** may be *able to create* one or more **Playlists**.
One **Playlist** must be *made by* one and only one **Account**.

One **Playlist** must be *including* one or more **Songs**.
One **Song** may be *added into* one or more **Playlists**.

Normalization

Plan (PlanID (key), PlanType, PlanPrice, PlanFeatures, PlanDuration)

Account (AccountID (key), Username, Email, Password, DateCreated, PlanID (fk), CustomerID (fk))

Customer (CustomerID (key), FirstName, LastName, Gender, DateOfBirth, Street, City, State, ZipCode, PhoneNumber)

PaymentInfo (PaymentID (key), Name, CardNumber, ExpirationDate, CardSecurityCode, CardType, CustomerID (fk))

Playlist (PlaylistID (key), PlaylistName, DateCreated, Privacy, AccountID (fk))

Song (SongID (key), SongTitle, Artist, Album, Duration, ReleaseDate)

Plan Relation

Plan (PlanID (key), PlanType, PlanPrice, PlanFeatures, PlanDuration)

FD1: PlanID (key) \rightarrow PlanType, PlanPrice, PlanFeatures, PlanDuration

1NF: Yes, because it meets the definition of a relation.

2NF: Yes, because there is no partial key dependency.

3NF: Yes, because there is no transitive dependency.

Account Relation

Account (AccountID (key), Username, Email, Password, DateCreated, PlanID (fk), CustomerID (fk))

FD1: AccountID (key) \rightarrow Username, Email, Password, DateCreated, PlanID (fk), CustomerID (fk)

1NF: Yes, because it meets the definition of a relation.

2NF: Yes, because there is no partial key dependency.

3NF: Yes, because there is no transitive dependency.

Customer Relation

Customer (CustomerID (key), FirstName, LastName, Gender, DateOfBirth, Street, City, State, ZipCode, PhoneNumber)

FD1: CustomerID (key) \rightarrow FirstName, LastName, Gender, DateOfBirth, Street, City, State, ZipCode, PhoneNumber

1NF: Yes, because it meets the definition of a relation.

2NF: Yes, because there is no partial key dependency.

3NF: Yes, because there is no transitive dependency.

Payment Info Relation

PaymentInfo (PaymentID (key), Name, CardNumber, ExpirationDate, CardSecurityCode, CardType, CustomerID (fk))

FD1: PaymentID (key) \rightarrow Name, CardNumber, ExpirationDate, CardSecurityCode, CardType, CustomerID (fk)

1NF: Yes, because it meets the definition of a relation.

2NF: Yes, because there is no partial key dependency.

3NF: Yes, because there is no transitive dependency.

Playlist Relation

Playlist (PlaylistID (key), PlaylistName, DateCreated, Privacy, AccountID (fk))

FD1: PlaylistID (key) \rightarrow PlaylistName, DateCreated, Privacy, AccountID (fk)

1NF: Yes, because it meets the definition of a relation.

2NF: Yes, because there is no partial key dependency.

3NF: Yes, because there is no transitive dependency.

Song Relation

Song (SongID (key), SongTitle, Artist, Album, Duration, ReleaseDate)

FD1: SongID (key) \rightarrow SongTitle, Artist, Album, Duration, ReleaseDate

1NF: Yes, because it meets the definition of a relation.

2NF: Yes, because there is no partial key dependency.

3NF: Yes, because there is no transitive dependency.

Playlist Song Relation

PlaylistSong (PlaylistID (fk), SongID (fk))

FD1: PlaylistSong \rightarrow PlaylistID (fk), PlaylistID (fk)

1NF: Yes, because it meets the definition of a relation.

2NF: Yes, because there is no partial key dependency.

3NF: Yes, because there is no transitive dependency.

