Contents

[Chapter 2 Analysis 2](#_Toc15317724)

[2.1 Analysis Methodology 2](#_Toc15317725)

[Analysis Methodology Steps 3](#_Toc15317726)

[DFD 4](#_Toc15317727)

[2.2 Information Gathering 6](#_Toc15317728)

[2.3 Feasibility Study 8](#_Toc15317729)

[2.4 SRS (Software Requirements and Specifications) 9](#_Toc15317730)

[2.4.1 Functional Requirements 9](#_Toc15317731)

[2.4.2 Non-Functional Requirements 10](#_Toc15317732)

[2.4.3 MOSCOW prioritization 11](#_Toc15317733)

[2.4.4 Hardware/Software Spec 13](#_Toc15317734)

[2.5 Use Case Diagram 14](#_Toc15317735)

[2.6 System Architecture 16](#_Toc15317736)

[3-tier Architecture 16](#_Toc15317737)

[2.6.1 NLA (Natural Language Analysis) 17](#_Toc15317738)

[2.6.2 Initial Class Diagram 18](#_Toc15317739)

# Chapter 2 Analysis

Analysis means evaluation of examination of any data. In other words, Analysis is a process of dividing a complex topic into smaller parts so that every detail of that topic can be covered. It is done specially to gain every knowledge associated by determining its nature or essential features.

## 2.1 Analysis Methodology

Hard Approach methodology is used for the analysis of this project (Sports Event Management System). Hard approach methodology is the way that focuses on the sequential or highly structured approach to analysis. It is mainly used for large and complex projects and also can be used for small projects where technical aspects need to be considered more than the human aspects. So, in this project as well technical or logical aspects are to be focused more than the human aspects.

Therefore, according to the nature and requirements of the project, **SSADM** (**Structured System Analysis and Design Methodology**) is used.

Mainly SSADM focuses on the three Views of a system. They are;

**Process View**

It describes the process that is carried out in the system i.e. how data moves and are processed into the system.

**Data View**

It describes all the data that is used by the system.

**Event View**

It describes the events that runs the system processes and the effects caused by the external events on data.

The project is based on water fall methodology. And SSADM is similar to the Water fall Model. Water fall methodology is suitable for that type of projects in which the requirements are pre-defined and second step is started only after the completion of first step.

Likewise in Waterfall Model there are various steps in SSADM as well. They are;

### Analysis Methodology Steps

**Analysis and Requirement Specifications**

During this step all the aspects such as hardware and software are analyzed and as per the analysis the requirements are defined I.e. technical specifications are produced.

**Design**

As per the analysis done and requirements gathered in the previous step all the aspects of the system are designed i.e. a software is coded, interface as well as database is designed in this step.

**Implementation**

In this step a designed software is implemented as a set of programs or as a unit. Unit testing is also done under this phase to examine if each of the units meet its requirements/specifications or not.

**Testing**

During this phase a system is tested through different testing methods in order to find errors if any. All the program units are integrated and is tested as a whole system to verify whether the system’s requirements are met. After the successive testing, the software is delivered.

**Maintenance**

After the deployment of the software there might arise different errors or problems which was not discovered during the development time of the software. And maintenance is process of fixing such type of errors to improve or enhance the performance of the system. So, this is the phase in which the software is maintained i.e. upgrades and updates are done if required. And all the bugs are fixed.

### DFD

DFD (Data Flow Diagram) is a technique used in this methodology i.e. SSADM that gives an overview or represents the flow of data in the system.

Followings are the DFD for admin and user separately.

