ARTS & SPORTS MANAGEMENT

PROJECT REPORT

Submitted by:

ARSHIDHA AJ	**(Reg No. : AZATSCS017)**
AZIN FATHIMA KHAN	**(Reg No. : AZATSCS018)**
RIFA CP	**(Reg No. : AZATSCS026)**
RIYA CP	**(Reg No. : AZATSCS027)**

For the award of the Degree of

Bachelor of science (B.Sc.)

In Computer Science

(University of Calicut)

AMAL COLLEGE OF ADVANCED STUDIES, NILAMBUR

(Affiliated to the University of Calicut)

Santhigramam,Myladi

Eranhimangad P.O

Nilambur, Kerala 679329

MARCH – 2022

ACKNOWLEDGMENT

First of all, I thank the **ALMIGHTY GOD** for showering his immense blessing and giving strength to complete my project work successfully.

I wish to express my sincere gratitude to **Dr. Zacaria TV,** Principal for providing us adequate facilities to do this work. I am grateful to **Mr. Shakeel Juman**, Head of the Department of Computer Science in bringing this report to a successful completion. .Last but not the least, I express my sincere thanks to all of the staffs in Department of Computer Science, all of my friends and my parents who have patiently extended all sorts of help for accomplishing this undertaking

Date:

ARSHIDHA AJ
RIFA CP
RIYA CP
AZIN FATHIMA KHAN

DECLARATION

We, here by declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person or material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text

Place: NILAMBUR
Date:

Signature:
Name: ARSHIDHA AJ
(Reg No: AZATSCS017)
RIFA CP
(Reg No: AZATSCS026)
RIYA CP
(Reg No: AZATSCS027)
AZIN FATHIMA KHAN
(Reg No: AZATSCS018)

CERTIFICATE

This is to certify that the project report entitled ARTS AND SPORTS MANAGEMENT submitted by ARSHIDHA AJ (Register Number: AZATSCS017), RIFA CP (Register Number: AZATSCS0026), RIYA CP (Register Number: AZATSCS027), AZIN FATHIMA KHAN (Register Number: AZASCS018) to University of Calicut for the award of the degree of Bachelor of Science (B.Sc.) in Computer Science is a bonafide record of the project work carried out by him/her under my supervision and guidance. The content of the report, in full or parts have not been submitted to any other Institute or University for the award of any other degree or diploma.

Signature

Mr.Shakkeel Juman

Head of the Department

Department of computer science
Amal college of advanced studies
Myladi, Nilambur

Place:

Date:

(Office seal)

Certified that the candidate was examined by us in the Project Viva Voce Examination held on..... and his/her Register Number is .....

Examiners:

1.

2.

CONTENTS

Abstract

‘ARTS AND SPORTS MANAGEMENT SYSTEM’ is a web application which is used to completely manage Arts and Sports event at college level. It allows users to register for events, know the schedule and time of the events, view their own scores, their house’s scores and also see the colourful and different moments of these events as photos. There will be a super admin who has the complete privileges on the system. And there will be separate admin for arts and sports. Also, there will be a admin for media as well for manging photos.

Key features of the project

- Users can register for different events so easily.
- Users can have the live updates of the score table on this web-app.
- Can view the schedule and timings of all the events.
- Can view images of different images.
- User friendly.

List of figures

List of tables

CHAPTER 1

INTRODUCTION

‘ARTS AND SPORTS MANAGEMENT SYSTEM’ is a web application which is used to completely manage Arts and Sports event at college level. It allows users to register for events, know the schedule and time of the events, view their own scores, their house’s scores and also see the colourful and different moments of these events as photos. The students can register for an event at anytime from anywhere. By using this system student can save a lot of time an effort. The student can easily get the information from anywhere.

There will be a super admin who has the complete privileges on the system. And there will be separate admin for arts and sports. Also, there will be a admin for media as well for manging photos.

The key aim of this application is to make the management of the Arts and Sports events easily and in effective way.

CHAPTER 2

PROBLEM DEFINITION AND METHODOLOGY

2.1 Problem Definition

The project aims to solve the problem occurring when manging Arts and Sports events offline. For example-

One has to wait at the hall for knowing the results of the events to be announced.

One has to look up on the notice board for long to look for schedule and timings of events.

2.2 Project Overview

This web-app can be used to completely manage Arts and Sports event at college level. It allows users to register for events, know the schedule and time of the events, view their own scores, their house's scores and also see the colourful and different moments of these events as photos. The students can register for an event at anytime from anywhere. By using this system student can save a lot of time an effort. The student can easily get the information from anywhere. There will be a super admin who has the complete privileges on the system. And there will be separate admin for arts and sports. Also, there will be an admin for media as well for manging photos.

2.3 Methodology

AGILE methodology is a practice that promotes continuous iteration of development and testing throughout the software development lifecycle of the project. Both development and testing activities are concurrent unlike the Waterfall model. The agile software development emphasizes on four core values.

- Individual and team interactions over processes and tools.
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation Responding to change over following a plan.

Phases of Agile Model:

1. Requirements gathering: In this phase, you must define the requirements. You should explain business opportunities and plan the time and effort needed to build the project. Based on this information, you can evaluate technical and economic feasibility.

2. Design the requirements: When you have identified the project, work with stakeholders to define requirements. You can use the user flow diagram or the high level UML diagram to show the work of new features and show how it will apply to your existing system.

3. Construction/ iteration: When the team defines the requirements, the work begins. Designers and developers start working on their project, which aims to deploy a working product. The product will undergo various stages of improvement, so it includes simple, minimal functionality.

4. Testing: In this phase, the Quality Assurance team examines the product's performance and looks for the bug.

5. Deployment: In this phase, the team issues a product for the user's work environment.

6. **Feedback:** After releasing the product, the last step is feedback. In this, the team receives feedback about the product and works through the feedback.

2.4 Purpose

The purpose of this project is to make the managing of arts and sport events through online platform since that is an efficient method

2.5 Scope

Since each and everything is online these days the scope of this project is high and also after the COVID-19 pandemic, the necessity of services based online platforms are high. And since this is proven efficient also increases the scope.

CHAPTER 3

REQUIREMENT ANALYSIS AND SPECIFICATION

3.1 Existing System

Conventional method of managing Arts and Sports method is offline way. We have to register for events separately through some processes, look for the schedule and timings of the events on the notice board, wait in the venue until the results will be announced. In short there's a lot of time spending. Students are not able to get proper information about events. The student need to spend time to know more about these events.

Limitations

- Consumes a lot of student's time.
- There is a lot of procedures and efforts.

3.2 Proposed system

The web-app is supported to eliminate and, in some cases, to reduce the hardship faced by the existing system. The proposed system is a web-app of Arts and Sports management system which is used to completely manage Arts and Sports event at college level. It allows users to register for events, know the schedule and time of the events, view their own scores, their house's scores and also see the colourful and different moments of these events as photos. The students can register for an event at anytime from anywhere. By using this system student can save a lot of time an effort. The student can easily get the information from anywhere.

Advantages

- The students can register for an event at anytime from anywhere.
- Student can save a lot of time and effort.
- The student can easily get the information from anywhere.

3.3 Feasibility Study

After doing the project Arts and Sports management System, study and analysing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible- given unlimited resources and infinite time. Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the requirements and should be flexible enough so hat future changes can be easily done based the upcoming future requirements.

In this project feasibility tests include economical, technical, operational and behavioural feasibility of the system

3.3.1 Economical feasibility

This is an important aspect to be considered while developing a project. We decided the system to be based on minimum possible cost factor. We will be developing this in shortest possible time. We don't need to buy an external hardware or other components for the development purpose. Also, all the recourses are already available, it gives an indication of the system is economically possible for development.

3.3.2 Technical feasibility

This included the study of function performance and constraints that may affect the ability to achieve an acceptable system. The developing system must be evaluating with the technical capability. The project is feasible within the limits of current technology. The technology we used here is website technology, which is one of the widely used. We can easily provide the complete services provided by this application with available technology resource constraints. The latest versions of frameworks and IDEs are used for developing. So, this system is technically feasible.

3.3.3 Operational feasibility

The application provides effective and reliable way to utilize the online education. The website is highly accurate and efficient. The simple user interface provides easily manageable user experience, flexibility, accuracy to user. The server provides fast experience to the all the users all operations on the system is extremely fast. All the inputs taken are self-explanatory even to a layman.

3.3.4 Behavioural Feasibility

Normal human psychology of human beings indicates that people are resistant to change and computers are known to facilitate change. The project is mainly focused on informing correct and accurate educational information. The users can trust the application and simply look through various information. This application can provide a good user experience. So, we expect that the students will accept the project with their open heart.

3.4 Requirement specification

The aim of the project is to create a web-app using python. It is connected to PostgreSQL Server for database information.

Software requirement specification involves the study of the platform being used in detailed and in this case the platform being used is HTML. It also involves the detailed study of the various operations performed by the system and their relation-ship within and outside the system.

The html is a standard mark-up language for creating webpage and web application, with cascading style sheet (CSS) and java script. Its designers also leveraged may tried – and – true approaches proven to work in the wireless world. It's true that many of these features appear in existing in proprietary platform. Web browser receives html document from a web server or from a local storage and the document into multimedia web pages. HTML describes the structure of a web page semantically and originally includes cues for the appearance of document.

3.4.1 Functional Requirements

- Super Admin: Has the complete privileges of this web-app. Only super admin can add/view/delete other admins for this system
- Arts/Sports/Media Admin: Can add/view/delete the tables allowed by the Super Admin.
- User: Users can register for events, view event schedules, view scores, view photos of various events.

3.4.2 Non-Functional Requirements

- Performance: The website is compatible with major browsers, and will work perfectly
- Usability: The simple user interface, accuracy and flexibility of the website gives ease to use it
- Efficiency: when a user uses this website, then it will be easier for him to do different activities specified in the website within a minimum amount of time

3.5 Environmental Details

Environmental requirements for the smooth functioning of this product could be configured based on the requirement needed by the component of the operating environment that works as front – end system. Here we suggest minimum configuration for both hardware and software components.

3.5.1 Hardware specification

The following are the hardware used for development of the application.

- Processor : Intel core i3
- RAM : 4 GB
- Hard disk : 500 GB
- Input device : Standard QWERTY keyboard,
  Two button mouse.
- Output device : Monitor

3.5.2 Software specification

The following are the software needed for the development of the application

- IDE : Visual Studio Code
- Front-end : HTML, CSS, JavaScript
- Backend : Python
- Database : PostgreSQL
- Frameworks : Django, Tailwind CSS
- Operating system : Windows 10

3.5.3 Software description

HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages.

HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as and directly introduce content into the page. Other tags such as surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page. HTML can embed programs written in a scripting language such as JavaScript, which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via

speech based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.

In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

JavaScript

JavaScript often abbreviated as JS, is a programming language that conforms to the ECMA Script specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web.

JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behaviour, and all major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

Originally used only in web browsers, JavaScript engines are also now embedded in server side website deployments and non-browser applications. Although there are similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

PostgreSQL

PostgreSQL is a powerful, open source object-relational database system that uses and extends the SQL language combined with many features that safely store and scale the most

complicated data workloads. The origins of PostgreSQL date back to 1986 as part of the POSTGRES project at the University of California at Berkeley and has more than 30 years of active development on the core platform. PostgreSQL comes with many features aimed to help developers build applications, administrators to protect data integrity and build fault-tolerant environments, and help you manage your data no matter how big or small the dataset. In addition to being free and open source, PostgreSQL is highly extensible. For example, you can define your own data types, build out custom functions, even write code from different programming languages without recompiling your database. There are many more features that you can discover in the PostgreSQL documentation. Additionally, PostgreSQL is highly extensible: many features, such as indexes, have defined APIs so that you can build out with PostgreSQL to solve your challenges. PostgreSQL has been proven to be highly scalable both in the sheer quantity of data it can manage and in the number of concurrent users it can accommodate. There are active PostgreSQL clusters in production environments that manage many terabytes of data, and specialized systems that manage petabytes.

Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed. Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-testdebug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective. Python's features include –

- Easy to learn - Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly

- Easy to read - Python code is more clearly defined and visible to the eyes
- Easy to maintain – Python source code is fairly easy to maintain
- A broad standard library − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- Interactive Mode − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- Portable − Python can run on a wide variety of hardware platforms and has the same interface on all platforms
- Extendable − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient
- Databases − Python provides interfaces to all major commercial databases
- GUI Programming − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- Scalable − Python provides a better structure and support for large programs than shell scripting
- It supports functional and structured programming methods as well as OOP
- It can be used as a scripting language or can be compiled to byte-code for building large applications.

Django

Django is a Python-based free and open-source web framework that follows the model–template–views (MTV) architectural pattern. Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source. It is maintained by the Django Software Foundation (DSF), an independent organization established in the US as a 501(c)(3) non-profit. Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes reusability and "pluggability" of components, less code, low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings, files, and data models. Django also provides an optional administrative create, read, update and delete interface that

is generated dynamically through introspection and configured via admin models. Some well-known sites that use Django include Instagram, Mozilla, Disqus, Bitbucket, Nextdoor and Clubhouse. Django's features include –

- Ridiculously fast- Django was designed to help developers take applications from concept to completion as quickly as possible.
- Reassuringly secure- Django takes security seriously and helps developers avoid many common security mistakes.
- Exceedingly scalable- Some of the busiest sites on the web leverage Django's ability to quickly and flexibly scale.

Tailwind

Tailwind CSS is a CSS framework. It's somewhat similar to popular frameworks, like Bootstrap and Materialize, in that you apply classes to elements and it styles them. But it is also atomic CSS in that one class name does one thing. While Tailwind does have Tailwind UI for pre-built componentry, generally you customize Tailwind to look how you want it to look.
Tailwind other features include

- With Tailwind, you get thousands of out-of-the-box CSS classes that you just need to apply to your HTML elements.
- The names are simple and they do a good job of telling you what their functions are. For example, `text-sm` gives your text a small font size**.** This is a breath of fresh air for people that struggle with naming custom CSS classes
- By utilizing a mobile-first approach, responsiveness is at the heart of Tailwind's design. Making use of the `sm`, `md`, and `lg` prefixes to specify breakpoints, you can control the way styles are rendered across different screen sizes. For example, if you use the `md` prefix on a style, that style will only be applied to medium-sized screens and larger. Small screens will not be affected.

Tailwind might also not be for you if you are someone who prefers ready-made components to avoid stress and save time, or you are working on a project with a short deadline.

CHAPTER 4

SYSTEM DESIGN

4.1 User of the system

The main users of the system are:

- Super admin
- Art/Sport/Media admins
- User

Super Admin:

The super admin will have the complete privilege in the system. The super admin can do the following-

- Add/view/delete/modify admins.
- Add/view/delete/modify Arts/Sports events
- Add/view/delete/modify Arts/Sports event details
- Add/view/delete/modify Arts/Sports participants
- Add/view/delete/modify Arts/Sports houses
- Add/view/delete/modify Arts/Sports gallery

Art admin:

Art admins can only mange tables related to arts only. Their privileges include-

- Add/view/delete/modify Arts event
- Add/view/delete/modify Arts event details
- Add/view/delete/modify Arts participants
- View Arts Houses
- View Arts Gallery

Sport admin:

Sport admins can only mange tables related to sports only. Their privileges include-

- Add/view/delete/modify Sports event
- Add/view/delete/modify Sports event details
- Add/view/delete/modify Sports participants

- View Sports Houses
- View Sports Gallery

Media admin:

Media admins can mange gallery tables related to both Arts and Sports. Their privileges include-

- Add/view/delete/modify Arts Gallery
- Add/view/delete/modify Sports Gallery

Users

Normal website users once logged in can have the access to Register for events, view the details and schedules of events, view live scores and view the gallery.

4.2 Architectural design

The software architecture of a computing system is the structure of the system, which comprise application components, the externally visible properties of those components, and the relationships between them. The term also refers to documentation of a system's software architecture. Documenting software architecture facilitates communication between stakeholders, documents early decisions about high-level design, and allows reuse of design components and patterns between projects. Architecture is commonly defined in terms of components and connectors. Components are identified and assigned responsibilities that client components interact with through "contracted" interfaces. Component interconnections specify communication and control mechanisms, and support all component interactions needed to accomplish system behaviour.

4.2.1 Database design

Database name: artsport

1. Table name: User

Field name	Datatypes	Constraints
Id	Integer	Primary key
Username	Varchar	Not null, unique
Password	Varchar	Not null
First name	Varchar	Not null
Last name	Varchar	
Email	Varchar	Not null
Is_staff	Boolean	
Is_superuser	Boolean	
Date_joined	Timestamp with timezone	Not null
Last_login	Time stamp with timezone	

Table. 4.2.1.1

2. Table name: arts_house

Field name	Datatypes	Constraints
id	Int	Primary key
house_name	Varchar	Not Null

Table. 4.2.1.2

3. Table name: arts_event

Field name	Datatypes	Constraints
id	Int	Primary key
event_name	Varchar	Not Null

Table. 4.2.1.3

4. Table name: arts_participant

Field name	Datatypes	Constraints
id	Int	Primary key
participant_name	Varchar	Not Null
participant_department	Varchar	Not null
Participant_score	Int	
participant_event_1_id	Int	Foreign key, Not null
participant_event_2_id	Int	Foreign key
participant_event_3_id	Int	Foreign key
participant_email	Varchar	Not null
participant_mobile	Varchar	Not null
participant_year	Varchar	Not null
participant_house_id	Int	Foreign key, Not null

Table. 4.2.1.4

5. Table name: arts_eventdetail

Field name	Datatypes	Constraints
id	Int	Primary key
event_name_id	Int	Foreign key, Not null
event_dateTime	Timestamp with timezone	Not null
rank1_id	Int	Foreign key
rank2_id	Int	Foreign key
rank3_id	Int	Foreign key

Table. 4.2.1.5

6. Table name: arts_gallery

Field name	Datatypes	Constraints
Id	Int	Primary key
image_name	Varchar	Not Null
image_desc	Text	Not null
image_file	varchar	Not null

Table. 4.2.1.6

7. Table name: sports_house

Field name	Datatypes	Constraints
id	Int	Primary key
house_name	Varchar	Not Null

Table. 4.2.1.7

8. Table name: sports_event

Field name	Datatypes	Constraints
id	Int	Primary key
event_name	Varchar	Not Null

Table. 4.2.1.8

9. Table name: sports_participant

Field name	Datatypes	Constraints
id	Int	Primary key
participant_name	Varchar	Not Null
participant_department	Varchar	Not null
Participant_score	Int	
participant_event_1_id	Int	Foreign key, Not null
participant_event_2_id	Int	Foreign key
participant_event_3_id	Int	Foreign key
participant_email	Varchar	Not null
participant_mobile	Varchar	Not null
participant_year	Varchar	Not null
participant_house_id	Int	Foreign key, Not null

Table. 4.2.1.9

10. Table name: sports_eventdetail

Field name	**Datatypes**	**Constraints**
id	Int	Primary key
event_name_id	Int	Foreign key, Not null
event_dateTime	Timestamp with timezone	Not null
rank1_id	Int	Foreign key
rank2_id	Int	Foreign key
rank3_id	Int	Foreign key

Table. 4.2.1.10

11. Table name: sports_gallery

Field name	**Datatypes**	**Constraints**
Id	Int	Primary key
image_name	Varchar	Not Null
image_desc	Text	Not null
image_file	varchar	Not null

Table. 4.2.1.11

4.3 Logical Design

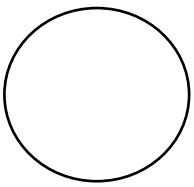
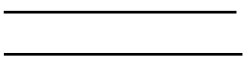
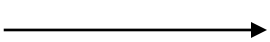
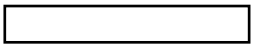
It defines the relationship between major structural elements of the program. This modular framework of a computer program can be derived from the analysis models and the interaction of the subsystems within the analysis model. The primary objective is to develop a modular program structure and represent a relationship between the modules. In addition, the logical design methods program structure and data structure, defining interface that enables data to flow throughout the program. The logical design of various modules in the CDAS is described using Data Flow Diagram.

