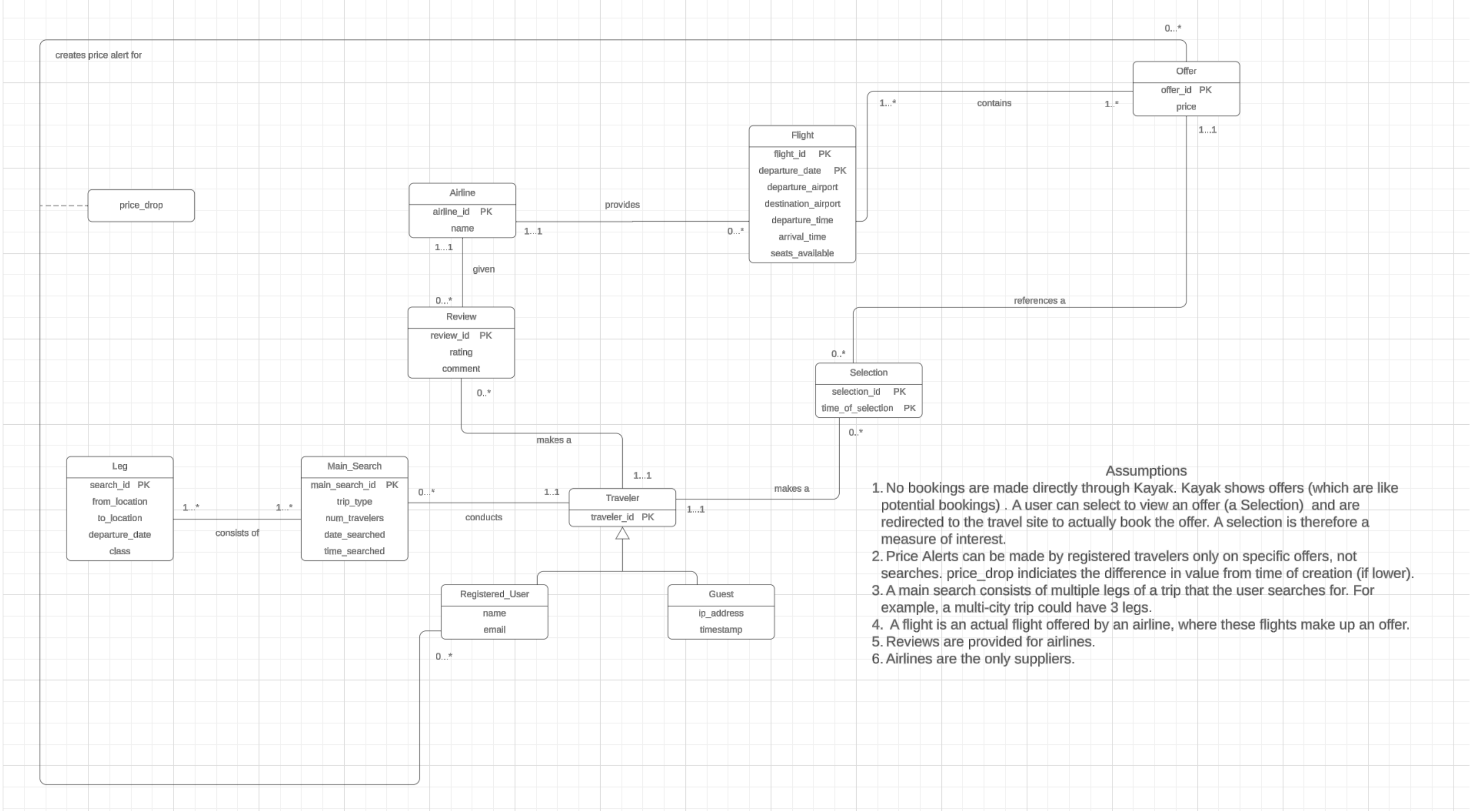
Phase 2 Report

Team #: 12

Team members: Ted Gershon, Ashley Liu

Updated Conceptual Model



Assumptions

1. No bookings are made directly through Kayak. Kayak shows offers (which are like potential bookings) . A user can select to view an offer (a Selection) and are redirected to the travel site to actually book the offer. A selection is therefore a measure of interest.
2. Price Alerts can be made by registered travelers only on specific offers, not searches. price_drop indiciates the difference in value from time of creation (if lower).
3. A main search consists of multiple legs of a trip that the user searches for. For example, a multi-city trip could have 3 legs.
4. A flight is an actual flight offered by an airline, where these flights make up an offer.
5. Reviews are provided for airlines.
6. Airlines are the only suppliers.

Updated User Stories

ID	Verb+Noun	As a <role>	I want <goal>	So that <reason>	Simple / Complex	Operational / Analytical
US1	Filter airlines	Traveler	To only see flights for a specific airline	I can book with my preferred airline and use loyalty points	Complex	Operational
US2	Create account	Traveler	To create an account profile	I can keep track of searches and prices, save flights, and get member-only deals	Simple	Operational
US3	Find lowest fare	Traveler	To find the cheapest flight	I spend the least amount of money on travel expenses	Complex	Operational
US4	Find quickest trip	Traveler	To find the quickest flight	I can spend the least amount of time on an airplane	Simple	Analytical
US5	Find popular searches	KAYAK Manager	To find which routes are currently the most searched	I can suggest trending recommendations to travelers	Complex	Analytical

