



GYM DATABASE MANAGEMENT SYSTEM

PROJECT REPORT

Prepared by:

BHAVYA TIWARI 2022UCS1574

HUZAIFA MULLA 2022UCS1510

PIYUSH LAHORI 2022UCS1580

SHOBIT MALHOTRA 2022UCS1505

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

- 1.1 Introduction of the Problem Area
- 1.2 Problem Statement/Purpose of the Project
- 1.3 Objective of the Project
- 1.4 Find and organize the information required(Requirement Analysis)

CHAPTER 2 HARDWARE REQUIREMENT

- 2.1 Hardware Tools

- 2.2 Software Tools

CHAPTER 3 DESIGN

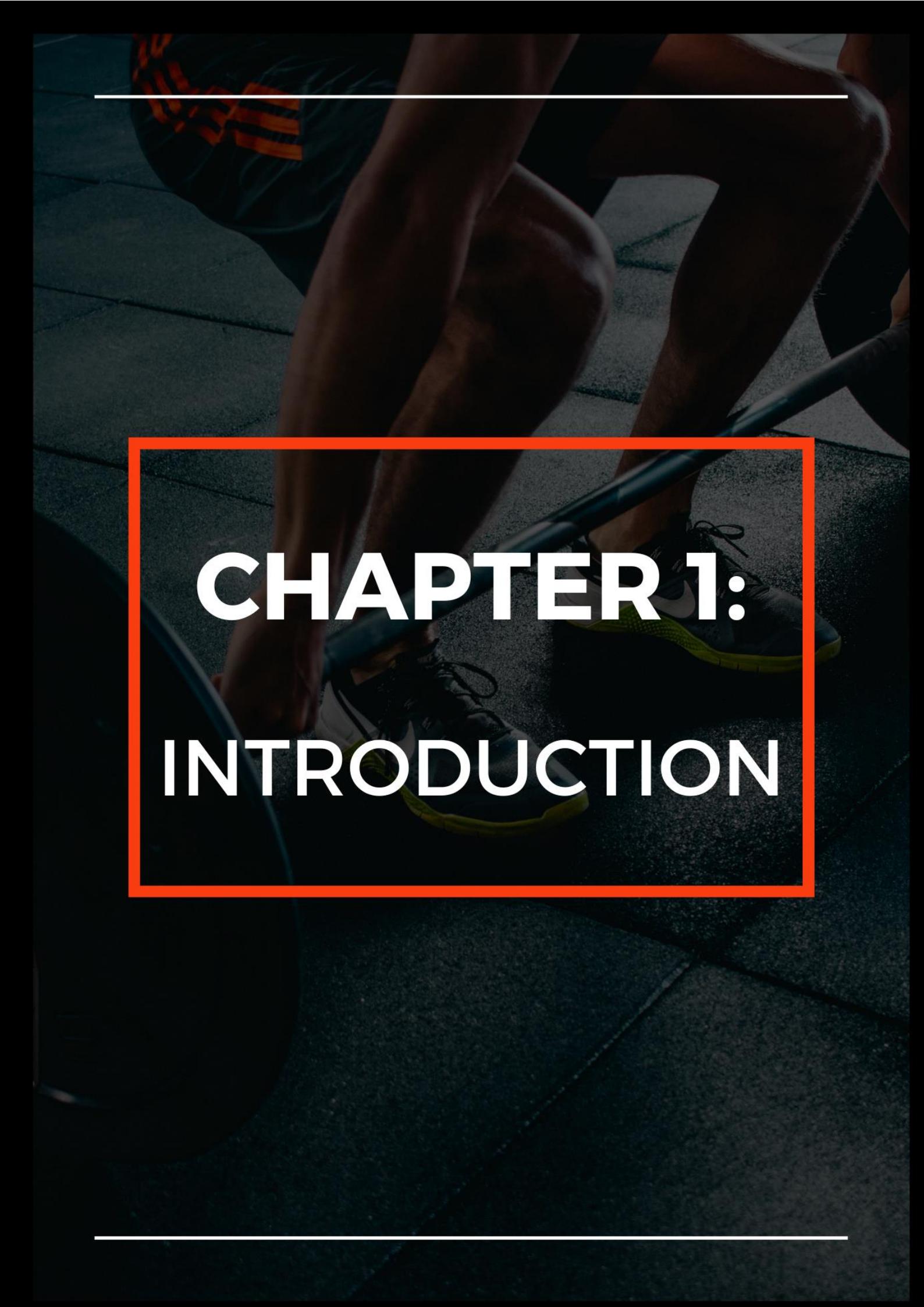
- 3.1 ER Model
- 3.2 ER to Schema {Divide the information into tables}
- 3.3 Functional Dependencies {Based on the requirement that matches with the sample data}
- 3.4 Specify the primary key
- 3.5 Set-up the table relationship
- 3.6 Refine your design by applying normalization rule till BCNF

CHAPTER 4 IMPLEMENTATION

- 4.1 Xampp Table
- 4.2 Application with code
- 4.3 Snapshots of running of application

CHAPTER 5 REFERENCES

- References



CHAPTER 1: INTRODUCTION

PROBLEM STATEMENT



1.1 INTRODUCTION OF THE PROBLEM AREA

Health and wellness of students is treated as an area of prime importance by the administration of NSUT, with respect to the welfare of students on campus, recognizing the pivotal role physical fitness plays in the holistic development of individuals. It is also something that young individuals are particularly passionate about, in this day and age. As students who regularly visit and use the college Gymnasium, we've identified several challenges that hinder its seamless operation and, consequently, impact the overall experience for fitness enthusiasts on campus.

From equipment tracking and maintenance to user engagement and scheduling inefficiencies, there exists a need for a comprehensive solution. This project endeavours to address these concerns by developing a Gym Database Management System, aiming to streamline operations, enhance user experiences, and contribute to the overall well-being of our student community.

1.2 PROBLEM STATEMENT

The problems faced daily are:

1. Manual data management:

The entries of students are done manually which gives rise to discrepancies. It also makes it nearly impossible to verify the identity of the student who is making the entry. This makes it difficult to hold someone responsible for the damages caused, if any.

2. Crowd management:

There is no cap on how many students are allowed to access the gym at a time. Similarly, there is no time constraint provided to the students for gym usage. Even if such constraints are imposed, there is no way to maintain order.

3. Absence of trainer:

Human bodies are fragile. Incorrect exercising patterns or over-exercising can often prove to be threatening. Since our gym is free of cost, it is accessible to anybody and everybody which includes gentry that may not be educated enough in this sphere..

4. Students bunking/ skipping classes:

There is no way to ensure whether or not a student is accessing the gym in the middle of their class time.

5. Inventory management:

Our gym has facilities to provide indoor games such as chess, carrom board, etc. It is also equipped with state-of-the-art cardio and strength equipment. There is no way to ensure its management.

6. Employee duty management:

There are guards/employees on duty in the gym at all times. The duty management of these individuals is difficult.

Thus, it was only natural for us to pick this real-world problem as the topic for our project and we came up with our own GYM MANAGEMENT SYSTEM.

1.3 OBJECTIVE OF THE PROJECT

- With a digital system, the reliability of the data increases.
- The students can be notified about the strength inside the gym and even alert them when they are exceeding the time limit of gym usage, with the help of a digital gym database management system.
- The digital system can analyse their training patterns through data and give them suggestions and general advice.
- With a digital system, their time slot in the gym can be compared with their time table and an alert can be provided when such a situation is observed.
- Management of inventory will become much easier with a digitized system.
- The management of the duty slots of employees will become convenient with a digitized gym management system.

1.4 REQUIREMENT ANALYSIS

Conducting a comprehensive requirement analysis, we surveyed students and consulted the Director of Physical Education at NSUT. These interactions provided valuable insights into user expectations, equipment needs, and operational challenges, shaping the foundation for the development of an effective Gym Management System tailored to our college's unique requirements.