Final Set of Relations

Plan (PlanID (key), PlanType, PlanPrice, PlanFeatures, PlanDuration)

Account (AccountID (key), Username, Email, Password, DateCreated, PlanID (fk), CustomerID (fk))

Customer (CustomerID (key), FirstName, LastName, Gender, DateOfBirth, Street, City, State, ZipCode, PhoneNumber)

PaymentInfo (PaymentID (key), Name, CardNumber, ExpirationDate, CardSecurityCode, CardType, CustomerID (fk))

Playlist (PlaylistID (key), PlaylistName, DateCreated, Privacy, AccountID (fk))

Song (SongID (key), SongTitle, Artist, Album, Duration, ReleaseDate)

PlaylistSong (PlaylistID (fk), SongID (fk))

Plan SQL

```
CREATE TABLE Plan
(
    PlanID          VARCHAR(10) NOT NULL,
    PlanType        VARCHAR(15),
    PlanPrice       MONEY,
    PlanFeatures    VARCHAR(20),
    PlanDuration    INTEGER,
    CONSTRAINT      pk_plan
        PRIMARY KEY (PlanID)
);

INSERT INTO Plan VALUES ( 'P101', 'Free', '0', 'Ads',
'999999' );

INSERT INTO Plan VALUES ( 'P102', 'Premium', "9.99",
'NoAds', '30' );

INSERT INTO Plan VALUES ( 'P103', 'Student Premium',
"4.99", 'NoAds', '30' );

INSERT INTO Plan VALUES ( 'P104', 'Family Premium',
"14.99", 'NoAds', '30' );
```

Customer SQL

```
CREATE TABLE Customer
```

```
(
    CustomerID          VARCHAR(10) NOT NULL,
    FirstName            VARCHAR(20),
    LastName             VARCHAR(20),
    Gender               VARCHAR(10),
    DateOfBirth          DATE,
    Street               VARCHAR(35),
    City                 VARCHAR(35),
    State                VARCHAR(35),
    Zipcode              INTEGER,
    PhoneNumber          VARCHAR(20),
    CONSTRAINT pk_customer
        PRIMARY KEY (CustomerID)
);
```

```
INSERT INTO Customer VALUES ( 'C101', 'Steven', 'Du',
'Male', "01/02/1997", "1234 86th St", 'Brooklyn', "New
York", '11214', "646-654-2524" );
```

```
INSERT INTO Customer VALUES ( 'C102', 'Stephen',
'Chan', 'Male', "02/03/1997", "2234 E 8th St",
'Manhattan', "New York", '10029', "646-123-6784" );
```

```
INSERT INTO Customer VALUES ( 'C103', 'Jackie', 'Mok',
'Male', "04/05/1997", "1243 8th Ave", 'Brooklyn', "New
York", '11219', "347-246-8312" );
```

```
INSERT INTO Customer VALUES ( 'C104', 'Denny', 'Huang',
'Female', "06/07/1997", "12 Elizabeth St", 'Manhattan',
"New York", '10013', "347-853-1267" );
```

```
INSERT INTO Customer VALUES ( 'C105', 'Andy', 'Ivanov',
'Male', "08/09/1997", "56-43 Main St", 'Flushing', "New
York", '11367', "347-1236-789" );
```

Account SQL

```
CREATE TABLE Account
```

```
(
    AccountID          VARCHAR(10) NOT NULL,
    Username            VARCHAR(30),
    Email               VARCHAR(255),
    Password            VARCHAR(255),
    DateCreated         DATE,
    PlanID              VARCHAR(15),
    CustomerID          VARCHAR(15),
    CONSTRAINT pk_account
        PRIMARY KEY (AccountID)
);
```

```
ALTER TABLE Account
```

```
    ADD CONSTRAINT fk_Account_Plan
        FOREIGN KEY (Planid)
        REFERENCES Plan (Planid)
```

```
ALTER TABLE Account
```

```
    ADD CONSTRAINT fk_Account_Customer
        FOREIGN KEY (Customerid)
        REFERENCES Customer (Customerid)
```

```
INSERT INTO Account VALUES ( 'A101', 'steven123',
"steven123@gmail.com", "thisismypassword", "04/22/2019", 'P103',
'C101' );
```

```
INSERT INTO Account VALUES ( 'A102', 'stephen123',
"stephen123@gmail.com", "donthackmeplease1", "03/22/2019",
'P104', 'C102' );
```

```
INSERT INTO Account VALUES ( 'A103', 'jmok123',
"jackiemok123@gmail.com", "123456789abc", "04/28/2019", 'P102',
'C103' );
```

```
INSERT INTO Account VALUES ( 'A104', 'denden123',
"dhuang123@gmail.com", "ilovenaruto123", "01/21/2019", 'P101',
'C104' );
```

```
INSERT INTO Account VALUES ( 'A105', 'theandy123',
"theandy123@gmail.com", "p@$s$w0rd", "04/22/2019", 'P104', 'C105'
);
```


