CS346 - Software Engineering Lab

Design

of the

Inter-IIT Tournament Management System

Prepared by-

Group No. 16

Annapurne Krishna Manik	180101009
Anshul Mittal	180101010
Munindra Naik	180101045
Sanket Kisan Pandhare	180101069

Introduction

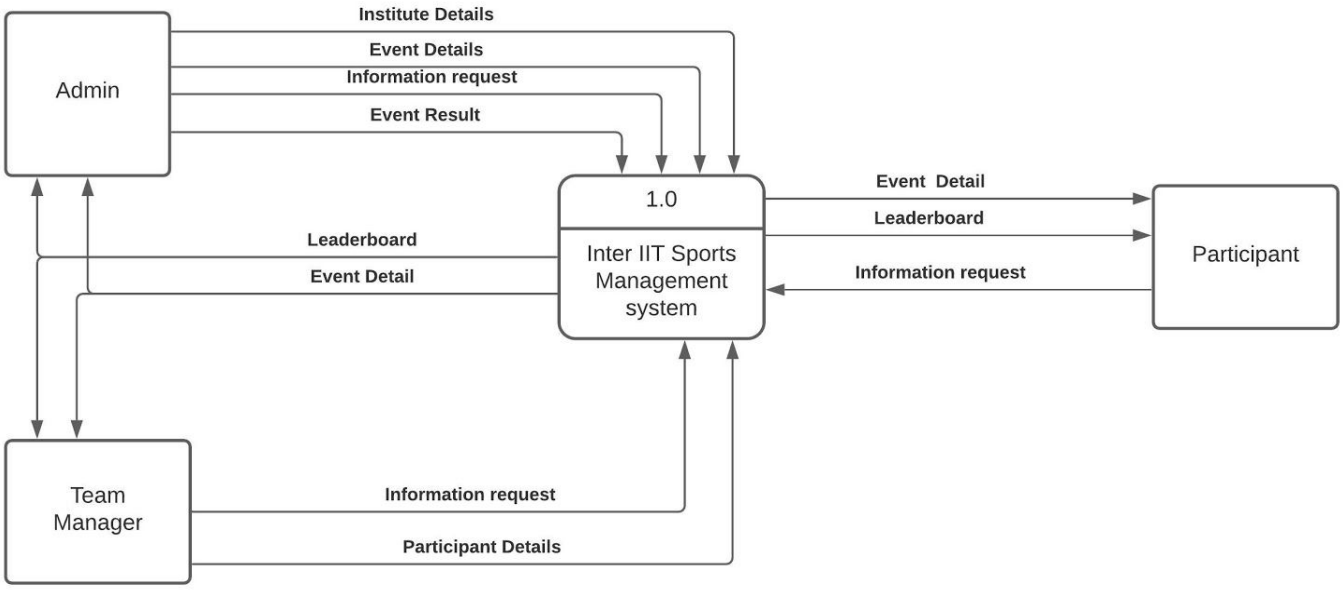
This document contains the design of the Inter-IIT Tournament Management System. The design of the management system is shown using DFD diagrams and the ER diagram for the databases which will be required to implement the software.

The design is shown in 3 stages. The three stages are as follows -

1) **Level 0 DFD**
   In this diagram the software is treated as a single module which consists of various modules inside it.
2) **Level 1 DFD**
   This diagram shows the breaks down of the management system into several modules which will be used by users to interact with the system
3) **Level 2 DFD**
   This diagram again breaks down the modules into its subparts where each subpart corresponds to an operation which is to be performed on the database to insert, modify, delete or view data by the user.

Followed by it will be the ER diagram of the database which we will use for InterIIT Tournament Management System. Then design justification on coupling and cohesion is mentioned as the last part.