4.3.1 Data Flow Diagram (DFD)

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. A data flow diagram can also be used for the visualization of data processing (structured design). It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context level DFD is then "exploded" to show more detail of the system being modelled.

Data Flow Diagram Notations

We can use different types of notations on Data Flow Diagram. There are four basic symbols in this notation

- Process 
- Data store 
- Data flow 
- External entity 

DFD Level 0

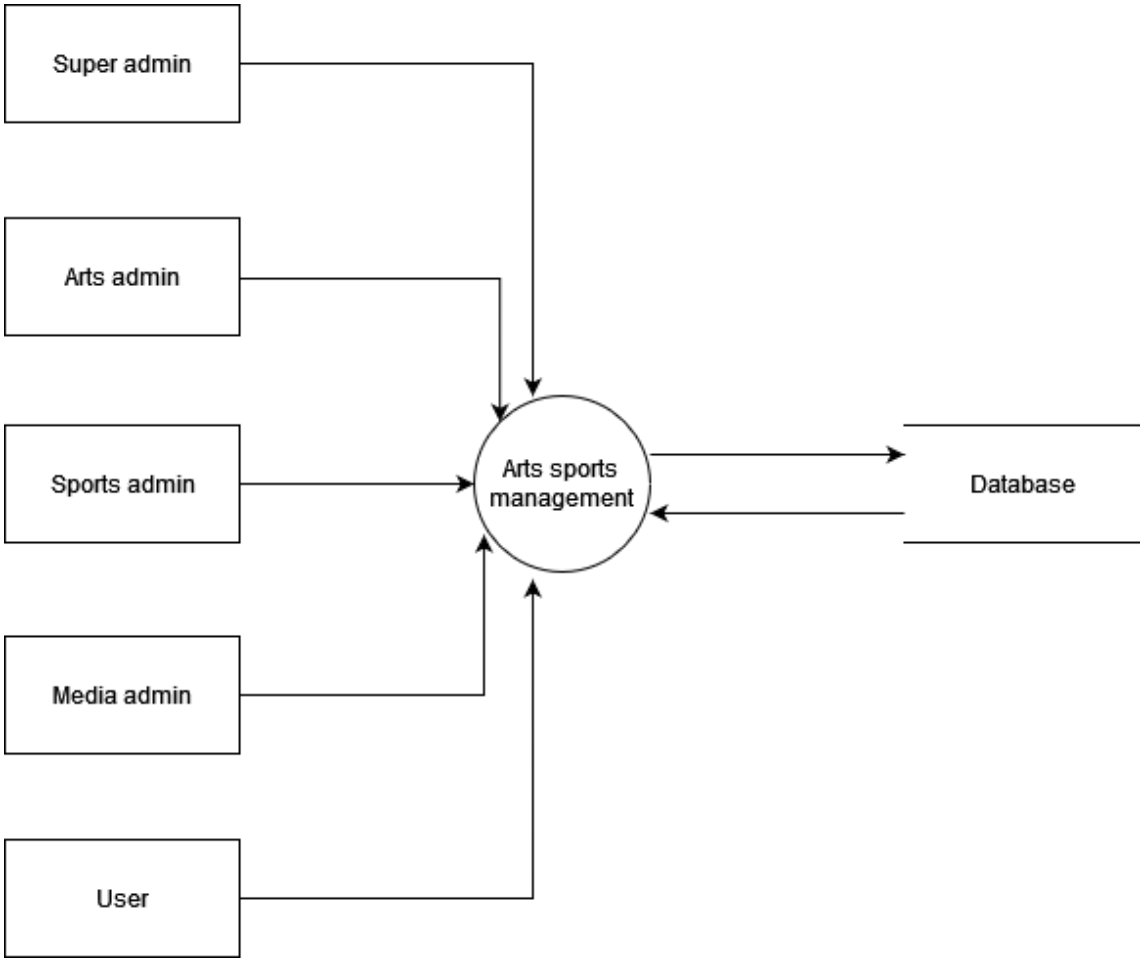


Fig:4.3.1.1

DFD Level 1.1(Super admin)

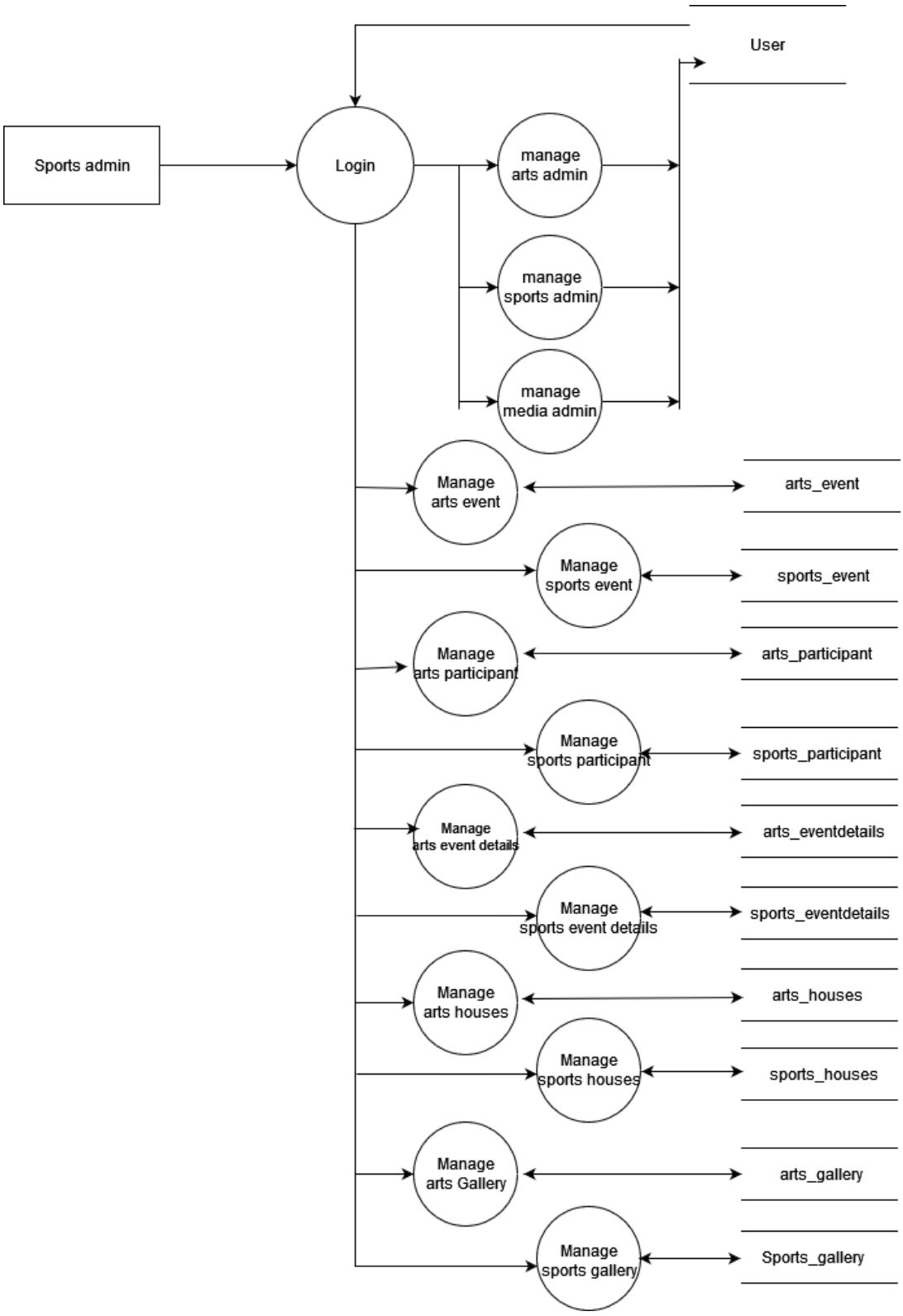


Fig:4.3.1.2

DFD Level 1.2(Arts admin)

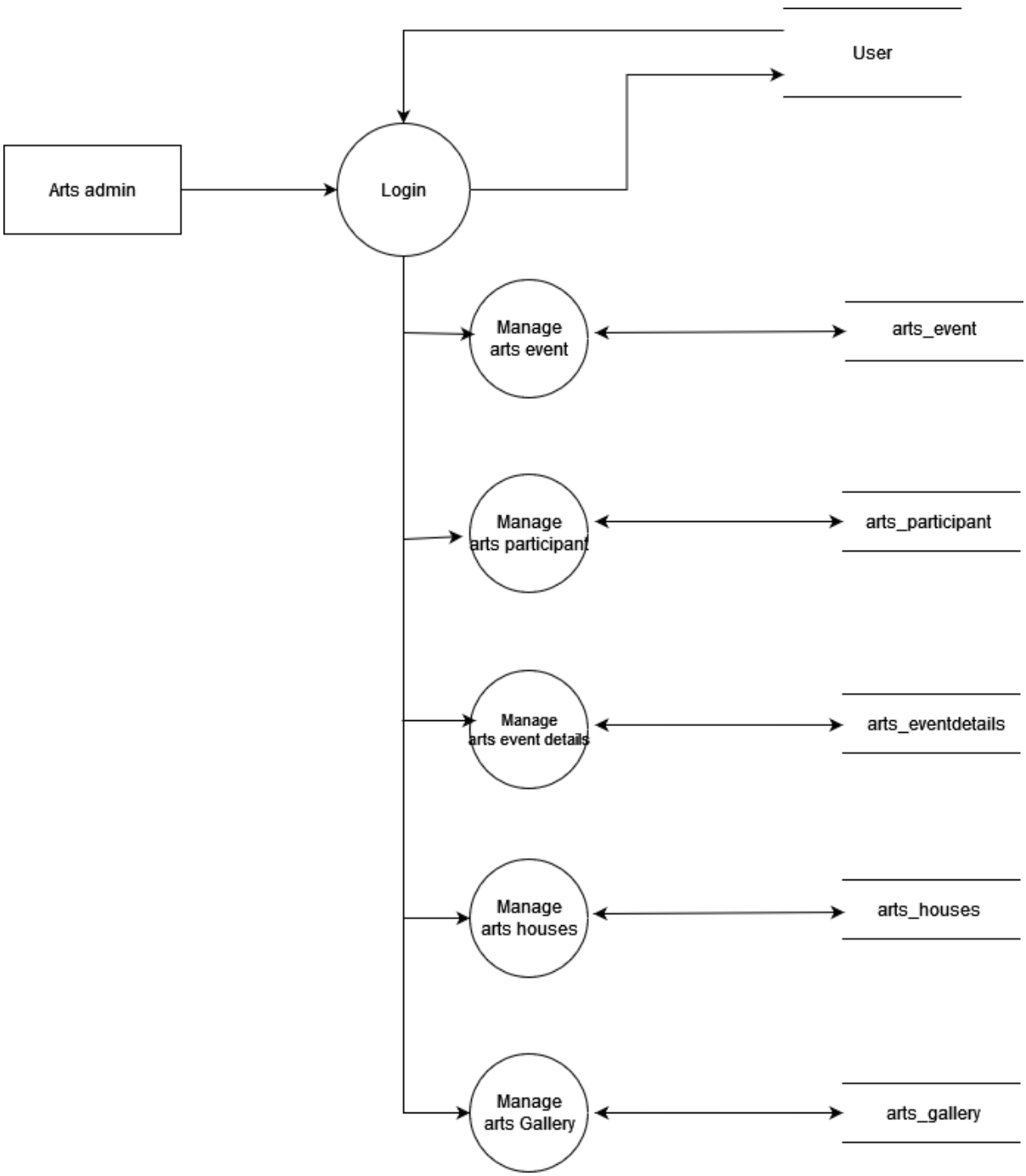


Fig:4.3.1.3

DFD Level 1.3(Sports admin)

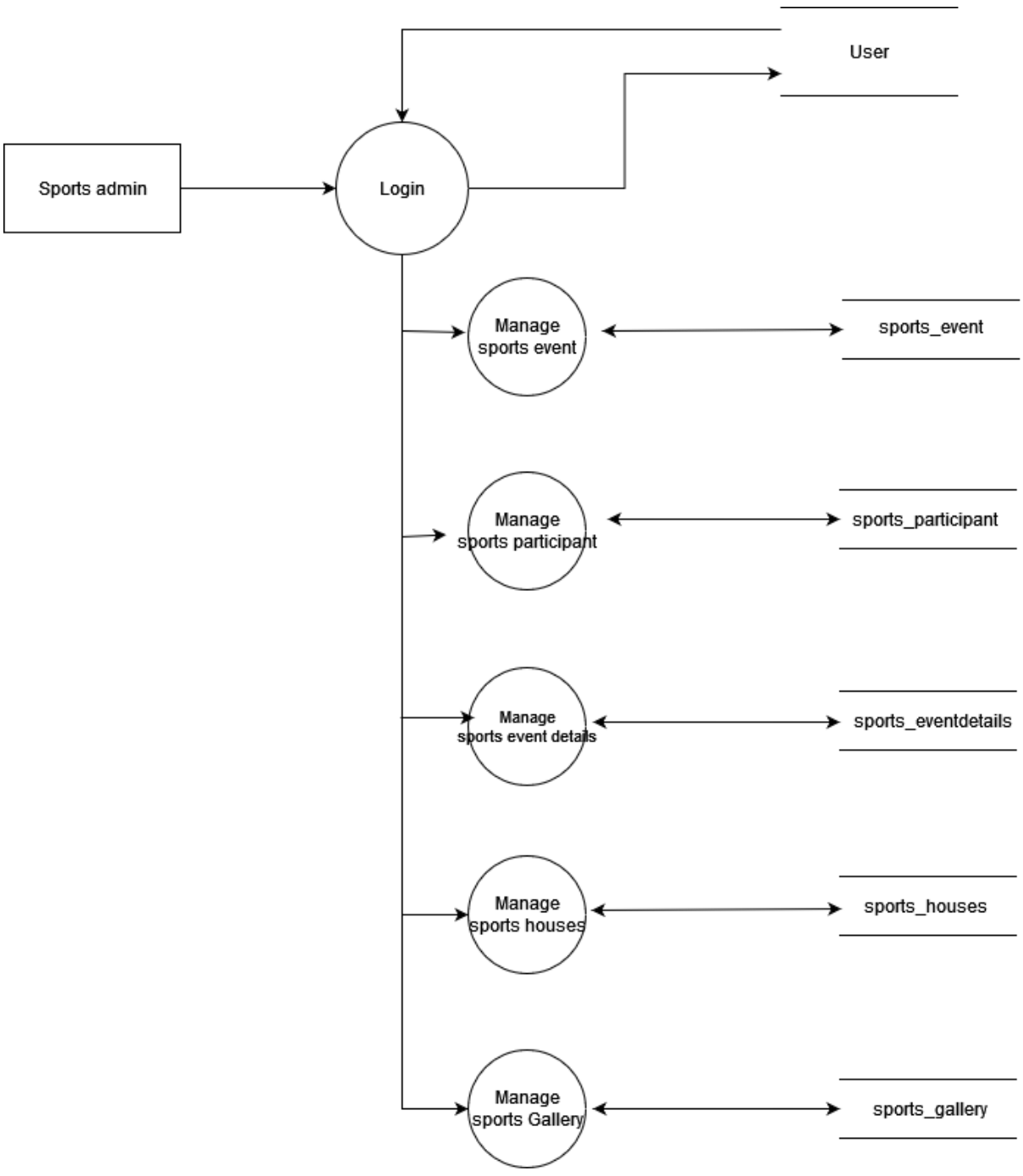


Fig:4.3.1.4

DFD Level 1.4(Media admin)

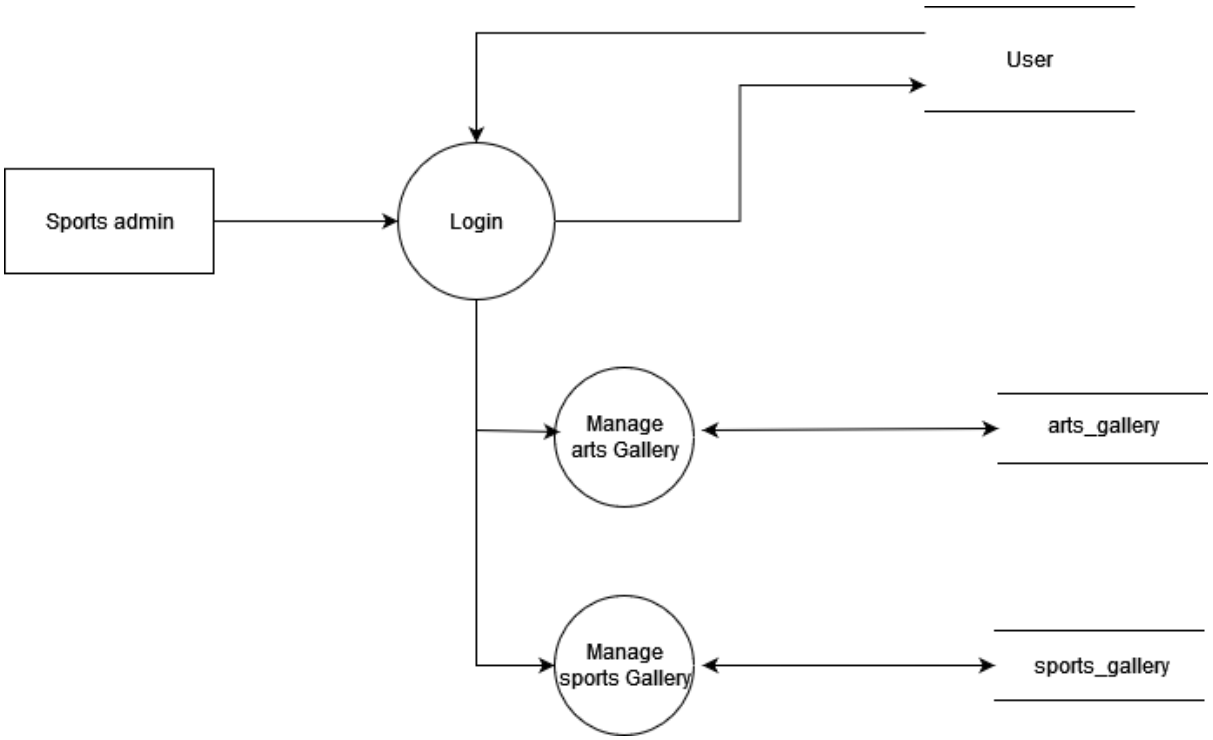


Fig:4.3.1.5

DFD Level 1.5(Users)