SURVEY QUESTION: “What are your thoughts on the digitization of the college gym? This is for a project.”



“I think your idea is very futuristic indeed. Also, considering that this is a university of “technology”, it only makes sense for the gym to be digitized. As a female, I think I would feel comfortable and more secure knowing that everyone in the gym has their identity verified and nobody is using another person’s name. It just makes it easier to find the person responsible in case of a mishap. Furthermore, the administration and management sort of becomes easier. Best wishes for the project guys!”

-Tiya Chadha, EE-1 (2022UEE4568)

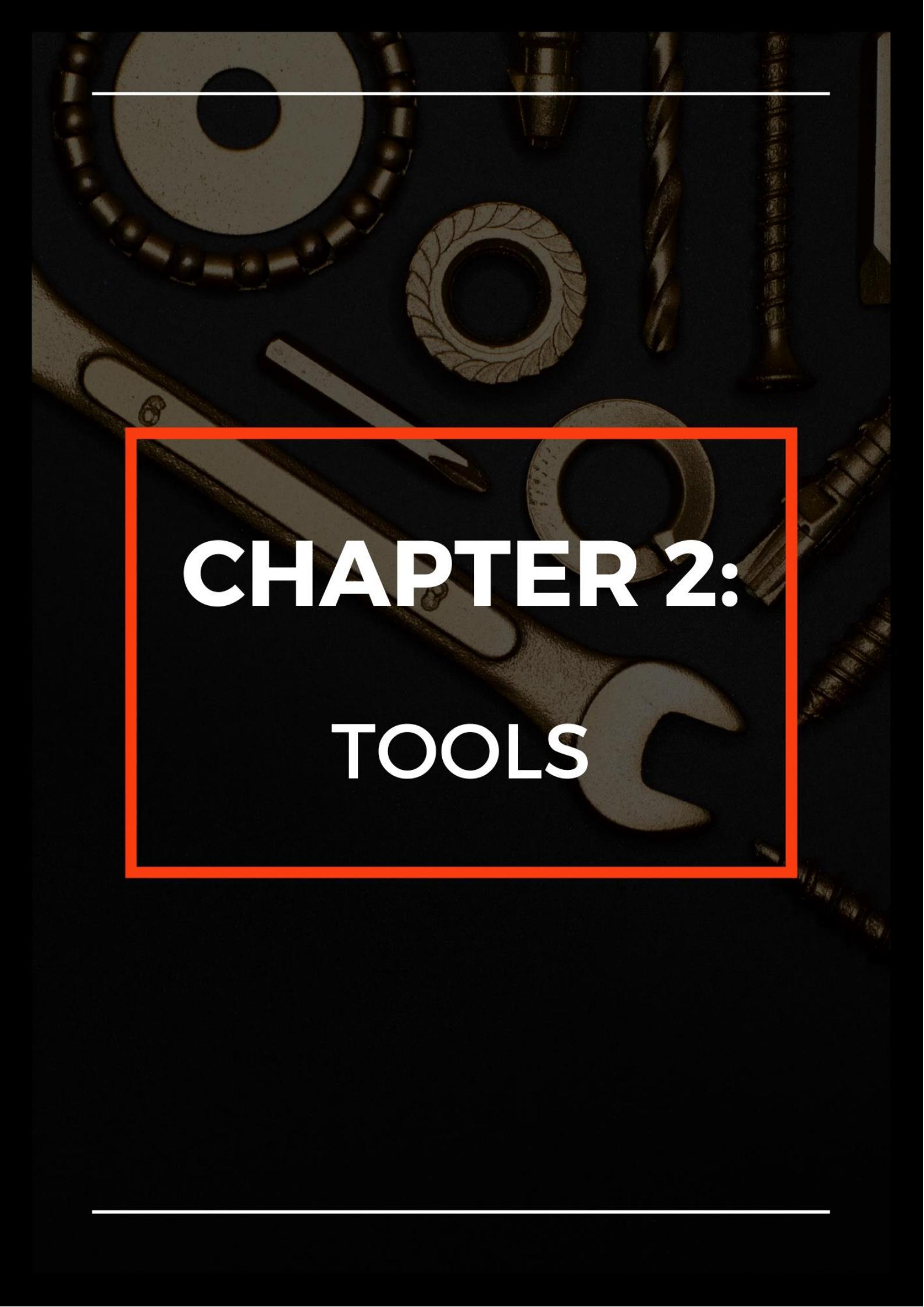
“My major issue is with how much the gym is crowded, especially in the evening. If that can somehow be managed, I will stop getting any second thoughts about finding a private gym outside college. Good luck with your vision!”

-Harshit Kaushik, IT-1 (2022UIT3010)



"Your idea is very good. Now that I am in my final year, I wish this would have been done sooner, considering how we are in the middle of a digital revolution. This is the era where manual data management should be a long forgotten concept. From administrative point of view, I feel this will ease the human effort and workload. Obviously this will help with data integrity, reliability, consistency, security and I can continue listing the benefits. From the student point of view, the problems that you have listed, especially crowd management, must be dealt with. However, the most appealing to me was the way you are thinking of analysing data and giving recommendations to students. Future scope is endless. All the best!"

-Rohit, CSE-1 (2020UCO1530)



CHAPTER 2: TOOLS

For the Gym Management System application, both hardware and software tools are essential to ensure seamless functionality.

2.1 HARDWARE REQUIREMENTS:

1. Computers or Tablets:

For administrative staff to manage the system and for users to access features such as scheduling and progress tracking.

2. Password-encoded entry machines:

For secure user identification of students when they want to use the gym.

2.2 SOFTWARE REQUIREMENTS:

1. Database Management System (DBMS):

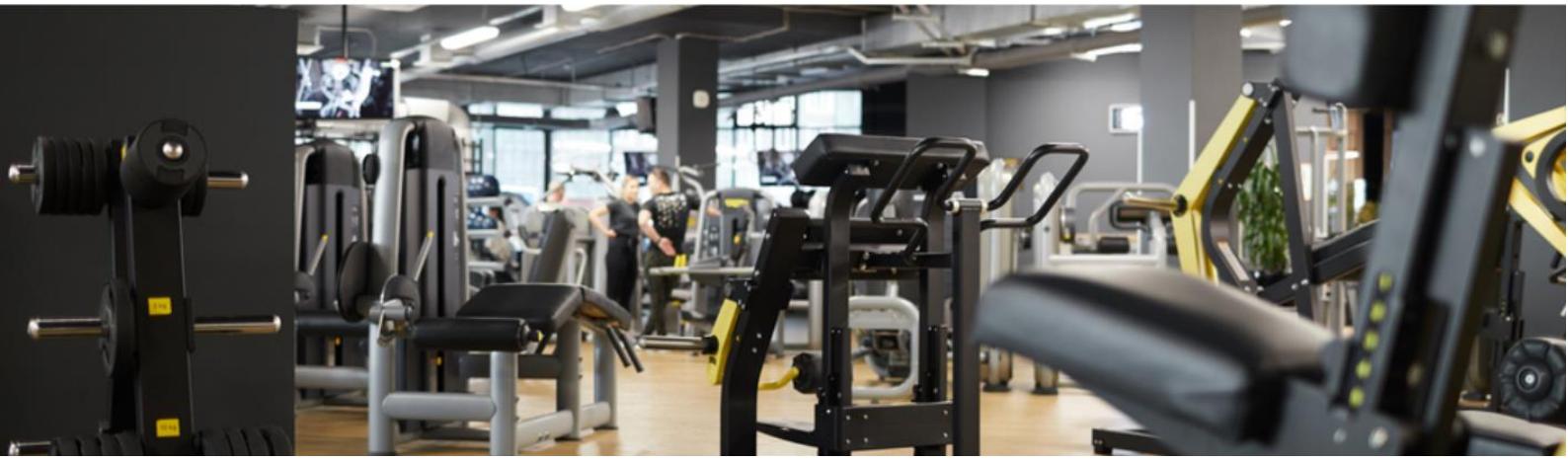
MySQL is used to efficiently store and retrieve data related to users, equipment, and schedules. And for trainers to enter their daily entry updates and their work timings.

