

CLUB MANAGEMENT SYSTEM

A dissertation submitted in partial fulfillment of the
requirements for the award of degree of

B.Sc. COMPUTER SCIENCE of UNIVERSITY OF CALICUT



Submitted by

MUHAMMED ABDUL BASITH (RCAVSCS006)

JUNAID MP (RCAVSCS003)

MUHAMMED ANAS. K (RCAVSCS007)

MUFEEHDA P (RCAVSCS001)

Under the Guidance of

Mrs. RASEENA KM

Dept of B.Sc. Computer Science

RCSH MUNDAMPARAMBA



**Department of B.Sc. Computer Science
REGIONAL COLLEGE OF SCIENCE AND HUMANITIES
KIZHISSERI, MALAPPURAM
2021-2024**



Department of B.Sc. Computer Science
REGIONAL COLLEGE OF SCIENCE AND HUMANITIES

Kizhisseri, Puliyakode (P.O), Malappuram (Dist.)

CERTIFICATE

This is to certify that **Mr. MUHAMMED ABDUL BASITH** bearing reg no: **RCAVSCS006**, **Mr. JUNAID MP** bearing reg no: **RCAVSCS003**, **Mr. MUHAMMED ANAS. K** bearing reg no: **RCAVSCS007**, **Ms. MUFEEDHA P** bearing reg no: **RCAVSCS001** has completed their final semester project work entitled “**CLUB MANAGEMENT SYSTEM**” as a partial fulfillment for the award of B.Sc. Computer Science Degree, during the academic year 2023-2024 under our joint supervision. It is certified that all correction/suggestion indicated for internal assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said degree.

Head of the Department

Mr. ABDUL LATHEEF AP

Signature of Internal Guide

Mrs. RASEENA KM

(Office Seal)

EXTERNAL VIVA

Name of the Examiners

- 1.
- 2.

DECLARATION

We **MUHAMMED ABDUL BASITH, JUNAID MP, MUHAMMED ANAS. K, MUFEEDHA P**, do hereby declare that this project entitled **CLUB MANAGEMENT SYSTEM** has been prepared by us in fulfilment of the requirements for the B.Sc COMPUTER SCIENCE degree by University of Calicut. We also declare that this project report contains no material previously published or written by another person which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning expect where due acknowledgement has been made in the text.

MUHAMMED ANAS. K (RCAVSCS007)

MUHAMMED ABDUL BASITH (RCAVSCS006)

JUNAID MP (RCAVSCS003)

MUFEEDHA P (RCAVSCS001)

Place: MUNDAMPARAMBA

Date:

ACKNOWLEDGEMENT

We would like to start this acknowledgment note by thanking to the God for the constant love and grace he has showered upon us since lifetime. We thank Mr. Abdul Latheef AP, Head, Dept. of Computer Science and Mrs. Raseena KM our internal guide for their effective guidance, timely suggestions and encouragements in completing this work.

Heartfelt thanks to all teachers and staff of Computer Science Department who have advised, helped and given valuable suggestions to bring our project into reality. The successful completion of the system wouldn't be possible without our parents and friends. Let us end this note of gratitude by thanking all of them who has been there for using the course of study and work of the system.

JUNAID MP(RCAVSCS003)

MUHAMMED ABDUL BASITH (RCAVSCS006)

MUHAMMED ANAS. K (RCAVSCS007)

MUFEEDHA P(RCAVSCS001)

ABSTRACT

We propose to build a software system that can efficiently handle and manage various activities of a club. All these activities will be happening under the supervision of the administrator. The administrator account controls everything.

Club management project is designed with the motive of managing a club. This software system consists of various online booking and management functionalities needed by a club. It provides various options like ground bookings, member registrations, notice posting, member registration, batch registration and more. This software system assures efficient management and maintains the functioning on a club.

CONTENTS

CHAPTERS	Page No
1. INTRODUCTION	01
1.1 About the Project	02
1.2 Methodology	02
1.3 Timeline	02
1.4 Model Used	02
2. SYSTEM SPECIFICATION	03
2.1 System Configuration	04
2.1.1 Hardware specification	04
2.1.2 Software Specification	04
2.1.3 Tools	04
2.2 Requirement Specification	05
2.2.1 Functional Requirements	05
2.2.2 Non-Functional Requirements	05
3. SYSTEM ANALYSIS	06
3.1 Feasibility Study	07
3.1.1 Technical Feasibility	07
3.1.2 Operational Feasibility	08
3.1.3 Economical Feasibility	08
3.1.4 Behavioral Feasibility	09
3.2 Existing System	09
3.3 Proposed System	09
3.4 Purpose of the System	10
4. SYSTEM DESIGN	11
4.1 Architectural Design	12
4.2 Entity Relationship Diagram	12
4.2.1 Er Diagram	14
4.3 Dataflow Diagram	15
4.3.1 DFD Level 0	16
4.3.2 DFD Level 1	17
4.3.3 DFD Level 2	18
4.4 Input Design	19
4.5 Output Design	19

5. DATABASE DETAILS	20
5.1 Database Details	21
5.2 Normalization	21
5.2.1 First Normal Form	21
5.2.2 Second Normal Form	22
5.2.3 Third Normal Form	22
5.3 Database Tables	23
5.3.1 User profile	23
5.3.2 Club member	23
5.3.3 Member Account	23
5.3.4 Sports Item	24
5.3.5 Club Profile	24
5.3.6 Club Gallery	24
5.3.7 Batch	24
5.3.8 Batch Time	25
5.3.9 Batch Register	25
5.3.10 Purpose	25
5.3.11 Ground Booking	25
5.3.12 NewsTb	26
6. OVERVIEW OF THE SOFTWARE	27
6.1 Front End Software Visual Studio Code	28
6.1.1 Advantages of Visual Studio	29
6.2 Back-End Software	30
6.2.1 PyCharm	30
6.3 Database	30
6.3.1 Sqlite-3	30
7. SYSTEM IMPLEMENTATION	32
7.1 System Implementation	33
7.2 Implementation Procedures	33
8. SYSTEM TESTING	34
8.1 Implementation	35
8.2 Testing	35
8.2.1 Types of Testing	36
8.2.2 Testing Strategies	36
8.2.3 Verification and Validation	38
8.2.4 System Security	38
8.2.5 System Maintenance	38

9. FUTURE ENHANCEMENT AND CONCLUSION	40
9.1 Future Enhancement	41
9.2 Conclusion	41
10. SCREENSHOTS	42
10.1 Web Screenshots	43
10.1.1 Login	43
10.1.2 Home Page	43
10.1.3 Profile	44
10.1.4 View Members	44
10.1.5 Sports Item	45
10.1.6 Gallery	45
10.1.7 Batch	46
10.1.8 News	46
10.1.9 Purpose	47
10.1.10 Ground Booking	47
10.1.11 Batch Booking	48
10.2 App Screenshots	49
10.2.1 Login	49
10.2.2 Home Page	49
10.1.3 Profile	49
10.2.4 View Gallery	50
10.2.5 News	50
10.2.6 Ground Booking	50
10.2.7 Batch Booking	51
10.2.8 My Bookings	51
10.2.9 My Batches	51
10.2.10 About us	52
11. BIBLIOGRAPHY	53
11.1 Bibliography	54
11.2 Online Reference	54

CHAPTER 1

INTRODUCTION

1.1 ABOUT THE PROJECT

The project is a proposal for a Club Management System, a software designed to efficiently handle and manage various activities within a club under the supervision of an administrator. It includes features such as admin and user modules, online member registration, ground booking, and automation of club activities. The software and hardware requirements are also outlined, along with the advantages of implementing the system. Overall, the project aims to streamline club operations and enhance efficiency through the implementation of various online booking and management functionalities.

1.2 METHODOLOGY

The major steps carried out through the project are as follows:

1. Detailed study of overall system
2. Identifying the problem
3. Collection of requirements
4. Design the system
5. Program development
6. Program testing and modification
7. Preparation of report

1.3 TIMELINE

The project was started in December 2023. Initial planning and requirement engineering were completed by the end of the month. The coding had started in mid-January, and the testing went on parallel. The core features got finished in last March 2023.

