**3) Subsystem Services**

**3.1) Player**

The player component is in the model and view of our MVC architecture. This component includes HeadDesign, Jersey, Coordinates and Status classes. Link between Interaction/Physics component and Player component provide the data about location that a player has. Player component takes the data that about the coordinations it and transfer the view controller about the location. Moreover, Player class also provides data about which animation is required for an interaction.

**3.2) Input Manager**

Input Manager component handles the inputs of the users and it is in the controller of our MVC architecture.. Basically, it collects the input data of the users from keyboard and tmanipulates our model, which controls the movements of their player.

**3.3) Interaction / Physics**

Interactions component is in the model architechtural desgin. It includes Collision, Power-up, Goals and Sound classes, and also implements Physics class. The data about interactions that player has during the game play is in the interaction component. It interacts with game manager and player component. The interaction/physics component provides the data about coordination and physics which is needed for both view and control architectures. In game, Powerups and physics are controlled and transfered to UI. Moreover, When required, sound class is used for notifying the users, like a shoot, goal or menu navigation sounds to add a new sense to the game.

**3.4) User Interface**

The user interface component is in the view of our MVC architecture. This component takes data from all other components on condition that direct and indirect. Player, Ball, Game manager,input manager and menu data components have the direct interaction with user interface component. The input manager provides hardware inputs, such as movements, powerup usages and menu option button usage to the user interface component. All movements of the players, balls etc. are displayed here, with control of the interaction/physics.

**3.5) Game Manager**

The game manager is in the model of MVC architecture. It handles all the game play features. The game manager interacts with interaction/physics and user interface components. It takes the game data from interaction component and sends the data of the game to user interface component.

**3.6) Menu Data**

Menu Data will be in the model of MVC architecture. This system consists of Help, Settings, Credits, Modes and Pause Menu and classes. Menu Data component will allow players to choose what type of game options they want to pick, namely characters, goal sizes, ball settings, randomising options etc. Also, reflecting changes of settings and transition between screens, Menu Data provides information for different menu screens (Pause menu, main menu, etc. )to user interface component.

**3.7)Ball**

Ball will be in the Model and view of MVC architecture. This system consists of Shape and Properties classes. The ball will be main focus on the gameplay and also be in the design. The location and physics of the ball will be calculated via interaction/physics and its data will be sent to UI. Different types of balls will be used and their properties will be checked on here. Moreover, the goal check will be used from here, since the coordinates of the ball defines the goal.

**3.8)Background**

Background will be in View of MVC architecture. This system consists of animation, Scoreboard and goals classes. Since the shapes of goals and the medium change during each round in the random mode, the change and properties will be controlled by here. Also, scores will be displayed in the background and this makes it interact with the interaction and UI classes.