Software Architecture Model

Phase 1 Report

1) UseCase:

1) Actors:

In the usecase there are 6 actors : (Competitor , staff , officials , emergency response service , referee , System)

2) UseCases: (All bolded words considered usecases)

Competitors interact with the competition system by **registering** into the system and **login**, to complete this process the system interact as actor with the competition system as it **validate the input data** from the user before submitting it to review either it is accepted or not.

After the staff search for competitor by his number he fill his score then the competitor can search for his results by his number, then both of them can print out report of details about that competitor, the staff record competition details, after the competition is over the staff can request result details, the staff can access the GUI.

As not all staff have the same access level so officials inherit from staff all his access and have their own level of access (managing the competitors by either register them or remove them)

Referee role in the system is to ensure the fair play in all competitions between all competitors

Emergency response service **provide support** for referee , staff , officials and competitors

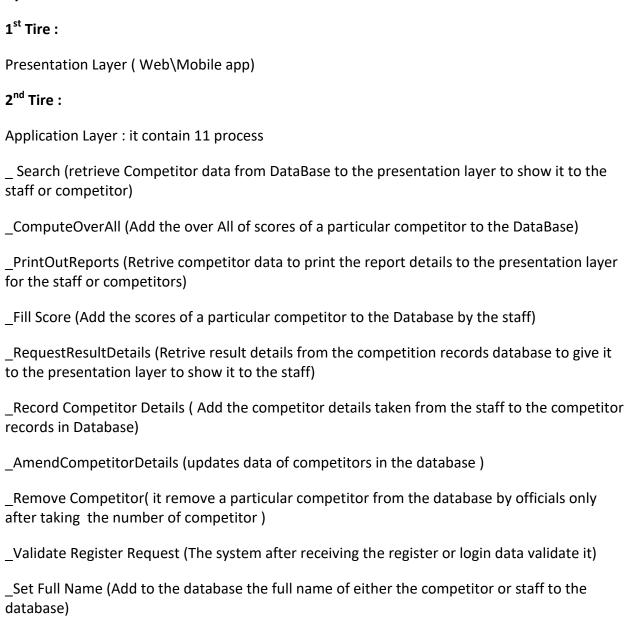
2) Class Diagram:

There are 6 classes in the class diagram

- 1) Name Class have private attributes (first name and second name), operations SetFullName that set the full name by adding the first name into the second name, get full name that gets the first name + second name
- **2) competitor class** inherit name (have its attributes and operations) it have private attributes (ID , Email, Password , competition type , age , country, scores, level) and it have operations
- Login that validate the input data from the user to login
- _Register allow competitor to input their data for further login into the system
- _SearchForCompetitor allow the competitor to search for his details and it have Boolean to print out report or not after searching
- _PrintOutReports it Print out reports for details about the competitor after searching
- _ComputeOverAll it take the scores of the competitor and compute its overall score
- _RequestSupport return a Boolean whether support is needed or not from the emergency response service
- **3) staff class** inherit name (have its attributes and operations) it have private attribute(id) it have operations :
- _FillScore that fill the scores of the competitors in the competition so the whole staff and competitors can search for this competitors
- RequestResultDetails provide the staff by the competition result details of the competitors
- _SearchForCompetitor the staff can search for a particular competitor to get his details , he can print out the report details
- _PrintOutReport it gets a Boolean from search for competitor operation to print particular competitor report details
- _RecordCompetitorDetails the staff after getting the details of the competitor he record those details to the system
- _RequestSupport return a Boolean whether support is needed or not from the emergency response service
- AmendCompetitorDetails the staff update the competitor records by this operation

- 4) Emergency response service class it only have one operation
- _ProvideSupport it accept the support request from staff and competitors
- 5) Officials class inherit staff (have its attributes and operations) it have two operations :
- _RegisterCompetitor the officials manage the registration process of the competitor
- _RemoveCompetitor the official have access to remove the competitor from the system
- **6) System** it have only one operation :
- _ValidateInputData it control the register and login process by validating them

3) Architecture Model



Get Full Name (it gets from the database the name of either a staff or competitor)

3rd Tire

DataBase Layer

It contains the database of the system

_Competitor records

_Staff records

_Competition Records