1.4 MODEL USED

Software Development Process Model represent the order in which the activities of software will be undertaken. It describes the sequence of in which the phases of the software lifecycle will be performed. In this project development I choose iterative model.

CHAPTER 2

SYSTEM SPECIFICATION

2.1 SYSTEM CONFIGURATION

A main purpose of software requirement specification is the clear definition and specification of functionality of the software product. It allows the developer to be carried out, performance level to be obtained and corresponding interface be established.

2.1.1 Hardware specification

Hardware requirements for the installation and smooth functioning of this project could be configured based on the requirements needed by the component of the operating environment that works as front-end system, here we suggest minimum configuration for the hardware components.

- Processor: Intel Pentium IV or above, INTEL DUAL CORE or above
- Monitor: Min. 14
- RAM: 1 GB
- Hard Disk: 5 GB
- Keyboard: Standard 104 Keys
- Mouse: Serial mouse

2.1.2 Software Specification

This is the one of the system requirements and software requirements for the installation and smooth functioning of this project. Here we are suggesting minimum configuration for the software components.

- IDE: JETBRAINS PYCHARM, VS CODE
- OS: Windows 10 or above / Linux New Versions
- Languages: Python, Dart
- Other Tools: SQL lite 3,
- Android OS: Android M or above
- Android Ram Capacity: 1Gb or above.
- WEB BROWSER: GOOGLE CHROME

2.1.3 Tools

1. Code editor: PyCharm, Vs code
2. Communication with other members: Mailing list.
3. Documentation: Ms word, Draw.io
4. Database server: SQL lite 3

2.2 REQUIREMENT SPECIFICATION

A system requirement specification is a complete description of the behavior of the system to be developed. It includes a set of use cases that describe all of the interaction that the users will have with the software. In software engineering approach there are two type of system requirement specification

- Functional Requirement.
- Non-Functional Requirement.

2.2.1 Functional Requirements

It is the statement of services the system should provide and also what the system should not do. These statement services the system should provide, how the system should react to particular inputs and how system should behave in particular situation.

Functional Requirement should include:

- Description of system reports or other outputs.
- Description of operations performed by each screen.
- Description of workflows performed by the system.
- Description of data to be entered into system.

2.2.2 Non-Functional Requirements

It is the requirements that are not directly concerned with the specific functions delivered by the system.

Non-Functional Requirement should include:

- Software should be user friendly.
- It must be reliable.
- It should be maintainable and customizable without any extra effort.
- Safety.
- Backup in case of failure.
- Security.
- Portability.
- Functionality.
- Feature set, Capabilities, Generality, Security.
- Usability.
- Human factors, Aesthetics, Consistency, Documentation.
- Reliability.
- Speed, Efficiency.
- Testability, Adaptability, Maintainability, Portability.

CHAPTER 3

SYSTEM ANALYSIS

3.1 FEASIBILITY STUDY

All projects are feasible when given unlimited resources and infinite time. It is both unnecessary and prudent to evaluate the feasibility of a project at the earliest possible time. A feasibility study is not warranted for systems in which economic justification is obvious, technical risk is low, few legal problems are expected and no reasonable alternative exists. An estimate is made of whether the identified user needs may be satisfied using current software and hardware technologies. The study will decide if the proposed system will be cost effective from the business point of view and if it can be developed in the given existing budgetary constraints. The feasibility study should be relatively cheap and quick. The result should inform the decision of whether to go ahead with a more detailed analysis. Before any user request is to be accepted, it is mandatory to check whether the new system is feasible or not. The major purpose of the analysis is to see that the development is technically and operationally helpful to the organization or not. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for developed. The key considerations involved in the feasibility are:

1. Technical Feasibility
 2. Operational Feasibility
 3. Economic Feasibility
 4. Behavioral Feasibility

3.1.1 Technical Feasibility

A study of function, performance and constraints may improve the ability to create an acceptable system. Technical feasibility is frequently the most difficult to achieve at the stage of product engineering process. Considering that are normally associated with the technical feasibility include:

1. Development Risk
2. Resource Availability
3. Technology

Technical feasibility study deals with the hardware as well as software requirements. The scope was whether the work for the project is done with the current equipment and the existing software technology has to be examined in the feasibility study. The outcome was found to be positive. The proposed system "**CLUB MANAGEMENT SYSTEM**" is technically feasible because the software required are easy to deploy and handle. The necessary software platform is already there. Connection available to all the systems anyways. So, the web application is quite easy to handle.

3.1.2 Operational Feasibility

Operational feasibility is a measure of how well a proposed system solves the problem and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. To ensure success, desired operational outcomes must be imparted during design and development.

These include such design-dependent parameters such as reliability maintainability, usability supportability, disposability, sustainability, affordability and others. Therefore, operational feasibility is a critical aspect of systems engineering that needs to be an integral part of the early design phases.

The system is operationally feasible since software requirements and installation can easily handle and there is no additional effort is to be made to train and educate the user son the new way of the system. The proposed system "**CLUB MANAGEMENT SYSTEM**" requires minimum effort for development and easily operable. So, I can say that in above aspects my proposed system fit and match is operationally feasible.

3.1.3 Economical Feasibility

Justification of any capitals outlay is that it will reduce expenditure or improve the quality of service, which in turn may be expected to provide increased profits and reputation. The technique of cost benefit analysis is often used as a basis for assessing economic feasibility. Cost benefit feasibility includes both hardware and software feasibility

In this system no initial Investment is needed. The proposed system "**CLUB MANAGEMENT SYSTEM**" was developed with the available resources. Since cost input is almost nil the output of the software is always profit. Hence the system is economically feasible. The purpose of the economic feasibility assessment is to determine the positive economic. Benefits to the society that the proposed system will provide. It is very essential because the main goal of the proposed system includes quantification and identification of all the benefits expected. This assessment typically involves a cost/benefits analysis, long term cooperative income strategies and cost of resources needed for development.

3.1.4 Behavioral Feasibility

People are inherently resistant to changes and computer is known for facilitating changes. An estimate should be made of how strongly the user staffs react towards the developments of the computerized system. In the proposed system the manpower is reduced so unnecessary burden is reduced. Therefore, the system is behaviorally feasible.

3.2 EXISTING SYSTEM

In the era before digital automation, businesses relied on manual call booking systems to manage appointments, reservations, and bookings. These systems involved human interaction via telephone calls. Handling phone calls can be time-intensive. Staff members need to manually coordinate and organize bookings, which can lead to delays and inefficiencies. Phone-based systems create a bottleneck. Customers have to wait in line or on hold, especially during peak hours. Manual data entry increases the risk of errors. Misunderstandings, double bookings, or incorrect information can occur. Phone-based systems don't easily allow for upselling additional services or products during the booking process. Online systems can seamlessly offer upgrades or related items. Maintaining phone lines and hiring staff to handle calls can be expensive. Online booking systems can be more cost-effective in the long run. Phone-based systems rely heavily on staff availability. If a staff member is absent or busy, it affects the booking process.

Here are some of the problems of the current system:

- Time-Consuming.
- Customers have to wait in line or on hold
- Human Error and Misunderstandings
- Maintaining phone lines and hiring staff to handle calls can be expensive
- Phone-based systems rely heavily on staff availability
- Handling payments over the phone can be cumbersome

3.3 PROPOSED SYSTEM

In today's digital age, businesses across various sectors are embracing online booking systems to simplify their operations and enhance customer satisfaction. These systems offer several advantages, making them a smart choice for managing reservations and appointments.

Advantages

- Contactless
- User friendly

- Extremely efficient
- Smooth Operation
- Real-time updates

3.4 PURPOSE OF THE SYSTEM

- Instant access to orders
- Progress tracking
- Feedback system
- Table oriented bills
- Instant waiter request
- Reduce waiting time
- Bright menu easy to read
- More information on menu items

The proposed system has mainly contained five modules they are,

1. ADMIN
2. USER

FUNCTIONS OF ADMIN

- Login
- Manage club details
- View bookings (Approve Disapprove Bookings)
- View payment
- Update club news
- Add and manage sports item
- View member details
- Add and manage booking purpose

FUNCTIONS OF CUSTOMER

- Login
- Signup
- Ground Booking for event
- Book amount calculation acc to period of booking
- Vacation and regular batch registration
- View Club news
- Members Area

CHAPTER 4

SYSTEM DESIGN

4.1 ARCHTECTURE DESIGN

Architecture is an overall structure of a system. Architecture takes into consideration the overall working of the system; large system can be decomposed into sub-systems that provide some related set of services. The initial design process of identifying these sub-systems and establishing a framework for sub-systems control and communication is called architecture design.