PaymentInfo SQL

```
CREATE TABLE PaymentInfo
(
    PaymentID          VARCHAR(10) NOT NULL,
    Name               VARCHAR(40),
    CardNumber         VARCHAR(16),
    ExpirationDate     VARCHAR(5),
    CardSecurityCode   INTEGER,
    CardType           VARCHAR(30),
    CustomerID         VARCHAR(15),
    CONSTRAINT pk_payment_info
        PRIMARY KEY (PaymentID)
);

ALTER TABLE PaymentInfo
    ADD CONSTRAINT fk_PaymentInfo_Customer
        FOREIGN KEY (CustomerID)
            REFERENCES Customer (CustomerID)

INSERT INTO PaymentInfo VALUES ( 'PA101', "Steven Du",
    '4054829344627004', "10/27", '743', 'VISA', 'C101' );

INSERT INTO PaymentInfo VALUES ( 'PA102', "Stephen
Chan", '6011123450029003', "11/22", '679', 'DISCOVER',
    'C102' );

INSERT INTO PaymentInfo VALUES ( 'PA103', "Jackie Mok",
    '4003152367899002', "03/25", '920', 'VISA', 'C103' );

INSERT INTO PaymentInfo VALUES ( 'PA104', "Denny
Huang", '4552348985201245', "12/03", '920', 'VISA',
    'C104' );

INSERT INTO PaymentInfo VALUES ( 'PA105', "Andy
Ivanov", '3221005236951124', "05/21", '852', "American
Express", 'C105' );
```

Playlist SQL

```
CREATE TABLE Playlist
```

```
(
```

```
    PlaylistID          VARCHAR(10) NOT NULL,
```

```
    PlaylistName        VARCHAR(255),
```

```
    DateCreated          DATE,
```

```
    Privacy              VARCHAR(10),
```

```
    AccountID            VARCHAR(15),
```

```
    CONSTRAINT           pk_playlist
```

```
        PRIMARY KEY (PlaylistID)
```

```
);
```

```
ALTER TABLE Playlist
```

```
    ADD CONSTRAINT fk_Playlist_Account
```

```
        FOREIGN KEY (AccountID)
```

```
            REFERENCES Account (AccountID)
```

```
INSERT INTO Playlist VALUES ( 'PL101', "Good Vibes",  
"04/16/2015", 'Public', 'A101' );
```

```
INSERT INTO Playlist VALUES ( 'PL102', "Discover  
Weekly", "04/22/2019", 'Public', 'A102' );
```

```
INSERT INTO Playlist VALUES ( 'PL103', "Korean OST",  
"04/28/2019", 'Private', 'A101' );
```

```
INSERT INTO Playlist VALUES ( 'PL104', "Jazz Vibes",  
"04/01/2019", 'Public', 'A103' );
```

```
INSERT INTO Playlist VALUES ( 'PL105', "ASMR",  
"04/20/2019", 'Private', 'A104' );
```

