Selected design patterns:

1. Proxy - The proxy pattern allows bridging between two existing legacy components in a coupled relationship. This situation is reminiscent of data management from the server to the client in the original Facebook application. The proxy constitutes a polymorphic item for one of the legacy and basically impersonates it. In this way one can apply more logic which could not be applied earlier.

I the current project a cache proxy was used in order to make the application more resilient to internet disconnections. The proxy caches the information on the first server engagement so it could be used it in an abrupt cut off.

1. Builder – The builder pattern is a method to build complex objects with a comfortable configuration interface. In this way replacing and reusing both the interface and the building component is easy by comparison.

In the current project it will be used in order to create the Midpoint object, since it is it is a made from the city object and the group list object. Both are also composite more objects. Furthermore, the midpoint object is very likely to be extended or altered and for those reasons, the builder pattern is almost required for this case.

1. dsa