Architecture design usually comes before detailed system specification. Architecture decomposition is necessary to structure and organize the specification. There is no generally accepted process depends on application knowledge and on the skill and intuition of the system architecture.

System design is the process of designing the elements of a system such as the architecture, modals and components, the different interfaces of those components and the data that goes through that system. This is the most creative and challenging phase important too. It improves the understanding and the procedural details necessary for implementing the new system. System design is a process contains asset of required to efficiently transform user need in to an effective software solution. This helps define overall system architecture

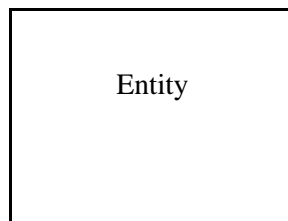
4.2 ENTITY RELATIONSHIP DIAGRAM

An ER diagram can express the overall structure of the database graphically. ER diagrams are simple and clear. ER diagram often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes.

Entity Relationship Diagram Notations

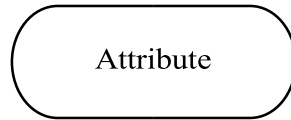
Entity:

An entity is an object or concept about which you want to store information



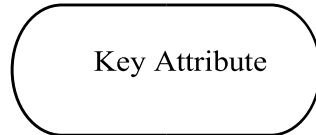
Attribute:

Each entity has attribute, or particular properties that describe the entity.



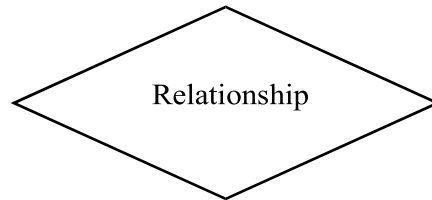
Key Attribute:

A key attribute is the unique, distinguishing characteristics of the entity.

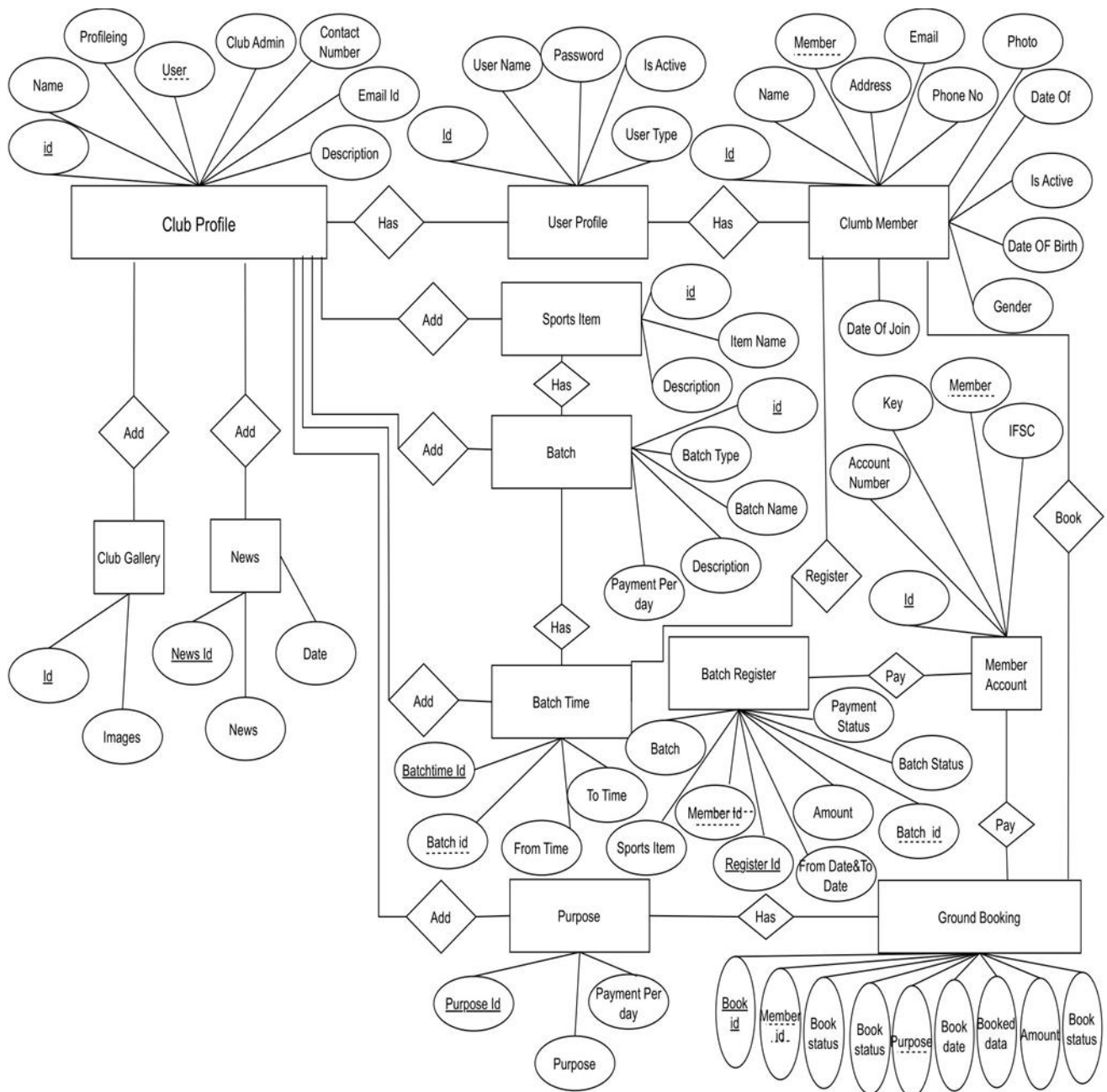


Relationships:

Relationships illustrate how two entities share information in the database structure.



4.2.1 Er Diagram



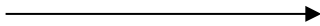
4.3 DATAFLOW DIAGRAM

DFD DESIGN NOTATIONS

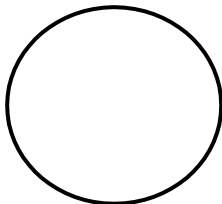
A system's data and how the data interacts with the system in DFD, there are four main symbols Data Flow Diagram is the graphical description of the system's data and how the processes transform the data. Data Flow Diagram depicts information flow, and the transforms that are applied as data move from the input to output. It is the starting point of the design phase that functionally decomposes the requirement specifications down to the lowest level of details. Thus, a DFD describes what data flows (logical) rather than how they are processed. Unlike detailed flowchart, Data Flow Diagrams don't supply detailed description of the modules but graphically describes:



- Source or Destination of data



- Flow of Data



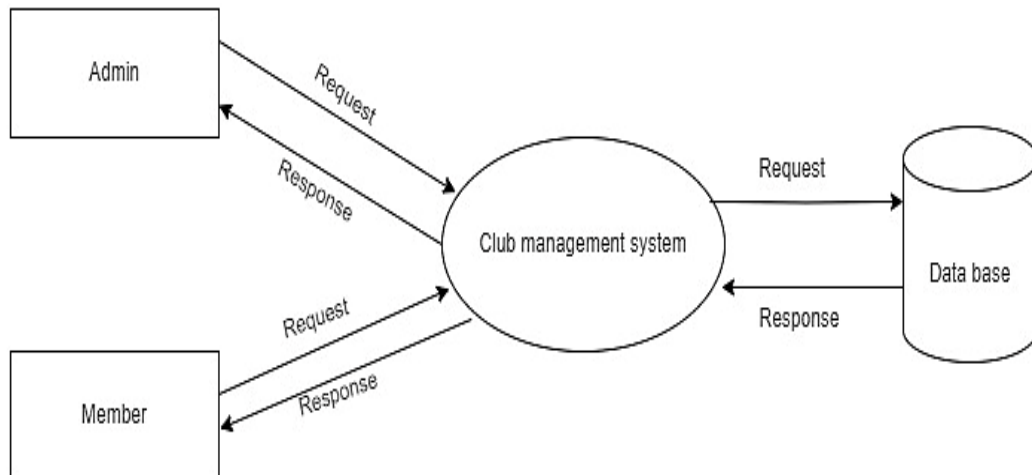
-Process transforming data



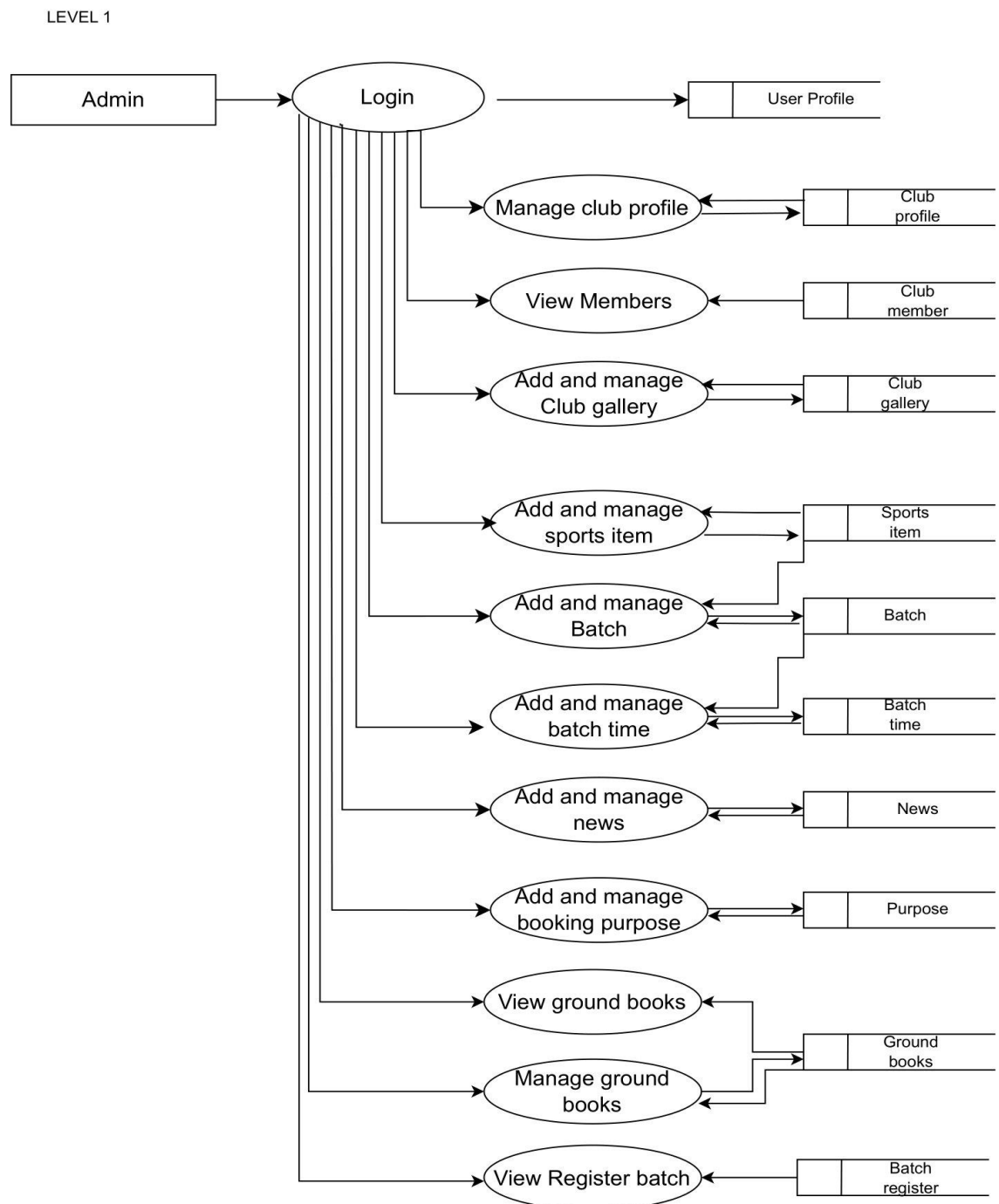
-Temporary repository of data

4.3.1 DFD Level 0

Level 0

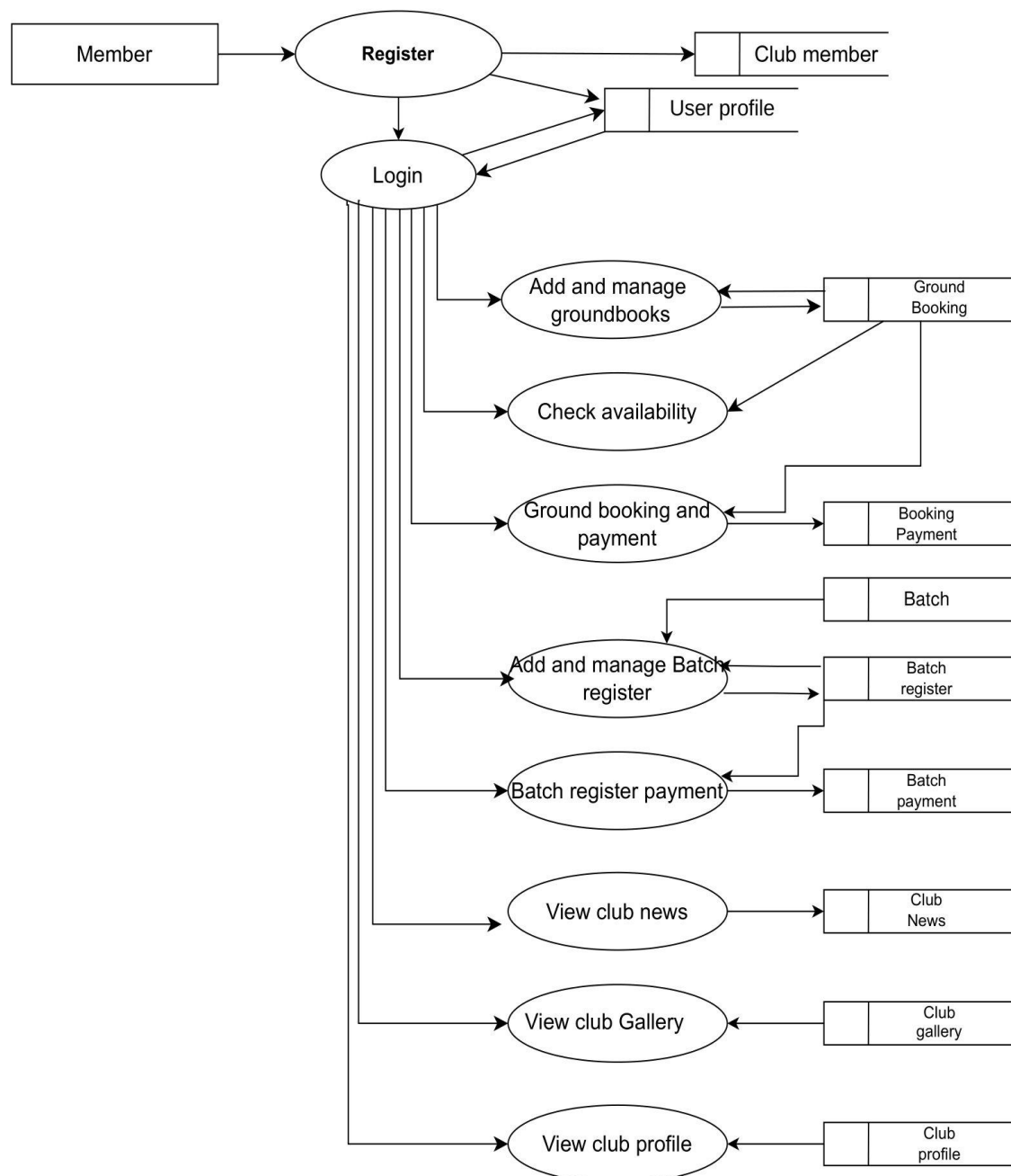


4.3.2 DFD Level 1



4.3.3 Level 2

LEVEL 2



4.4 INPUT DESIGN

Input design is the link that ties the information system into the world of its user. Inaccurate input data are the most common cause of errors in data processing. Error entered by data entry operators can be controlled by input design. It is the process of converting user originated input to a computer-based format. The input design involves determining what the inputs are, how the data should be performed, how to validate data, how to minimize data entry and how to provide a multi-user facility. The input data are collected and organized into groups of similar data. In input data design, the designing of source document that capture the data and then selected the media used to enter them into the computers. The source document may be entered into the system through punch cards, dice or even directly from keyboard.