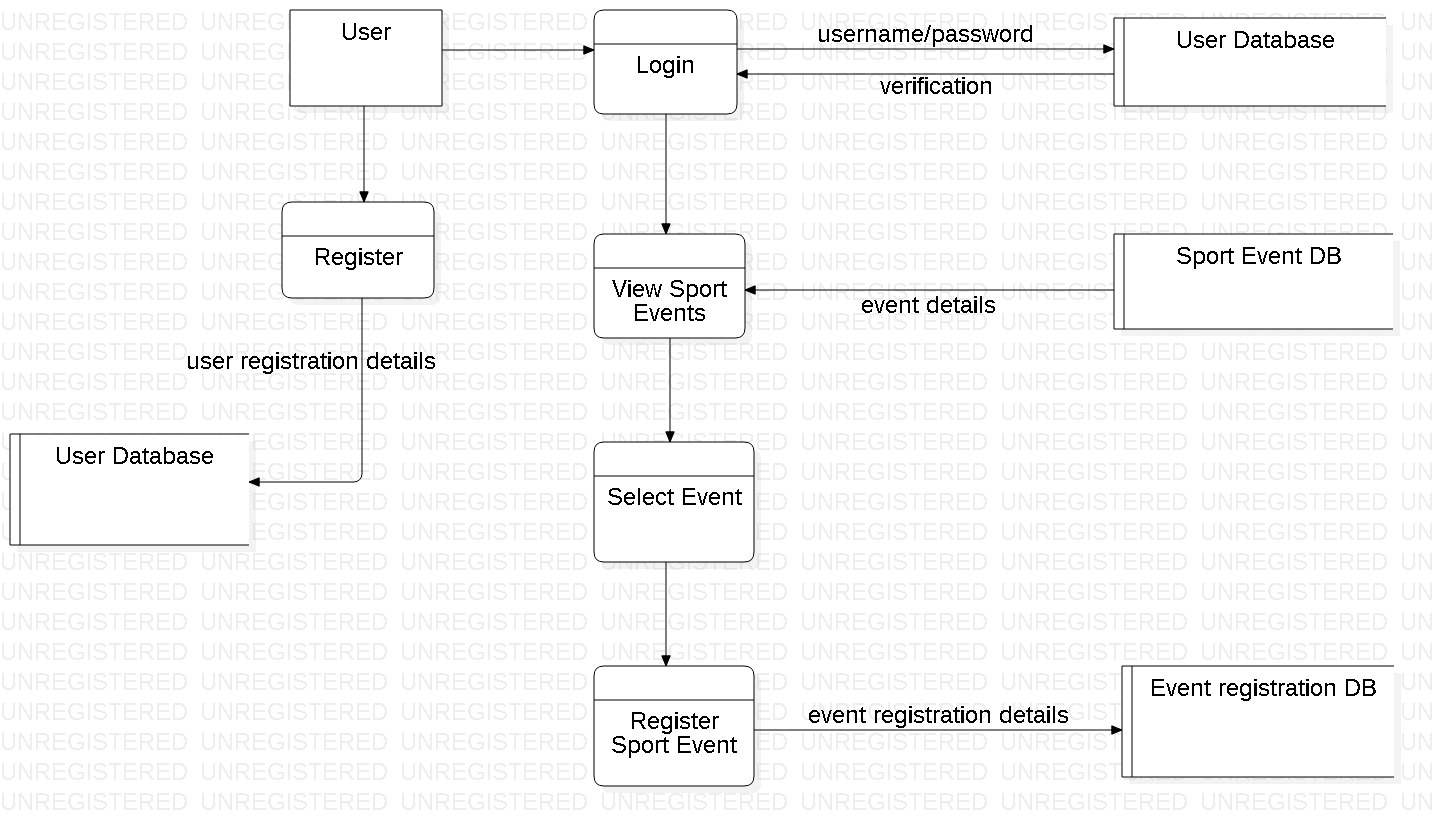


Figure 1: DFD for user

This is a Data flow diagram indicating how data flows in the system while user is using it. It does not provide detail functions but gives an overview of how the data are flowing in the system while user is accessing it.