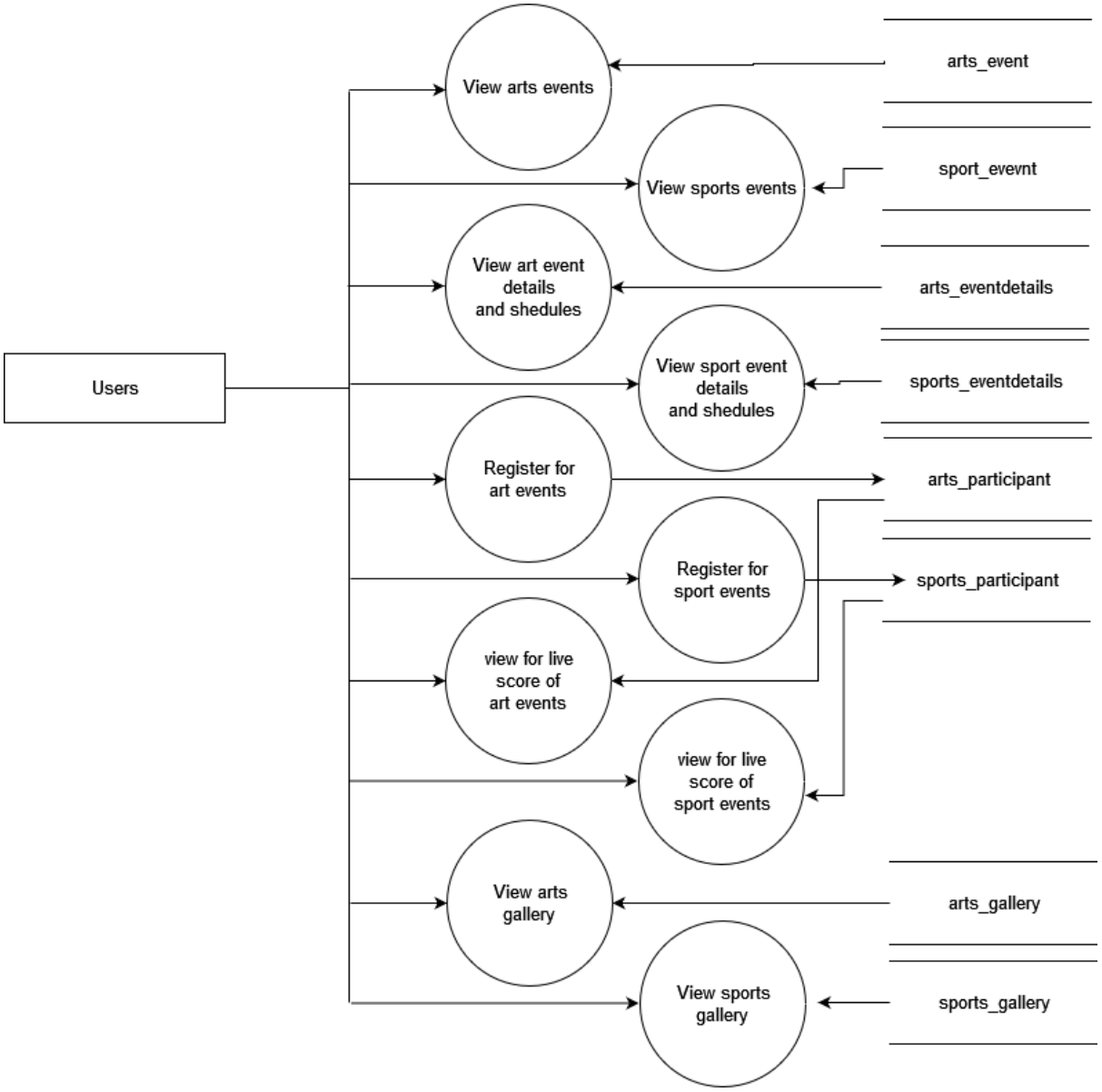


Fig:4.3.1.6

DFD Level 2.1.1(Super admin manage arts admin)

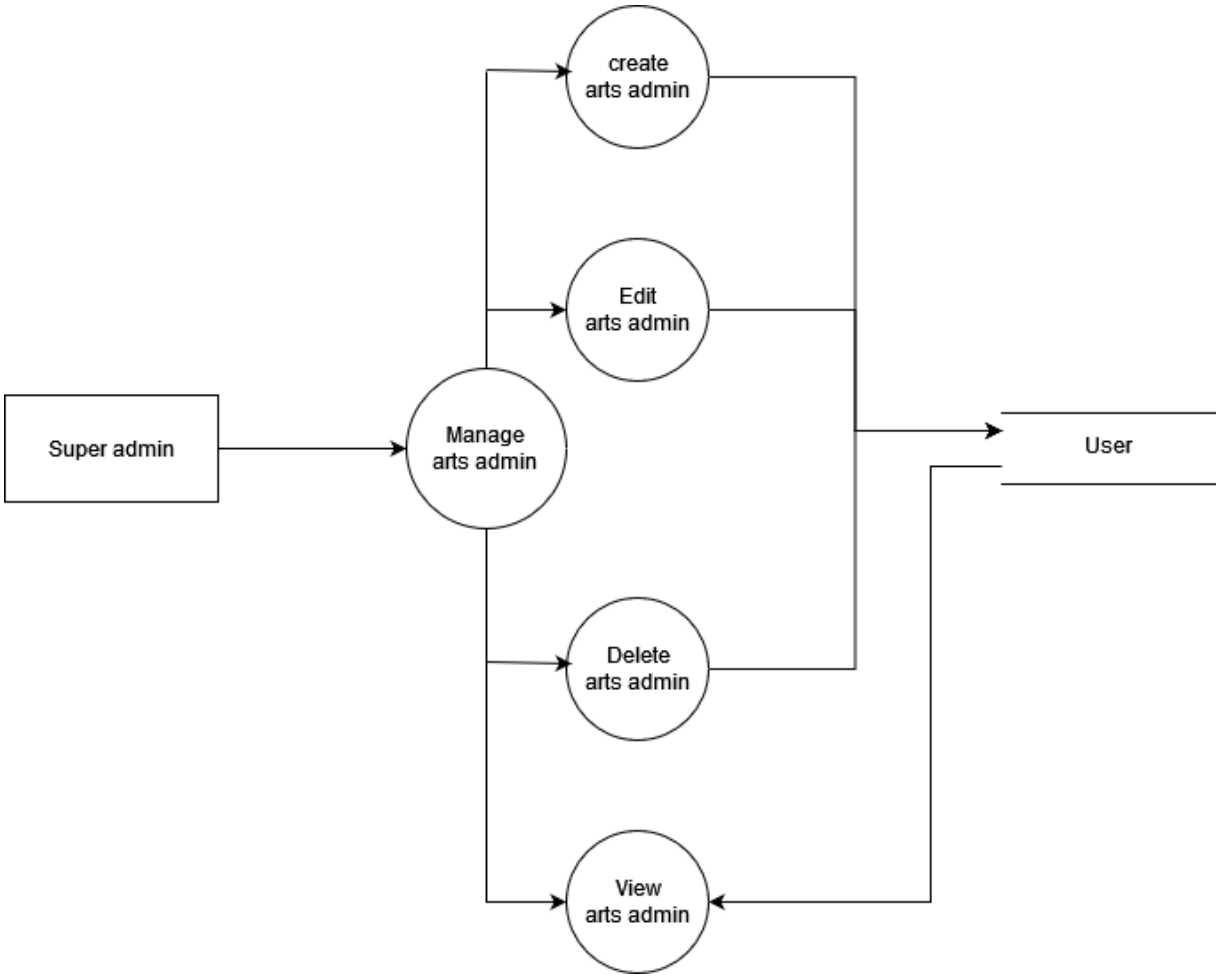


Fig:4.3.1.7

DFD Level 2.1.2(Super admin manage sports admin)

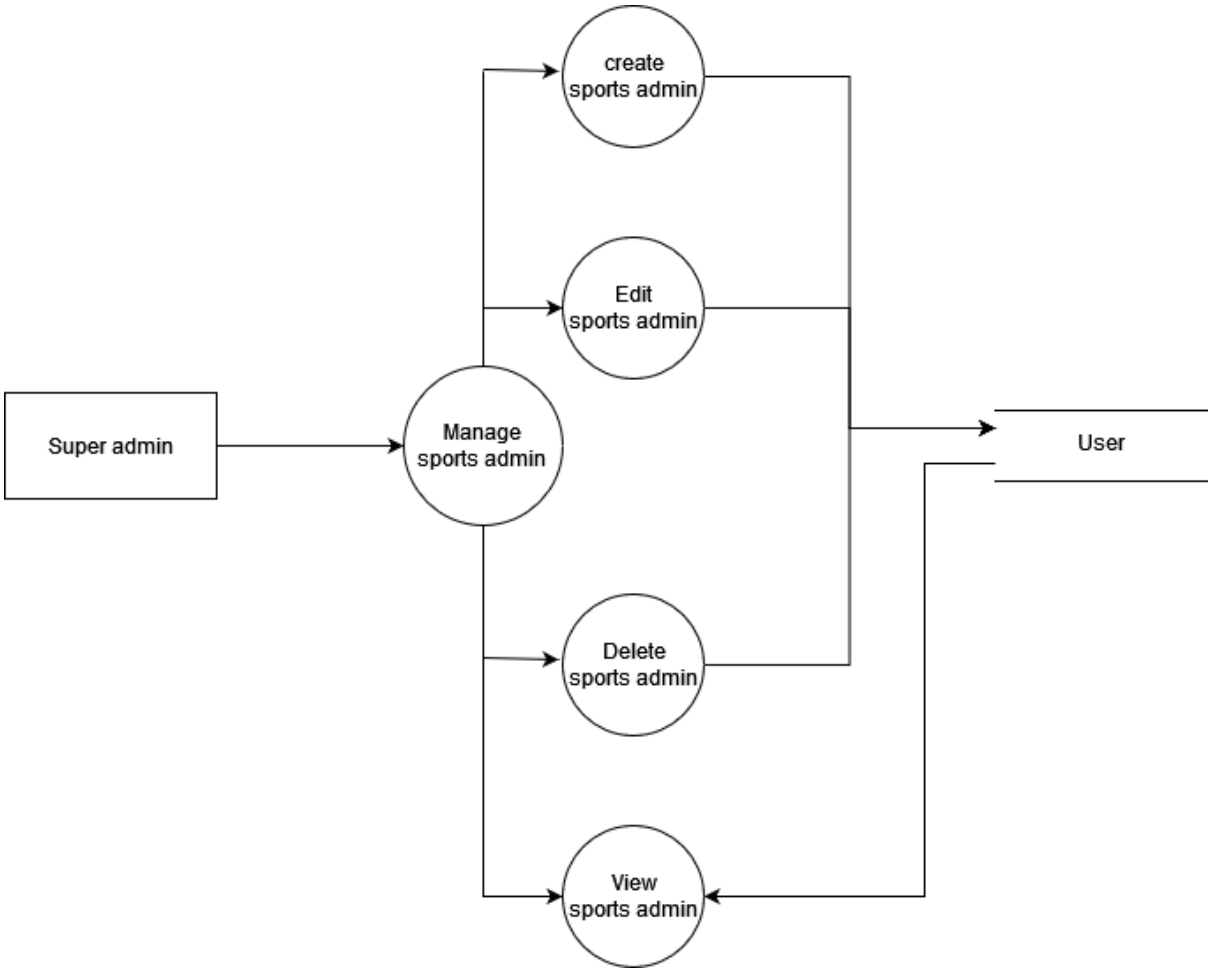


Fig:4.3.1.8

DFD Level 2.1.3(Super admin manage media admin)

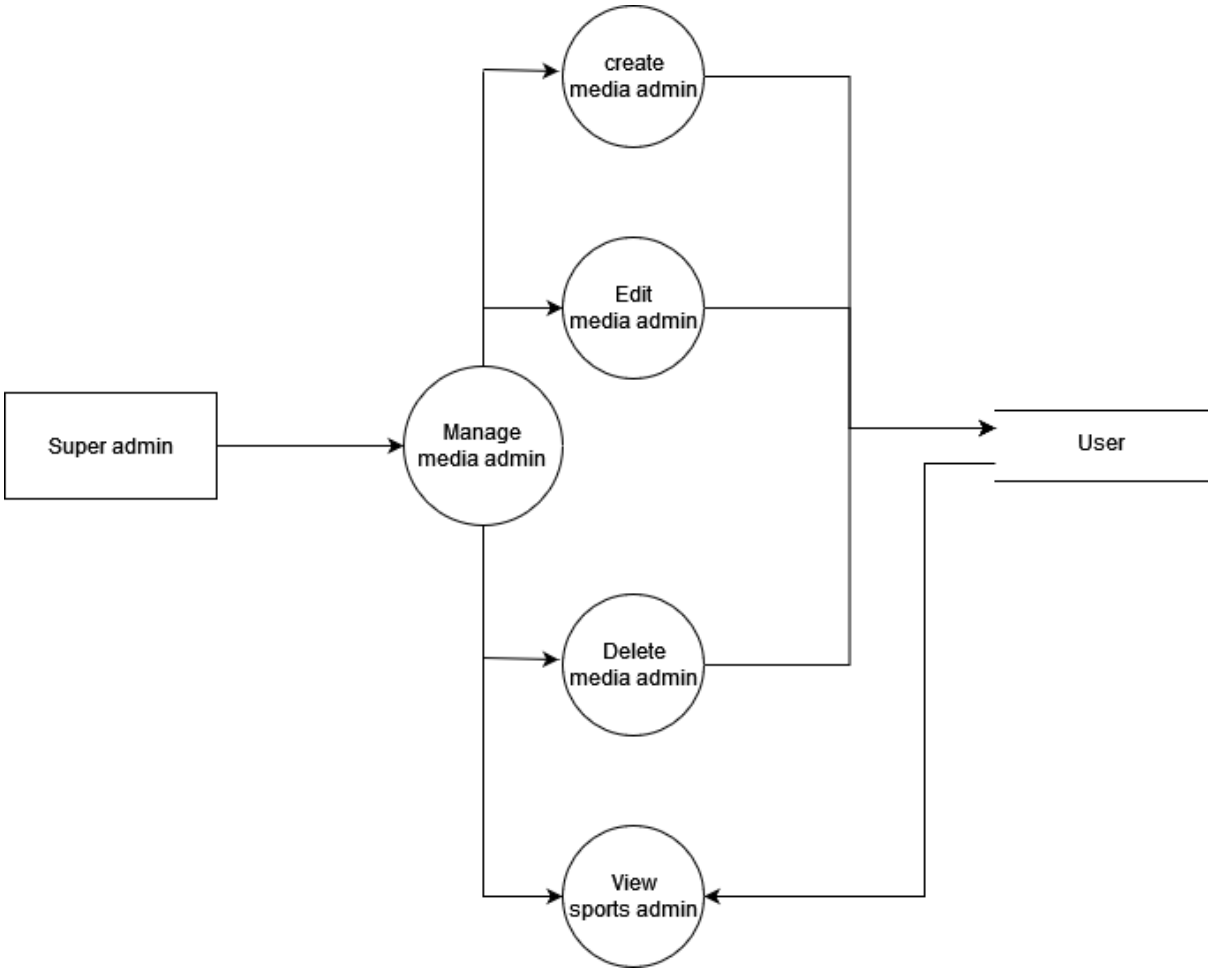


Fig:4.3.1.9

DFD Level 2.1.4(Super admin manage arts events)

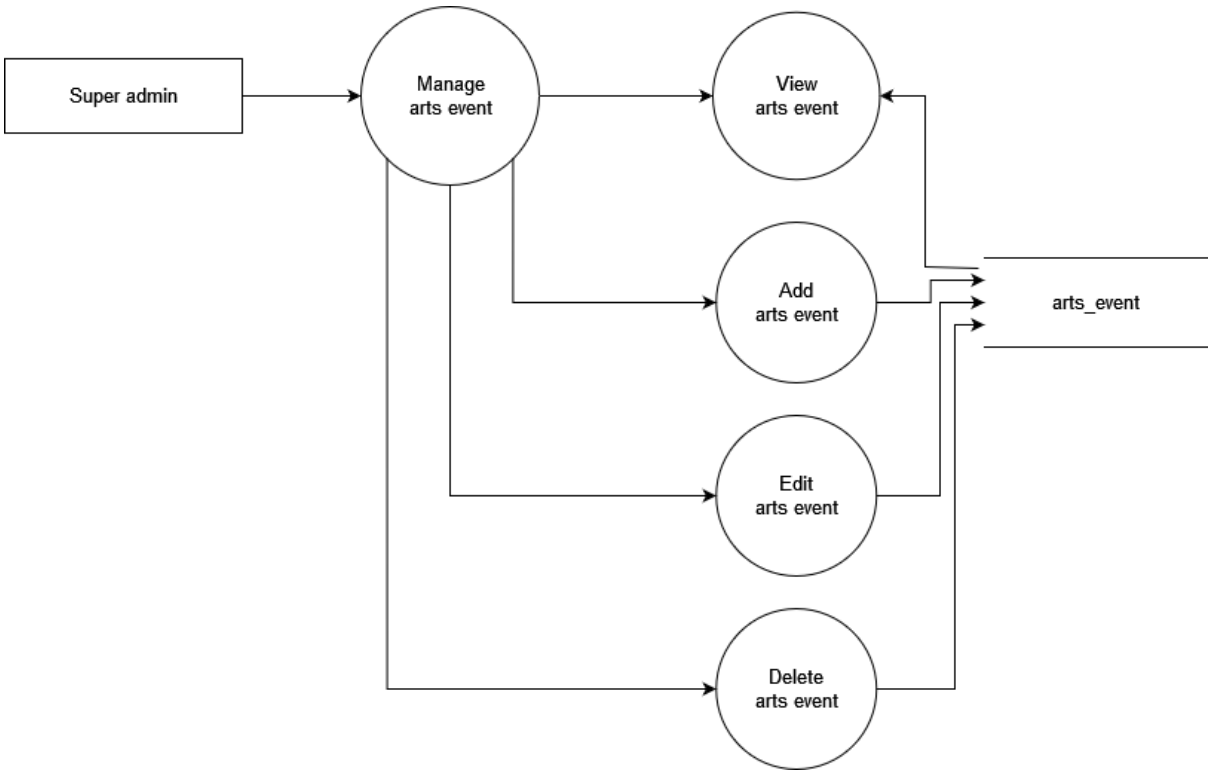


Fig:4.3.1.10

DFD Level 2.1.5(Super admin manage sports events)