4.5 OUTPUT DESIGN

Computer output is the most important and direct source of implementation of the user. Efficient intelligible, output design should improve the system's relationship with the user and help in decision making. It is a very important phase because the output needs to be in an attractive manner. The success or failure of software is decided by the integrity and correctness of output that is produced from the system. One of the objectives behind the automation of business system itself is the fast and prompt generation of reports in a short period of time. In today's competitive world of business, it is very important for companies to keep themselves up to date about the happening in the business. Prompt and reliable reports are considered to be the lifeline of every business today. So the output report generated by the software system is of paramount importance.

CHAPTER 5

DATABASE DETAILS

5.1 DATABASE DETAILS

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the tables and views. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system. The process of doing database design generally consists of a number of steps which will be carried out by the database designer.

Usually, the designer must

- Determine the relationships between the different data elements
- Superimpose a logical structure upon the data on the basis of these relationships

The following factors are to be considered while designing a database

- Organize data into columns
- Decide the primary key
- Normalization.

5.2 NORMALISATION

Normalization is the process of decomposing a set of relations with anomalies to produce smaller and well-structured relations that contain minimum redundancy. It is a formal process of deciding which attributes should be grouped together in a relation.

5.2.1 First Normal Form

First Normal form (1NF) is now considered to be part of the formal definition of relational model. 1NF is designed to disallow multivalued attribute, composite attributes, and their combinations. It states that the domain of an attribute must include only atomic values. A domain is atomic, if elements of the domain are considered to be indivisible units. We say that a relational schema R is in 1NF if the domain of all attributes of R' is atomic.

5.2.2 Second Normal Form

Second Normal form (2NF) is based on the concept of functional dependency. A relation R is in 2NF if it is in 1NF and every non key attribute A of R is fully dependent on the primary key. That is, relation is said to be in 2NF if each attribute A in R meets one of the following criteria:

- (a) It appears in the primary key.
- (b) It is fully functionally dependent on the primary key

5.2.3 Third Normal Form

Third Normal form (3NF) is based on the concept of transitive dependency. A relation is said to be in 3NF if it is in 2NF and has no transitive dependencies. That is all the non key attribute should be functionally determined by the primary key. In the proposed system all attributes of tables are fully depends on the primary key only that is all non key attributes are mutually independent.

5.3 DATABASE TABLES

5.3.1 USERPROFILE

TABLE 5.3.1

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Id	Auto field		PRIMARY KEY
Username	Charfield	50	
Usertype	Charfield	50	
Is active	Boolean Field		

5.3.2 CLUMBMEMBER

TABLE 5.3.2

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Id	Auto field		PRIMARY KEY
Name	Charfield	50	
Member	Charfield		FOREIGN KEY
Address	Text field		
Email	Charfield	50	
Phonenumber	Integer Field		
Date of birth	Date Field		
Gender	Text field	50	
Date of join	DateTime field		
Photo	Image field		
Is active	Boolean Field		

5.3.3 MEMBERACCOUNT

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Id	Auto field		PRIMARY KEY
Account number	Text field	50	
IFSC	Charfield	50	
Key	Charfield	50	
Member	Integer Field		FOREIGN KEY
Amount	Charfield	50	

5.3.4 SPORTSITEM

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Id	Autofield		PRIMARY KEY
Item name	Charfield	50	
Description	Textfield	100	

5.3.5 CLUB PROFILE

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Id	Autofield		PRIMARY KEY
Name	Charfield	50	
Profileimg	ImageFiled		
Clubadmin	Charfield	50	
Contact number	Integer Field	10	
Email Id	Email Field		
Description	Text Field	100	
User	Integer Field		FOREIGN KEY

5.3.6 CLUBGALLERY

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Id	Auto Field		PRIMARY KEY
Img	Image Field		

5.3.7 BATCH

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Id	Auto Field		PRIMARY KEY
Batch type	Integer Field		FOREIGN KEY
Batch name	Char Field	50	
Description	Text Field		
Payment per day	Integer Field		

5.3.8 BATCH TIME

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Batch time id	Auto Field		PRIMARY KEY
Batid	Integer Field	50	FOREIGN KEY
From time	Date Field		
To time	Date Field		

5.3.9 BATCH REGISTER

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
RegId	Auto Field		PRIMARY KEY
Membid	Integer Field		FOREIGN KEY
Sportsitem	Char Filed	50	
Batch	Char Field	50	
Batchid	Integer Field		FOREIGN KEY
Batchtime	Char Filed	50	
Fromdate	Date Field		
Todate	Date Field		
Amount	Integer Field		
Paymentstatus	Char Field	50	
Batchstatus	Char Filed	50	

5.3.10 PURPOSE

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Purposeid	Auto Field		PRIMARY KEY
Purpose_name	Char Filed	50	
Payment_perday	Integer Field	50	

5.3.11 GROUNDBOOKING

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Bookid	Auto Field		PRIMARY KEY
Membid	Integer Field	50	FOREIGN KEY

Purpose	Integer Field	50	FOREIGN KEY
Bookdate	Date Field		
Bookeddate	Date Field		
Amount	Integer Field		
Bookstatus	Char Field	50	
PaymentStatus	Char Filed	50	

5.3.12 NEWSTB

COLUMN NAME	DATA TYPE	SIZE	CONSTRAINT
Newsid	Auto Field		PRIMARY KEY
News	Text Field		
Date	Date Field		

CHAPTER 6

OVERVIEW OF THE SOFTWARE

6.1 FRONT END SOFTWARE VISUAL STUDIO CODE (flutter)

The story of Visual Studio Code begins with Microsoft's recognition of the growing need for a lightweight, versatile, and cross-platform code editor. In the early 2010s, the software development landscape was evolving rapidly, with a surge in popularity of open-source tools and platforms. Developers were increasingly seeking tools that provided a seamless coding experience across different operating systems and languages.

In 2011, Microsoft launched Visual Studio Online Monaco, a web-based code editor, as part of its broader cloud development strategy. While Monaco served its purpose, Microsoft realized that there was a demand for a more powerful, standalone, and locally installable code editor that could cater to a broader audience of developers.

In April 2015, Microsoft unveiled Visual Studio Code during the Build conference. Unlike the traditional Visual Studio IDE, Visual Studio Code was designed to be a lightweight, fast, and cross-platform code editor. It was released as a free and open-source product under the MIT License.

The key motivations behind the development of Visual Studio Code were:

- **Cross-Platform Compatibility:** Recognizing the increasing diversity of developer environments, Microsoft aimed to create a tool that would work seamlessly on Windows, macOS, and Linux. This decision was a departure from the Windows-centric approach of the traditional Visual Studio IDE.
- **Lightweight and Fast:** Developers needed a code editor that was quick to install, responsive during use, and didn't require extensive system resources. Visual Studio Code was built with a minimalistic design and a focus on performance.
- **Open Source and Extensible:** Microsoft embraced the open-source ethos with Visual Studio Code, encouraging community contributions and providing extensive documentation. The extension system allowed developers to customize and enhance their coding experience by adding language support, themes, and other features.
- **Developer Productivity:** Visual Studio Code aimed to boost developer productivity with features like IntelliSense for intelligent code completion, built-in Git integration, and a rich set of extensions. The goal was to create an editor that could adapt to various programming languages and workflows.

Over the years, Visual Studio Code has undergone continuous development and refinement. The extension ecosystem has flourished, with a wide range of third-party extensions created by both Microsoft and the community. The active and engaged developer community has played a significant role in shaping the evolution of Visual Studio Code, providing feedback, reporting issues, and contributing to its growth.

Today, Visual Studio Code stands as one of the most popular and widely used code editors in the software development industry, embodying Microsoft's commitment to supporting diverse development environments and fostering collaboration within the developer community.

