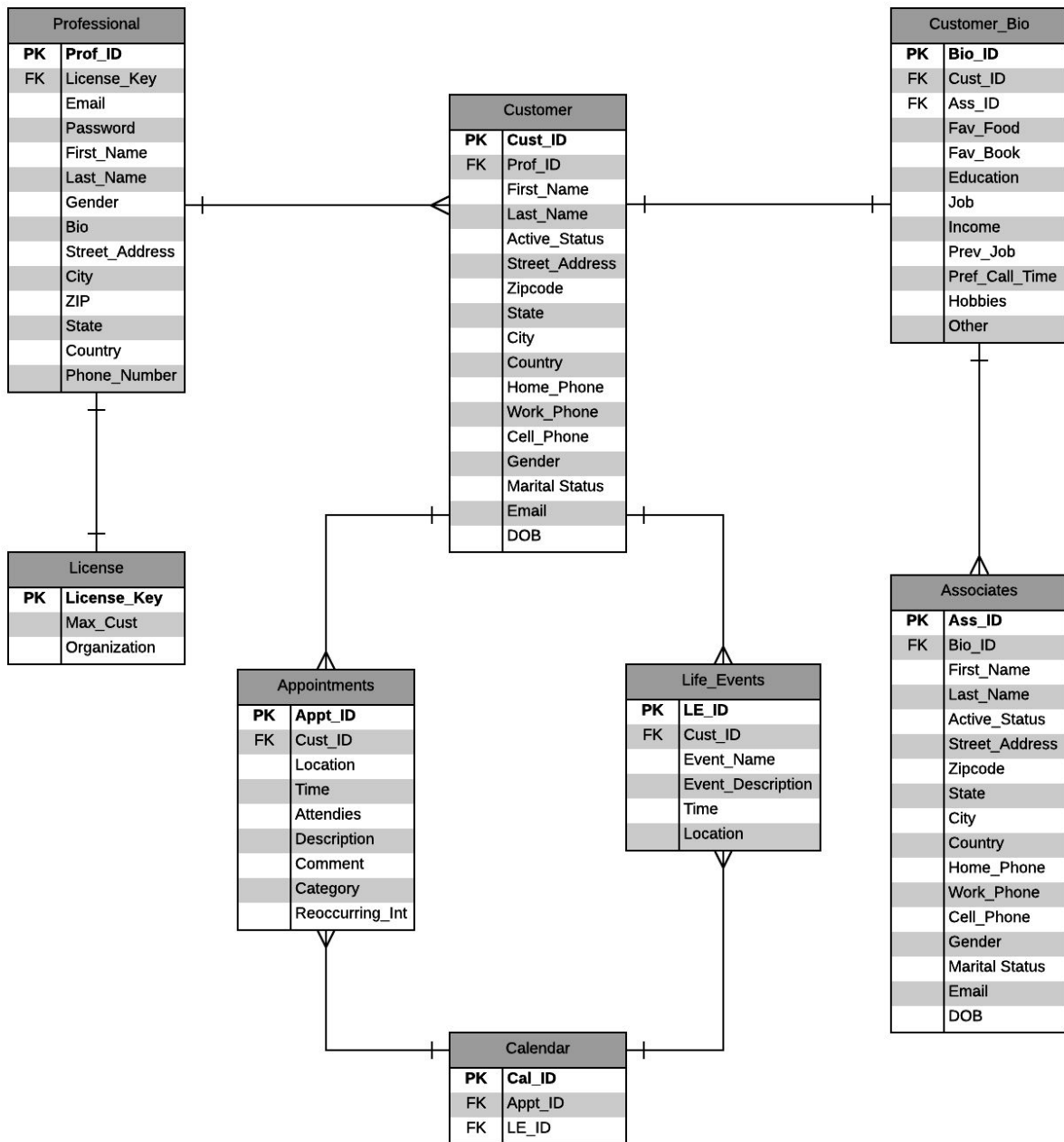


ER Diagram & Documentation

In this document we, The Salty Groundhogs, have broken down our ER diagram into a universally communicable explanation of our tables and relationships.