Level 0 DFD Diagram -



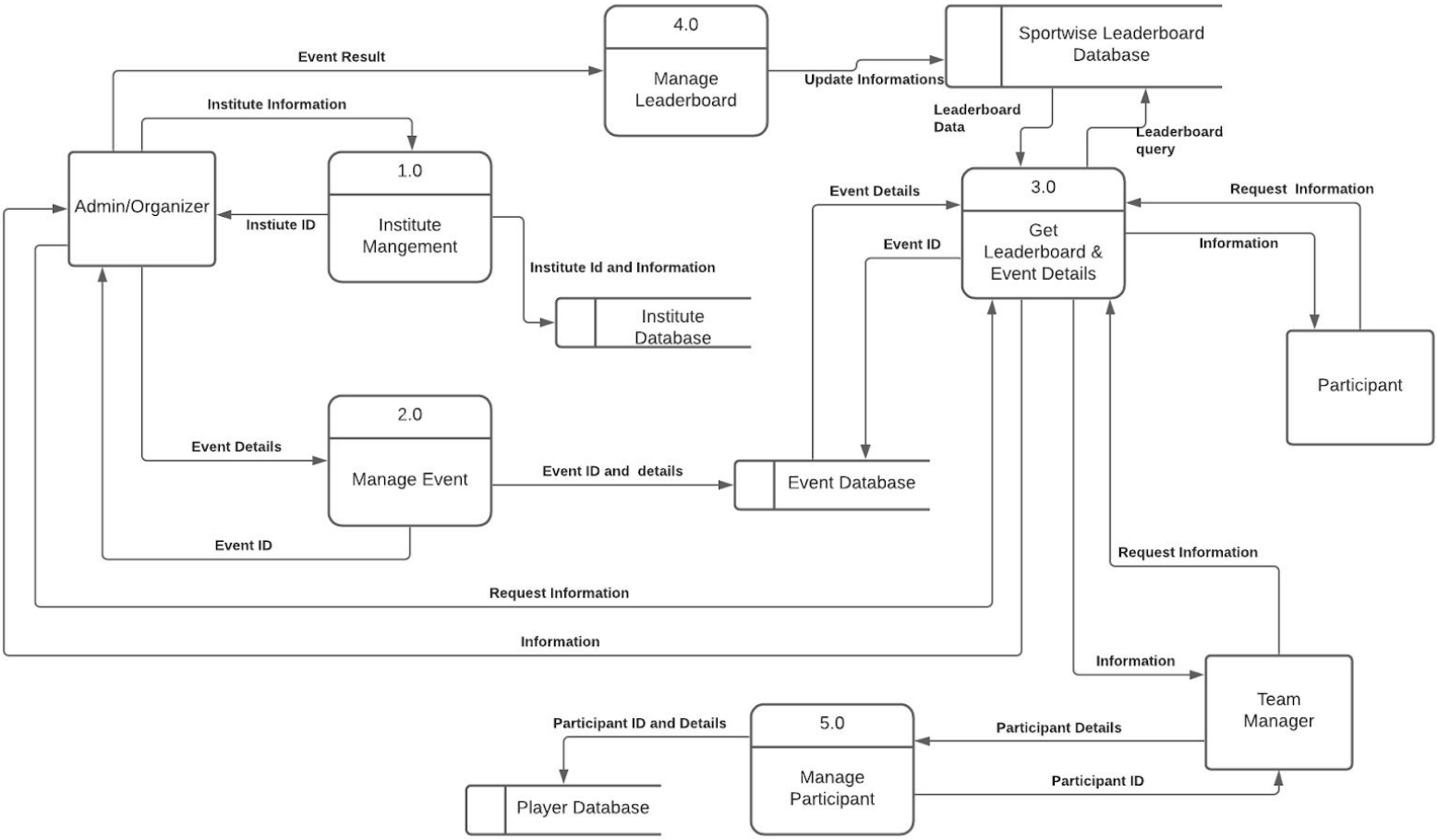
Three types of users interact with the Inter IIT Sports Management System, viz., -

- Admin
- Team Manager
- Participant

This diagram shows the system as a single unit where each type of user will give command to perform specific operation and the system will perform the operation and return the output or status of the operation.