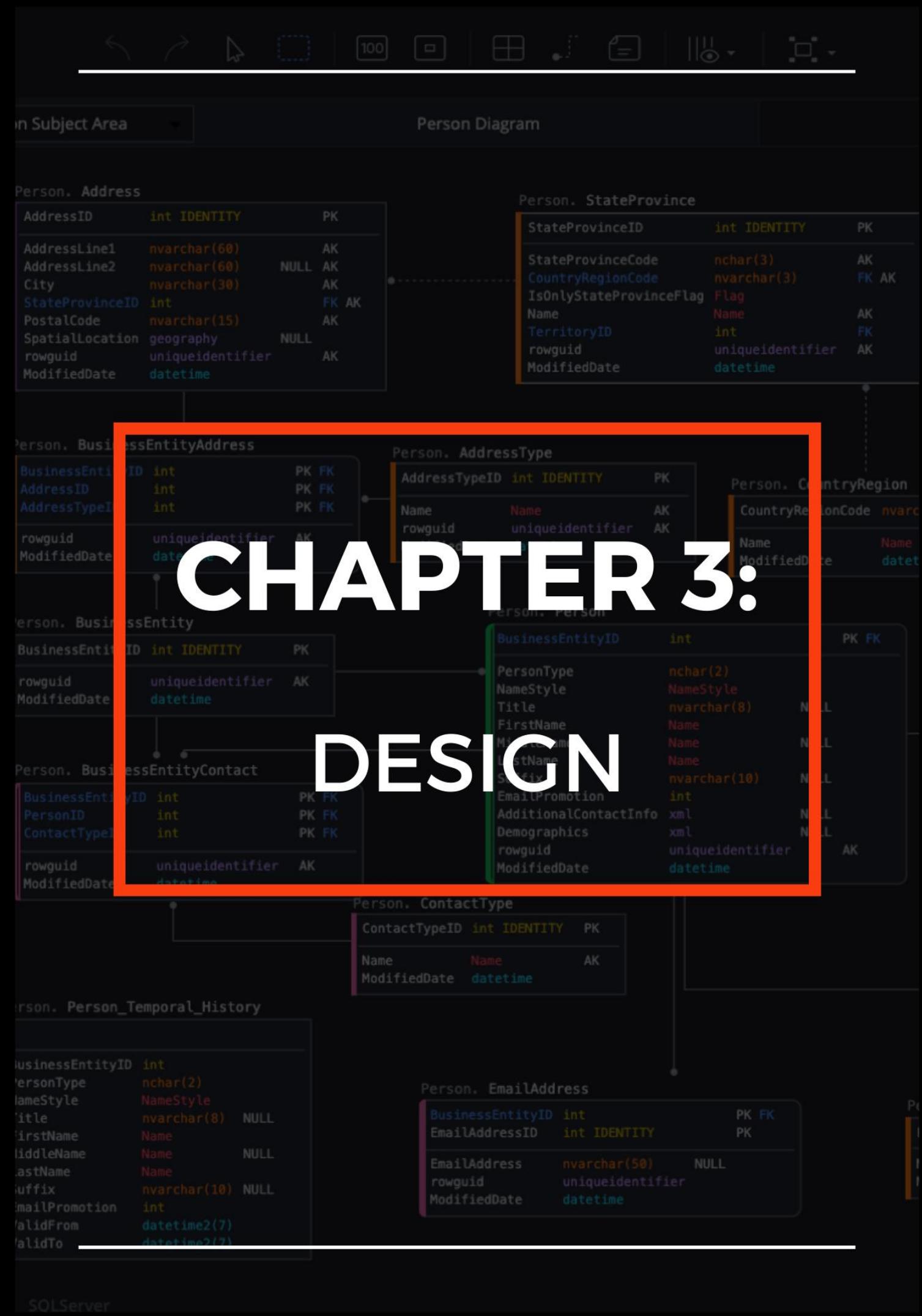
2. Website:

To build the website - HTML for framework, CSS for designing, and JavaScript for giving functionality.

3. Website connectivity:

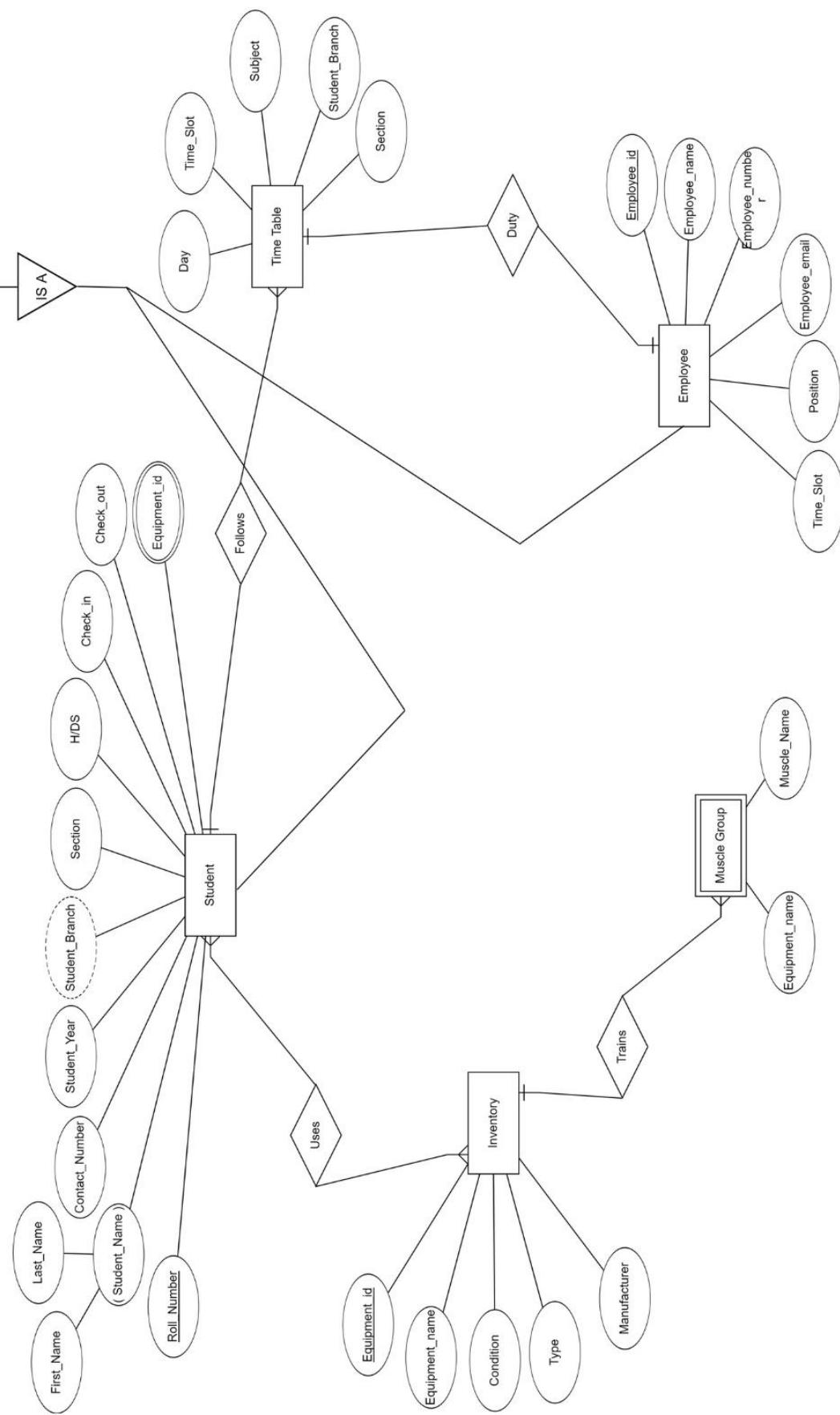
PHP is used for connecting the database to the website to display the content of the database on the website.