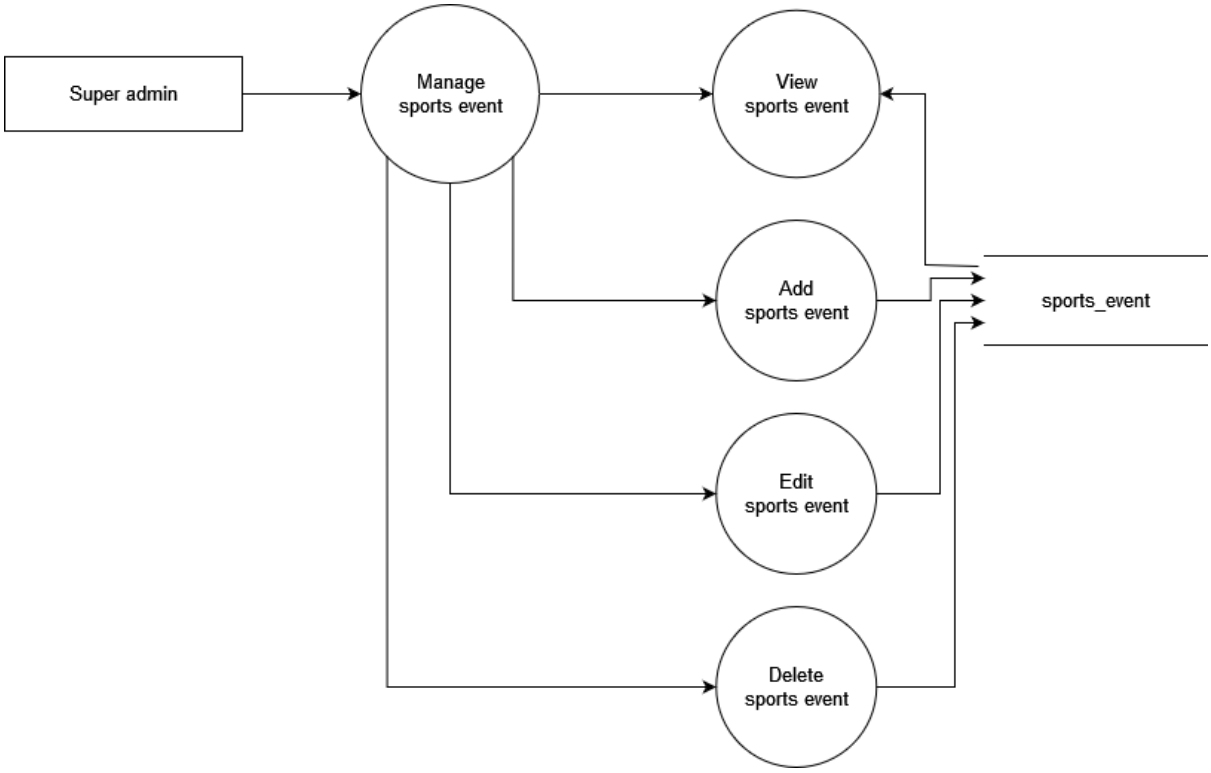


Fig:4.3.1.11

DFD Level 2.1.6(Super admin manage arts participant)

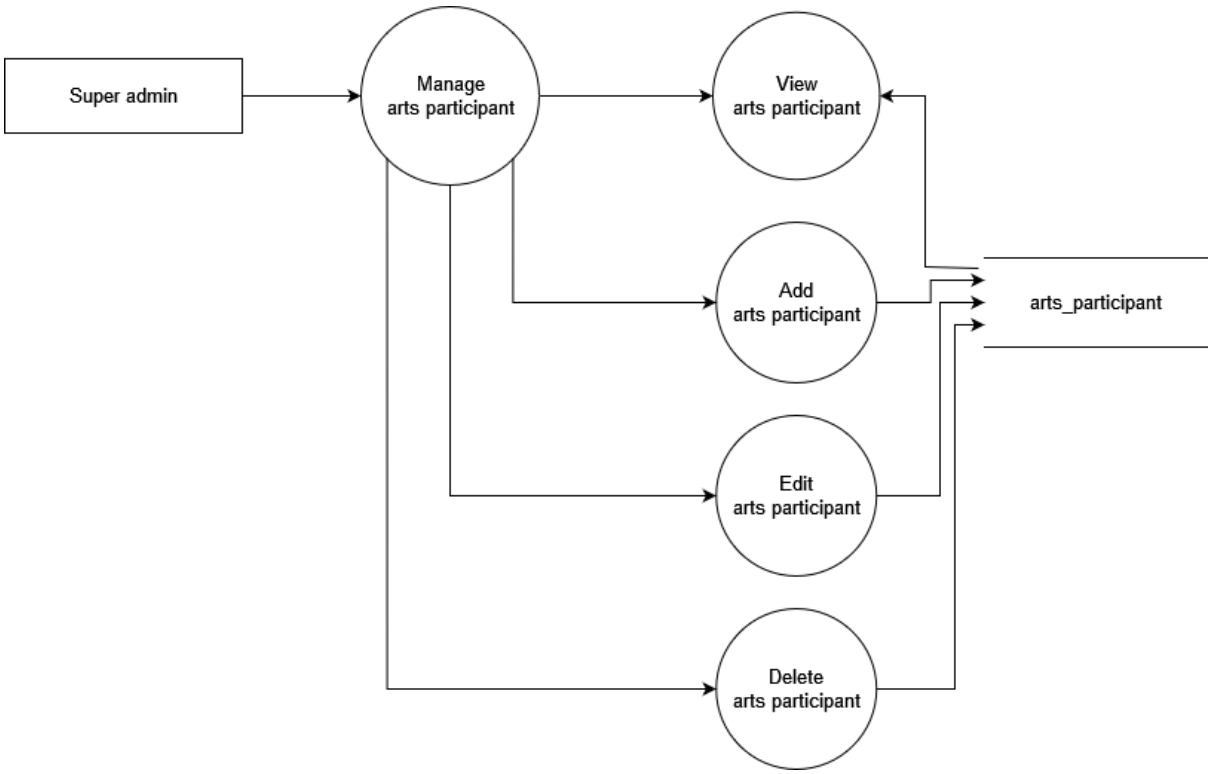


Fig:4.3.1.12

DFD Level 2.1.7(Super admin manage sports participant)

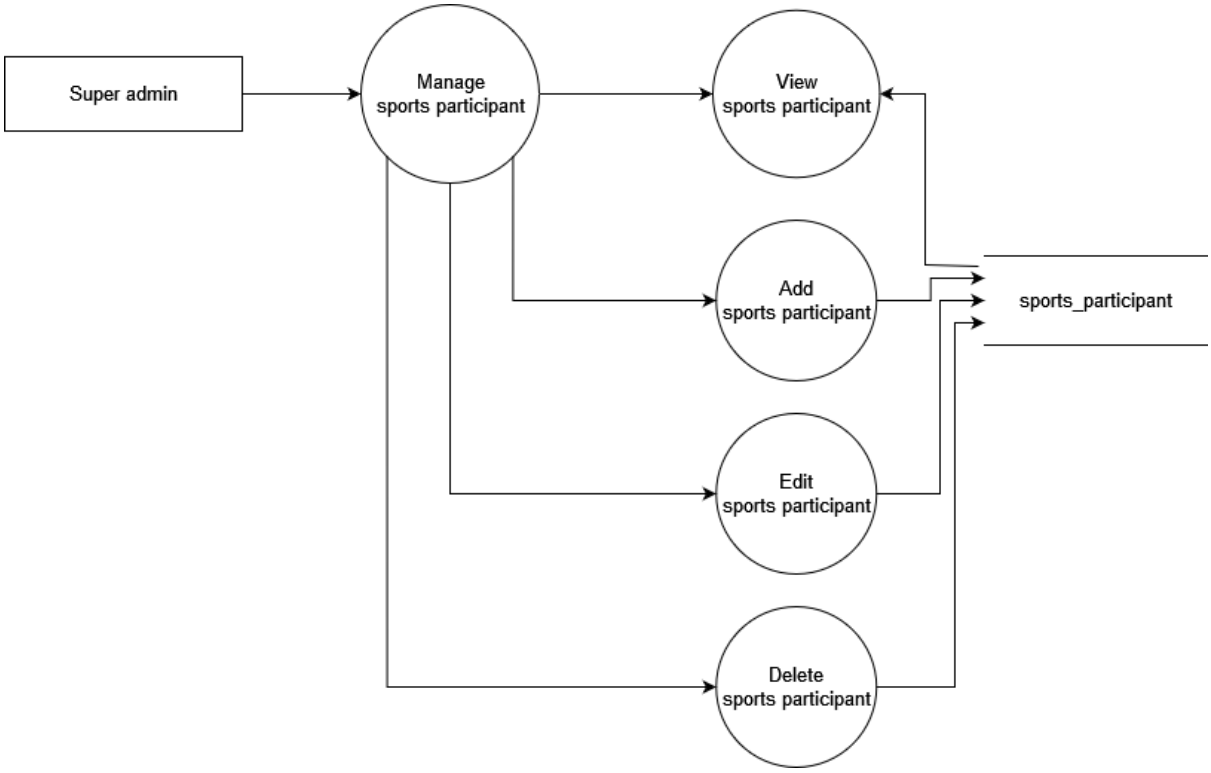


Fig:4.3.1.13

DFD Level 2.1.8(Super admin manage arts event details)

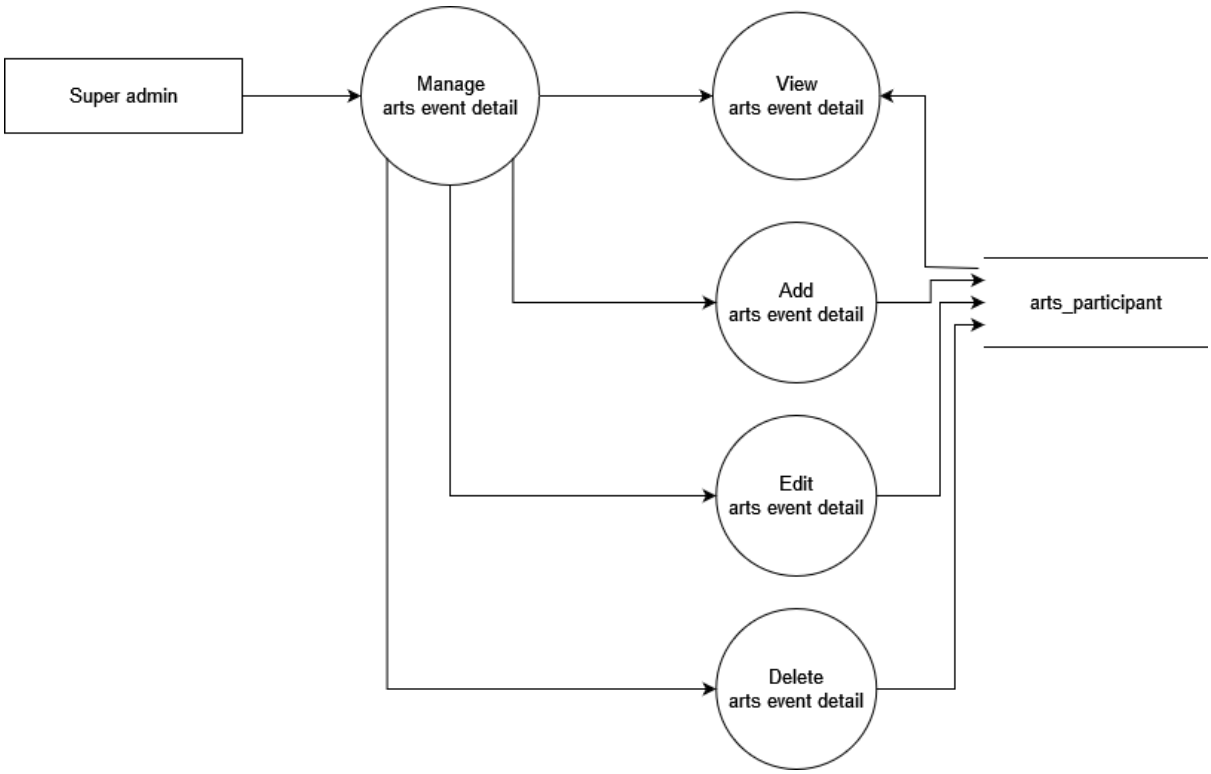


Fig:4.3.1.14

DFD Level 2.1.9(Super admin manage sports event details)

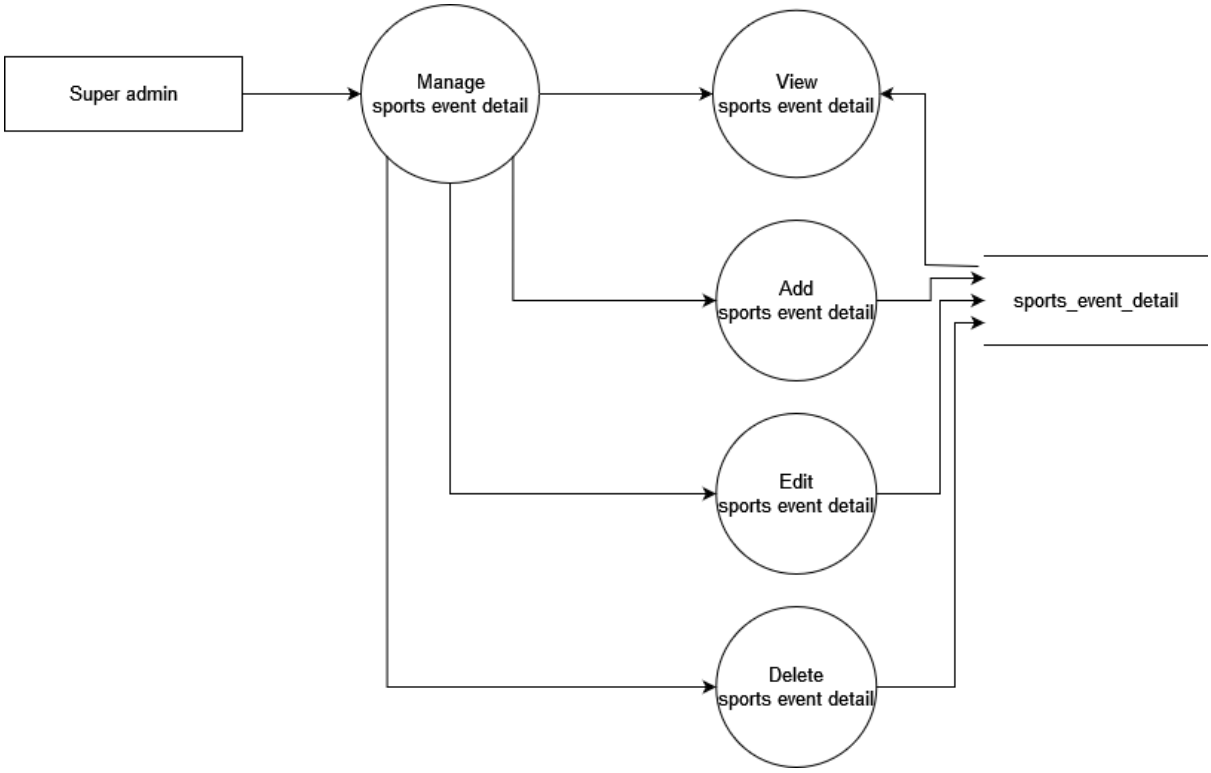


Fig:4.3.1.15

DFD Level 2.1.10(Super admin manage arts house)

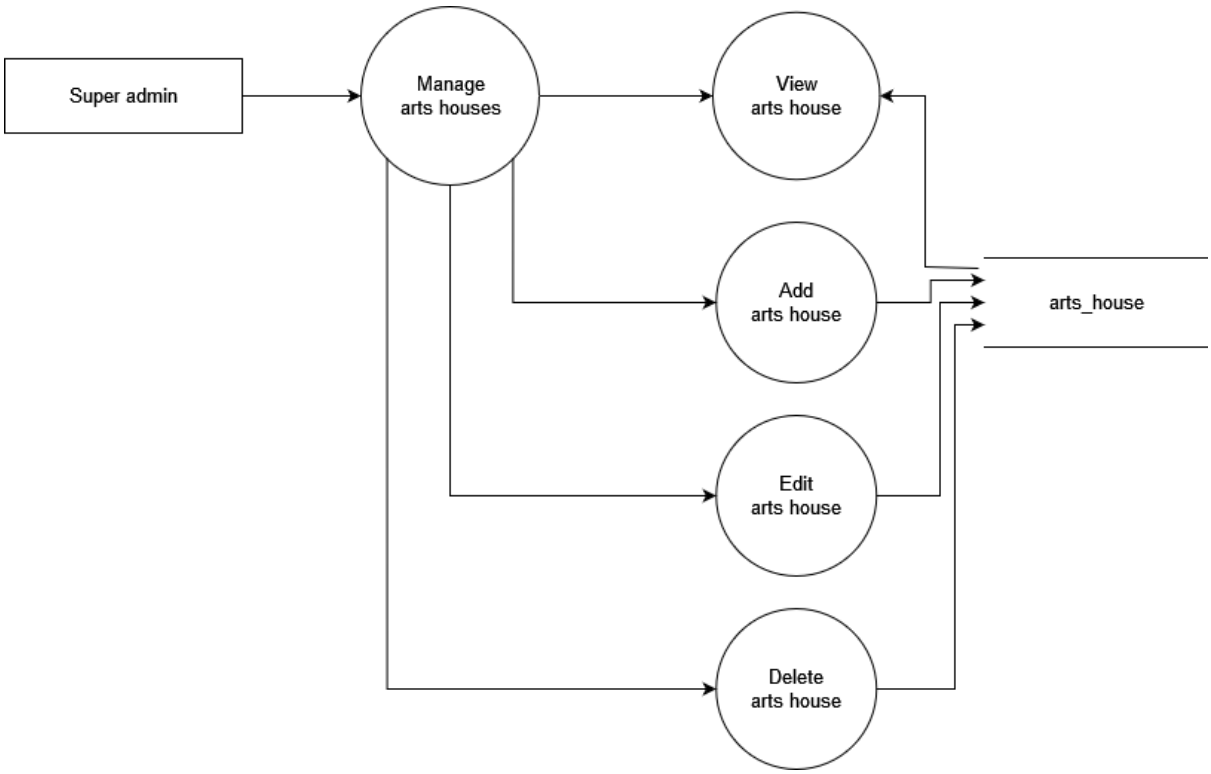


Fig:4.3.1.16

DFD Level 2.1.11(Super admin manage sports house)

