Project Statement

This project demo allows users to navigate destinations records, and the logged restaurants at each destination.

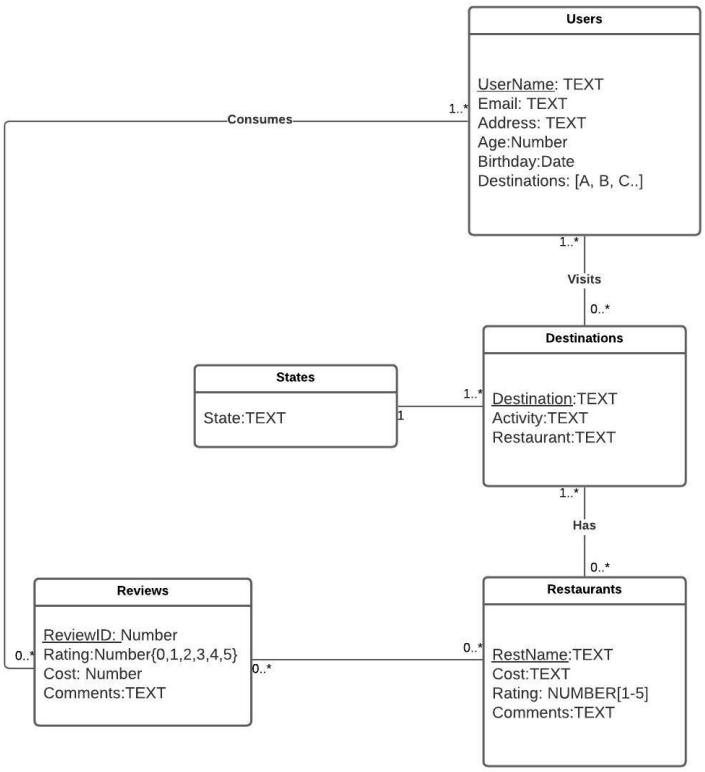
Rules:
Navigation Bar: Direct users to restaurants list and destination lists.

Manager page:
Users can add destination, and it's state with given state ID(see appendix 1);
Users can search destination in the search bar;
Users can edit and delete destination in the manager pag, clicking "Edit" will direct user to edit destination page.

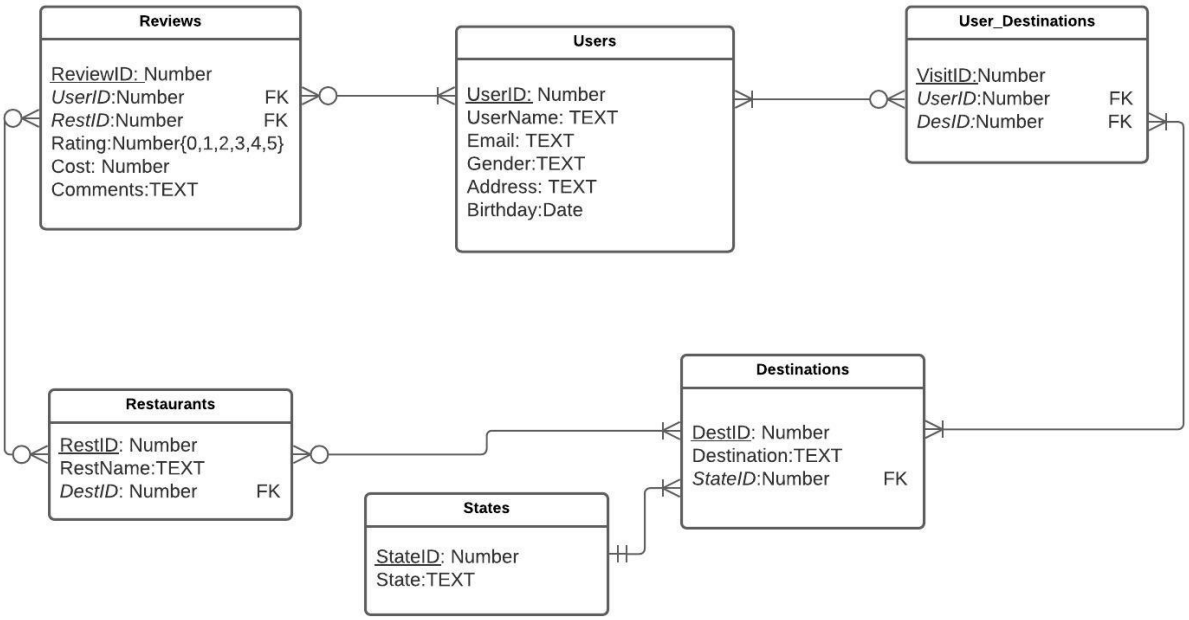
Edit Destination Page:
Users can update destinations and state with give state ID;
User can delete destinations with delete button;
Users can add and delete restaurants under the specific destination;
Users can click link of restaurants and directed to restaurant info page.

Restaurants Page:
This page display restaurant information of average rating, average cost, and reviews from other users.

Conceptual Modeling:



Logical Modeling:



Relational Schema:

Destinations(DestID, Destination, *StateID*)
DestID→Destinations

States(StateID, *State*)
StateID→State

Restaurants(RestID, RestName, *DestID*)
RestID→RestName
DestID→RestID
Restaurants located at specific Destinations, it's one to mange relationship(one destination can have many restaurant) and this is constrained as foreign key.

Reviews(ReviewID, *UserID, RestID,* Rating, cost, comments)
ReviewID→ UserID, RestID, Rating, cost, comments

Users(UserID, username, email, gender, address, birthday, *visitID*)
UserID→username, email, gender, address, birthday, visitID

User_Destinations(VisitID, *UserID, DestID*)
VisitID →UserID, DestID
Users and Destination is many to many relationship, user_Destination table is used to track.

Appendix 1 StateID and State information lookup

1	CA
2	CT
3	OH
4	MO
5	DC
6	PA
7	WA
8	AZ
9	NV
10	NY
11	RI
12	NC
13	FL
14	TX
15	VA
16	MA
17	AL
18	TN
19	MI
20	OK
21	CO
22	IL