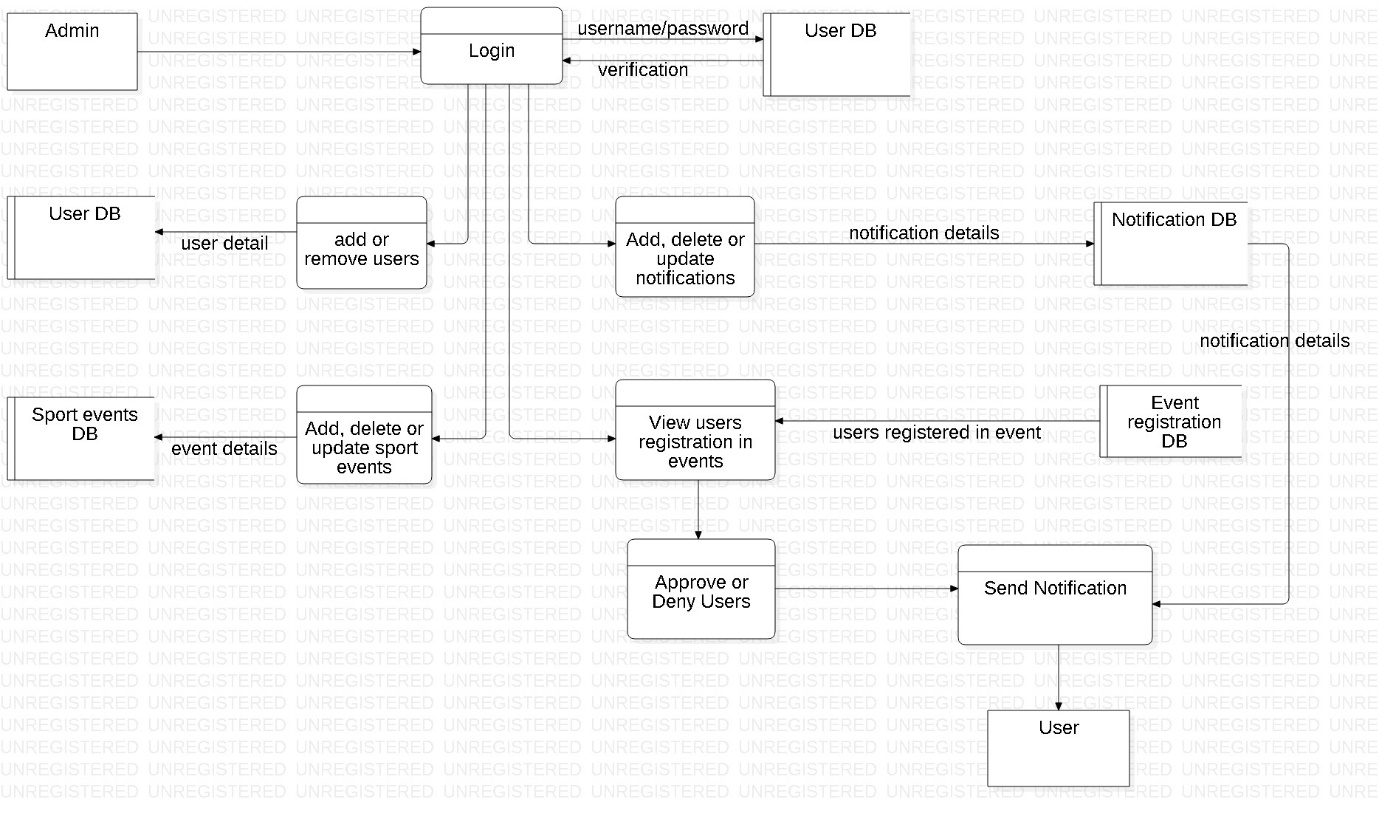


Figure 2: DFD for Admin

This is a Data flow diagram indicating how the data flows in the system while admin is accessing it. This diagram provides an overview of how different functions are flowing in the system and what are the functions or data that are accessed by the admin.

## 2.2 Information Gathering

Information gathering is a process of gathering or collecting different data or information against the targeted system. It is considered as one of the effective way to collect the required information. There are several techniques of gathering information such as Questionnaires, Observations, Focused groups, Interviews, Surveys etc.

And from these techniques I have choose to conduct interview for gathering the required information for the further development of system (Sport Event Management System).

**Interview**

It is an information gathering technique in which various predefined questions based on specific agenda is asked to the targeted victim and the answer is noted. And on the basis of that answer various requirements of the system are analyzed and prioritized. Some of the advantage of taking interview are;

**Advantages**

* High response rate
* More detail questions can be asked.
* More detail information can be gathered.

**Plan for the interview with 10 Persons of Sport Events Management field.**



Various subjects based on those agendas were discussed with the number of peoples in belief of gaining positive outcome.

After analyzing the outcome (i.e. answers given by the interviewees) of the interview, it is found that 80 percent of the interviewees have recommended similar type of requirements. However, 20 percent of them have suggested different variety of requirements. So, as more percentage of interviewees recommended similar type of requirements or features such as login, registration, crud functions, searching, filtering etc., I have categorized them as an important features that are to be implemented in the system.

Therefore, conducting interview has assisted a lot in the process of gathering information related to the Sports event Management System.