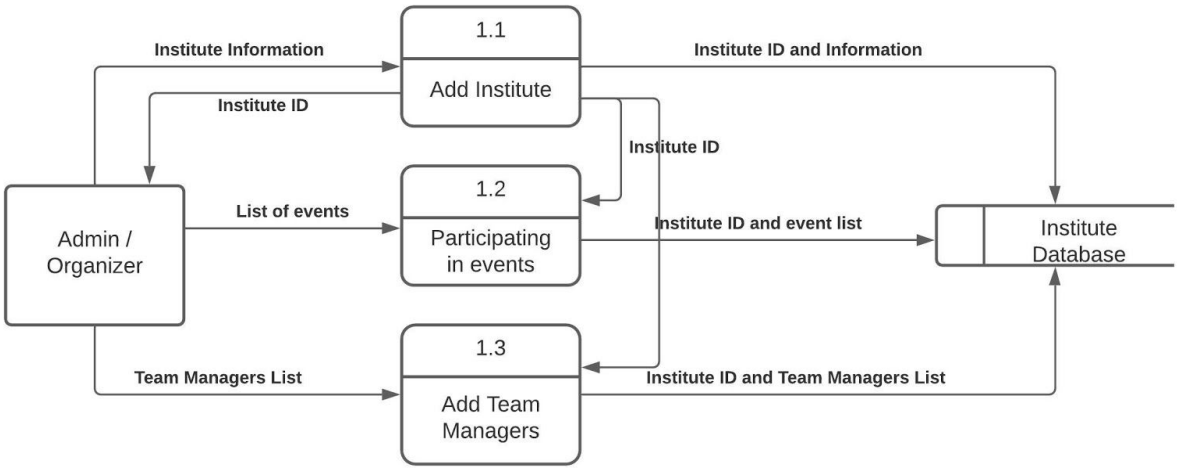
As shown in the diagram above, the Admin, Participant and Team manager can give information requests as a command to get Leaderboard and Event details.