ER DIAGRAM

3.1





UNDERSTANDING THE ER MODEL

ENTITIES:

We have 6 entities in our database, namely:

1. User: This entity is specialised into 2 other entities, Student and Employee. It has Login_Id and Password as its attributes that determine the type of user.
2. Student: The table of this entity holds information about the students of our college who use the gym.
3. Employee: The table of this entity holds information about the staff who is put on duty in the college gym and their duty slots.
4. Inventory: The table of this entity holds information about the equipment and board games available in the gym, who has used/ borrowed anything and the condition it is in.
5. Muscle group: The table of this entity holds information about which equipment is used to train which muscle group in our body, for data analysis purposes.
6. Time Table: The table of this entity holds information about the class schedules of the students.

RELATIONSHIPS:

We have 4 relationships in our database, namely:

1. Uses: Depicts a relationship between Student and Inventory.
2. Follows: Depicts a relationship between Student and TimeTable.
3. Trains: Depicts a relationship between Inventory and MuscleGroup.
4. Duty: Depicts a relationship between TimeTable and Employee.

MAPPING CARDINALITIES:

1. Many-to-one(N:1) Relationship:

- Student to TimeTable: This relation follows a many-to-one relationship because multiple students can follow the same timetable.

- Inventory to MuscleGroup: This is a many-to-one relationship as there are multiple equipment which targets a single muscle group and a muscle group can be trained by many equipments.
2. Many-to-Many (N:M) Relationship:
- Student to Inventory: This is a many-to-many relationship. There is a relation of 'Use' between the 2 entities, which signifies that each student can use multiple equipment present in the gym and each equipment can be used by multiple students.
3. One-to-One (1:1) Relationship:
- Timetable to Employee: This is a one-to-one relationship; each timetable entry corresponds to one employee in the 'Employee' table indicating the duty assigned during that time slot.

These relationships are crucial for maintaining data integrity and establishing the connections necessary to represent the logic of the management system. They promote efficient management.

SPECIALIZATION:

The specialization of the user entity into student and employee means creating two distinct types of users, each with its unique characteristics and roles, while sharing common attributes like login ID and password. It helps better capture the unique characteristics and requirements of individuals in these different roles. It allows for more efficient data management and improved system functionality.

MULTIVALUED ATTRIBUTE:

In the gym students have the option to use multiple pieces of equipment during a single workout session. To track all the equipment they use, "equipment ID" is a multivalued attribute in the "Student" entity.

DERIVED ATTRIBUTE:

The "Student_Branch" is a derived attribute, the values of which are derived from the Roll_Number attribute of the "Student" entity. Example:

If Roll_Number = '2022UCS1574', then Student_Branch = 'CSE'

If Roll_Number= '2022UCA1543', then Student_Branch = 'CSAI'

COMPOSITE ATTRIBUTE:

The "Student_Name" is a composite attribute because it has sub-attributes "First_Name" and "Last_Name".

WEAK ENTITY SET:

A weak entity is an entity that does not have a primary key attribute of its own. Instead, it relies on a relationship with another entity, called parent entity, to give it meaning. Here in our database, we have 'MuscleGroup' as a weak entity set as it does not have a primary key of its own. It relies on 'Equipment_name' which is foreign key referencing 'Equipment_name' in 'Inventory' table. The combination of 'Equipment_name' and the corresponding entry in the Inventory table provides a unique identifier of entries in the 'MuscleGroup' table.

CONSTRAINTS:

1. Primary Key Constraint: A primary key is a unique identifier for each record in a table. In the provided schema:
 - Roll_Number is the primary key in the Student table.
 - Equipment_id is the primary key in the Inventory table.
 - Employee_id is the primary key in the Employee table.
2. Unique Constraint: A unique constraint ensures that a column or combination of columns contains unique values across all records in a table. In the schema:
 - Employee_email have unique constraints to prevent duplication of email.
 - Login_id from User table has unique constraints to prevent duplication of login id's.
3. Foreign Key Constraint: A foreign key constraint establishes a link between two tables by ensuring that values in a column in one table correspond to values in a primary key column in another table. In the schema:
 - Equipment_name in the MuscleGroup table is a foreign key referencing the Equipment table
 - .
4. Not Null Constraint: A not null constraint ensures that a column cannot contain null values, meaning it must have a value for every record. In the schema:
 - The Check_in column in the Student table are marked as not null as these values are necessary in the table and cannot be null.



ER TO RELATIONAL MODEL CONVERSION

RELATIONAL SCHEMA

```
mysql> desc student;
```

Field	Type	Null	Key	Default	Extra
Roll_Number	varchar(50)	NO	PRI	NULL	
First_Name	varchar(50)	YES		NULL	
Last_Name	varchar(50)	YES		NULL	
Student_MONO	varchar(50)	YES		NULL	
Student_Year	int	YES		NULL	
Student_Branch	varchar(50)	YES		NULL	
Section	int	YES		NULL	
HorDS	varchar(5)	YES		NULL	
Check_in	time	NO		NULL	
Check_out	time	YES		NULL	
Equipment_id	int	YES		NULL	

```
mysql> desc employee;
```

Field	Type	Null	Key	Default	Extra
Employee_id	varchar(10)	NO	PRI	NULL	
Employee_Name	varchar(50)	YES		NULL	
Employee_MONO	varchar(50)	YES		NULL	
Employee_email	varchar(50)	YES		NULL	
Position	varchar(50)	YES		NULL	
Time_Slot	varchar(50)	YES		NULL	

```
mysql> desc inventory;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| Equipment_id | int | NO | PRI | NULL |
| Equipment_name | varchar(50) | YES | | NULL |
| Equipment_Condition | varchar(50) | YES | | NULL |
| TypeofEquipment | varchar(50) | YES | | NULL |
| Manufacturer | varchar(50) | YES | | NULL |
+-----+-----+-----+-----+-----+
```

```
mysql> desc musclegroup;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| Equipment_name | varchar(50) | YES | | NULL |
| Muscle_Name | varchar(50) | YES | | NULL |
+-----+-----+-----+-----+-----+
```

```
mysql> desc timetable;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| Day | varchar(50) | YES | | NULL |
| Time_Slot | varchar(50) | YES | | NULL |
| Subject | varchar(50) | YES | | NULL |
| Student_Branch | varchar(50) | YES | | NULL |
| Section | int | YES | | NULL |
+-----+-----+-----+-----+-----+
```

```
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| Roll_Number | varchar(50) | YES | | NULL |
| Equipment_id | int | YES | | NULL |
+-----+-----+-----+-----+-----+
```



3.3 FUNCTIONAL DEPENDENCIES

Functional dependencies in a relational database describe the relationships between attributes in a table. Some functional dependencies are as follows:

1. *Student Table:*

$\text{Roll_Number} \rightarrow \text{First_Name}, \text{Last_Name}, \text{Student_MONO}, \text{Student_Year}, \text{Student_Branch}, \text{Section}, \text{HorDS}, \text{Check_in}, \text{Check_out}$

2. *Inventory Table:*

$\text{Equipment_id} \rightarrow \text{Equipment_name}, \text{Equipment_Condition}, \text{TypeofEquipment}, \text{Manufacturer}$

3. *MuscleGroup Table:*

$\{\text{Equipment_name}, \text{Muscle_Name}\} \rightarrow \{\text{Equipment_name}, \text{Muscle_Name}\}$

4. *Employee Table:*

$\text{Employee_id} \rightarrow \text{Employee_Name}, \text{Employee_MONO}, \text{Employee_email}, \text{Position}, \text{Time_Slot}$

5. *TimeTable Table:*

$\{\text{Day}, \text{Time_Slot}, \text{Student_Branch}, \text{Section}\} \rightarrow \text{Subject}$

6. *USER Table:*

$\text{Login_Id} \rightarrow \text{TypeofUser}, \text{Password}$

7. * EquipmentUsed Table:*

$\{\text{Roll_Number}, \text{Equipment_id}\} \rightarrow \{\text{Roll_Number}, \text{Equipment_id}\}$

These functional dependencies are based on the assumption that attributes on the right side of the arrow are functionally dependent on the attributes on the left side.

