Dom Gordon (dlg2156), Mat Lirman (ml3707)
Andres Aguayo (aa3642), Julian Silerio (jjs2245)
ScheduLIT
Assignment 2

Diagram and CRC cards by Dom and Mat
Prose by Andres and Julian

User
The User class consists of a name, a unique ID, email, username-password login, and itinerary. A User can log-in, start a trip, remove a trip, view a trip, and view an itinerary. The User class consists of itineraries which he or she plans for a Trip, which itself has a collection of Users to denote who will be on said trip.

This class allows the client to interface with our application and begin planning the trip he or she would like to see. Because each User has a unique ID number as well as login info, the client will also be able to save his or her planned itineraries to refer to them at later points.

Trip
The Trip class consists of a name, a unique ID, a location, start and end date, a collection of Users on the Trip, and a collection of Itineraries. This class can also add and remove Users as well as set the location and start and end dates of the Trip.

The Trip class is the main structure of *ScheduLIT* around which much of the functionality is based. It's through this class that clients can compare the plans they'd like to make for their trips, which is the crux of our application.

Itinerary
The Itinerary class consists a unique ID, a name, and a collection of Events. This class can also create, remove, edit, and move Events as well get the owner (User object) of this itinerary.

Each User creates an Itinerary for the Trip they are going on in order to demonstrate the Events he or she would like to see as part of the Trip.

Event
The Event class consists of a start date and end date, name, a unique ID, and a description of the event. This class can also update its name, dates, and start and end times.

The Event class represents the actual items on the Itinerary for a User that he or she would like to see make it on to the overall schedule of the Trip.

Relationships

The User class *consists of* an Itinerary which the User users to plan his or her desired schedule for the Trip. Without the user class, the Itinerary would have little meaning standing on its own, hence the relationship between the User and Itinerary. Conversely, the User is a part of a collection for the Trip class, but because the User can plan multiple Trips, the Trip class only *has a* collection of Users instead of *consisting of* them.

The Trip class *has a* collection of Users to represent the catalog relationship that Trips share with Users, i.e. Users can exist without the Trip of which they are a part. Meanwhile, The Trip class *consists of* Itineraries because without the Trip, the Itineraries would have little meaning since the Itineraries are based around the Itinerary.

The Itinerary class *consists of* Events that a client would like to see on their Trip. Although an Event can be shared between multiple Itineraries, it does not carry enough weight to stand on its own without an Itinerary, hence the part-whole relationship that Events and Itineraries share.