## 2.3 Feasibility Study

To determine how conveniently or easily something can be done, feasibility study is undertaken. Feasibility study helps to analyze the different factors such as cost, time, social, legal, technical etc. and how will these factors are impacted or maintained during or after the project development. There are various type of feasibility study that can be performed. Some of them are as follows;

**Economic Feasibility**

The project is not such a large project so, a small amount of budget is sufficient for it. With the allocated budget my project will be completed on time, without lacking in financial resource. In order to determine the economic feasibility more precisely cost/benefit analysis is undertaken. Under this how much cost is allocated for the project i.e. Sports Event Management System and what will be its benefits and vice versa are estimated.

**Technical Feasibility**

Different technological resources are essential for different projects. For this project i.e. Sports Event Management System various technical resources such as Laptop with minimum Core i5 specification, text editor such as brackets, sublime etc. are required. So, all the technical resources to undertake the project are available.

**Time Feasibility**

This (**Sports Event Management System**) project is not so large. Therefore, it will not take long time to be completed. Within the allocated or estimated time, the project will be concluded.

**Social Feasibility**

Underlying on various social constraints the project has been planned. And it will not have any negative impacts on society.

**Legal Feasibility**

The project is planned and will be developed within the legal constraints. It will not violate any rules or regulations. As well as, it is not associated with any of the illegal implications.

## 2.4 SRS (Software Requirements and Specifications)

### 2.4.1 Functional Requirements

The requirements without which a system cannot function or run properly is known as Functional requirements. They defines a functions/components of a system.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Title** | **Description** | **Rational** | **Dependency** |
| **FR01** | Login for Admin | Admin logins to the admin portal of the system | To control/manage the overall process of the system |  |
| **FR02** | Login for User | User logins to the system  through valid username and password | To get access to the validated contents/features of the system | **FR03** |
| **FR03** | Registration for User | Unregistered users register themselves | To be permitted for logging in into the system |  |
| **FR04** | Add sport events | Admin adds the sport events | Added events are viewed by the users |  |
| **FR05** | Update events | Admin updates the sport events if required | Updated events are viewed by the users | **FR04** |
| **FR06** | View events | Users will view the sport events added by admin | To get the detail information of the sport events | **FR02** |
| **FR07** | Send notifications | Notifications of important contents will be sent/received | To increase the availability and accessibility of contents of the system |  |
| **FR08** | Search events | Users can search the events as they want | To locate/select the favorite sport event | **FR02** |
| **FR09** | View registered events | Admin views the registration of users in various events | To manage the users as per registration |  |
| **FR10** | Adding & Removing users | Admin adds or removes the user | To manage the number of users | **FR03,FR01** |
| **FR11** | Approving users | Admin approves the valid users registration request | To make user eligible for participating in particular event |  |
| **FR12** | View score updates | User can view score after the sport event | To know the details of that particular event | **FR02** |
| **FR13** | View schedules | User can view schedules of the events | To know the detail information of the events | **FR02** |
| **FR14** | Registration in events | User can register them into the sport event they want | To take part in the interested sport event | **FR02** |
| **FR15** | Update profile | User can update their profile details | To change the details if required | **FR03,FR02** |
| **FR16** | Cancel registration of events | User can cancel their registration in events | To call of, their previous decision of registering |  |

### 2.4.2 Non-Functional Requirements

The requirements that assures the system’s quality rather than functionality is known as Non-Functional Requirements. They are also known as Quality Attributes of a system. Followings are some of the Non-functional requirements of the (Sport Event Management system) system.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Title** | **Description** | **Dependency** |
| **NFR01** | Security | To make the system secure different validations and authentications for the user and admin are done. |  |
| **NFR02** | Maintainability | Even after the completion of system if any bugs or errors arise then they are handled by upgrading or updating as required. |  |
| **NFR03** | Performance | Any tasks or process done by the system maintains the response time in order to deliver the better performance. | **NFR09** |
| **NFR04** | Data Integrity | Hashing algorithms are used for passwords to maintain the data integrity. | **NFR01** |
| **NFR05** | User friendly | To make the system more user friendly, unfamiliar or vague contents are not used. |  |
| **NFR06** | Availability | The system can be obtained or used in a quality manner. |  |
| **NFR07** | Serviceability | The system will provide a good service to its users. |  |
| **NFR08** | Reliability | The system will provide a consistent performance which can simultaneously gain the trust of the users. | **NFR04, NFR06** |
| **NFR09** | Scalability | As per the work lengths the system will be able to perform or the system is capable of performing in variable ranges. |  |

### 2.4.3 MOSCOW prioritization

MOSCOW is a method of prioritizing the features or requirements, on the basis of which a system can be developed in a quality way. It helps to prioritize the requirements in more details than the other methods.