3.4 SPECIFY THE PRIMARY KEY

- Roll_Number is the primary key in the Student table.
- Equipment_id is the primary key in the Inventory table.
- Employee_id is the primary key in the Employee table.

3.5 SET UP THE TABLE RELATIONSHIP

1. Student Table:

Attributes: (Roll_Number (PK), First_Name, Last_Name, Student_MONO, Student_Year, Student_Branch, Section, HorDS, Check_in, Check_out)

FK: (Day, Time_Slot, Student_Branch, Section) referencing TimeTable.Day, TimeTable.Time_Slot, TimeTable.Student_Branch, TimeTable.Section respectively.

2. Inventory Table:

Attributes: (Equipment_id (PK), Equipment_name, Equipment_Condition, TypeofEquipment, Manufacturer)

3. MuscleGroup Table:

Attributes: (Equipment_name, Muscle_Name)

FK: Equipment_name referencing Inventory.Equipment_name

4. Employee Table:

Attributes: (Employee_id (PK), Employee_Name, Employee_MONO, Employee_email, Position, Time_Slot)

5. TimeTable Table:

Attributes: (Day, Time_Slot, Subject, Student_Branch, Section)

6. USER Table:

Attributes: (TypeofUser, Login_Id (PK), Password)

7. EquipmentUsed Table:

Attributes: (Roll_Number , Equipment_id)

Primary Key: (Roll_Number, Equipment_id))

FK: Roll_Number referencing Student.Roll_Number, Equipment_id referencing Inventory.Equipment_id

3.6

BCNF FORM

To check if a database schema is in the **First Normal Form (1NF)**, we need to ensure that each table meets the following conditions:

Atomic Values: Each attribute (column) should have atomic values. Atomic values are indivisible and cannot be further subdivided. This is to avoid storing multiple values in a single field.

Unique Column Names: Each column in a table must have a unique name.

No Repeating Groups: Avoid storing multiple values in a single field, as this violates the atomicity rule. Each field should contain a single value.

Let's check each table against these criteria:

1. Student Table:

Appears to be in 1NF. Each attribute seems to contain atomic values, and column names are unique.

2. Inventory Table:

Appears to be in 1NF. Each attribute contains atomic values, and column names are unique.

3. MuscleGroup Table:

Appears to be in 1NF. The attributes Equipment_name and Muscle_Name seem to contain atomic values.

4. Employee Table:

Appears to be in 1NF. Each attribute contains atomic values, and column names are unique.

5. TimeTable Table:

Appears to be in 1NF. Each attribute contains atomic values, and column names are unique.

6. USER Table:

Appears to be in 1NF. Each attribute contains atomic values, and column names are unique.

7. EquipmentUsed Table:

Appears to be in 1NF. The attributes Roll_Number and Equipment_id seem to contain atomic values.

Based on the provided information, it seems that **all the tables meet the conditions for**

the First Normal Form (1NF).

To check if a database schema is in the **Second Normal Form (2NF)**, we need to ensure that it is already in 1NF and that **no partial dependencies exist**.

Partial dependency: Proper subset of a candidate key depends on non-prime attributes. In other words, all non-prime attributes should be fully functionally dependent on candidate keys.

Let's examine each table:

1. Student Table:

Appears to be in 2NF. All attributes seem to be functionally dependent on the candidate key (Roll_Number).

2. Inventory Table:

Appears to be in 2NF. All attributes seem to be functionally dependent on the candidate key (Equipment_id).

3. MuscleGroup Table:

Since Equipment_name is part of the candidate key, it is in 2NF by definition.

4. Employee Table:

Appears to be in 2NF. All attributes seem to be functionally dependent on the candidate key (Employee_id).

5. TimeTable Table:

The attributes Day, Time_Slot, Subject, Student_Branch, and Section seem to be functionally dependent on the composite candidate key (Day, Time_Slot), but not on the entire candidate key. It may need further normalization.

- Class Table:

Attributes: (Day, Time_Slot, Subject)

Primary Key: (Day, Time_Slot)

This table stores information about classes, with the candidate key being (Day, Time_Slot). The Subject attribute is fully functionally dependent on the candidate key.

- StudentClass Table:

Attributes: (Day, Time_Slot, Student_Branch, Section)

Primary Key: (Day, Time_Slot, Student_Branch, Section)

This table represents the association between students and classes. The candidate key is (Day, Time_Slot, Student_Branch, Section). This structure allows you to associate students with specific classes.

6. USER Table:

Since Login_Id is the primary key, and the other attributes seem to be functionally

dependent on it, the table is in 2NF.

7. EquipmentUsed Table:

The attributes Roll_Number and Equipment_id form the composite primary key, and both attributes seem to be functionally dependent on this composite key. The table is in 2NF.

Now it seems that **all the tables meet the conditions for the Second Normal Form (2NF)**.

To check if the database schema is in the **Third Normal Form (3NF)**, we need to ensure that there are no transitive dependencies and that each non-prime attribute is fully functionally dependent on the candidate key.

Let's analyze each table:

1. Student Table:

Appears to be in 3NF. All non-prime attributes (attributes other than the primary key) seem to be functionally dependent on the candidate key.

2. Inventory Table:

Appears to be in 3NF. All non-prime attributes seem to be functionally dependent on the candidate key.

3. MuscleGroup Table:

The MuscleGroup Table has a foreign key (Equipment_name) referencing another table (Inventory.Equipment_name). This relationship ensures that it is in 3NF.

4. Employee Table:

Appears to be in 3NF. All non-prime attributes seem to be functionally dependent on the candidate key.

5. Class Table:

Appears to be in 3NF. All non-prime attributes seem to be functionally dependent on the candidate key.

6. StudentClass Table:

Appears to be in 3NF. All non-prime attributes seem to be functionally dependent on the candidate key.

7. USER Table:

Appears to be in 3NF. All non-prime attributes seem to be functionally dependent on the candidate key.

8. EquipmentUsed Table:

Appears to be in 3NF. All non-prime attributes seem to be functionally dependent on the

candidate key (Roll_Number, Equipment_id).

Based on the provided information, it seems that **all the tables are in the Third Normal Form (3NF)**.

To check if the given tables are in **Boyce-Codd Normal Form (BCNF)**, we need to ensure that for every non-trivial functional dependency ($X \rightarrow Y$), X is a superkey. In simpler terms, every determinant of a non-trivial functional dependency should be a candidate key.

Let's analyze each table:

1. Student Table:

Appears to be in BCNF. All non-trivial functional dependencies seem to have superkeys on the left side.

2. Inventory Table:

Appears to be in BCNF. All non-trivial functional dependencies seem to have superkeys on the left side.

3. MuscleGroup Table:

The MuscleGroup Table has a foreign key (Equipment_name) referencing another table (Inventory.Equipment_name). This relationship ensures that it is in BCNF.

4. Employee Table:

Appears to be in BCNF. All non-trivial functional dependencies seem to have superkeys on the left side.

5. Class Table:

Appears to be in BCNF. All non-trivial functional dependencies seem to have superkeys on the left side.

6. StudentClass Table:

Appears to be in BCNF. All non-trivial functional dependencies seem to have superkeys on the left side.

7. USER Table:

Appears to be in BCNF. All non-trivial functional dependencies seem to have superkeys on the left side.