US6	Find popular airlines	KAYAK Manager	To view what airlines' offers receive the most user engagement and selections	I can share insights with airline partners regarding their user engagement	Complex	Analytical
US7	Receive Price Alert	Registered Traveler	To receive Price Alerts through email about an offer	I can book when there is a noteworthy price drop to save money	Complex	Operational
US8	View airline reviews	Traveler	To see airline reviews indicating level of customer satisfaction	I can inform my travel arrangements with which providers have reliable pricing and good customer service	Complex	Operational
US9	View engagement	Airline	To view traffic that comes from KAYAK redirections and my performance compared to competitors	I can view how much my company is profiting from using KAYAK and make informed decisions based on traveler demand/engagement	Complex	Analytical
US10 **NEW**	Provide seating availability	Airline	To tell users of Kayak how many seats are left in a specified flight for all flights	I can help users make more informed decisions about what and when to book	Simple	Operational

Relational Model

Leg (**search_id**, from_location, to_location, departure_date, class)
Leg_in_Search (**search_id**, **main_search_id**)
Main_Search (**main_search_id**, trip_type, num_travelers, date_searched, time_searched, traveler_id)
Traveler (**traveler_id**)
Review (**review_id**, rating, comment, airline_id, traveler_id)
Airline (**airline_id**, name)
Registered_User (**traveler_id**, name, email)
Guest_User (**traveler_id**, ip_address, timestamp)
Price_Alert (**traveler_id**, **offer_id**, price_drop)
Flight (**flight_id, departure_date**, departure_airport, destination_airport, departure_time, arrival_time, seats_available, airline_id)
Flight_Offer (**flight_id**, **departure_date**, **offer_id**)
Offer (**offer_id**, price)
Selection (**selection_id**, **time_of_selection**, traveler_id, offer_id)