Therefore, I have used MOSCOW prioritization method for categorizing the requirements.

**M**O**SC**O**W** stands for

**M-** Must have: The features that should not be neglected or without which a system is incomplete.

**S-** Should have: The features that are of high importance but not that vital.

**C-** Could have: The features that are nice to have but are not essential for the core functionality.

**W-** Won’t have: The features that are not suitable for current timeframe or can be implemented in future for further improvements.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Title** | **MOSCOW** | **Description** |
| **R1** | Login for Admin | **M** | It is a fundamental feature without which there will be no validation for admin-login and admin will not be able to manage the users and other contents of the system as well. |
| **R2** | Login for User | **M** | It is also a must have feature without which validation of the user cannot be done. |
| **R3** | Registration for User | **M** | Without this feature user cannot get enrolled into the system. So, it is essential. |
| **R4** | Add sport events | **M** | Without sport events system cannot be completed. |
| **R5** | Update events | **M** | While adding events if some details are missed then update should be done as quick as possible in a way that the user should not be aware about it. |
| **R6** | View events | **M** | It is a must have feature because without this user cannot view the events they are interested or may result in incomplete system. |
| **R7** | Send notifications | **C** | It is prioritized as could have because it is good to have but does not play a significant role in core functionality. |
| **R8** | Search events | **S** | It is important feature because if user does not find their desired sport event on viewing whole events, then they can search the specific sport event. So, it is prioritized as should have and not as must have. |
| **R9** | View registered events | **M** | All the registered events should be viewed by the admin to approve the valid one and to deny the invalid ones. So, it is a must have requirement. |
| **R10** | Adding & Removing users | **M** | Users must be added and removed by the admin in order to manage the limited number of users. So, it is also a must have requirement. |
| **R11** | Approving users | **M** | It is also a must have requirement because all the users are to be approved by the admin in order to maintain the validity of users. |
| **R12** | View score updates | **C** | It is a feature which could be implemented in the future. So, it is could have requirement. |
| **R13** | View schedules | **M** | It is also must have requirement because user must be able to view the schedules of the events that are going to be held. |
| **R14** | Registration in events | **M** | User must be able to register them in interested sport event to take part in it without which events cannot be commenced. So, it is a must have requirement. |
| **R15** | Update profile | **M** | If required user must be able to update their profile details in a quick manner. So, it is a must have requirement. |
| **R16** | Cancel registration of events | **S** | User may change their mind to reverse their decision of registering them in the events. So, feature of cancelling the registration should be implemented. |
| **R17** | Security | **S** | Login validations or contents validation maintains the security of the system from unauthorized access. So, it is should have requirement. |
| **R18** | Maintainability | **C** | It is could have requirement because it is done after the deployment of the system and will not impact in the functionality of the system. |
| **R19** | Performance | **S** | A system should maintain the response time of any processes for the better performance. So, it is should have requirement. |
| **R20** | Data Integrity | **S** | If the data of the system is changeable then there will be lack of consistency. So, it is should have requirement. |
| **R21** | User friendly | **C** | This is could have requirement because it is good to have more user friendly system but is not an essential part of core functionality. |
| **R22** | Availability | **S** | Without this requirement the system’s data or information could not be used/obtained in a quality manner. So, it is should have requirement. |
| **R23** | Serviceability | **C** | This is also a requirement good to have but is not much important for the main functionality of system. |
| **R24** | Reliability | **S** | A system should be reliable i.e. it should perform consistently and without this requirement a system will not be qualitative. So, it is should have requirement. |
| **R25** | Scalability | **S** | Without this requirement a system will not be capable of performing at different ranges/lengths as per work. So, it is should have requirement. |

### 2.4.4 Hardware/Software Spec

Hardware and software that will be required during the development or before the release of the system i.e. (Sports Event Management System) are as follows;

**Hardware Specification**

* **Processor:** Laptop with minimum core i5 processor
* **RAM:** Minimum 2GB
* **Hard disk:** Minimum 10 GB

**Software Specification**

* **Text editor (Brackets)**

It is used for coding the program.

