**Implementation decisions:**

All my work is mainly based on the given code. The only classes I created are Stages, ListLocation, Displacement2.

As displacement class can only help to calculate distance and bearing of a point from origin. A Displacement2 was created to calculate the distance and bearing between two locations, i.e. from a position to a waypoint.

A lots of HashMap and ArrayList were used. For example, in the Stage class, I want something that can display the leg and waypoint annotation at the current stage, therefore I used HashMap since personally I think HashMap is a convenient way to achieve this goal. Stage class was created to store detailed information of a tour, including: total number of legs, waypoints and stages, also the information of leg and waypoint corresponding to each stage.

In the controllerImp class, it will create a HashMap in the form of (HashMap <String, ArrayList <Displacement>>), and A problem initializing of ArrayList<displacement> was encountered. To solve this problem, ListLocation class was created to avoid the initializing of the ArrayList. ListLocation class can store the ArrayList <Displacement> first, and it is then used in the HashMap.

All these implementation decisions are trying to minimize the changes to the skeleton code. Since the skeleton code is enough to achieve most of the requirements, therefore only few new classes were added.