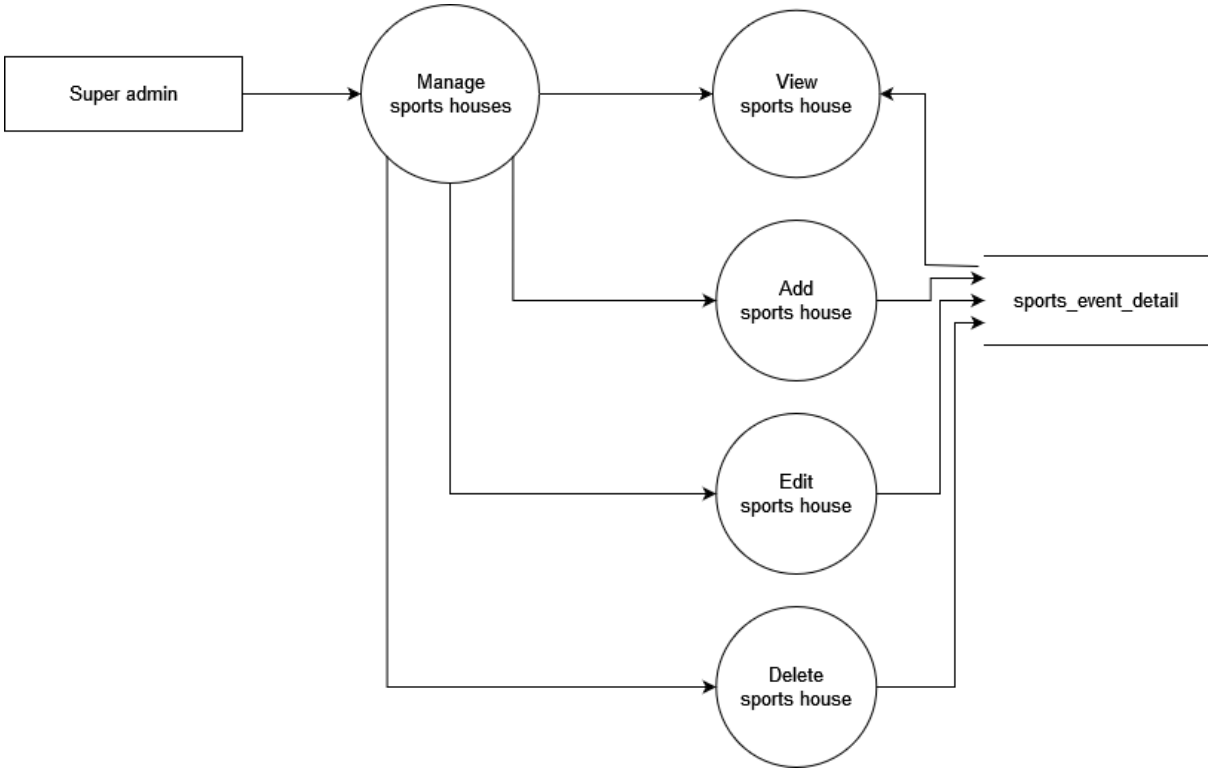


Fig:4.3.1.17

DFD Level 2.1.12(Super admin manage arts gallery)

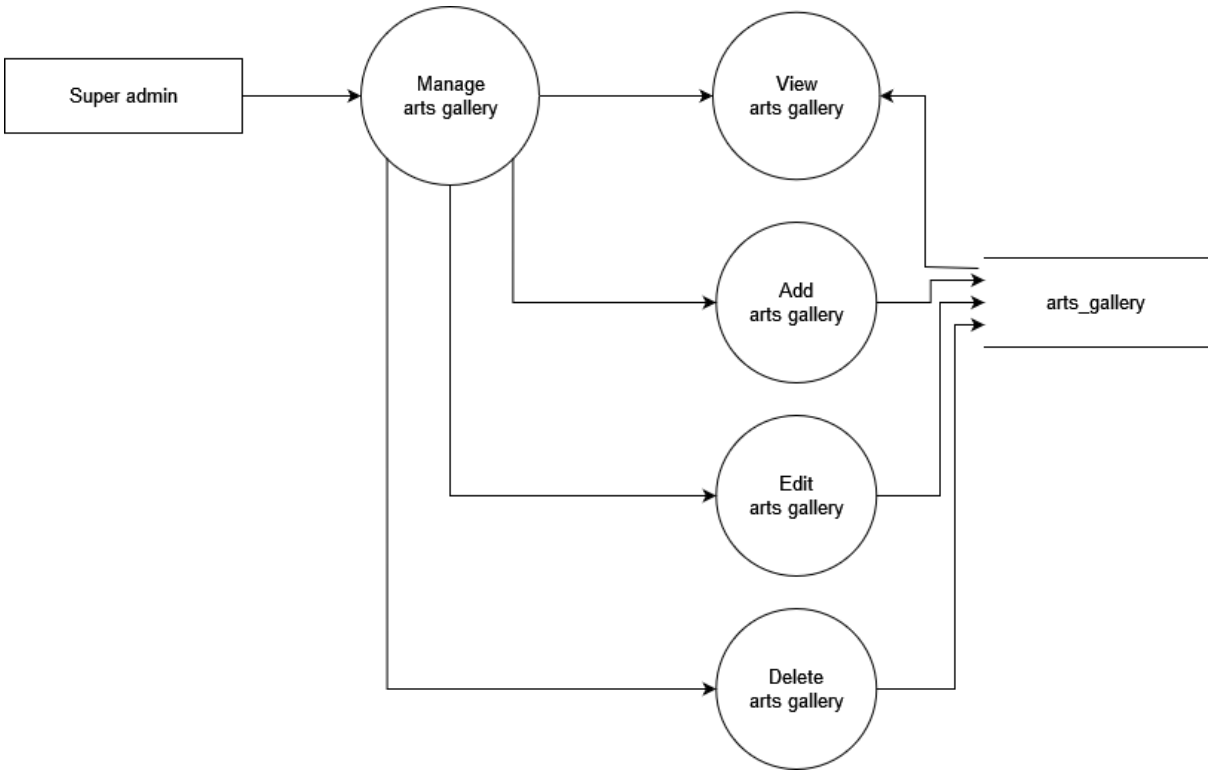


Fig:4.3.1.18

DFD Level 2.1.13(Super admin manage sports gallery)

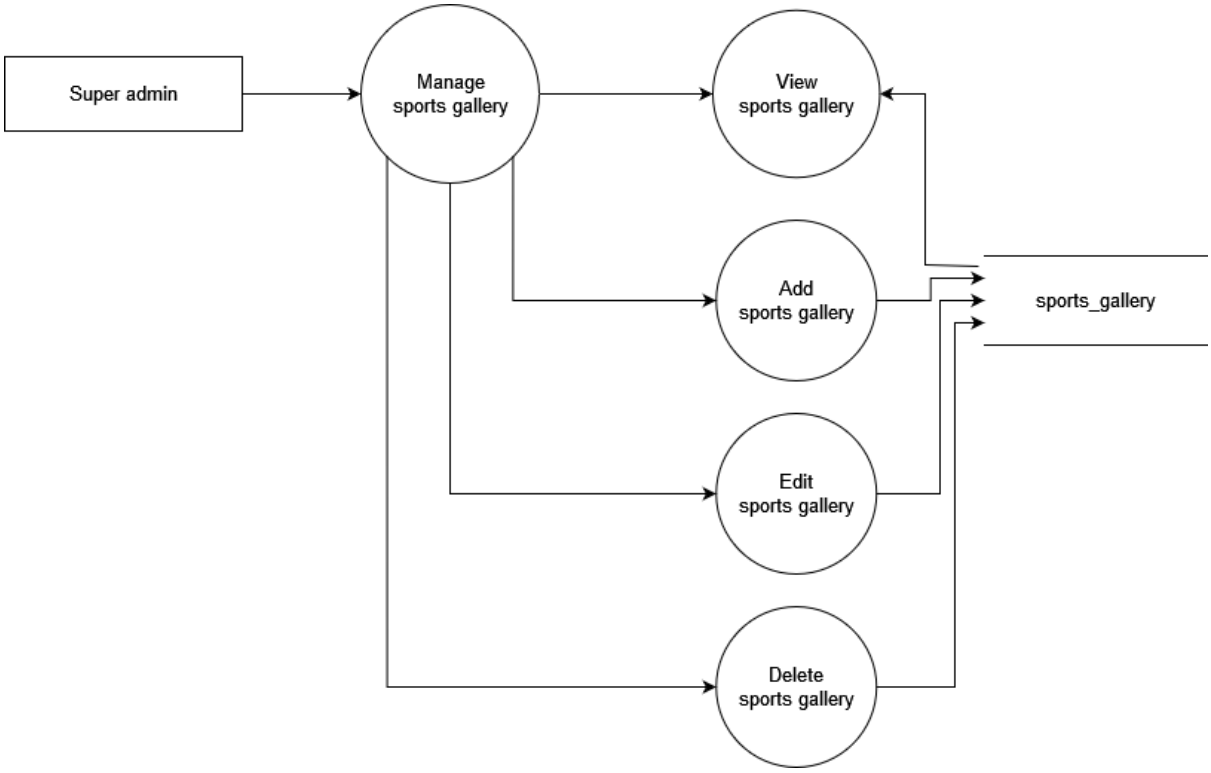


Fig:4.3.1.19

DFD Level 2.2.1(Arts admin manage arts event)

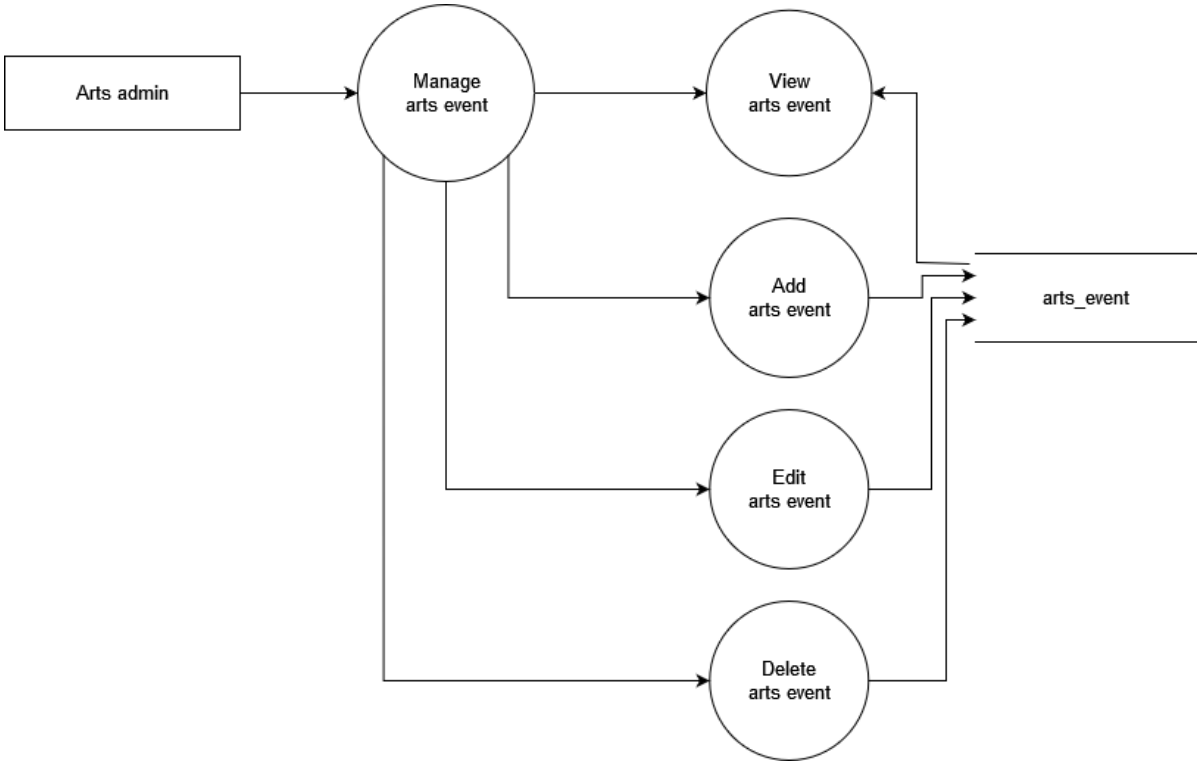


Fig:4.3.1.20

DFD Level 2.2.2(Arts admin manage arts participant)

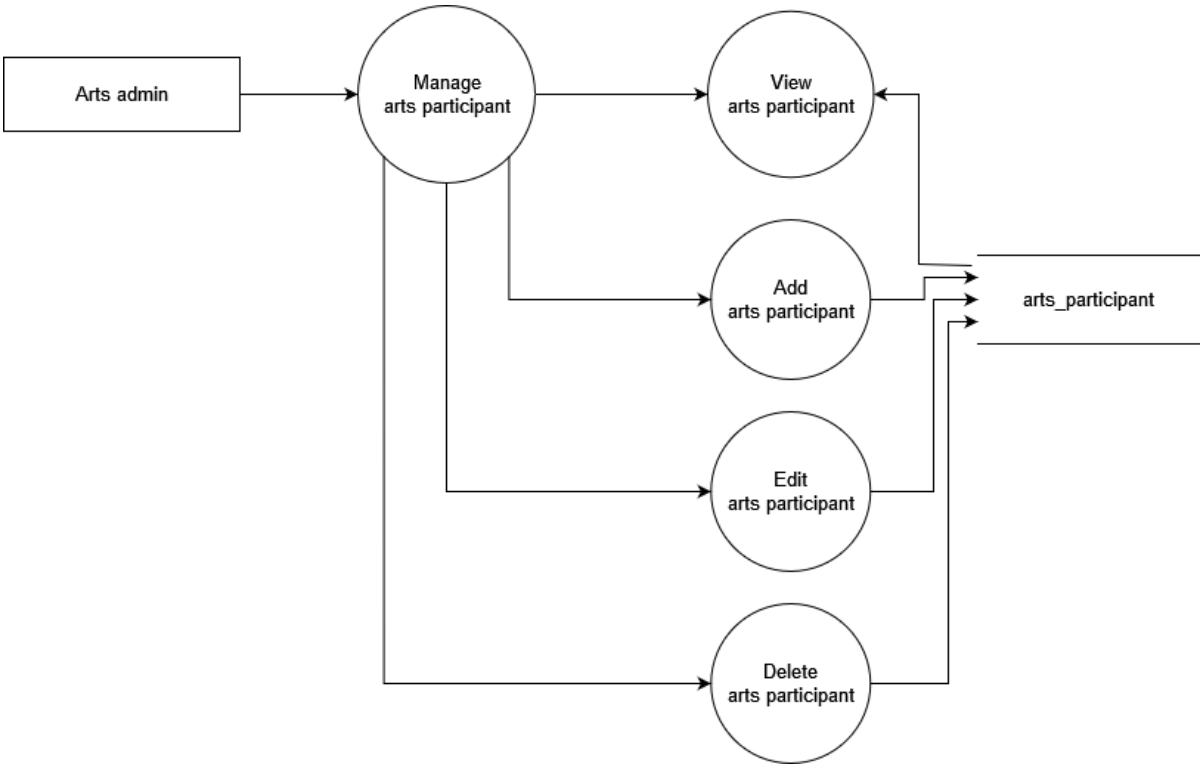


Fig:4.3.1.21

DFD Level 2.2.3(Arts admin manage arts event details)

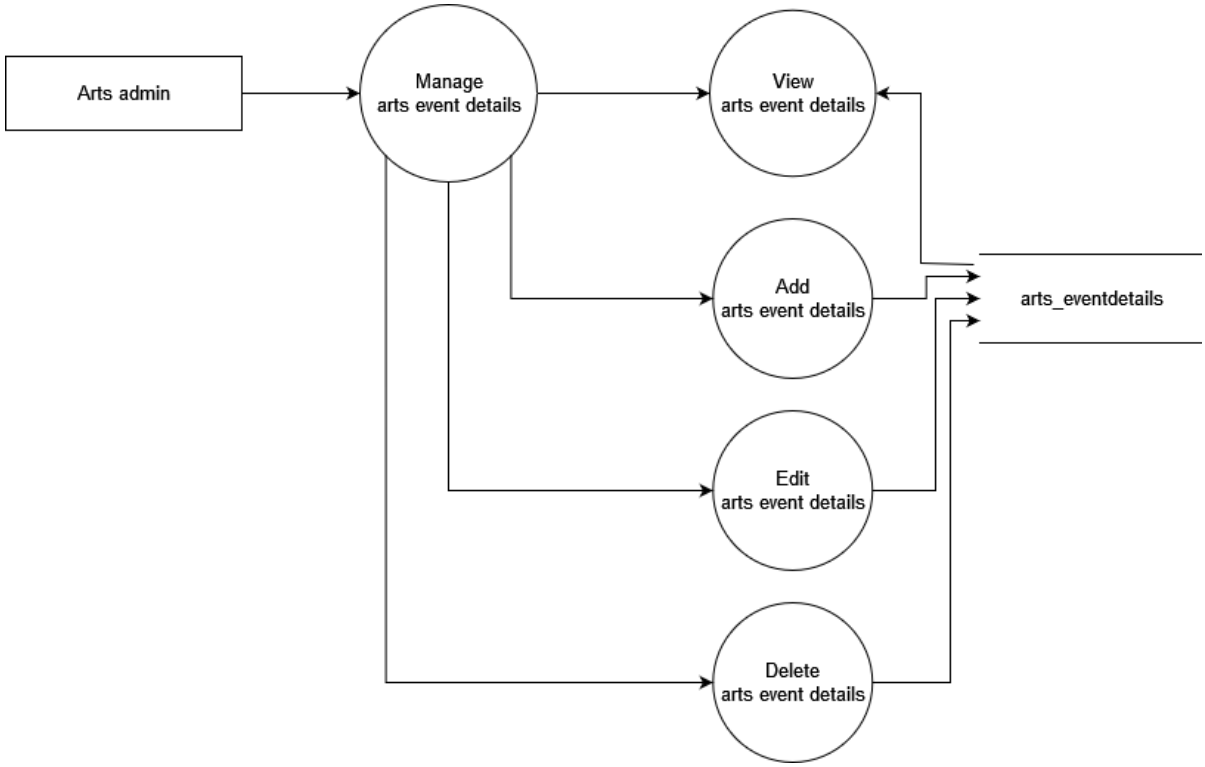


Fig:4.3.1.22

DFD Level 2.2.4(Arts admin manage arts houses)