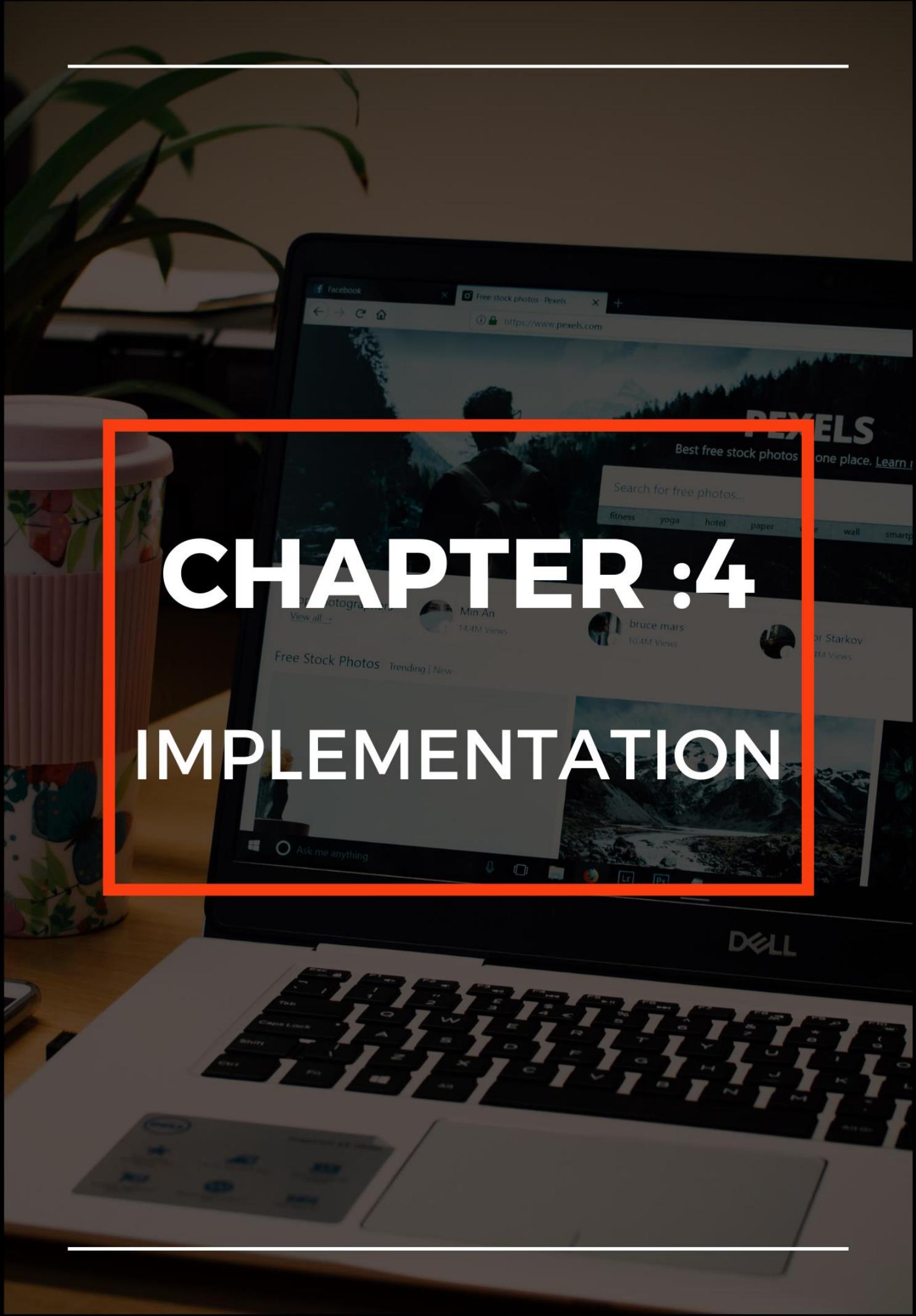
8. EquipmentUsed Table:

Appears to be in BCNF. All non-trivial functional dependencies seem to have superkeys on the left side.

Based on the provided information, it seems that **all the tables are in Boyce-Codd Normal Form (BCNF)**.

CHAPTER :4

IMPLEMENTATION



4.1 XAMPP TABLE

The screenshot shows the phpMyAdmin interface for the 'gym' database. The left sidebar lists tables: college, gym, employee, equipmentused, inventory, musclegroup, student, timetable, user, information_schema, mysql, performance_schema, phpmyadmin, and test. The 'student' table is selected. The main area displays 15 rows of student data with columns: Roll_Number, First_Name, Last_Name, Student_MONO, Student_Year, Student_Branch, Section, HorDS, Check_in, and Check_out. A 'Query results operations' section at the bottom includes links for Print, Copy to clipboard, Export, Display chart, and Create view.

Roll_Number	First_Name	Last_Name	Student_MONO	Student_Year	Student_Branch	Section	HorDS	Check_in	Check_out
2022UCA1827	Arnav	Chandwani	8816249878	2	CSAI	1 H	08:30:00	10:40:00	
2022UCA1843	Mitanish	Sharma	9343229364	2	CSAI	1 H	10:00:00	11:00:00	
2022UCA1849	Chiralityan	Yadav	9178583321	2	CSAI	1 H	10:00:00	11:00:00	
2022UCA1850	Tammay	Pawar	9213494677	2	CSAI	1 H	03:00:00	04:00:00	
2022UCA1886	Moenal	Garg	8002783309	2	CSAI	2 D	12:00:00	01:00:00	
2022UCA1915	Samarth	Garg	9900405959	2	CSAI	2 H	03:00:00	03:30:00	
2022UCS1504	Shobhit	Mallikar	9737866729	2	CSE	1 H	08:00:00	10:00:00	
2022UCS1510	Huzefa	Mulla	9289588773	2	CSE	1 DS	05:30:00	06:30:00	
2022UCS1562	Shreya	Kohli	9102394599	2	CSE	1 DS	03:00:00	04:45:00	
2022UCS1574	Arioli	Kamath	7356965508	2	CSE	1 DS	07:00:00	07:45:00	
2022UCS1574	Bhavya	Tiwari	8826777864	2	CSE	1 H	12:30:00	01:00:00	
2022UCS1580	Riyush	Lahori	8130072590	2	CSE	1 H	05:00:00	06:00:00	
2022UCS1580	Tammay	Jain	8853003878	2	CSE	2 DS	04:00:00	05:00:00	
2022UCS1800	Hritshabin	Thakur	9611529034	2	CSE	2 H	11:00:00	11:45:00	
2022UCS1900	Shivon	Jindal	7840594759	2	CSE	2 H	04:00:00	04:45:00	

The screenshot shows the phpMyAdmin interface for the 'gym' database. The left sidebar lists tables: college, gym, employee, equipmentused, inventory, musclegroup, student, timetable, user, information_schema, mysql, performance_schema, phpmyadmin, and test. The 'employee' table is selected. The main area displays 4 rows of employee data with columns: Employee_id, Employee_Name, Employee_MONO, Employee_email, Position, and Time_Slot. A 'Query results operations' section at the bottom includes links for Print, Copy to clipboard, Export, Display chart, and Create view.

Employee_id	Employee_Name	Employee_MONO	Employee_email	Position	Time_Slot
EMP1010	Suresh Chandra	9910772620	schandra@staff.nstu.ac.in	STAFF	08:00-11:00
EMP1020	Hari Kishor	9972233599	hkishor@staff.nstu.ac.in	STAFF	03:00-05:00
EMP2010	Madan Lal	9826888064	mlal@security.nstu.ac.in	GUARD	11:00-01:00
EMP2020	Raghav Kumar	8810799095	rkumar@security.nstu.ac.in	GUARD	05:00-08:00

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=gym&table=inventory

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Showing rows 0 - 21 (22 total, Query took 0.0002 seconds.)

