**Traceability Matrix**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Cases** | **Features** | | | | | | |
|  | User Functions | Team functions | Tournament | Schedule | Brackets | Scoring System |
| Create Tournament |  |  | ⮠ |  |  |  |
| Create Brackets |  |  | ⮠ | ⮠ | ⮠ |  |
| Edit Brackets |  |  |  | ⮠ | ⮠ |  |
| Create Schedule |  |  | ⮠ | ⮠ |  |  |
| Edit Schedule |  |  |  | ⮠ |  |  |
| Delete Team |  | ⮠ |  |  |  |  |
| Edit Score |  |  |  |  |  | ⮠ |
| Enter Score |  |  | ⮠ |  |  | ⮠ |
| Register Account | ⮠ | ⮠ |  |  |  |  |
| View Published Scores |  |  |  |  |  | ⮠ |
| View Schedule |  |  | ⮠ | ⮠ |  |  |
| View Bracket |  |  | ⮠ | ⮠ | ⮠ |  |
| Register Team |  | ⮠ | ⮠ |  |  |  |

**Key Features:**

* Identify and prioritize features of the system you are supposed to implement

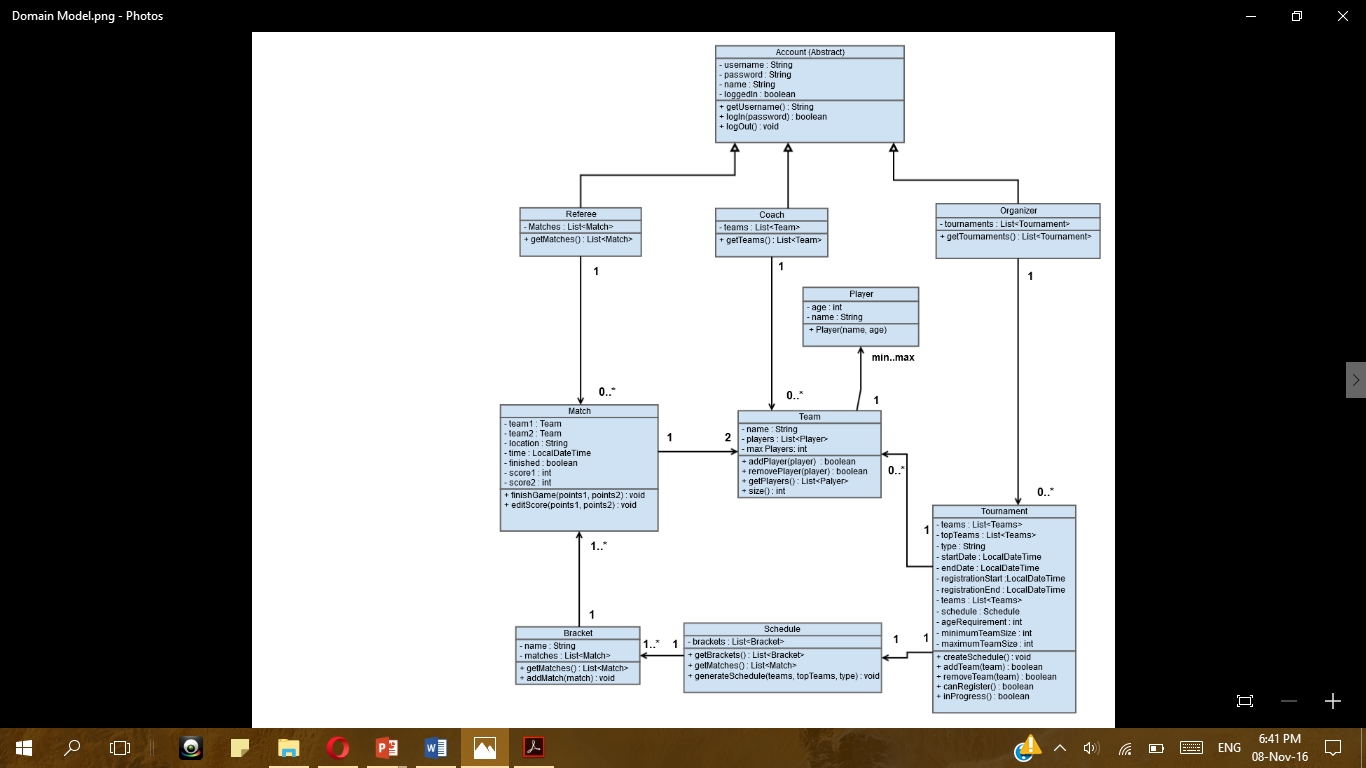
Functions of the system:

1. **Create a Tournament**
2. **Add/remove a team**
3. **Register an account for user**
4. **Add/remove players**
5. **Create bracket**
6. **Generate schedule**
7. **Add/edit scores**

Using the 3 Q’s of architecture we have managed to narrow the features down to only the architecturally significant features of the system. The 3 Q’s of architecture are the questions you ask in order to narrow down the features to be able to concentrate on the essential parts of the system.   
These questions are;   
**Q1 – Is it part of the essence of the system?  
Q2 – What does it really mean?  
Q3 – How can I implement it?**

**1 – Create a Tournament – Q1, Q3 –** The whole point of the system is to be able to create a tournament; this feature is the main core base of the system, but at the same time the implementation of this might be dependent on the other features within the system as creating a tournament.  
 **2 – Add/remove a team – Q1 –** In order for a tournament to generate a schedule, there must be a certain number of teams and it is essential to be able to get a tournament created. **3 – Register an account for user – Q3 –** It is not essential for now, but it is still something that needs figuring out in order to implement properly. **4 – Add/remove players – Q2 –** There are certain elements of adding a player that needs to be considered such as who can add the players? How many teams a player play for? Time is needed to fully consider the regulations for this feature. **5 – Create bracket – Q2, Q3 –** Creating a bracket comes after creating a schedule, this is a vague feature for now as it depends on a lot of other things. For example it is highly dependent on the format of the tournament and also we are still uncertain of how we can include all the necessary formats so time is needed to figure out what needs to be done. **6 – Generate Schedule – Q2, Q3 –** Generating a schedule is within creating a bracket, in order to create a schedule again a lot of things needs to be considered in the previous processes which means there are still a lot of uncertainties of how exactly the algorithm will be written. **7 – Add/edit scores – Q3 –** Again, this will only be able to feature within the system after a schedule has been created and for matches to have been played, so implementation for this feature will come in later after most of the other features have been figured out.

From this we have narrowed down the main features to;   
**1- Creating a Tournament   
2- Add/remove a team**

These two features are essential to the system as in order for a tournament to create a schedule teams will need to be added into it, so as a base of the system these two features will provide the core for the other features to be able to be implemented.

In terms of our domain model, the Tournament and Team classes are necessary to carry out the identified architecturally significant features of the system.

In terms of implementation for now we can ignore the accounts part as they are a planned feature that is for now not essential to the core and the implementation will mainly include the Tournament and Team class.