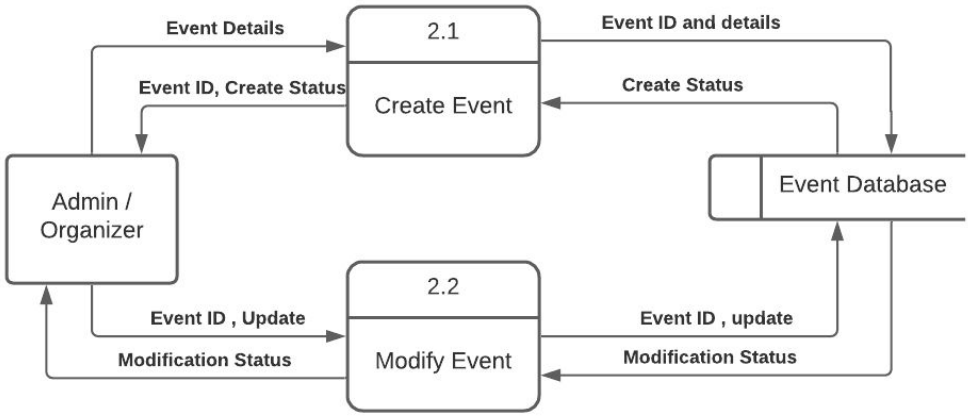
Level 1 DFD Diagram -

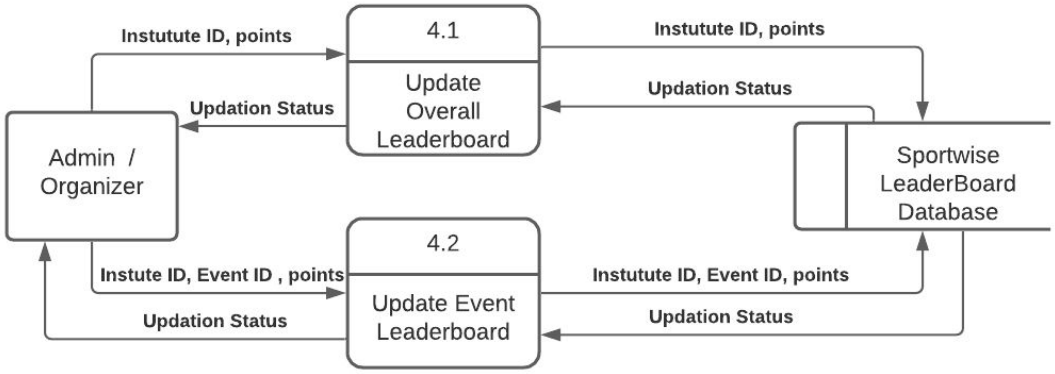


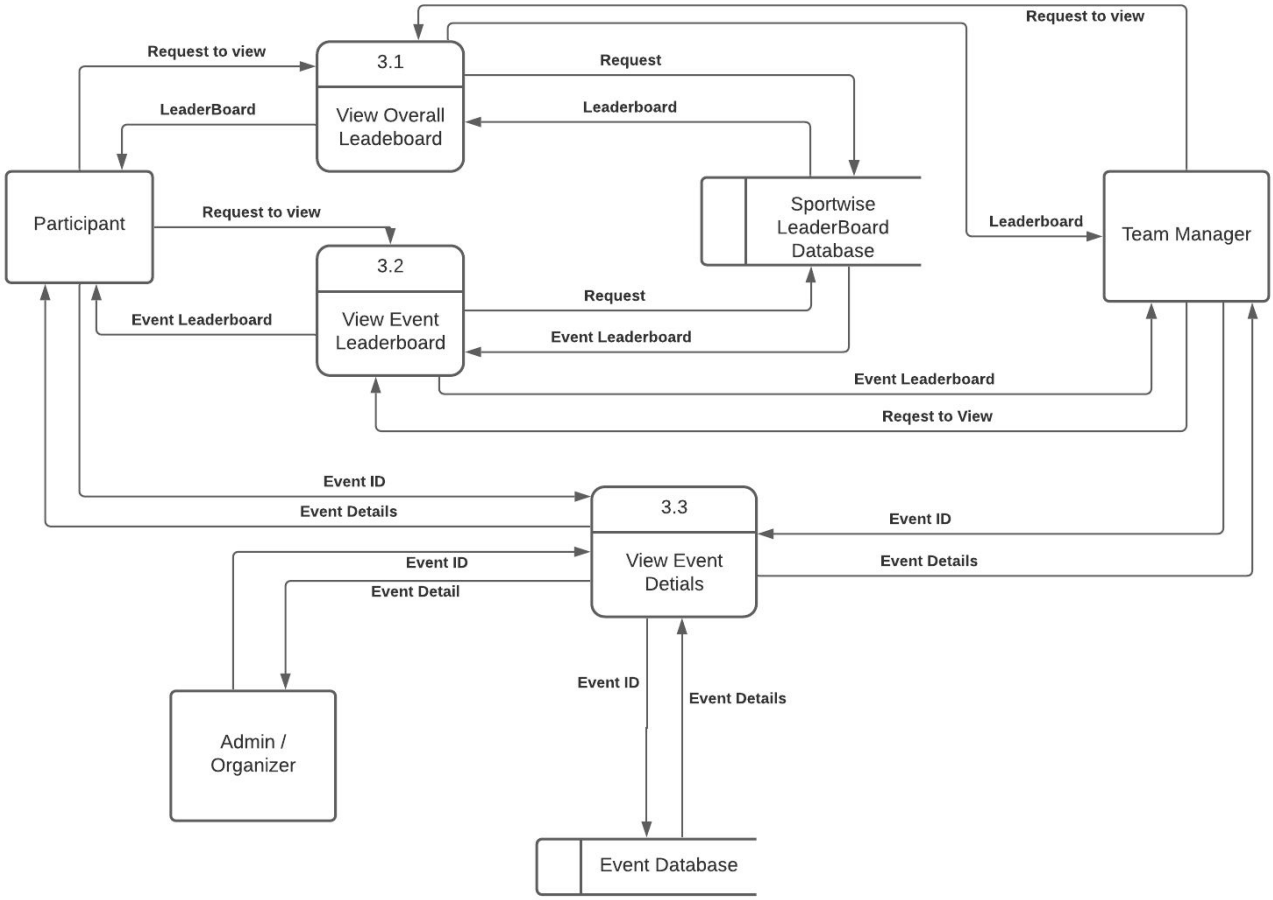
This Level 1 Diagram gives us the idea of all the modules, databases which will be implemented as specified in the Software Requirement Specification Document where each type of user will use those modules to perform operations interacting with the information provided to those modules and the databases.