SELECT * FROM `inventory`

Profile Edit inline | Edit Explain SQL Create PHP code Refresh

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

	Equipment_id	Equipment_name	Equipment_Condition	TypeofEquipment	Manufacturer
<input type="checkbox"/>	1001	LEG EXTENTION	WORKING	GYM	FLEXNEST
<input type="checkbox"/>	1002	SEATED CABLE ROW	WORKING	GYM	MS
<input type="checkbox"/>	1003	SHOULDER PRESS	WORKING	GYM	FITNESS WORLD
<input type="checkbox"/>	1004	CHEST PRESS	WORKING	GYM	HEALTHLINE
<input type="checkbox"/>	1005	LAT PULLDOWN	WORKING	GYM	MS
<input type="checkbox"/>	1006	CHEST FLY	NOT-WORKING	GYM	HEALTHLINE
<input type="checkbox"/>	1007	HAMSTRING CURL	WORKING	GYM	FLEXNEST
<input type="checkbox"/>	1008	HAMSTRING CURL	NOT-WORKING	GYM	FLEXNEST
<input type="checkbox"/>	1009	INCLINE BENCH	WORKING	GYM	HEALTHLINE
<input type="checkbox"/>	1010	FLAT BENCH	WORKING	GYM	HEALTHLINE
<input type="checkbox"/>	1011	LAT PULLDOWN	NOT-WORKING	GYM	MS
<input type="checkbox"/>	1012	SQUAT RACK	WORKING	GYM	MS
<input type="checkbox"/>	1013	5KG DUMBBELL	WORKING	GYM	BULLROCK
<input type="checkbox"/>	1014	10KG DUMBBELL	WORKING	GYM	BULLROCK
<input type="checkbox"/>	1015	15KG DUMBBELL	WORKING	GYM	BULLROCK
<input type="checkbox"/>	1016	20KG DUMBBELL	WORKING	GYM	BULLROCK
<input type="checkbox"/>	1017	CARAMBOARD	WORKING	GAMES	VOLATILITY
<input type="checkbox"/>	1018	CARAMBOARD	NOT-WORKING	GAMES	VOLATILITY
<input type="checkbox"/>	1019	CARAMBOARD	WORKING	GAMES	VOLATILITY
<input type="checkbox"/>	1020	CHESS	WORKING	GAMES	MTM WOOD
<input type="checkbox"/>	1021	CHESS	WORKING	GAMES	MTM WOOD
<input type="checkbox"/>	1022	CHESS	WORKING	GAMES	MTM WOOD

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=gym&table=musclegroup

Browse Structure SQL Search Insert Export Import Privileges Operations Triggers

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Showing rows 0 - 10 (11 total, Query took 0.0002 seconds.)

SELECT * FROM `musclegroup`

Profile Edit inline | Edit Explain SQL Create PHP code Refresh

Show all Number of rows: 25 Filter rows: Search this table

Equipment_name	Muscle_Name
LAT PULLDOWN	BACK
SEATED CABLE ROW	BACK
SHOULDER PRESS	SHOULDER
CHEST PRESS	CHEST
CHEST FLY	CHEST
FLAT BENCH	CHEST
INCLINE BENCH	CHEST
LEG EXTENTION	LEG
HAMSTRING CURL	LEG
SQUAT RACK	LEG
SMITH-MACHINE	LEG

Show all Number of rows: 25 Filter rows: Search this table

Query results operations

Print Copy to clipboard Export Display chart Create view

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=gym&table=timetable

Showing rows 0 - 32 (32 total, Query took 0.0002 seconds.)

SELECT * FROM `timetable`

Profile [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Day	Time_Slot	Subject	Student_Branch	Section
Monday	08:00-09:00	Web Technology	CSAI	1
Monday	09:00-10:00	MPMC	CSAI	1
Monday	10:00-12:00	CAO TUT	CSAI	1
Monday	12:00-01:00	DAA	CSAI	1
Monday	01:00-02:00	DBMS	CSAI	1
Monday	02:00-03:00	NULL	CSAI	1
Monday	03:00-04:00	CAO	CSAI	1
Monday	04:00-06:00	NULL	CSAI	1
Monday	08:00-09:00	Web Technology	CSAI	2
Monday	09:00-10:00	DAA	CSAI	2
Monday	10:00-12:00	CAO TUT	CSAI	2
Monday	12:00-01:00	MPMC	CSAI	2
Monday	01:00-02:00	DBMS	CSAI	2
Monday	02:00-03:00	NULL	CSAI	2
Monday	03:00-04:00	CAO	CSAI	2
Monday	04:00-06:00	NULL	CSAI	2
Monday	08:00-09:00	Web Technology	CSE	1
Monday	09:00-10:00	MPMC	CSE	1
Monday	10:00-12:00	DBMS LAB	CSE	1
Monday	12:00-01:00	DAA	CSE	1
Monday	01:00-02:00	CAO	CSE	1
Monday	02:00-03:00	NULL	CSE	1
Monday	03:00-04:00	DBMS	CSE	1
Monday	04:00-06:00	NULL	CSE	1
Monday	08:00-09:00	Web Technology	CSE	2

localhost/phpmyadmin/index.php?route=/sql&pos=0&db=gym&table=equipmentused

Showing rows 0 - 24 (29 total, Query took 0.0002 seconds.)