Song SQL

```
CREATE TABLE Song
```

```
(  
    SongID          VARCHAR(10) NOT NULL,  
    SongTitle       VARCHAR(255),  
    Artist          VARCHAR(255),  
    Album           VARCHAR(255),  
    Duration        VARCHAR(255),  
    ReleaseDate     DATE,  
    CONSTRAINT pk_song  
        PRIMARY KEY (SongID)  
);
```

```
INSERT INTO Song VALUES ( 'S101', "Earth", "Lil Dicky",  
"Earth", "4m 41s", "04/18/2019" );
```

```
INSERT INTO Song VALUES ( 'S102', "Universe", "EXO",  
"Universe - Winter Special Album, 2017", "4m 24s",  
"12/26/2017" );
```

```
INSERT INTO Song VALUES ( 'S103', "Thotiana",  
"Blueface", "Famous Cryp", "2m 9s", "12/18/2018" );
```

```
INSERT INTO Song VALUES ( 'S104', "It's You", "Henry",  
"당신이 잠든 사이에 Pt. 2 Original Television Soundtrack",  
"3m 52s", "10/04/2017" );
```

```
INSERT INTO Song VALUES ( 'S105', "Spectrum", "Zedd",  
"Clarity", "4m 3s", "01/01/2012" );
```

Playlist Song SQL

```
CREATE TABLE PlaylistSong
(
    PlaylistID    VARCHAR(15),
    SongID        VARCHAR(15)
);

ALTER TABLE PlaylistSong
    ADD CONSTRAINT fk_Playlist
        FOREIGN KEY (PlaylistID)
            REFERENCES Playlist (PlaylistID)

ALTER TABLE PlaylistSong
    ADD CONSTRAINT fk_Song
        FOREIGN KEY (SongID)
            REFERENCES Song (SongID)

INSERT INTO PlaylistSong VALUES ( 'PL103', 'S104' );

INSERT INTO PlaylistSong VALUES ( 'PL101', 'S105' );

INSERT INTO PlaylistSong VALUES ( 'PL103', 'S102' );
```

Account Form

Account Form

AccountID: A101 DateCreated: 4/22/2019

Username: steven123

Password: thisismypassword

Plan: Premium

PlaylistSong

Open Spotify

PlaylistID	SongID	PlaylistName	DateCreated	Privacy	SongTitle	AccountID
PL101	S105	Good Vibes	4/16/2015	Public	Spectrum	A101
PL103	S102	Korean OST	4/28/2019	Private	Universe	A101
PL103	S104	Korean OST	4/28/2019	Private	It's You	A101
*						

Records: 1 of 3 No Filter Search

Selecting a Plan

Account Form

AccountID: A101 DateCreated: 4/22/2019

Username: steven123

Password: thisismypassword

Plan: Premium

PlaylistSong

Open Spotify

PlaylistID	SongID	PlaylistName	DateCreated	Privacy	SongTitle	AccountID
PL101	S105	Good Vibes	4/16/2015	Public	Spectrum	A101
PL103	S102	Korean OST	4/28/2019	Private	Universe	A101
PL103	S104	Korean OST	4/28/2019	Private	It's You	A101
*						

Records: 1 of 3 No Filter Search

Spotify Premium

Thank you for joining Premium!

OK

Upgrading from Free Plan to Premium Plan

AccountID: A101 DateCreated: 4/22/2019

Username: steven123

Password: thisismypassword

Plan: Free

Open Spotify

PlaylistID	SongID	PlaylistName	DateCreated	Privacy	SongTitle	AccountID
PL101	S105	Good Vibes	4/16/2015	Public	Spectrum	A101
PL103	S102	Korean OST	4/28/2019	Private	Universe	A101
PL103	S104	Korean OST	4/28/2019	Private	It's You	A101

Record: 1 of 1

Spotify Free

Upgrade to Premium now for no Ads!

Yes No

Account Form

AccountID: A101 DateCreated: 4/22/2019

Username: steven123

Password: thisismypassword

Plan: Premium

Open Spotify

PlaylistID	SongID	PlaylistName	DateCreated	Privacy	SongTitle	AccountID
PL101	S105	Good Vibes	4/16/2015	Public	Spectrum	A101
PL103	S102	Korean OST	4/28/2019	Private	Universe	A101
PL103	S104	Korean OST	4/28/2019	Private	It's You	A101

Record: 1 of 3

Spotify Premium

Welcome to Premium!

