Problem Statement:

Our team landmarkEvent, consisting of three members Pratik Jaipuria, Bhushan Mohite and Varun Nandu, is addressing the problem of finding and exploring current happening events in the city. Our app will contain Persons (Users) which are divided into four types: - Host, performer, Eventee and Admin. Persons contains the following fields personId, firstName, lastName, dateOfBirth, Email and Phone number. Admin has complete control over the application and can perform any CRUD operation on any of the pages in the app. We are exploring the following types of domain objects namely Ticket, Review, Events, Venues. The relationships between different domain objects and users are defined below. An Eventee attends events hosted by Host. An Eventee can also follow a performer and thus a performer has an additional field followed by which will contain list of eventees that follows him. An eventee can also find the performers upcoming schedule. A Venue has an availability field which shows which dates are available for an EventHost to book it. An Eventee checks in for an Event at a particular Venue. All the Persons will have a primary and secondary phone numbers. Apart from persons even a Venue will have a primary and secondary phone number. An eventee can follow other eventees. Thus, an eventee will have two additional fields, one referring to eventees he follows and one referring to eventees that follow him.

Performer performs in the Host Event. The Host sends invites to Performers to perform at the event. Inversely, the Performers can also request an Invitation from Event Host. When this request is made a portfolio is sent with the invite. Thus, portfolio field is inside Performer which is of type String. An eventee also reviews events. Events can be reviewed only by ticket where one ticket can add maximum of one review for the event. An Eventee can also review performers for which there is no need for a ticket. An eventee purchases tickets from the Host. Event host posts events, sells tickets for events, invites performers and can accept/decline invite request by performers. Additionally, an Event Host will

post events, sells tickets, invites performers and Accept/decline invitation of performer. An Event host will also contain two additional fields namely one which contains a list of past events hosted by him and an average rating of all the past events denoting the rating of the host. In our application an event will have one to many tickets. An event can take place in only one venue. An event can have many reviews. A venue can also have many reviews. One ticket lets an event post one review about the host. We are using [EventFul API](http://api.eventful.com/) to fetch data of events occurring by location.