* **Star UML**

It is used for designing or modelling activity diagram, class diagram, use case diagram, etc.

* **My SQL**

It is used for creating database.

* **Operating system:** Windows 10 (34 or 64 bit)

## 2.5 Use Case Diagram

A use case diagram is a graphic depiction of the interactions among the elements of a system.

Elements used in use case diagrams;

**System Boundary:** It separates an internal system with respect to external actors.

**Actor:** It may be anyone that interacts with the system or in some case system itself.

**Use case:** Different functions of the system that are accessed by the actors are identified as use cases and are associated to the respective actors.

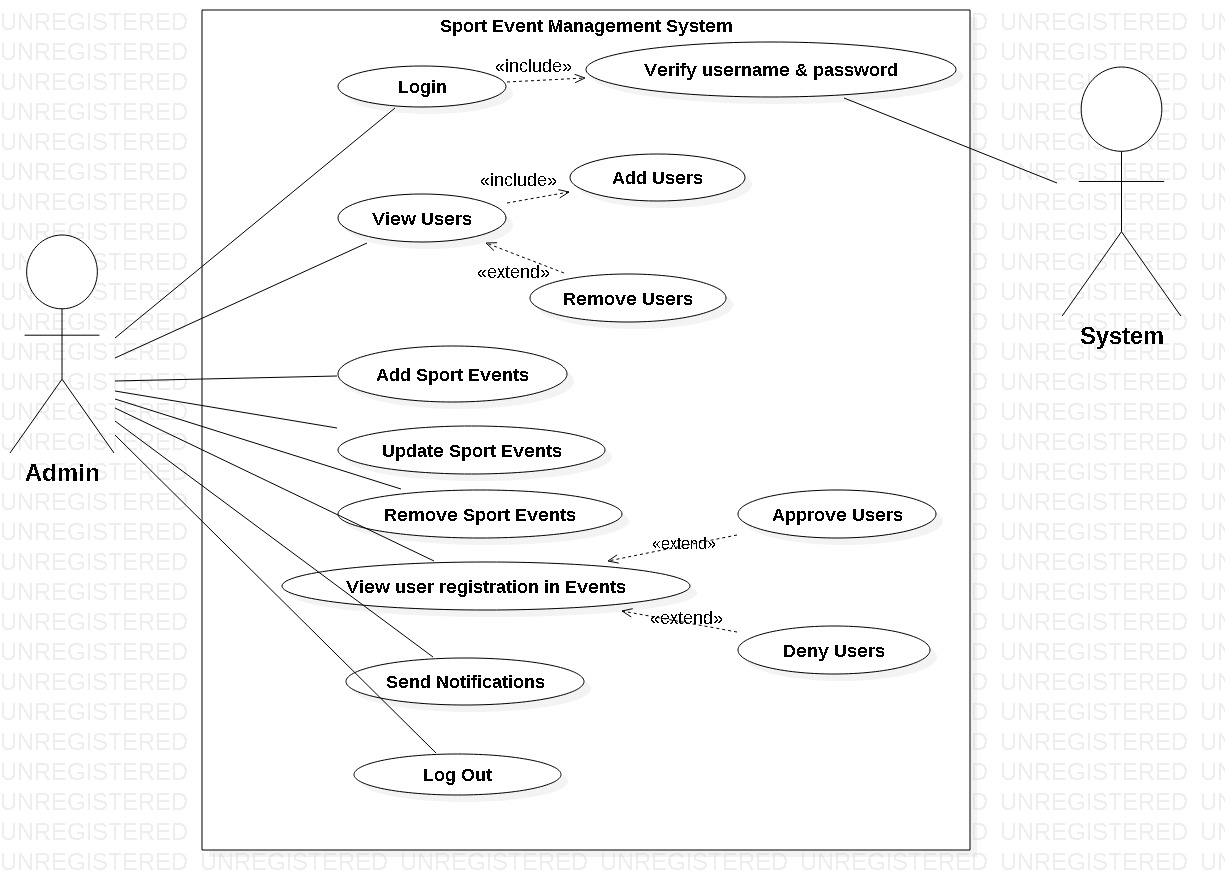
****

Figure 3: Use case diagram for admin

**Actor 1:** Admin is the first actor in this use case diagram. All the features of the system that are accessed by admin are identified as use cases and associated in the above use case diagram.