ER Diagram



Professional Table

The *Professional Table* stores all the data for the professionals. We have chosen “Prof_ID” to be our primary key so we can uniquely identify all professionals in the table. There is a foreign key, “License_Key”, which references the *License Table*, and is used to enforce that all professionals on True Course have paid and are legitimate users. The *License Table* and the *Professional Table* share a one-to-one relationship which means there is one license per professional and one professional per license. The *Professional Table* and the *Customer Table* share a one-to-many relationship from the professional table. This relationship allows one professional to have many customers.

License Table

The *License Table* stores all the data for user access licenses. We have chosen “License_Key” to be our primary key so we can uniquely identify all licenses that True Course gives out. The *License Table* and the *Professional Table* share a one-to-one relationship, which means there is one license per professional and one professional per license

Customer Table

The *Customer Table* stores all the data on True Course customers. We have chosen “Cust_ID” to be our primary key so we can uniquely identify all customers in the table. There is a foreign key, “Prof_ID”, which references the *Professionals Table*, and is used to assign customers to their professionals. The *Customer Table* and the *Professional Table* share a one-to-many relationship originating from the professionals table. This relationship allows one professional to have many customers. The *Customer Table* and the *Customer_Bio Table* share a one-to-one relationship, which enforces that one customer has one bio. The *Customer Table* and the *Life_Events Table* share a one-to-many relationship, from the *Customer Table* side, which allows for a customer to have many life events. The *Customer Table* and the *Appointments Table* share a one-to-many relationship, from the *Customer Table* side, which allows one customer to have many appointments.

Customer_Bio Table

The *Customer_Bio Table* stores all of the customers biography data that is relevant to True Course. We have chosen “Bio_ID” to be our primary key so all customer bios can be uniquely identified. There is a foreign key “Cust_ID”, which references the *Customer Table*, and is used to connect customer bios to their respective customers. The *Customer Table* and *Customer_Bio Table* share a one-to-one relationship, which enforces that one customer has one bio. There is a foreign key “Ass_ID”, which references the *Associates Table*, and is used to connect associates to customer bios. The *Customer_Bio Table* and the *Associates Table* share a one-to-many relationship, from the *Customer_Bio Table*, which allows one bio to have many associates.

Associates Table

The *Associates Table* stores all the data for the customers associates. We have chosen “Ass_ID” to be our primary key so all associates can be uniquely identified. There is a foreign key “Bio_ID”, which references the *Customer_Bio Table*, and is used to connect customer bios to their associates. There is a one-to-many relationship between the *Customer_Bio Table* and the *Associates Table*, from the *Customer_Bio Table* side, which allows one customer bio to have many associates.

Life_Events Table

The *Life_Events Table* stores all the data for customer life events. We have chosen “LE_ID” to be our primary key so all life events can be uniquely identified. There is a foreign key “Cust_ID”, which references the *Customer Table*, and is used to connect customers to their life events. There is a one-to-many relationship between the *Customer Table* and the *Life_Events Table*, from the *Customer Table* side, which allows one customer to have many life events. There is a one-to-many relationship between the *Calendar Table* and the *Life_Events Table*, from the *Calendar Table* side, which allows the calendar to store many life events.

Appointments Table

The *Appointments Table* stores all the data for customer appointments. We have chosen “Appt_ID” to be our primary key so all appointments can be uniquely identified. There is a foreign key “Cust_ID”, which references the *Customer Table*, and is used to connect customers to their appointments. There is a one-to-many relationship between the *Customer Table* and the *Appointments Table*, from the *Customer Table* side, which allows one customer to schedule many appointments. There is a one-to-many relationship between the *Calendar Table* and the *Appointments Table*, from the *Calendar Table* side, which allows a calendar to store many appointments.

Calendar Table

The *Calendar Table* stores all data for managing customer calendars. We have chosen “Cal_ID” to be our primary key so all calendar events can be uniquely identified. There is a foreign key “Appt_ID”, which references the *Appointments Table*, and is used to connect appointments to the calendar. There is a one-to-many relationship between the *Calendar Table* and the *Appointment Table*, from the *Calendar Table* side, which allows for a calendar to store many appointments. There is a foreign key “LE_ID”, which references the *Life Events Table*, and is used to connect life events to the calendar. There is a one-to-many relationship between the *Calendar Table* and the *Life Events Table*, from the *Calendar Table* side, which allows for a calendar to store many life events.