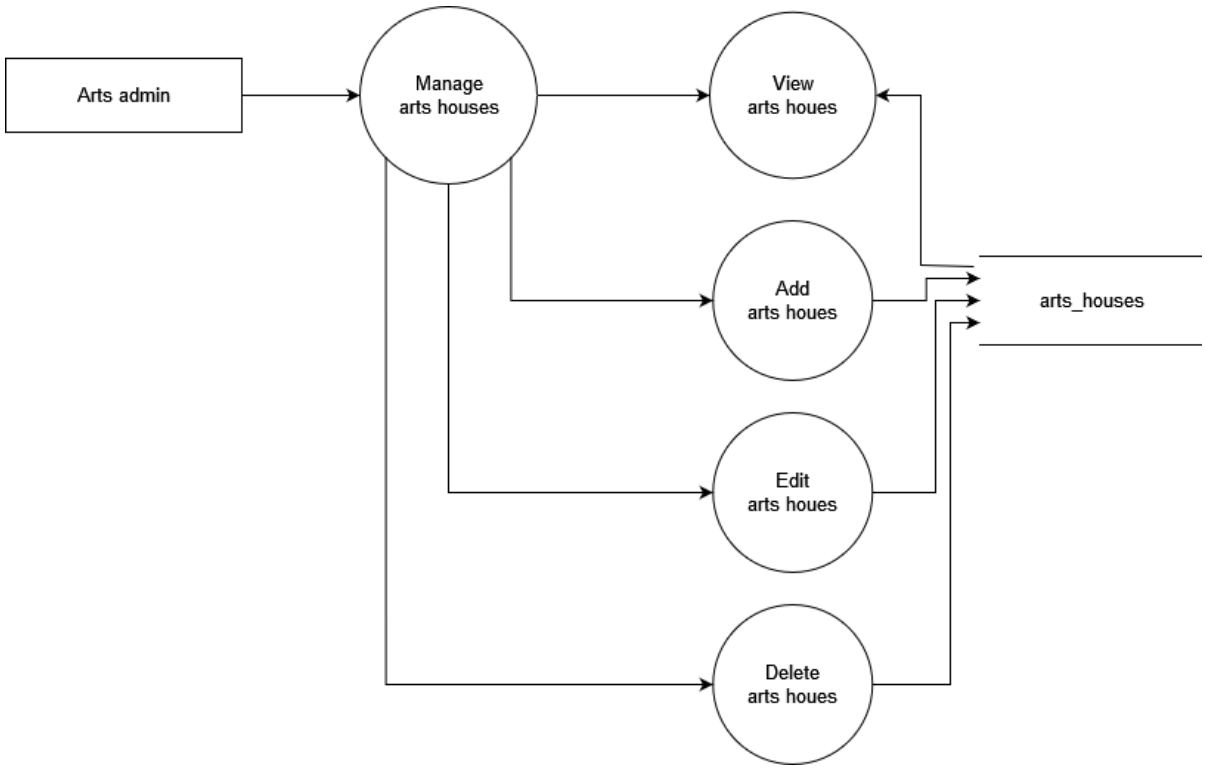


Fig:4.3.1.23

DFD Level 2.2.5(Arts admin manage arts gallery)

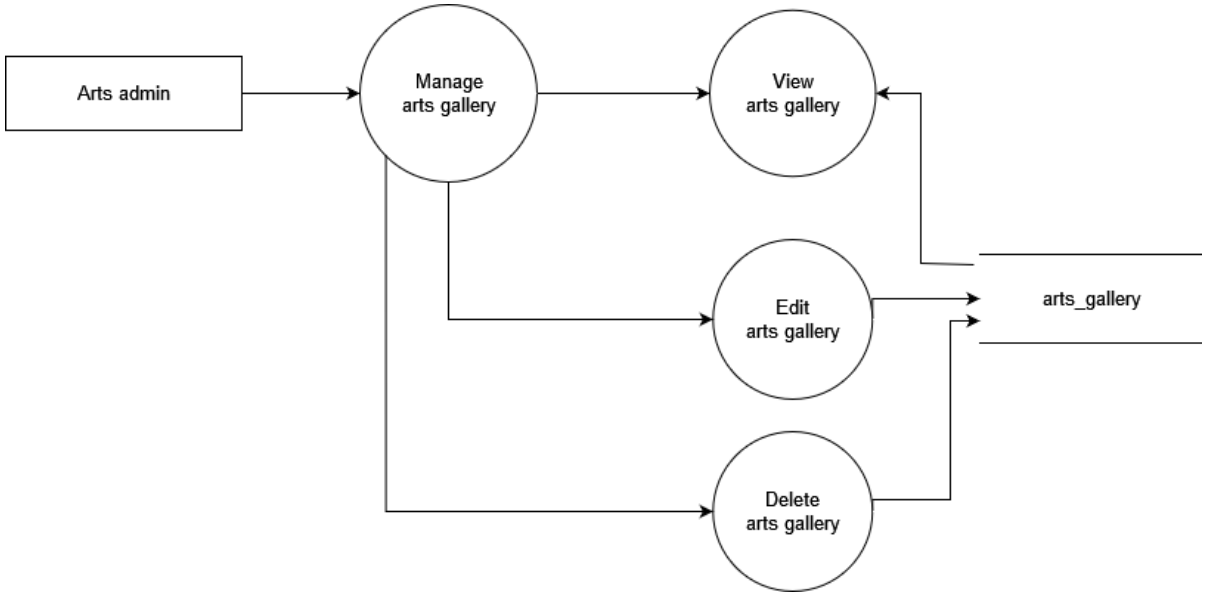


Fig:4.3.1.24

DFD Level 2.3.1(Sports admin manage sports events)

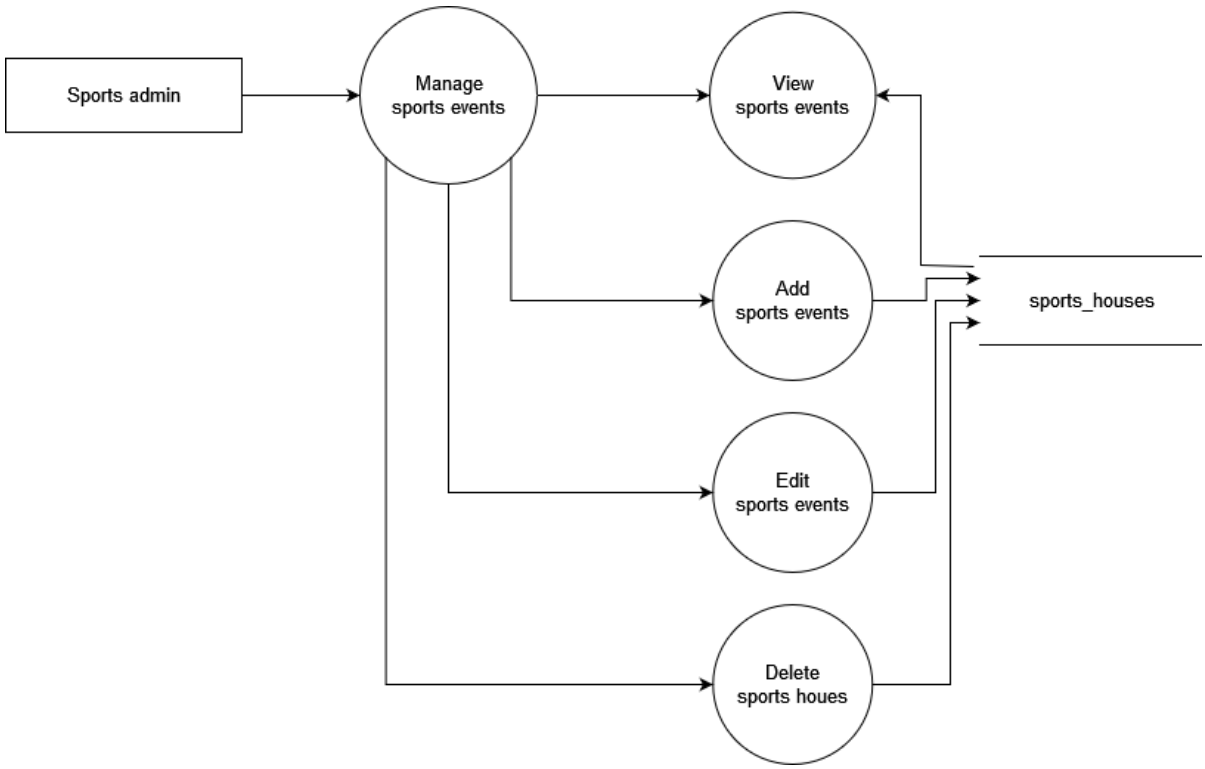


Fig:4.3.1.25

DFD Level 2.3.2(Sports admin manage sports participants)

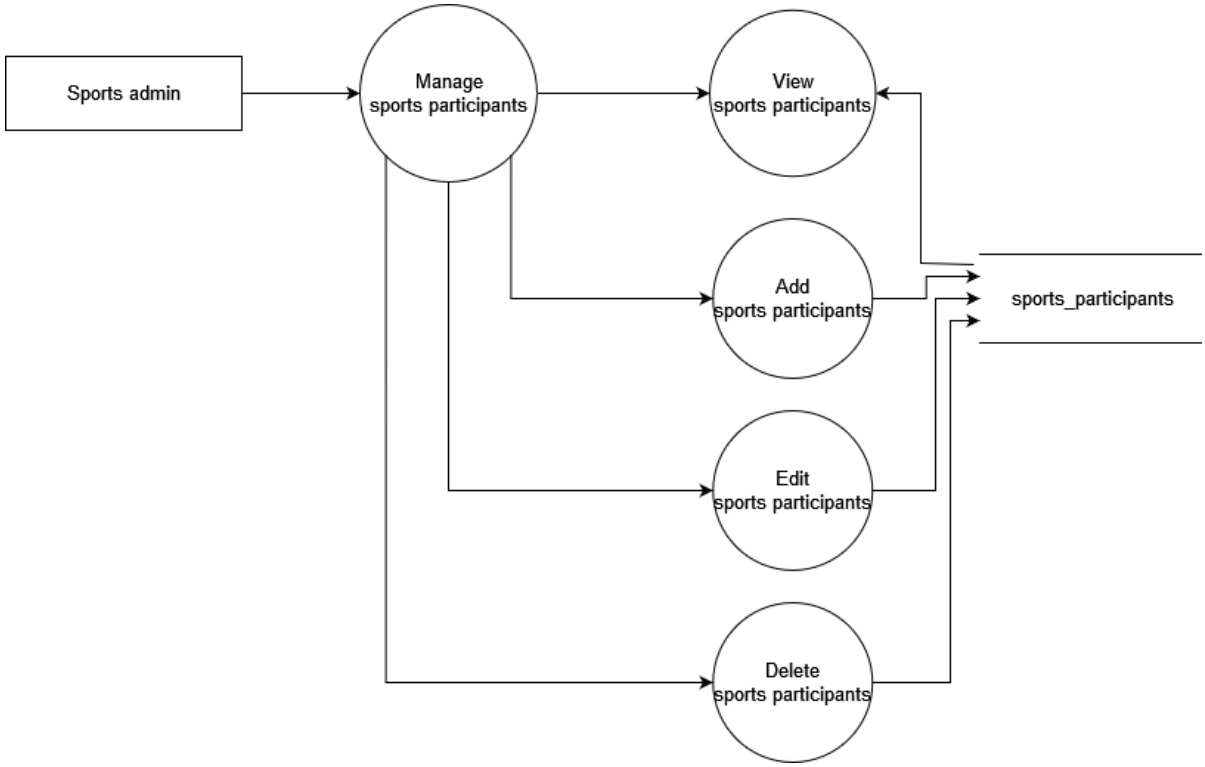


Fig:4.3.1.26

DFD Level 2.3.3(Sports admin manage sports event details)

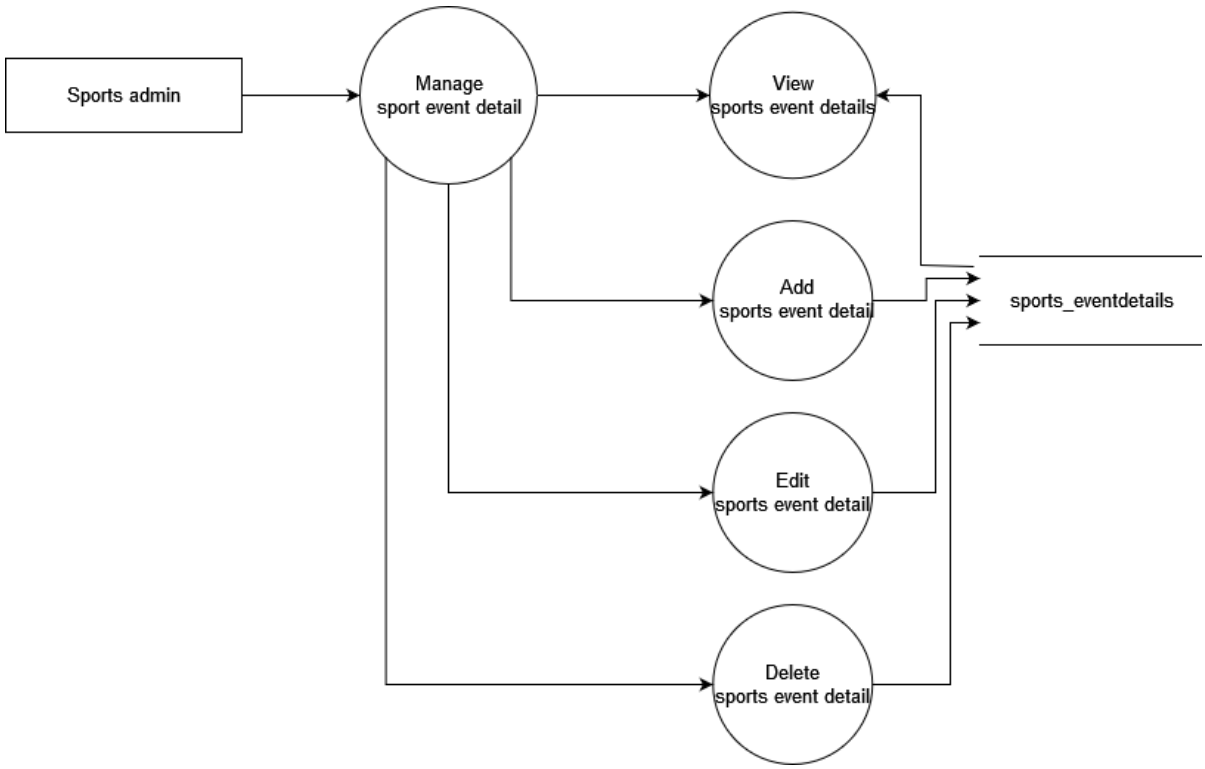


Fig:4.3.1.27

DFD Level 2.3.4(Sports admin manage sports houses)

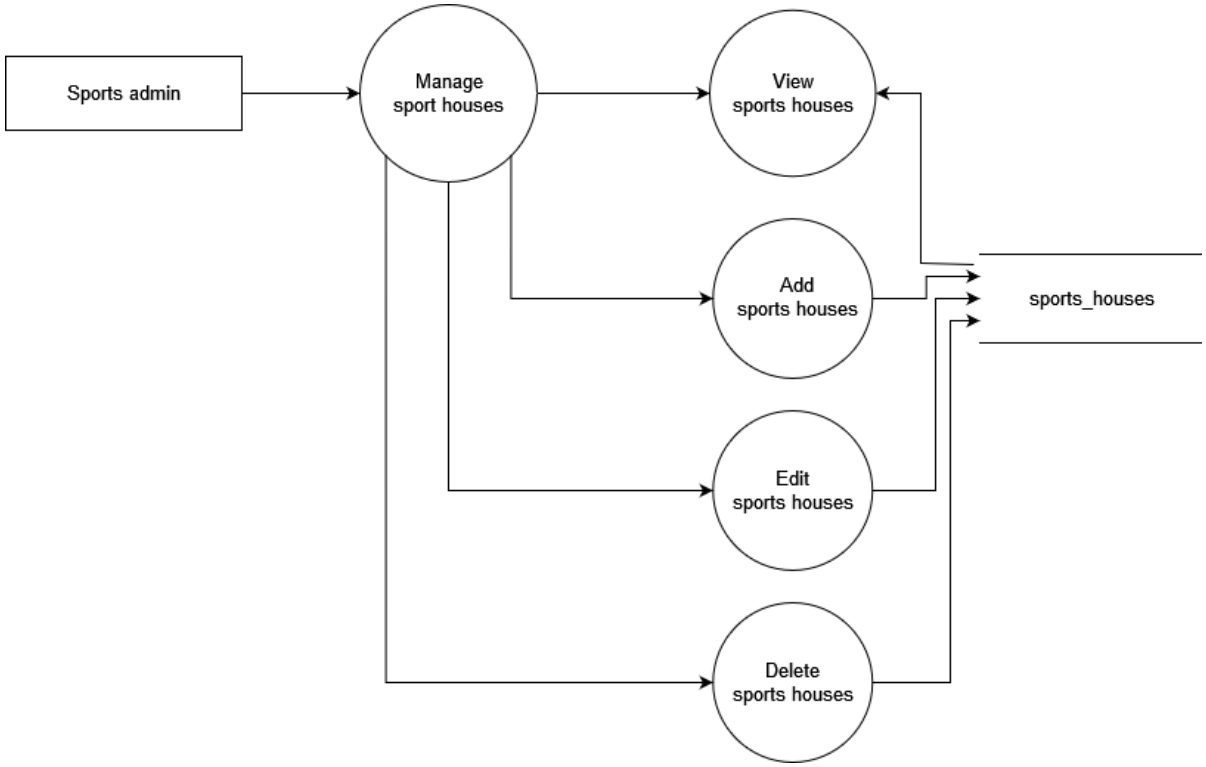


Fig:4.3.1.28

DFD Level 2.3.5(Sports admin manage sports gallery)

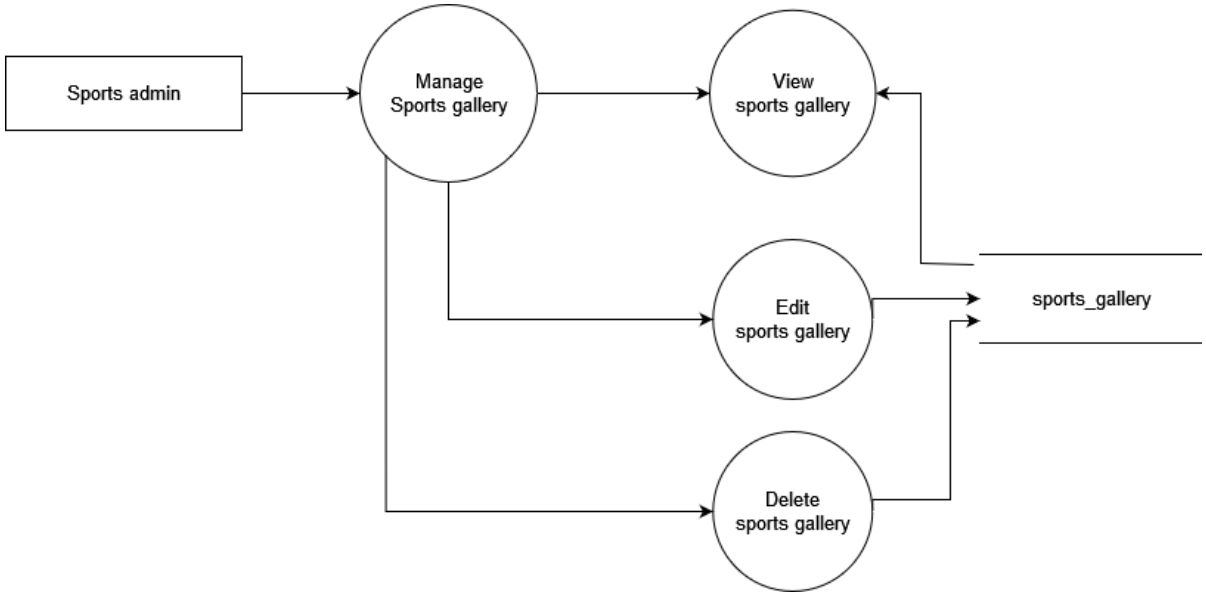


Fig:4.3.1.29

DFD Level 2.4.1(Media admin manage arts gallery)