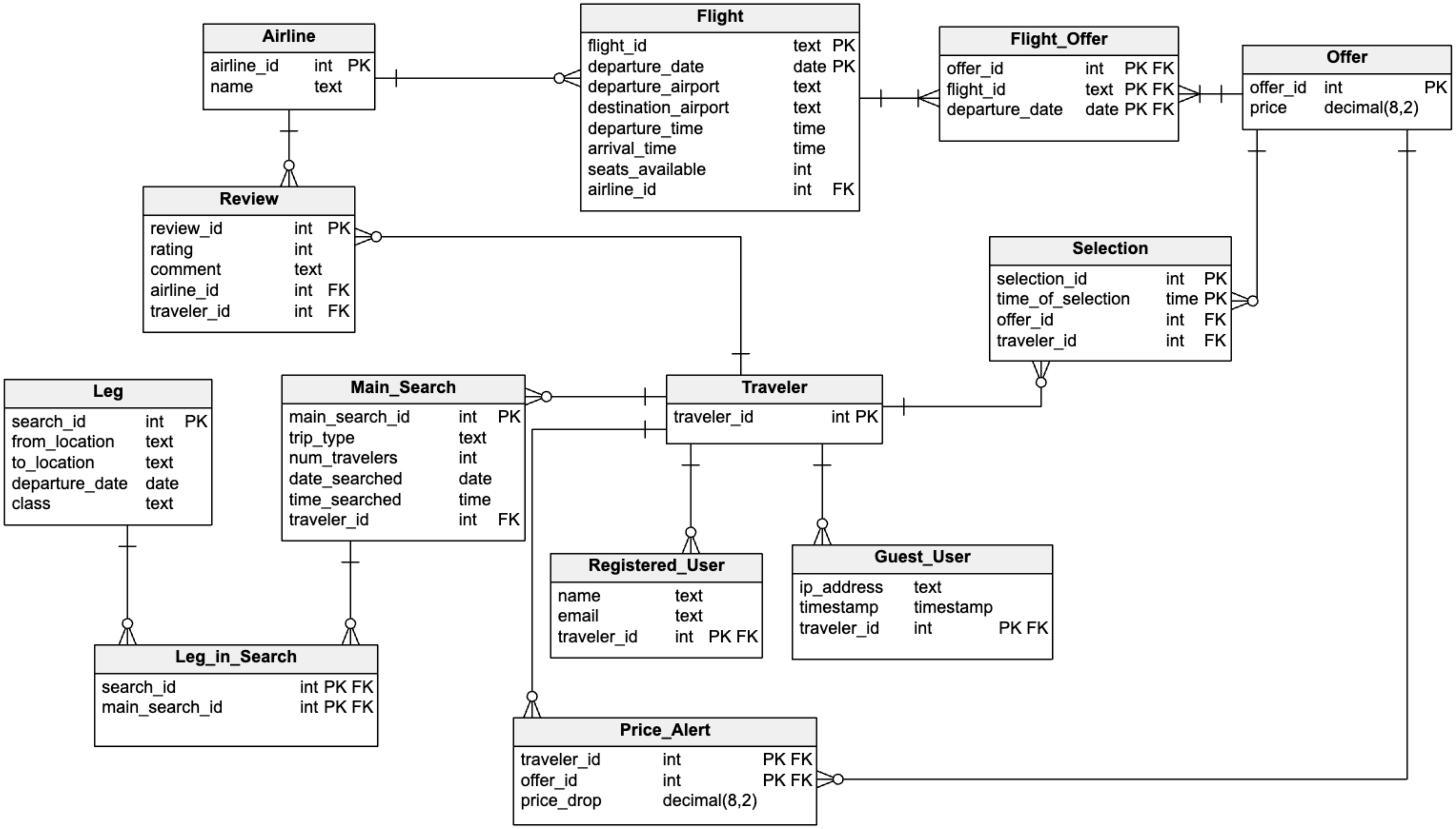
Functional Dependencies and Normalization