**Actor 2:** Second actor in this use case diagram is System itself. All other functions are done by the admin whereas the verification of the admin is done by the system.

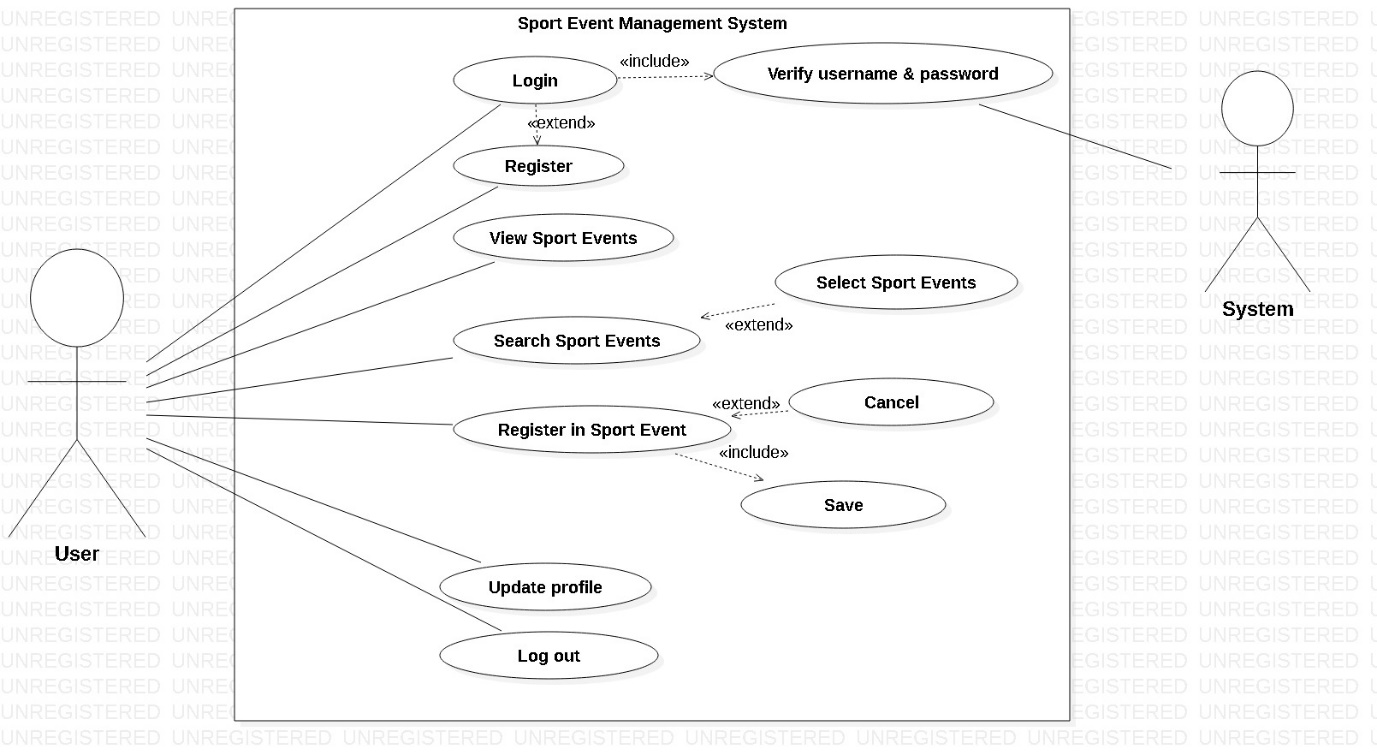


Figure 4: Use Case diagram for User

**Actor 1:** In this use case diagram user is the first actor and all the features or functions of the system that are accessed by them are identified as use cases and are associated.

**Actor 2:** System itself is the second actor in this use case diagram that verifies the user to permit them for accessing other functions of the system.

## 2.6 System Architecture

### 3-tier Architecture

A 3-tier is a client-server architecture that facilitates the use of object oriented concepts and data security through application layer.