6.1.1 ADVANTAGES OF VISUAL STUDIO

Cross-Platform Compatibility

Visual Studio Code is designed to work seamlessly across Windows, macOS, and Linux, providing a consistent coding experience on different operating systems.

Lightweight and Fast

VS Code is known for its minimalistic design and quick startup time. It consumes fewer system resources compared to traditional IDEs, making it an excellent choice for developers on less powerful machines.

Extensibility

Similar to Visual Studio IDE, Visual Studio Code supports a rich ecosystem of extensions. Developers can customize their editor with language support, themes, and various productivity tools.

Integrated Git Support

VS Code includes built-in Git integration, allowing developers to manage version control directly within the editor. This simplifies collaboration and code repository management.

Intelligent Code Completion

VS Code features IntelliSense, providing intelligent code completion and suggestions for a wide range of programming languages. This enhances coding productivity and reduces errors.

Integrated Terminal

The editor includes an integrated terminal, allowing developers to run command-line tools and scripts without leaving the coding environment.

Active Community

Visual Studio Code has a vibrant and active community. Regular updates, a wealth of online resources, and extensive documentation contribute to its popularity and ongoing improvement.

6.2 BACK-END SOFTWARE

6.2.1 PyCharm

- PyCharm is an integrated development environment (IDE) used in computer programming, specifically for Python language
- It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes) and supports web development with Django
- Coding assistance and analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes
- Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages.
- Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional.
- programming Python is often described as a "batteries included" language due to its comprehensive standard library

6.3 DATABASE

6.3.1 Sqlite-3

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Micro systems now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MYSQL project to create Maria DB. MYSQL is a component of the LAMP web application software stack (and others), which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database-driven web applications including Drupal, Joomla, php BB and WordPress. MySQL is also used by many popular websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

Cross-platform support

- Stored procedures, using a procedural language that closely adheres to SQL/PSM
- Triggers
- Cursors
- Updatable views
- Online Data Definition Language (DDL) when using the InnoDB Storage Engine
- Information schema
- Performance Schema that collects and aggregates statistics about server execution and query performance for monitoring purposes.

- Unicode support

Backup software

MySQL dump is a logical backup tool included with both community and enterprise editions of MySQL. It supports backing up from all storage engines. MySQL Enterprise Backup is a hot backup utility included as part of the MySQL Enterprise subscription from Oracle, offering native InnoDB hot backup, as well as backup for other storage engines

High availability software

Oracle MySQL offers nowadays an high availability solution with a mix of tools including the MySQL router and the MySQL Shell all based on Group Replication, open source tools

Cloud deployment

See article: Cloud database

MySQL can also be run on cloud computing platforms such as Microsoft Azure Amazon EC2, Oracle Cloud Infrastructure

CHAPTER 7

SYSTEM IMPLEMENTATION

7.1 SYSTEM IMPLEMENTATION

Implementation phase is the phase, which involves the process of converting a new system design into an operational one. It is the key stage in achieving a successful new system. Implementation is the stage of the project, where the theoretical design is turned into a working system. At this stage, the workload, the greatest upheaval and major impact on controlled carefully,

It can cause chaos. The implementation stage is a system project in its own right. It involves careful planning, investigation of the current system and its constraints on the implementation, design methods the implementation plan consists of the following steps:

Testing the developed system within the sample data

- Detection and correctness of errors
- Making necessary changes in system
- Training and involvement of user personnel
- Installation of software utilities

7.2 IMPLEMENTATION PROCEDURES

The implementation phase is less creative than system design. A system project may be dropped at any time prior to the implementation although it becomes more difficult when it goes to the design phase. The final report to the implementation phase includes procedural charts, record layout and a workable plan for implementing the candidate system. Implementation is used to the process of converting a new or revised system design into an operational one. Conversion is one aspect of implementation. Several procedures are unique to the implementation phase Conversion begins with a review of the project plan, the system test documentation, and implementation plan.

CHAPTER 8

SYSTEM TESTING

8.1 IMPLIMENTATION

System implementation is the stage where the theoretical design is turned into a working system. The system can be implemented only after thorough testing is done and if it is found to work according to specifications. The following methods were undergone.

- Testing developed programs with updating • Correction of errors identified.
- Creating the tables of the system with actual data. . Making necessary changes with actual data.
- Doing a parallel run of the system to find out any errors identified and to correct them.
- Training of user personnel's

8.2 TESTING

For software that is newly developed, primary importance is given to testing the system. It is the last opportunity for the developer to detect the possible errors in the software before handing over it to the customer. Testing is the processes by which the developer will generate a set of data, which gives the maximum probability of finding all types of error, occur in the software.

The various steps in testing the system can be listed as below:

- Running the program to identify any errors that might have occurred while feeding the program into the system
- Applying the screen formats to regulate users to extend, so that the screens are comprehensible to the user.
- Presenting the formats to the administration for the purpose of obtaining approval and checking if any modification has to be done. Obtaining feedbacks from users and analyzing the scope for improvement.
- Checking the data accessibility from the data server and whether any improvement or not. s needed or not.
- Testing is a methodology for evaluating the project. The good test has a high probability of finding an error. Testing is generally two types- Black box testing and White box testing.

8.2.1 TYPES OF TESTING

White Box Testing

White box Testing (also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing) is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. White box testing techniques analyses the internal structures the used data structures, internal design, code structure and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing. White box testing is a testing technique, that examines the program structure and derives test data from the program logic/code. The other names of glass box testing are clear box testing, open box testing, logic driven testing or path driven testing or structural testing. The white box testing helps to forces test developer to reason carefully about implementation and reveals errors in "hidden" code and spots the Dead Code or other issues with respect to best programming practices.

Black Box Testing

Black-box testing is a method of software testing that examines the functionality of an application based on the specifications. It is also known as Specifications based testing. Independent Testing Team usually performs this type of testing during the software testing life cycle. This method of test can be applied to each and every level of software testing such as unit, integration, system and acceptance testing. There are different techniques involved in Black Box testing is Equivalence Class, Boundary Value Analysis, Domain Tests, Orthogonal Arrays, Decision Tables, State Models, Explore tory Testing and All-pairs testing. Its also known as Behavioral Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional.

8.2.2 TESTING STRATEGIES

Unit testing

Unit testing focuses on verification effort on the smallest limit of software design. Using the unit test plan prepared in the design phase of the system, important control paths are tested to uncover the errors within the module. This testing was carried out during the coding itself. In this testing step each module is going to be working satisfactorily as the expected output from the module.

Integration testing

It is the systematic technique for constructing the program structure to uncover errors associated with the interface. The objective is to take unit-tested module and built the program structure that has been dictated by design. All modules are combined in this step. Then the entire program whole. If a set of errors is encountered connection is difficult because the isolation of causes is tested as a complicated by vastness of the entire program. Using this test plan preparing the design phase of the system, the integration was carried out. All the errors found in the system were corrected for the next testing step.

System testing

In system testing, the software and other system elements are tested as a whole. System testing is actually a series of different test whose primary purpose is to fully exercise the computer-based system.

Validation testing

All the end of the integration testing, software is completely assembled as package, interfacing errors have been uncovered and corrected and final series of software validation testing begins. Validation testing can be defined in any ways, but a simple definition is that validation succeeds when the software function is a manner that can be reasonably accepted by the user.

Software validation is achieved through a serious of black box test that demonstrate conformity with requirements. After validation test has been completed one of the following two conditions exists:

- 1) The function or performance characteristics confirm to specification and are accepted.
- 2) A deviation from a specification is uncovered and a deficiency list it's created.

Output Testing

After performing validation testing the next step is to perform the output testing of the proposed system. Since no system could be useful if it does not produce the required output in the specified format.

User Acceptance Testing

User Acceptance of a system is a key factor to the success of any system. The system under consideration was tested for user acceptance by constantly keeping in touch with the prospected system set at the time of developing and making changes wherever required. This is done with regard to the points.

1. Input screen design.
2. Output screen design.
3. Online message to guide the user

8.2.3 VERIFICATION AND VALIDATION

The goals of verification and validation activities are to access and improve the quality of the work product generated during development and modification of this software. Quality attributes include correctness, completeness, consistency, reliability, usefulness, usability, and efficiency, conformance to standard and overall cost effectiveness. There are two types of verification. They are follows:

- Life cycle verification is the process of determining the degree to which the work products of a given phase of the development cycle fulfil the specification established during prior phases.
- Formal verification demonstrates that source code confirms to its specification. This check both the verification.