Relation	***Functional Dependencies***	***Normal Form/Justification***	***Assumptions for FDs***
Leg (**search_id**, from_location, to_location, departure_date, class)	search_id -> from_location, to_location, departure_date, class	BCNF because there are no bad FDs {search_id}+ = {search_id, from_location, to_location, departure_date, class}	
Leg_in_Search (**search_id**, **main_search_id**)	No FDs	BCNF because there are no bad FDs	
Main_Search (**main_search_id**, trip_type, num_travelers, date_searched, time_searched, traveler_id)	main_search_id -> trip_type, num_travelers, date_searched, time_searched, traveler_id	BCNF because there are no bad FDs {main_search_id}+ = {main_search_id, trip_type, num_travelers, date_searched, time_searched, traveler_id}	
Traveler (**traveler_id**)	No FDs	BCNF because there are no bad FDs	
Review (**review_id**, rating, comment, airline_id, traveler_id)	review_id -> rating, comment, airline_id, traveler_id	BCNF because there are no bad FDs {review_id}+ = {review_id, rating, comment, airline_id, traveler_id}	
Airline (**airline_id**, name)	airline_id -> name	BCNF because there are no bad FDs {airline_id}+ = {airline_id, name}	
Registered_User (**traveler_id**, name, email)	traveler_id -> name, email	BCNF because there are no bad FDs {traveler_id}+ = {traveler_id, name, email}	

Guest (**traveler_id**, ip_address, timestamp)	traveler_id -> ip_address, timestamp	BCNF because there are no bad FDs {traveler_id}+ = {traveler_id, ip_address, timestamp}	
Price_Alert (**traveler_id**, **offer_id**, price_drop)	traveler_id, offer_id -> price_drop	BCNF because there are no bad FDs {traveler_id, offer_id}+ = {traveler_id, offer_id, price_drop}	A registered user creates a price alert for a specific offer which keeps track of how much the price has changed (price_drop) from the traveler's original price
Flight (**flight_id, departure_date**, departure_airport, destination_airport, departure_time, arrival_time, seats_available, airline_id)	flight_id, departure_date -> departure_airport, destination_airport, departure_time, arrival_time, seats_available, airline_id	BCNF because there are no bad FDs {flight_id, departure_date}+ = {flight_id, departure_date, departure_airport, destination_airport, departure_time, arrival_time, seats_available, airline_id}	flight_id has unique characters in beginning that signal airline_id (i.e. all Alaska Airline flights begin id w/ "AS")
Flight_Offer (**flight_id**, **departure_date**, **offer_id**)	No FDs	BCNF because there are no bad FDs	
Offer (**offer_id**, price)	offer_id -> price	BCNF because there are no bad FDs {offer_id}+ = {offer_id, price}	
Selection (**selection_id, time_of_selection**, traveler_id, offer_id)	selection_id, time_of_selection -> traveler_id, offer_id	BCNF because there are no bad FDs {selection_id, time_of_selection}+ = {selection_id, time_of_selection, traveler_id, offer_id}	

*The initial set of relations are already in BCNF, so no normalization is needed.

Normalized Schema



Airline		
airline_id	int	PK
name	text	

Flight		
flight_id	text	PK
departure_date	date	PK
departure_airport	text	
destination_airport	text	
departure_time	time	
arrival_time	time	
seats_available	int	
airline_id	int	FK

Flight_Offer		
offer_id	int	PK FK
flight_id	text	PK FK
departure_date	date	PK FK

Offer		
offer_id	int	PK
price	decimal(8,2)	

Review		
review_id	int	PK
rating	int	
comment	text	
airline_id	int	FK
traveler_id	int	FK

Selection		
selection_id	int	PK
time_of_selection	time	PK
offer_id	int	FK
traveler_id	int	FK

Leg		
search_id	int	PK
from_location	text	
to_location	text	
departure_date	date	
class	text	

Main_Search		
main_search_id	int	PK
trip_type	text	
num_travelers	int	
date_searched	date	
time_searched	time	
traveler_id	int	FK

Traveler		
traveler_id	int	PK

Registered_User		
name	text	
email	text	
traveler_id	int	PK FK

Guest_User		
ip_address	text	
timestamp	timestamp	
traveler_id	int	PK FK

Leg_in_Search		
search_id	int	PK FK
main_search_id	int	PK FK

Price_Alert		
traveler_id	int	PK FK
offer_id	int	PK FK
price_drop	decimal(8,2)	