OK

VBA Code

Spotify Welcome to Premium and Ask to Upgrade if on Free Plan

```
Private Sub Combo20_AfterUpdate()  
Dim message As String  
  
Select Case Me.Combo20  
Case "Free"  
message = MsgBox("Upgrade to Premium now for no Ads!", vbYesNo,  
"Spotify Free")  
  
    If message = vbYes Then  
        Combo20.Value = "Premium"  
        message = MsgBox("Welcome to Premium!", vbOKOnly,  
"Spotify Premium")  
    End If  
  
Case "Premium"  
message = MsgBox("Thank you for joining Premium!", vbOKOnly,  
"Spotify Premium")  
  
Case ("Student Premium")  
message = MsgBox("Thank you for joining Premium!", vbOKOnly,  
"Spotify Premium")  
  
Case ("Family Premium")  
message = MsgBox("Thank you for joining Premium!", vbOKOnly,  
"Spotify Premium")  
  
End Select  
  
End Sub
```

Open Spotify

```
Private Sub Command22_Click()  
  
Dim myPath As String  
  
myPath = "C:\Users\AppData\Roaming\Spotify\Spotify.exe"  
  
Call Shell(myPath, 1)  
  
End Sub
```

Account Subform

Account Subform

AccountID	A101
Email	steven123@gmail.com
Username	steven123
Password	thisismypassword



Customer Form

Customer

Customer

AccountID	A101
CustomerID	C101
FirstName	Steven
LastName	Du
Gender	Male
DateOfBirth	1/2/1997
Street	1234 86th St
City	Brooklyn
State	New York
Zipcode	11214
PhoneNumber	646-654-2524

Customer1

Customer Payment

CustomerID

C101

FirstName

Steven

LastName

Du

PaymentInfo

PaymentID	Name	CardNumber	ExpirationDate	CardSecurityCode	CardType	CustomerID
PA101	Steven Du	4054829344627004	10/27	743	VISA	C101
*						C101

Record: 1 of 1

No Filter
Search

Customer Payment Subform

PaymentInfo Subform						
PaymentID ▾	Name ▾	CardNumber ▾	ExpirationDate ▾	CardSecurityCode ▾	CardType ▾	CustomerID ▾
PA101	Steven Du	4054829344627004	10/27	743	VISA	C101
PA102	Stephen Chan	6011123450029003	11/22	679	DISCOVER	C102
PA103	Jackie Mok	4003152367899002	03/25	920	VISA	C103
PA104	Denny Huang	4552348985201245	12/03	920	VISA	C104
PA105	Andy Ivanov	3221005236951124	05/21	852	American Exp	C105
*						

Playlist Song Subform

PlaylistSong Subform						
PlaylistID ▾	SongID ▾	PlaylistName ▾	DateCreated ▾	Privacy ▾	SongTitle ▾	AccountID ▾
PL101	S105	Good Vibes	4/16/2015	Public	Spectrum	A101
PL103	S102	Korean OST	4/28/2019	Private	Universe	A101
PL103	S104	Korean OST	4/28/2019	Private	It's You	A101
*						

Navigation Form

Navigation Form

Customer
Account Form
Customer Payment

Customer

AccountID	A101
CustomerID	C101
FirstName	Steven
LastName	Du
Gender	Male
DateOfBirth	1/2/1997
Street	1234 86th St
City	Brooklyn
State	New York
Zipcode	11214
PhoneNumber	646-654-2524

Navigation Form

Customer
Account Form
Customer Payment

AccountID: A101 DateCreated: 4/22/2019

Username: steven123

Password: thisismypassword

Plan:

PlaylistSong

Open Spotify

PlaylistID	SongID	PlaylistName	DateCreated	Privacy	SongTitle	AccountID
PL101	S105	Good Vibes	4/16/2015	Public	Spectrum	A101
PL103	S102	Korean OST	4/28/2019	Private	Universe	A101
PL103	S104	Korean OST	4/28/2019	Private	It's You	A101
*						

Navigation Form

Customer
Account Form
Customer Payment

Customer Payment

CustomerID: C101

FirstName: Steven

LastName: Du

PaymentInfo

PaymentID	Name	CardNumber	ExpirationDate	CardSecurityCode	CardType	CustomerID
PA101	Steven Du	4054829344627004	10/27	743	VISA	C101
*						