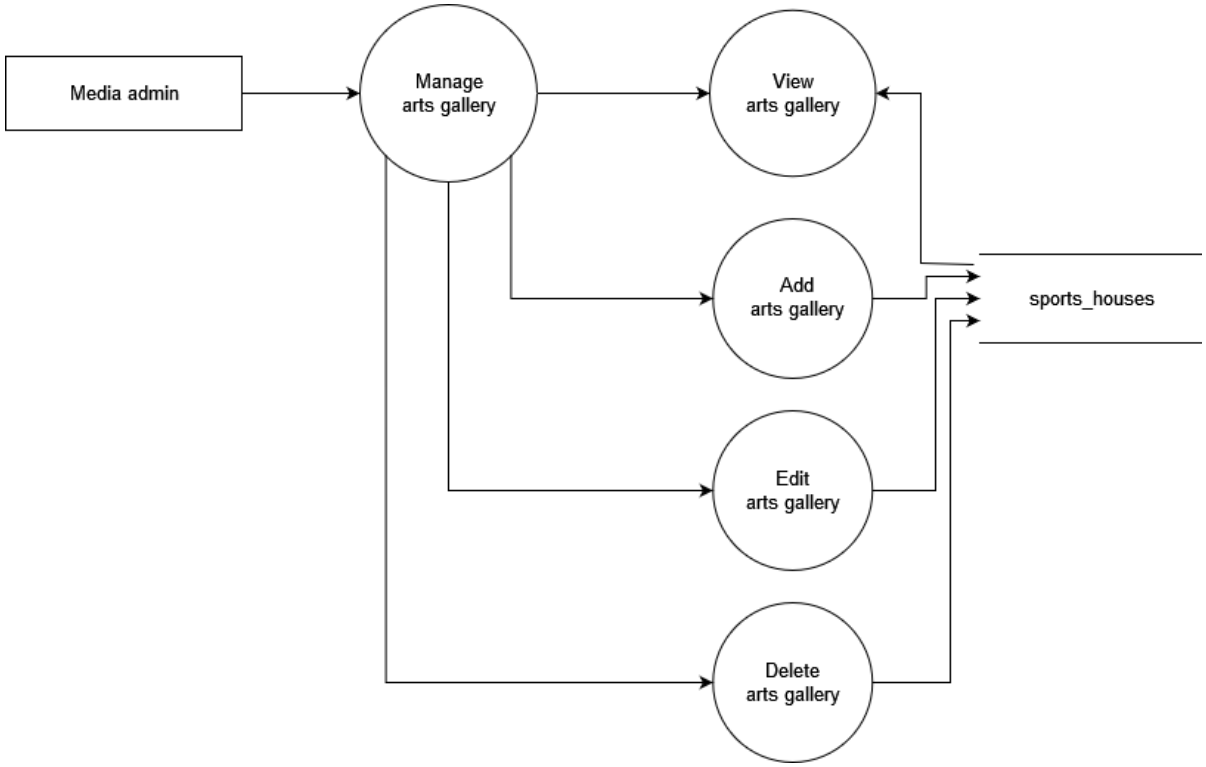


Fig:4.3.1.30

DFD Level 2.4.2(Media admin manage sports gallery)

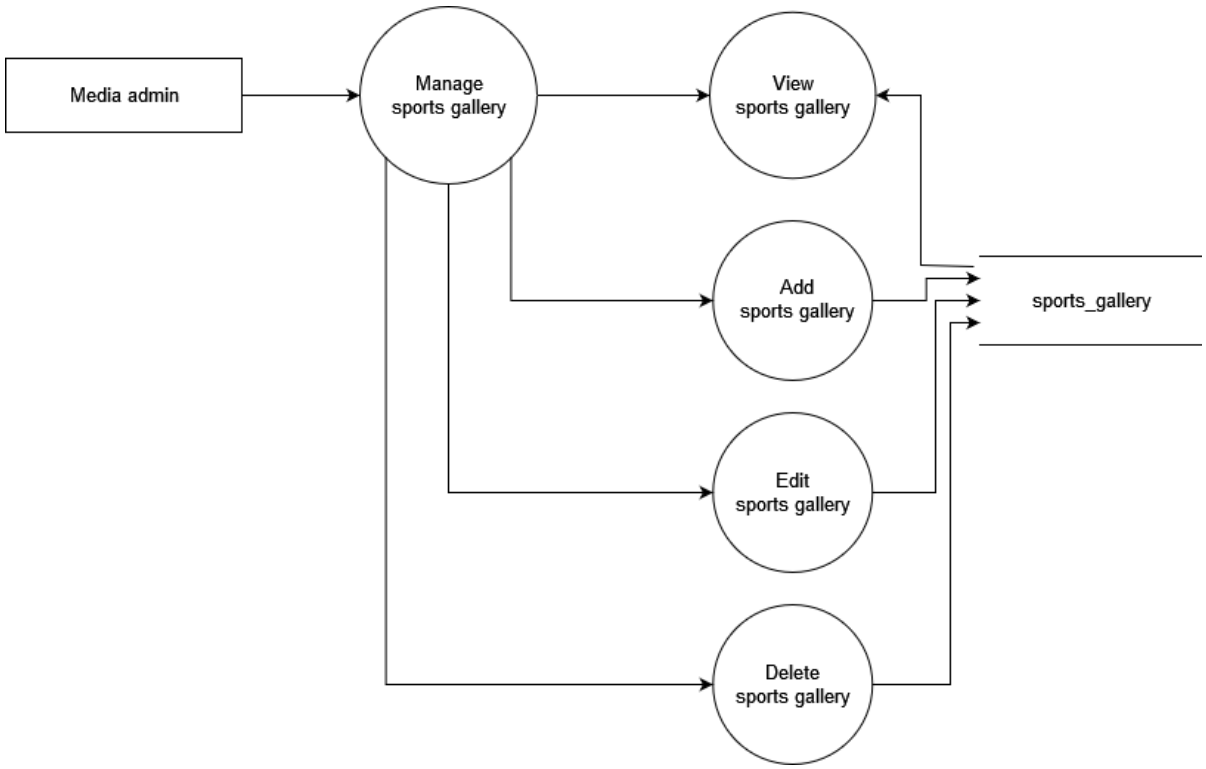
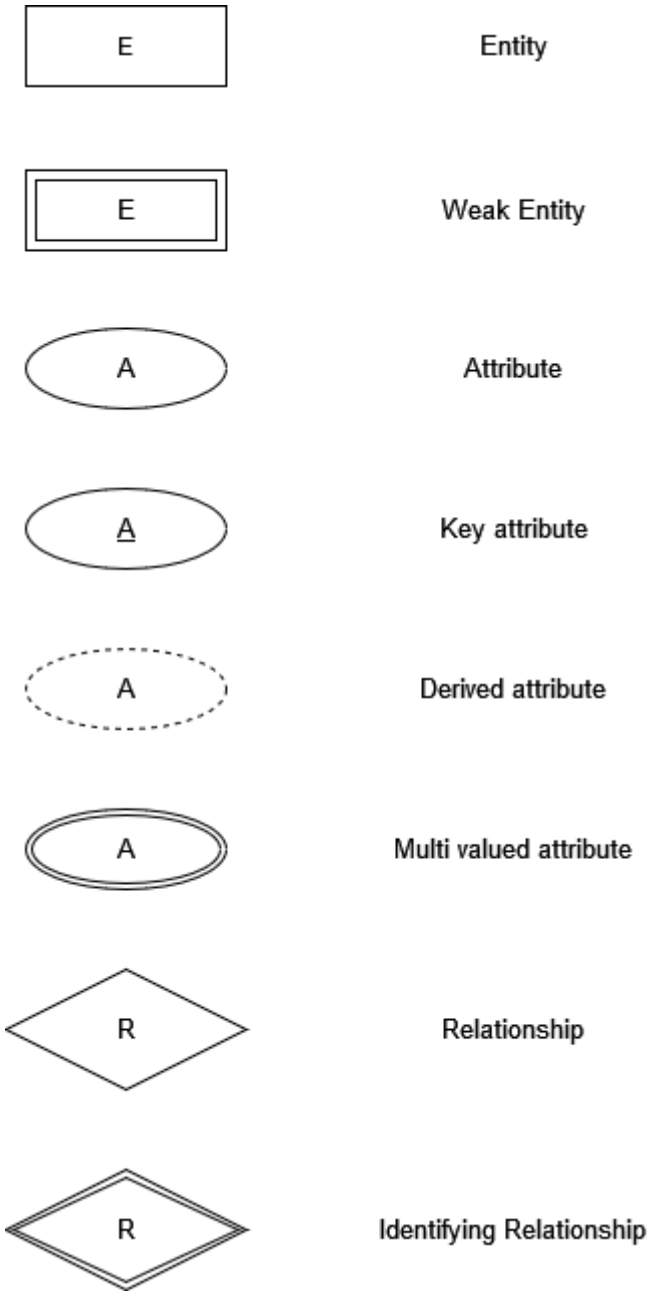


Fig:4.3.1.31

4.3.2 Entity-Relationship Diagram (ER diagram)

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how "entities" relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes

Entity Relationship Diagram Notation:



ER diagrams

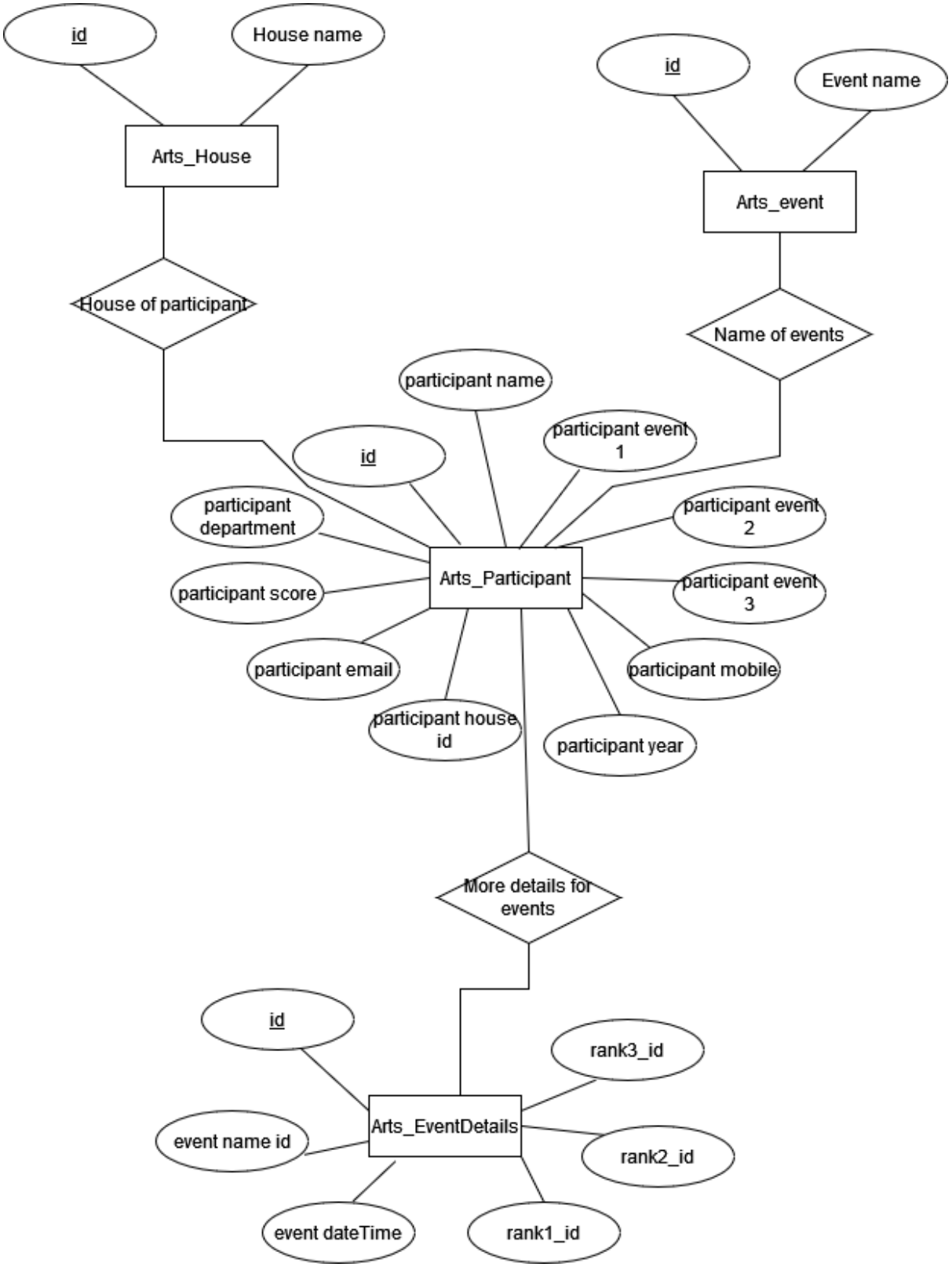


Fig:4.3.2.1

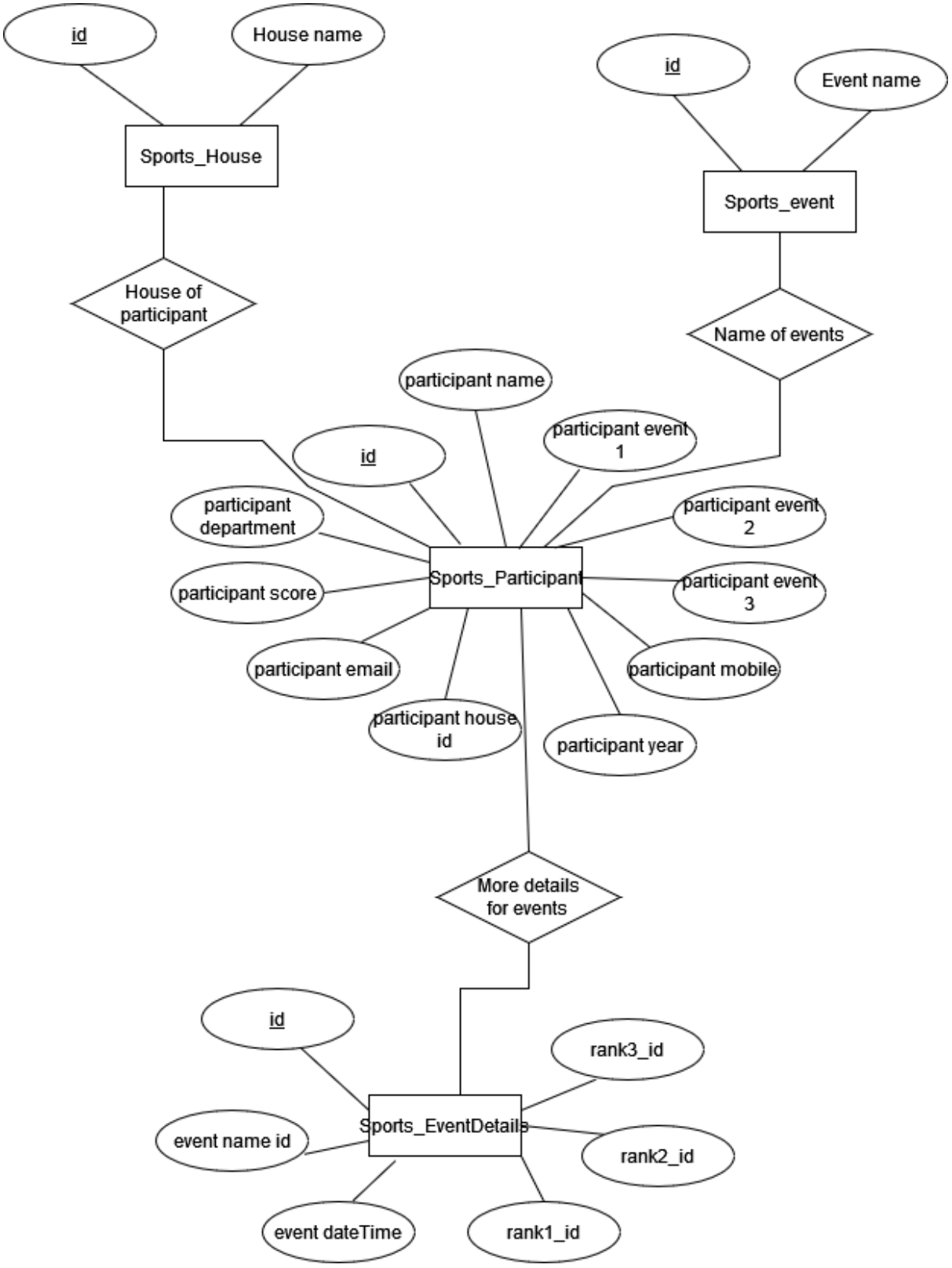

id
House name
Sports_House
House of participant
id
Event name
Sports_event
Name of events
participant name
id
participant event 1
participant department
participant event 2
Sports_Participant
participant score
participant event 3
participant email
participant mobile
participant house id
participant year
More details for events
id
rank3_id
Sports_EventDetails
event name id
rank2_id
event dateTime
rank1_id

Fig:4.3.2.2

CHAPTER 5

IMPLEMENTATION

The implementation is the important phase. It involves user training, system testing and successful running of the developed proposed system. The user tests the developed systems and changes are made according to their needs.

The implementation phase of software development involves translating of design specification into source code, and debugging, documenting and unit testing the source code.

Implementation is the process of converting a new or a revised system design in to an operational one. Conversion means changing from one system to another. The objective is to put the system into operation while holding costs, risks and personnel irritation to minimum. It involves

- Creating computer compatible files.
- Training operational staff.
- Installing terminals and hardware.