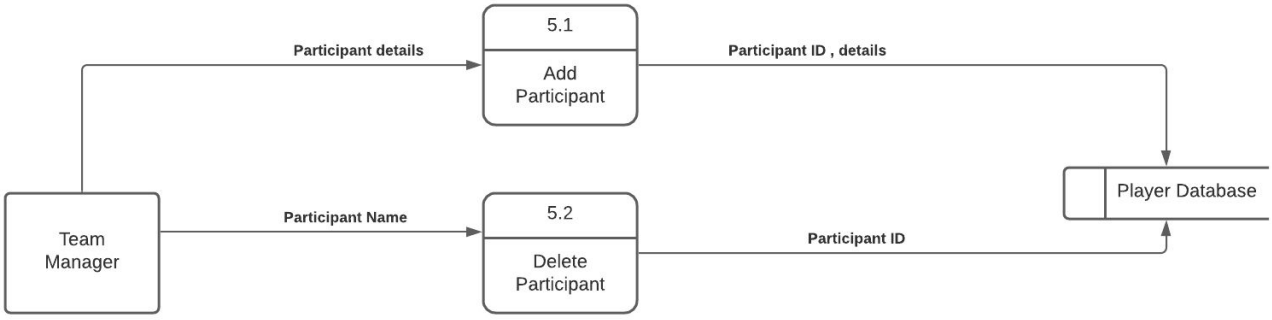
Level 2 DFD Diagrams



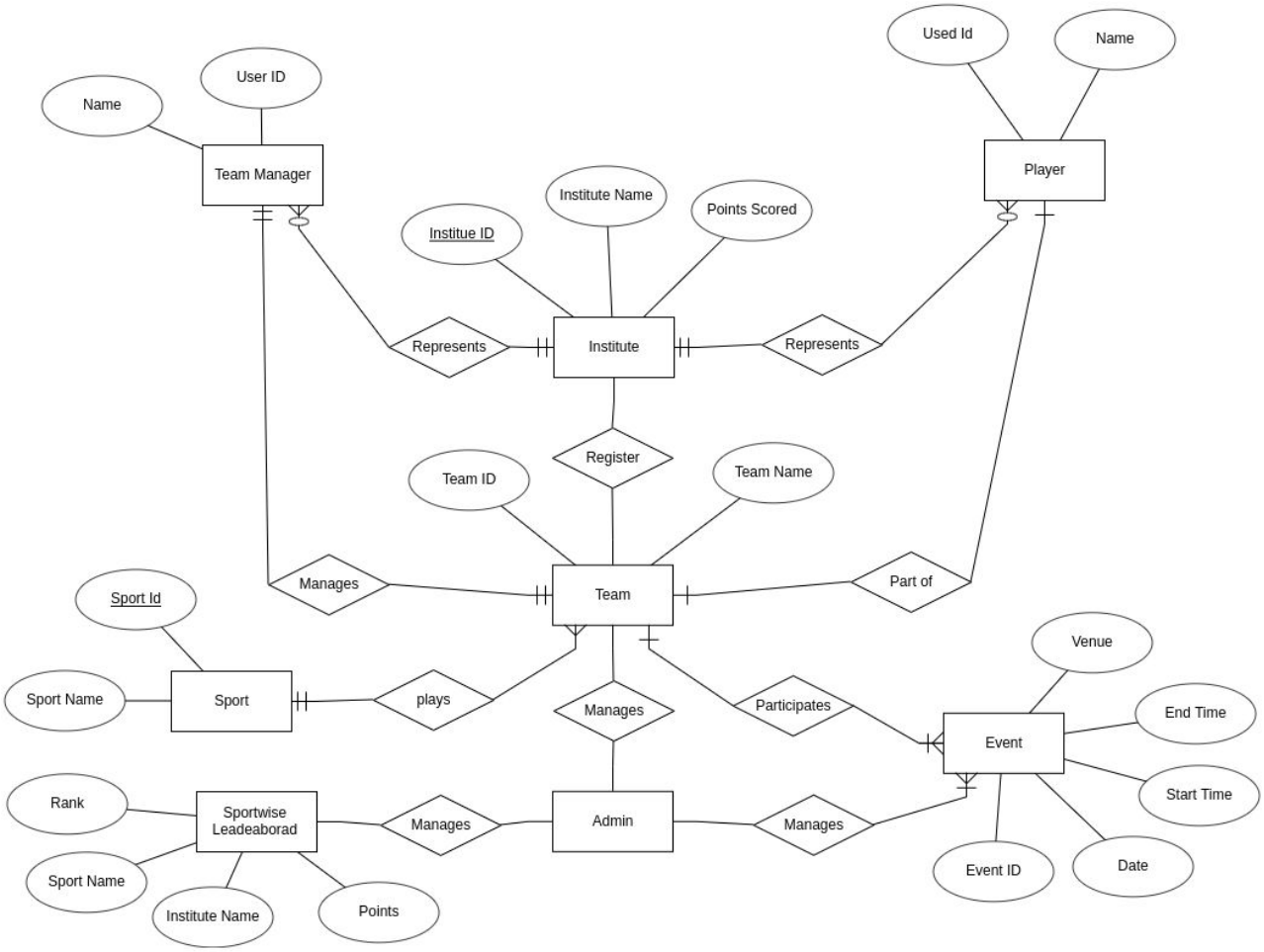





Request to view
Request to view
3.1
View Overall Leadeboard
Request
LeaderBoard
Leaderboard
Participant
Sportwise LeaderBoard Database
Leaderboard
Team Manager
Request to view
3.2
View Event Leaderboard
Request
Event Leaderboard
Event Leaderboard
Event Leaderboard
Reqest to View
Event ID
Event Details
3.3
View Event Detials
Event ID
Event Details
Event ID
Event Detail
Event ID
Event Details
Admin / Organizer
Event Database


5.1
Add Participant
Participant details
Participant ID , details
Player Database
Team Manager
Participant Name
5.2
Delete Participant
Participant ID

ER Diagram:



This ER diagram shows the relationships between entities which will be used by the management system to store and retrieve information related to tournaments, leaderboard, teams and participants.

2) Usability Document -

In view of **Shneiderman's "Eight Golden Rules of Interface Design"** we implemented following points to make our software more user friendly -

- Informative feedback is given at every point. Status is shown for each module if it was successful or some failure occurred during the process after the module is invoked.
- For each module, the same set of procedures and information is required whenever the module is invoked.
- The naming of each module is such that it is easily understandable to the functionality of the operation by a novice user.
- Short term memory load is minimum in our design so that users need not remember too many things. Users need to remember only 1-2 things such as sport name, institute name to view the details.
- Most of the actions are easily reversible- an added event can be deleted immediately, leaderboard can be updated at any time, participants can be added or removed at any time.

Further points are related to the interface of the system which will be present in the prototype document.

3) Design Justification -

All the modules are described below with their subparts and their aim and how they achieve that required task.

A) **Manage Institute** -

This module deals with the registration of the institute for the various types of sports which are to be played during the tournament. Institute Management has 3 subparts, viz., Add Institute, Add Team Managers and Participate in Events, which take information about the institute and its teams and register them for individual sports with their team managers by adding them in the database.

B) **Manage Event** -