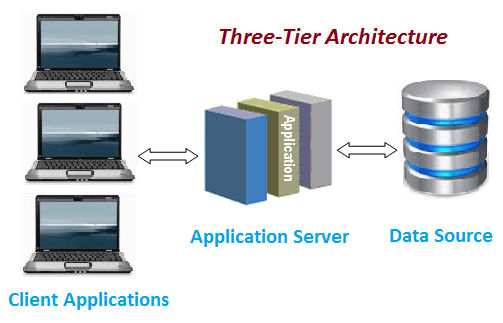




Figure 5: 3-tier architecture

This system i.e. Sport events management system consists of various services and resources which is to be presented to the clients or users via server as requested. And 3-tier architecture is the best to fulfill this requirement of a system as it is a client-server architecture. It also enables the faster migration to new graphical environment and object-oriented concepts. Similarly application layer provides data security and changes made to one layer will not have any impact on other layers.

Therefore, 3-tier architecture is a most suitable architecture to this system.

### 2.6.1 NLA (Natural Language Analysis)

It is a process of analyzing the provided scenario to find out all the nouns and verbs that are present. In which nouns represent class and verbs represent method.

**Scenario**

XYZ is a Sport events managing company. It is not a big and reputed company as they are rookie in this field. They manage different Categories of Sport events such as Basketball, Cricket, Football, etc. They are doing all the functions manually which is not benefiting for the growth of their business. So, to upgrade their business they are seeking for an automated computer based Sport event management system. And the system should be able to do the primary requirements such as admin can add, remove, update sport events and users as well as can send notifications, users can view and register in events only after valid registration and login.

Followings are the steps that are considered while performing NLA.

**Step I: Listing of all the nouns / verbs**

**Nouns:** Sports events, company, company, field, Category, Sport events, business, business, computer, Admin, Sport events, Admin, users, user, and notification

**Verbs:** manage, big, manage, type, add, update, delete, add, delete, send, receive, view, login, and register

**Step II: Repeated nouns / verbs as well as synonyms are removed**

**Nouns:** Sport events, company, field, Category, business, Admin, user, notification

**Verbs:** manage, big, type, add, update, delete, send, receive, view, login, register

**Step III: Nouns / Verbs that does not give specific meaning or out of scope are removed**

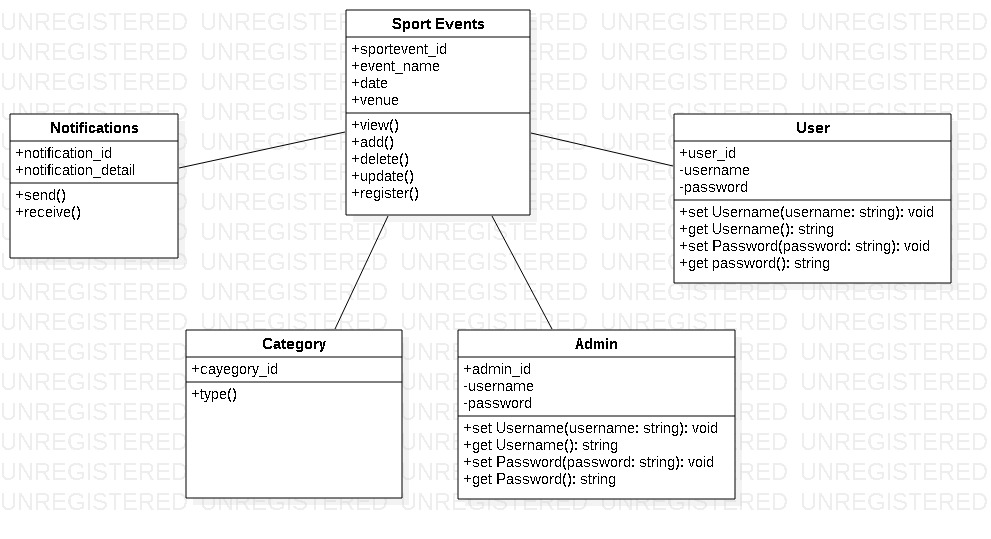
**Nouns:** Sport events, Category, Admin, User, Notification

**Verbs:** add, update, delete, send, receive, view, login, and register

**Actual Class:** Sport events, Admin, Category, User, Notification

**Actual Methods:** add, update, delete, send, receive, view, login, and register

### 2.6.2 Initial Class Diagram



Class diagram is the blue print of aspects to build the whole system. It defines how the classes will be working together to develop a system. So, above is the initial class diagram of the Sport events management system that includes main classes along with some attributes and methods. This is only the indication of how the class diagram will look like. But final class diagram will be developed further.