**Visualizer component of CengBall Project**

**Requirements**: First of all, we need ‘percept’ class’s datatype. Thus, we need to decide datatypes of this class.

**Issues for short time:**

* Filling ‘percept’ class according to analyzed datatypes.
* Interface implementation for first iteration which only consist DEMO functionality.
* Creating json log file for initial data transfer from simulator to visualizer. This initial json file will be read only once. This json consists team, player relationship and its components. Its main purpose to evade logging player objects entities. With this json, we will match player ids with player facilities.
* Creating json percept files for demo visualization. DEMO functionality of visualizer uses demo.json log file and according to pitch-time-distance measurement of project visualize it.

**Other issues relating visualizer**

**Commentary issue:** Simulator will check current and brand new created percepts and according to the analyzed action, simulator attack commentary comments to percept object. Also simulator will be able to use player action class to ease analyzing.

**Do function issue:** Do function will return “player action class”. Sample player action object is below, former details will be explained later.

[Ball action: x, Type: shoot, direction: (x,y), power: p] (enumerated constants will be used here.)

**Measurement issue:** According to real football time-distance-pitch values global variables will be measured and defined. For example, a footballer can run on a pitch from one side to another approximately 30 seconds, then this measurement have to be scaled according to simulators values. For timing problem, for 10 minutes game and 24 frame/second, we need 14400 frames. Suppose 25% of them will be exact percepts, then we need 3600 percepts for a 10 minutes game, half of them are for one team. Also to keep consistency in visualizer, we will divide one percept to 4 another because of 25% measurement constraint.