Validation is the process of evaluating software at the software development process to determine compliance with the requirements. This system did a lot of validation activities. High quality cannot be achieved through testing of source code alone. The source code errors are the measures of quality, testing alone cannot guarantee the absence of errors in a program. The number of bugs remaining in a program is proportional to the number of already discovered. This project minimizes the number of errors in a program is to catch and remove the errors during analysis and design, so that few errors are introduced into the source code

8.2.4 SYSTEM SECURITY

System security is a branch of technology known as information security as applied to computer and networks. The objective of system security includes protection of information and property from theft, corruption, or natural disaster, while allowing the information and property to remain accessible and productive to its intended users. The terms system. means the collective process and mechanisms by which sensitive and valuable information and services are protected from publication, tampering or collapse by unauthorized activities or untrustworthy individuals and unplanned events respectively

8.2.5 SYSTEM MAINTENANCE

Maintenance corresponds to restoring something to original condition covering a wide range of activity, including correcting coding, design errors, updating user support. Better the system design, easier to maintain the system. Maintenance is performed most often to improve the existing software rather than to respond to a crisis or system failure. According to user needs and operational environment change maintenance should be done in parallel, otherwise the system could fail. Provision must be made for environmental changes, which may affect

either the computer or other parts of a computer-based system such as activity are normally called maintenance. It includes both improvement of system functions and the correction of fungus that arise during the operation of a system Maintenance activity may require the continuing involvement of a large proportion of computer department resources. Most changes arise in two ways.

As part of the normal running of the system when errors are found, users ask for improvement of external requirements change. As the results of specific investigation and review of the systems performance, maintenance involves the software industry captive.

typing of system resources. It means restoring something to its original condition Maintenance was done after the successful implementation. Maintenance is continued till the product is re-engineered or deployed to another platform. Maintenance is also done based on fixing the problem reported, changing the interface with other software or hardware enhancing the software. Any system developed should be secured and protected against possible hazards. Security measures are provided to prevent authorized access of the database at various levels: An uninterrupted power supply should be provided so that power failure or voltage fluctuations will not erase the data in the files.

CHAPTER 9

FUTURE ENHANCEMENT AND CONCLUTION

9.1 FUTURE ENHANCEMENT

As future enhancement **“CLUB MANAGEMENT SYSTEM”** we planned to improve the user experience part of the Software and also aim to make include multiple ground.

9.2 CONCLUSION

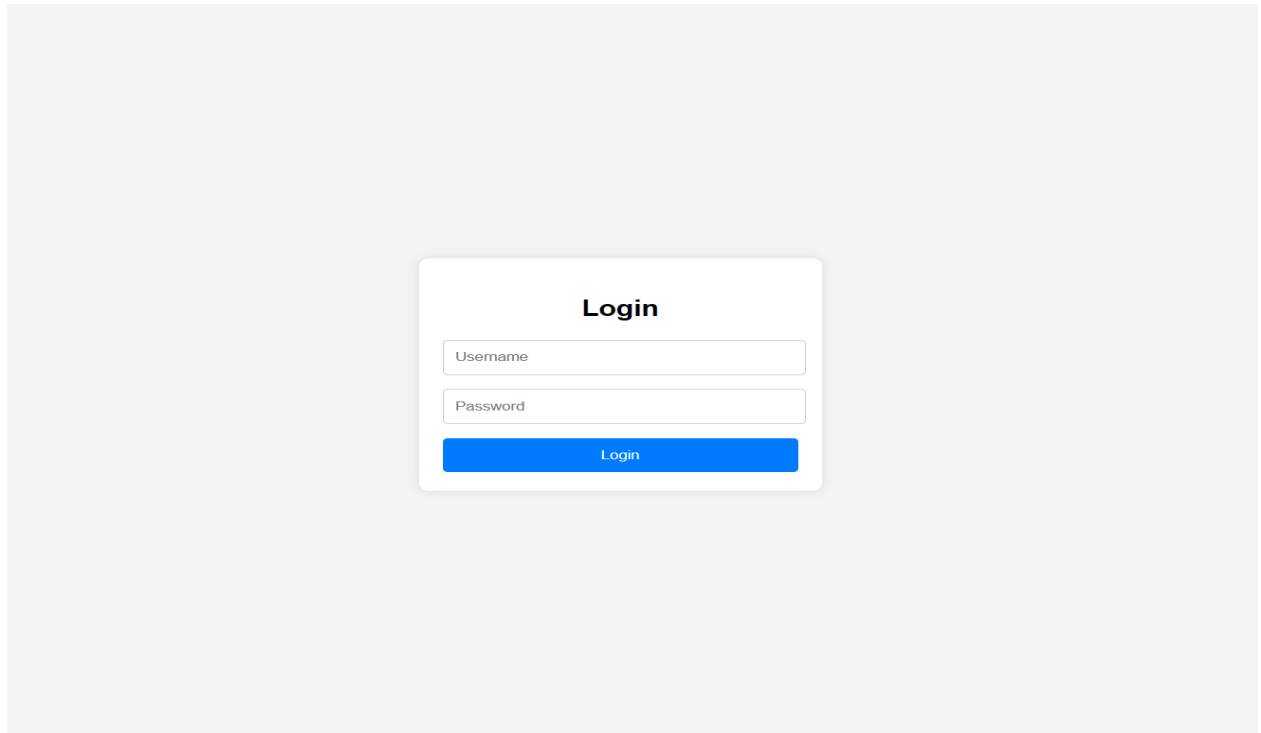
As per the requirement specified proposed system is successfully implemented and tested it has been a great pleasure for me to work on this existing and challenging project. This project proved good for me as it provided practical knowledge of programming in android and python. The Project achieved entire goal the practice of **“CLUB MANAGEMENT SYSTEM”** the proposed application helps to place orders with ease.

CHAPTER 10

SCREENSHOTS

10.1 WEB SCREENSHOTS

10.1.2 Login



A screenshot of a login form titled "Login" centered on a light gray background. The form is a white rounded rectangle containing two input fields: "Username" and "Password", each with a small eye icon for password visibility. Below the fields is a blue "Login" button.

10.1.3 Home Page



10.1.3 Profile

Club Management System

Home

Profile

View Members

Sports Item

Gallery

Batch

News

Purpose

Ground booking

Batch register

Logout



Muhammed Abdul Basith

Club Admin Muhammed Abdul Basith

Email basiibnumoideen@gmail.com

Phone Number 8590882253

Description A very good Academy and we provide a wonderful ground



© 2024 Your Website. All rights reserved.

10.1.4 View members

Club Management System

Home

Profile

View Members

Sports Item

Gallery

Batch

News

Purpose

Ground booking

Batch register

Logout

ID	NAME	ADDRESS	EMAIL	PHONE NUMBER	DATE OF BIRTH	GENDER	PHOTO	DATE OF JOIN	
5	anas	areethode	anasbichu2002@gmail.com	9072470985	Sept. 26, 2002	Male		March 2, 2024, 8:20 a.m.	VIEW
6	Muhammed Abdul Basith	Thadathil Appartment, Eliyaparambu	basiibnumoideen@gmail.com	8590882253	Dec. 22, 2003	Male		March 5, 2024, 7:47 a.m.	VIEW



© 2024 Your Website. All rights reserved.

Page | 44

10.1.5 Sports item

Club Management System

+

ID	ITEM NAME	DESCRIPTION		
1	Cricket	Cricket is a bat-and-ball game played between two teams of eleven players on a field at the centre.v	Edit	Delete
2	Football	Association football, more commonly known as football or soccer, is a team sport played between two	Edit	Delete
3	Volleyball	Volleyball is a team sport in which two teams of six players are separated by a net. Each team tmimm	Edit	Delete

Home

Profile

View Members

Sports Item

Gallery

Batch

News

Purpose

Ground booking

Batch register

Logout







© 2024 Your Website. All rights reserved.