Arts and Sports Management System is a web-based application system, so it requires some space in the server. To implement the new system, it has to be deployed in the local host. It is also possible to run the system in the local network. First up all build the website, then publish website. After publishing the project, the files will be integrated into a package. These files and database will be published into the online server. Then it is possible to access the functionalities just by using the URL. Only a web browser is required to perform the operation. The project is developed in a framework, but the client system doesn't require any framework to be installed. This method also offers the greatest security.

For the complete implementation of the system, we should introduce this project to the public. Then the super admin can login to the system by entering the username and password on the admin portal. Then he/she can manage the system.

5.1 SYSTEM DEVELOPMENT

The development phase of the software design consists of different task to be done sequentially for obtaining the desired results. The different phases are:

- Creating the database:

A database is created, in which all the tables are defined which required to do the different operations such as storage and retrieval of information. Database is designed in such a way it can handle the different database queries. Users and admin can retrieve required details from the system on clicking on the links and buttons.

- Creating graphical user interface:

Graphical user interface is created in Visual studio code for a user-friendly interface. It is intended for two purposes. First is to create a user-friendly interface for the software. Having a good user interface make it easier for the user to use and understand the different functionalities of the website. Secondly, the user interface hides the end users from the complexities in the working of the software. So, the user is made unaware of how a task is performed when he chooses to perform it.

- Creating system environment:

For the intended project to work on, we need to implement its required hardware and software requirements. This system is build using Visual studio code and based on Windows Operating System. Memory and Hard disk should confirm according to hardware mentioned.

CHAPTER 6

SYSTEM TESTING

Testing is the process of examining the application to compare the actual behaviour with that of the excepted behaviour. The major goal of software testing is to demonstrate that faults are not present. In order to achieve this goal, the tester executes the program with the intent offending errors. Though testing cannot show absence of errors but by not showing their presence it is considered that these are not present. The following method of testing are carried out to assure the correctness and reliability of Arts and Sports Management System.

6.1 Smoke Testing

Smoke testing is the most cost-effective method for identifying and fixing defects in application. It is a preliminary to further testing, intended to reveal simple failures severe enough to reject a prospective application release. In this case the smoke is metaphorical. A subset of test cases that cover the most important functionality of a component or system are selected and run, to ascertain if the most crucial functions of a program work correctly. The purpose is to determine whether the application is so badly broken that further testing is unnecessary. Smoke tests can either be performed manually or using an automated tool. In Arts and Sports Management System smoke tests are performed by manually. Smoke tests can be broadly categorized as functional tests or unit tests.

6.2 Functional testing

Functional tests exercise the complete program with various inputs. Functional tests may be a scripted series of program inputs, possibly even with an automated mechanism for controlling mouse movements. It ensures that the program physically works the way it was intended and all required menu options are present. It also ensures that the program conforms to the industry standards relevant to that environment.

6.3 Unit testing

Unit testing is carried out in parallel to coding. Functionality of each of the individual modules we have developed is tested using test cases. Unit testing is done according to the test plans prepared for each of the module. Test plan prepared have covered all functional areas, actual input and expected outputs. Modules are tested using different test cases. Actual outputs are compared with expected outputs. The results are also recorded.

6.4 Integration Testing

The major concerns of integration testing are developing incremental strategies that will the complexity of entire actions among components as they are added to the system. Though each program works individually; they should work after linking them together. This is also referred to as interfacing. Data may be lost across interface and one module can have adverse effect on another. Integration testing is a systematic technique for constructing program structure while at the same time, conducting test to uncover errors associated with the interface. In the testing, the programs are constructed and tested in small segments.

6.5 Performance Testing

Performance Testing is designed to test the runtime performance of the integrated system. CDAS can process large amount of data with minimum memory and time is tested here.

6.6 Test Plan

A test plan is a systematic approach to testing a system. The plan typically contains a detailed understanding of what the eventual workflow will be. Normally, testing of any large systems will be in two parts.

- The functional verification and validation against the requirement specification.
- Performance evaluation against the indicated requirements.

Testing activity is involved right from the beginning of the project. At the very first stage of testing, the goals and objectives are set. This simplifies the limits or borders of testing process. Before testing, the tester should plan what kind of data he is giving for test. Give data inputs as functional, boundary, stress, performance, usability values etc.

6.7 Characteristics of a Good Test

- Tests are likely to catch bugs
- No redundancy
- Not too simple or too complex

6.8 Test Cases

Specific set of steps and data along with expected results for a particular test objective. A test case should only test one limited subset of a feature or functionality. Test case documents for each functionality/testing area will be written, reviewed and maintained separately in Excel Sheet. In system testing, test data should cover the possible values of each parameter based on the requirements. Since testing every value is impractical, a few values should be chosen from each equivalence class. An equivalence class is a set of values that should all be treated the same. Ideally, test cases that check error conditions are written separately from the functional test cases and should have steps to verify the error messages and logs. Realistically, if error test cases are not yet written, it is OK for testers to check for error conditions when performing normal functional test cases. It should be clear which test data, if any, is expected to trigger errors.

Example

Test cases	**Test conditions**	**Test data**	**Execution setup**	**Expected result**	**Pass/Fail**
1	Admin login	Username and password	Click on login button	Admin page displayed	Pass
2	Event Registration	Name and other details	Click on register button	Redirects to home page	Pass

Table 6.8.1

CHAPTER 7

CONCLUSION

Arts and Sports Management System, the web app is very essential for students' educational growth in current digital scenario. It allows users to register for events, know the schedule and time of the events, view their own scores, their house's scores and also see the colourful and different moments of these events as photos. The students can register for an event at anytime from anywhere. By using this system student can save a lot of time an effort. The student can easily get the information from anywhere. The program for caring out the various activities have been run and tested successfully to ensure that the software development will meet the needs satisfactory. As a whole we could conclude that this website has met all its objectives of being user friendly.

CHAPTER 8

FUTURE ENHANCEMENT

We have some future visions for the enhancement of the web-app. Now this system is only available as a bowser app. In future we will be developing the system for android and apple application. We will be implementing more features such as e-certificate generator built-in in the near future.

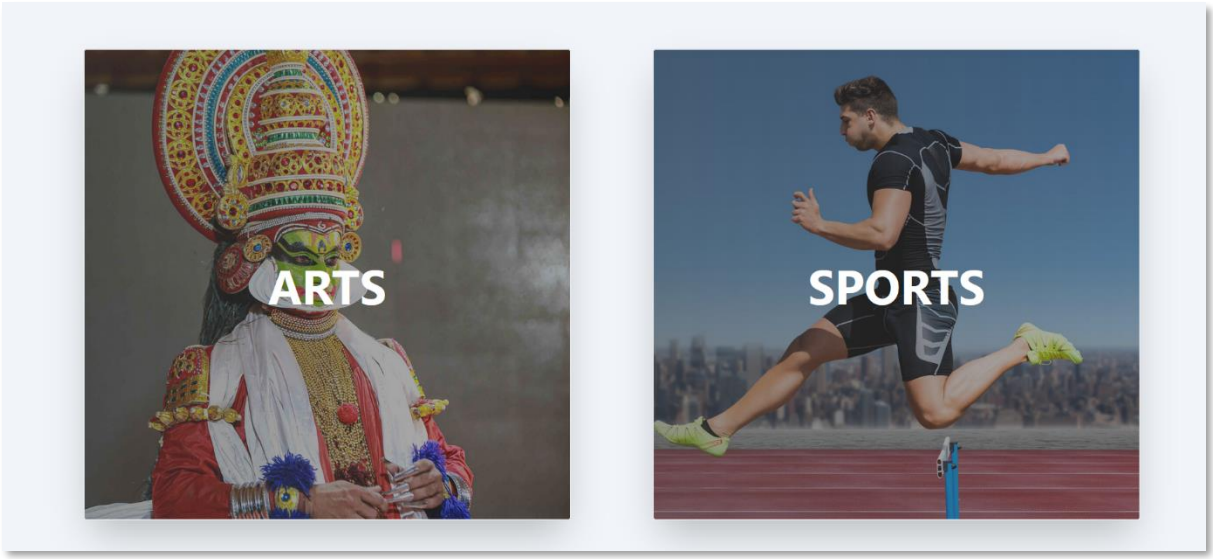
BIBLIOGRPHY

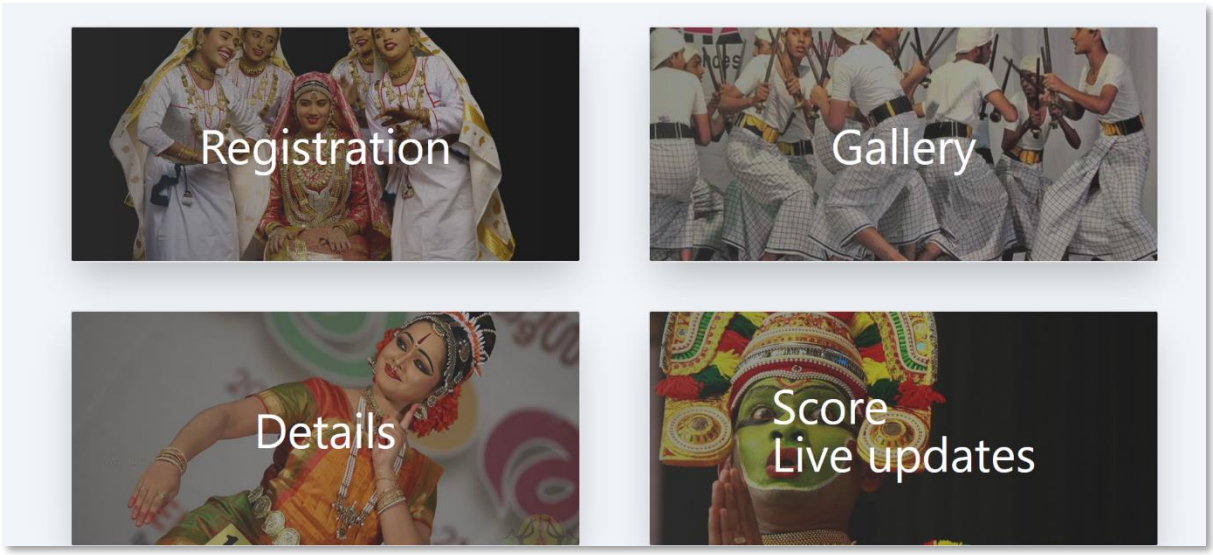
Websites

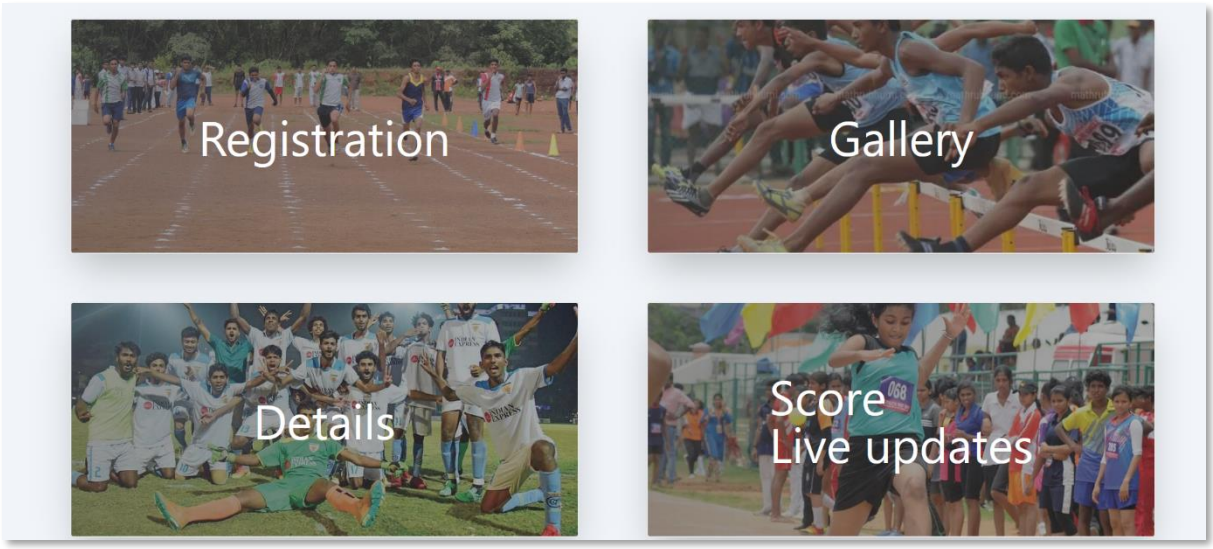
- Stack Overflow
- W3Schools
- Django project
- Tailwind CSS
- Real Python
- Medium
- YouTube

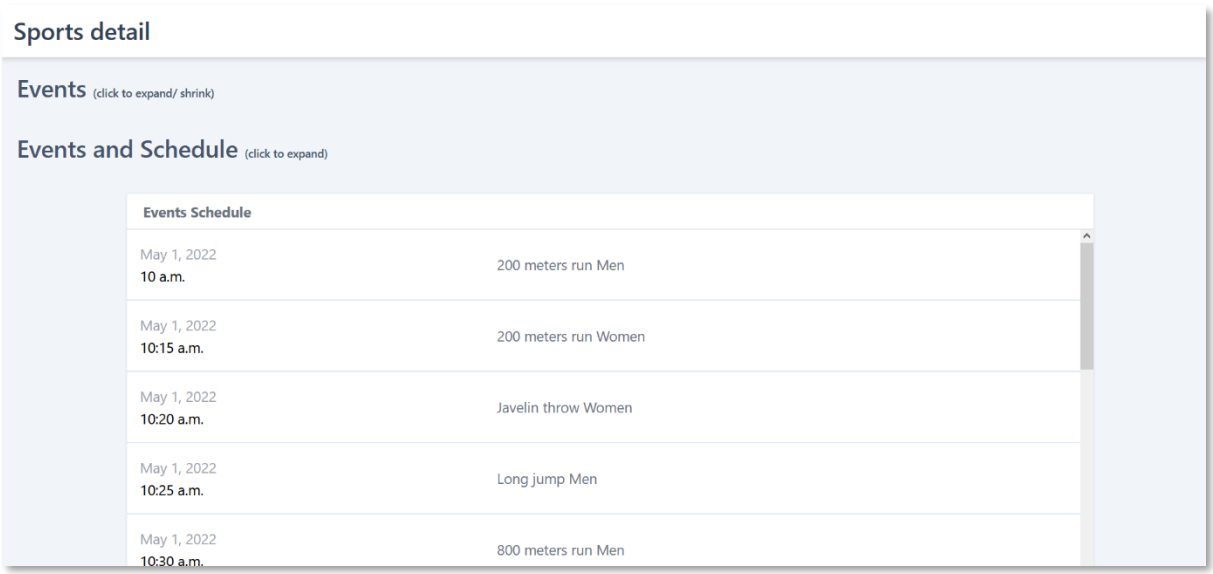
APPENDIX

Screenshots (Desktop view)

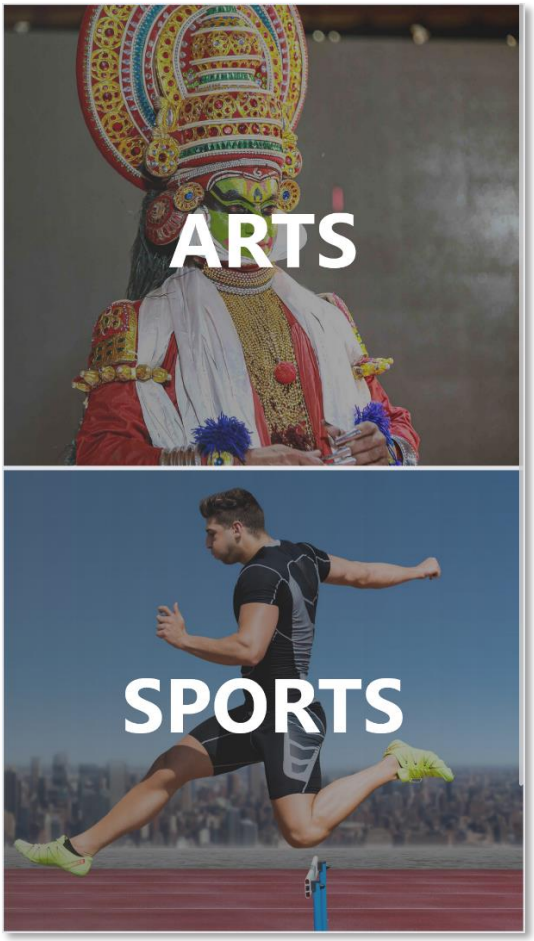


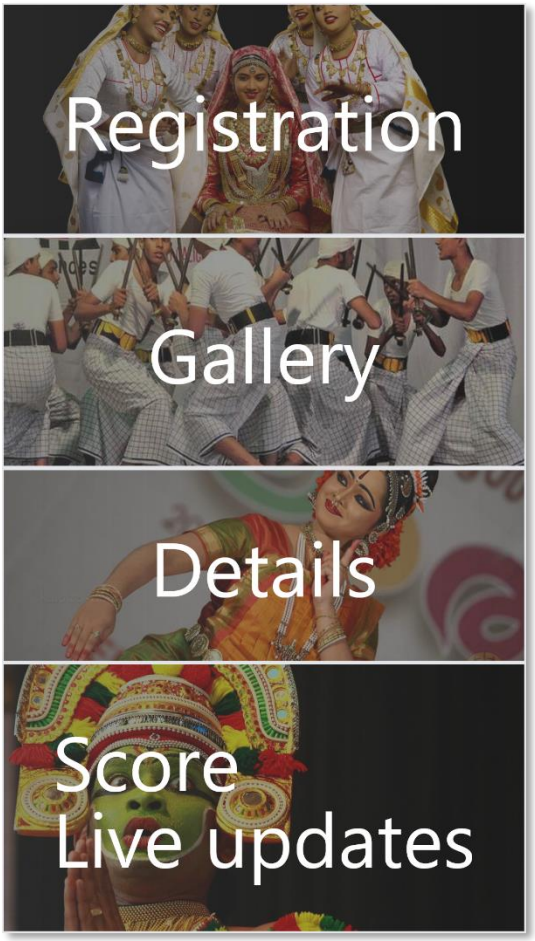



Registration
Gallery
Details
Score
Live updates


Sports detail
Events (click to expand/ shrink)
Events and Schedule (click to expand)
Events Schedule
May 1, 2022
10 a.m.
200 meters run Men
May 1, 2022
10:15 a.m.
200 meters run Women
May 1, 2022
10:20 a.m.
Javelin throw Women
May 1, 2022
10:25 a.m.
Long jump Men
May 1, 2022
10:30 a.m.
800 meters run Men

Screenshots (Mobile view)





Sports detail

Events (click to expand/ shrink)

Events and Schedule (click to expand)

Events Schedule	
May 1, 2022 10 a.m.	200 meters run Men
May 1, 2022 10:15 a.m.	200 meters run Women
May 1, 2022 10:20 a.m.	Javelin throw Women
May 1, 2022	Long jump Men