This module deals with event related operations which will occur during the tournament. It has 2 subparts, viz., Create Event and Modify Event, whose main function is to create a new event for which it will take details from the admin as input and add a new event to the database and the other subpart helps to change any information related to any specific event, if required. Both the subparts are related to events.

C) **Get Leaderboard and Event Details** -

It deals with the information details retrieval which can either be of leaderboard or related to specific events. It has 3 subparts, viz., View Overall Leaderboard, View Event Details and View Event Leaderboard, which shows the standings of the institutes according to their scores and information related to any event from the database. These all subparts deal with query of information.

D) **Manage Leaderboard** -

It deals with updation of the leaderboard which will get updated after the result of each event is entered into the system which will change the scores of the institutes. It has 2 subparts, viz., Update Overall Leaderboard and Update Event Leaderboard which will update the information in the database. These both parts deal with scores of the institutes which decide the leaderboard.

E) **Manage Participant** -

It deals with the information related to each player who will register as a participant on the system. It has 2 subparts, viz., Add Participant and Delete Participant, which will add participants to the database and delete from it as specified by the team managers who will be interacting with this module. Both the subparts deal with the participant's information on the database.

Cohesion is a measure of the degree to which the elements of the module are functionally related. It is the degree to which all elements directed towards performing a single task are contained in the component.

All the modules are made such that each module is related to a specific task and its subparts help it to achieve it through a number of steps with the information provided to them. Through the description of the modules we can say that cohesion is high in our design as each subpart of the module is related to corresponding specific tasks.

Coupling is the degree of interdependence between software modules; a measure of how closely connected two routines or modules are the strength of the relationships between modules.

As described above, each module is related to a specific task and does not use any other module as a part of it in any way or require some information to be passed from any to other module to it to perform its task for which it is made which states that dependence between any two modules is not present by which we come to the conclusion that Coupling is low in our design.