10.1.6 Gallery

Club Management System

+

ID	IMAGE	
4		Delete
5		Delete
6		Delete
7		Delete

Home

Profile

View Members

Sports Item

Gallery

Batch


News

Purpose

Ground booking

Batch register

Logout



© 2024 Your Website. All rights reserved.

10.1.7 Batch

Club Management System

Home

Profile

View Members

Sports Item

Gallery

Batch

News

Purpose

Ground booking

Batch register

Logout

+

Batch Type ALL SEARCH

ID	BATCH TYPE	BATCH NAME	DESCRIPTION	PAYMENT PER DAY			
4	Cricket	Morning	A cricket morning batch typically refers to a scheduled training session or practice session for cricket players that takes place in the morning hours. These batches are organized by cricket clubs, academies, schools, or other cricket training institutions to provide structured coaching and practice opportunities for players of various skill levels.	99		Edit	Delete
5	Cricket	Night	A cricket night batch typically refers to a scheduled training session or practice session for cricket players that takes place during the evening or nighttime hours. These batches are organized by cricket clubs, academies, schools, or other cricket training institutions to provide structured coaching and practice opportunities for players who may have other commitments during the day or prefer to train in cooler temperatures.	120		Edit	Delete
6	Football	Morning	A soccer morning batch typically refers to a scheduled training session or practice session for soccer players that takes place in the morning hours. These batches are organized by soccer clubs, academies, schools, or other soccer training institutions to provide structured coaching and practice opportunities for players of various ages and skill levels.	99		Edit	Delete
7	Football	Evening	A soccer evening batch typically refers to a scheduled training session or practice session for soccer players that takes place in the evening hours. These batches are organized by soccer clubs, academies, schools, or other soccer training institutions to provide structured coaching and practice opportunities for players who may have other commitments during the day or prefer to train in cooler temperatures.	120		Edit	Delete

© 2024 Your Website. All rights reserved.

10.1.8 News

Club Management System

Home

Profile

View Members

Sports Item

Gallery

Batch

News

Purpose

Ground booking

Batch register

Logout

+

ID	NEWS	DATE	
6	<div> <div>Important Announcement: Tomorrow's Batch Cancelled</div> <div> <p>Dear Morning Soccer Members, We regret to inform you that tomorrow's scheduled training batch has been cancelled due to essential maintenance work at our facilities. The maintenance is necessary to ensure the safety and quality of our training environment. Date of Cancellation: 20/3/2024 Reason for Cancellation: Maintenance Work We understand the inconvenience this may cause and apologize for the short notice. Please rest assured that we are working diligently to minimize any disruption to our training schedule. We kindly ask all participants to adjust their schedules accordingly. Rest assured, we will reschedule the missed session and communicate the new date and time as soon as possible. Thank you for your understanding and cooperation. Best regards, Manager Club Management System.</p> </div> </div>	March 19, 2024	Delete


© 2024 Your Website. All rights reserved.

10.1.9 Purpose

Club Management System

[Home](#)
[Profile](#)
[View Members](#)
[Sports Item](#)
[Gallery](#)
[Batch](#)
[News](#)
[Purpose](#)
[Ground booking](#)
[Batch register](#)
[Logout](#)

ID	Name	Payment Per Day		
1	Marriage	10000	Edit	Delete
4	DJ	20000	Edit	Delete
5	Auto Expo	10000	Edit	Delete



© 2024 Your Website. All rights reserved.

10.1.10 Ground Booking

Club Management System

[Home](#)
[Profile](#)
[View Members](#)
[Sports Item](#)
[Gallery](#)
[Batch](#)
[News](#)
[Purpose](#)
[Ground booking](#)
[Batch register](#)
[Logout](#)

BOOK ID	MEMBER	PURPOSE	BOOK DATE	BOOKED DATE	AMOUNT	BOOK STATUS	PAYMENT STATUS		
1	anas	Marriage	March 4, 2024	March 2, 2024	10000	Approved	Complete	Approve	Reject
2	anas	Marriage	March 5, 2024	March 2, 2024	10000	Reject	Complete	Approve	Reject



© 2024 Your Website. All rights reserved.

10.1.11 Batch register

Club Management System

[Home](#)[Profile](#)[View Members](#)[Sports Item](#)[Gallery](#)[Batch](#)[News](#)[Purpose](#)[Ground booking](#)[Batch register](#)[Logout](#)

Batch Type: ALL

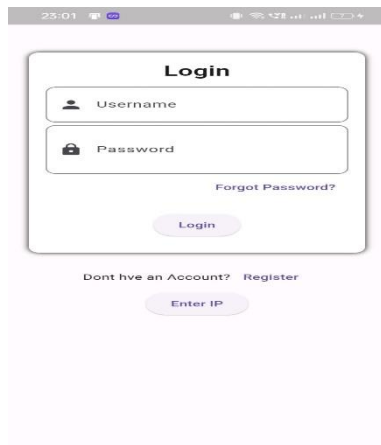
ID	MEMBERID	BATCHTIMEID	FROMDATE	TODATE	AMOUNT
5	anas	10	March 20, 2024	March 21, 2024	99



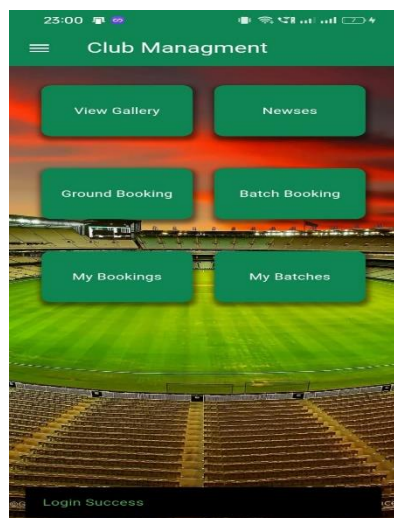
© 2024 Your Website. All rights reserved.

10.2 APP SCREENSHOTS

10.2.1 Login



10.2.2 Homepage



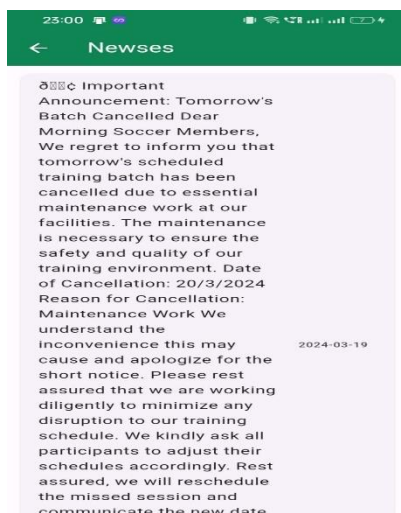
10.2.3 Profile



10.2.4 View Gallery



10.2.5 News



10.2.6 Ground Bookings

A screenshot of a mobile application interface for booking a ground. At the top, there is a green header bar with a back arrow and the text "Book your Event". Below the header, there is a light purple box containing three input fields. The first field is a dropdown menu with the text "select". The second field is a date picker with the text "Date". The third field is a text input with the text "Amount". Below these fields, there is a button with the text "Pay Now".

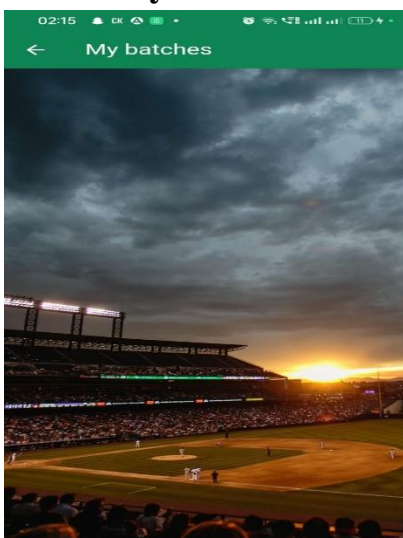
10.2.7 Batch Bookings



10.2.8 My Bookings



10.2.9 My Batches



10.2.10 About Us



CHAPTER 11

BIBLIOGRAPHY

10.1 BIBLIOGRAPH

Ian Somerville, Software Engineering, 5th edition, Pearson Education Programming
Android Java programming for the new Generation of mobile devices introduction to Machine
language with Python.

10.2 ONLINE REFERENCE

1. <https://en.wikipedia.org>
2. <https://www.android.com>
3. <https://www.python.org>