SELECT * FROM `equipmentused`

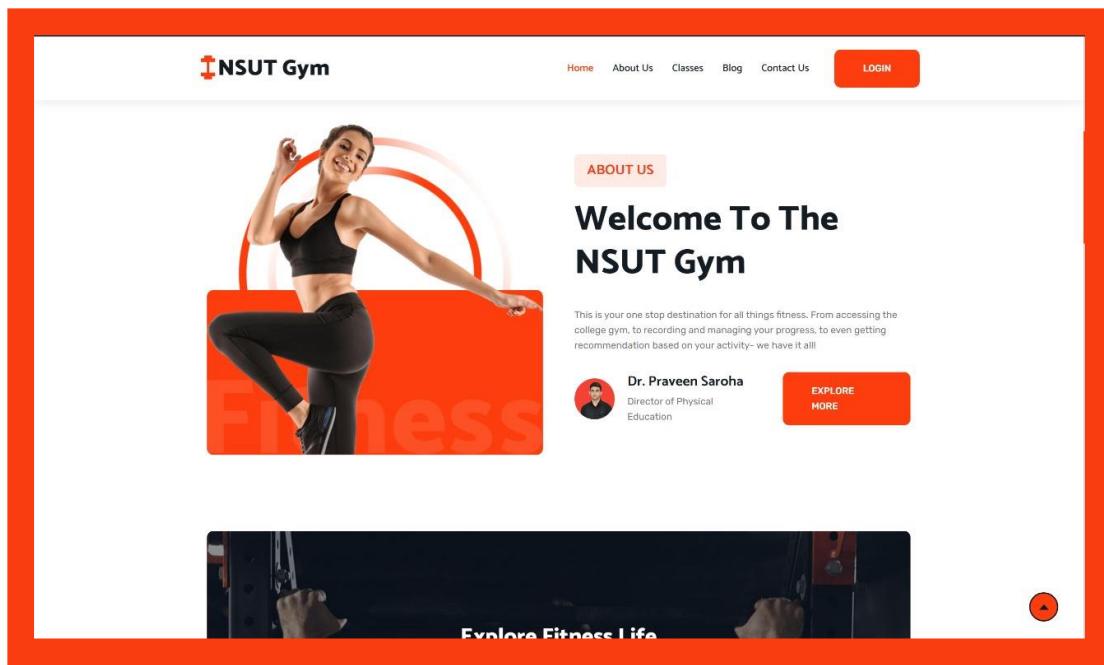
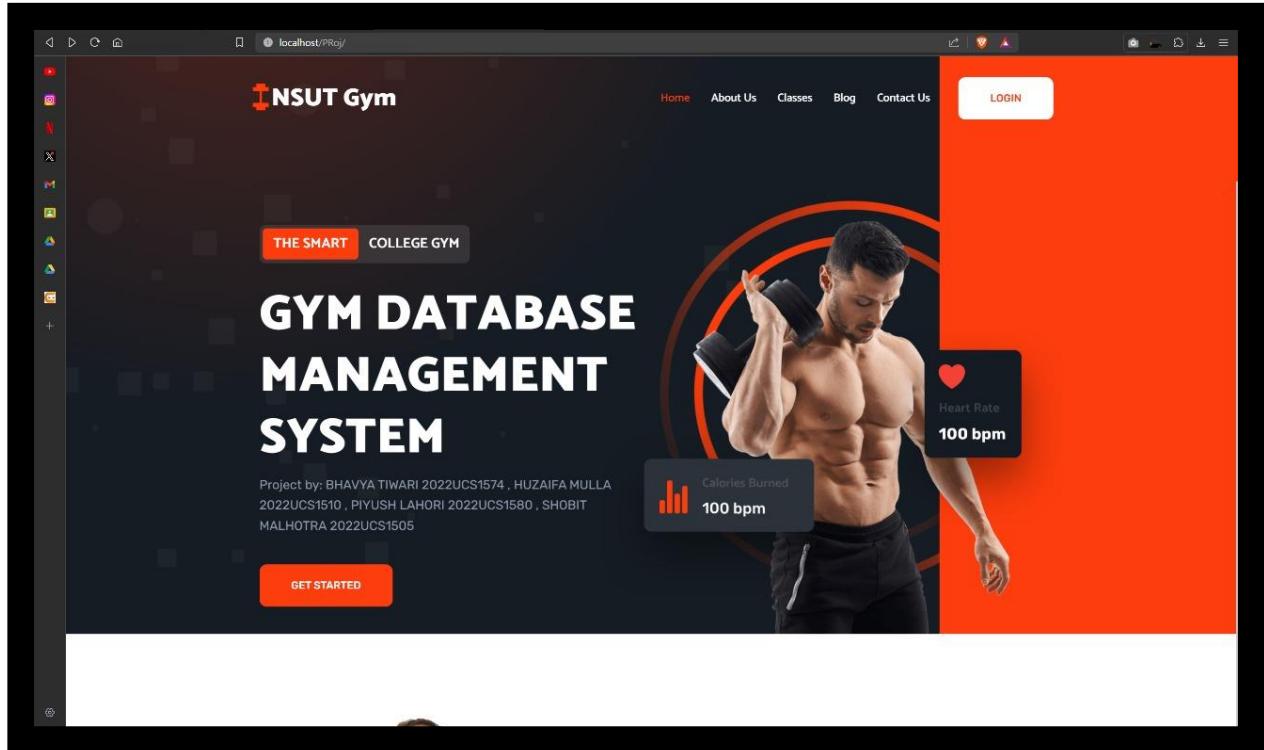
Profile [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Roll_Number	Equipment_Id
2022UCS1505	1010
2022UCS1505	1009
2022UCS1627	1003
2022UCA1827	1004
2022UCA1827	1006
2022UCA1849	1005
2022UCA1849	1004
2022UCA1843	1005
2022UCA1843	1002
2022UCS1600	1012
2022UCS1600	1001
2022UCS1600	1007
2022UCA1896	1020
2022UCS1574	1012
2022UCS1574	1001
2022UCS1562	1017
2022UCA1850	1018
2022UCA1915	1021
2022UCS1603	1010
2022UCS1603	1009
2022UCS1598	1005
2022UCS1598	1002
2022UCS1598	1003
2022UCS1580	1012
2022UCS1580	1001

4.2 APPLICATION WITH CODE

<https://github.com/v9pt/nsutgymmm>

4.3 SNAPSHOT OF RUNNING APPLICATIONS



NSUT Gym

Home About Us Classes Blog Contact Us LOGIN

OUR CLASSES

Fitness Classes For Every Goal

Weight Lifting

Weight lifting, also known as resistance training, involves lifting weights to build strength and muscle. It is a popular form of exercise that can improve overall fitness and help individuals achieve their fitness goals.

Class Full 85%

Cardio & Strength

Cardiovascular exercise, often referred to as cardio, focuses on improving the health of the heart and lungs through activities like running, cycling, and swimming. Strength training, on the other hand, aims to build muscle and increase physical strength by using resistance, such as weights or bodyweight exercises.

Class Full 70%

Power Yoga

Power yoga is a dynamic and intense form of yoga that combines traditional yoga poses with faster-paced movements and increased strength and flexibility. It offers a vigorous workout that can help improve cardiovascular fitness and promote muscle tone while also providing the mental benefits of mindfulness and stress reduction.

Class Full 90%

NSUT Gym

Home About Us Classes Blog Contact Us LOGIN

Latest Blog Feed

5 NOVEMBER 2023

Making the Gym Access Easier!

Your idea is very good. Now that I am in my final year, I wish this would have been done sooner, considering how we are in the middle of a digital revolution. This is the era where manual data management should be a long forgotten concept. From administrative point of view, I feel this will ease the human effort and workload. Obviously this will help with data integrity, reliability, consistency, security and I can continue listing the benefits. From the student point of view, the problems that you have listed, especially crowd management, must be dealt with. However, the most appealing to me was the way you are thinking of analysing data and giving recommendations to

READ MORE

5 NOVEMBER 2023

Futuristic Idea!!

"I think your idea is very futuristic indeed. Also, considering that this is a university of "technology", it only makes sense for the gym to be digitized. As a female, I think I would feel comfortable and more secure knowing that everyone in the gym has their identity verified and nobody is using another person's name. It just makes it easier to find the person responsible in case of a mishap. Furthermore, the administration and management sort of becomes easier. Best wishes for the project guys!" -Tiya Chahda, EE-1 (2022UEE4568)

READ MORE

5 NOVEMBER 2023

Solving the Crowd Problem.

"My major issue is with how much the gym is crowded, especially in the evening. If that can somehow be managed, I will stop getting any second thoughts about finding a private gym outside college. Good luck with your vision!" -Harshit Kaushik, IT-1 (2022UIT3010)

READ MORE

NSUT Gym Management System

Login

Student Login

Username:

Password:

Login

Employee Login

Username:

Password:

Login

(query – display student data)

Welcome Admin

[Not Working](#) [Employee Details](#) [Equipments In Use](#) [Muscle Groups](#) [Late](#)

[Logout](#)

Trainee Details

Roll_Number	First_Name	Last_Name	Student_MONO	Student_Year	Student_Branch	Section	HrDS	Check_in	Check_out
2022UCAI827	Arnav	Chandwani	8816249978	2	CSAI	1	DS	09:30:00	10:40:00
2022UCAI843	Mitansh	Sharma	9343229384	2	CSAI	1	H	10:00:00	11:00:00
2022UCAI849	Chaitanya	Yadav	9178563321	2	CSAI	1	H	10:00:00	11:00:00
2022UCAI850	Tanmay	Pawar	9213494677	2	CSAI	1	H	03:00:00	04:00:00
2022UCAI886	Meenal	Garg	8002783309	2	CSAI	2	DS	12:00:00	01:00:00
2022UCAI915	Samarth	Garg	9903405959	2	CSAI	2	H	03:00:00	03:30:00
2022UCS1505	Shobit	Malhotra	9737856729	2	CSE	1	H	09:00:00	10:00:00
2022UCS1510	Huzefa	Mulla	9289588773	2	CSE	1	DS	05:30:00	06:30:00
2022UCS1562	Shreya	Kohli	9102394599	2	CSE	1	DS	03:00:00	04:45:00
2022UCS1573	Arpit	Kamath	7356655506	2	CSE	1	DS	07:00:00	07:45:00
2022UCS1574	Bhavya	Tiwari	8826777864	2	CSE	1	H	12:30:00	01:00:00
2022UCS1580	Piyush	Lohori	8130072590	2	CSE	1	H	05:00:00	06:00:00
2022UCS1596	Tanmay	Jain	8853083576	2	CSE	2	DS	04:00:00	05:00:00
2022UCS1600	Hrishabh	Thakur	9611520034	2	CSE	2	H	11:00:00	11:45:00
2022UCS1603	Shiven	Jindal	7840594759	2	CSE	2	H	04:00:00	04:45:00

[Add Trainee](#)

New Register

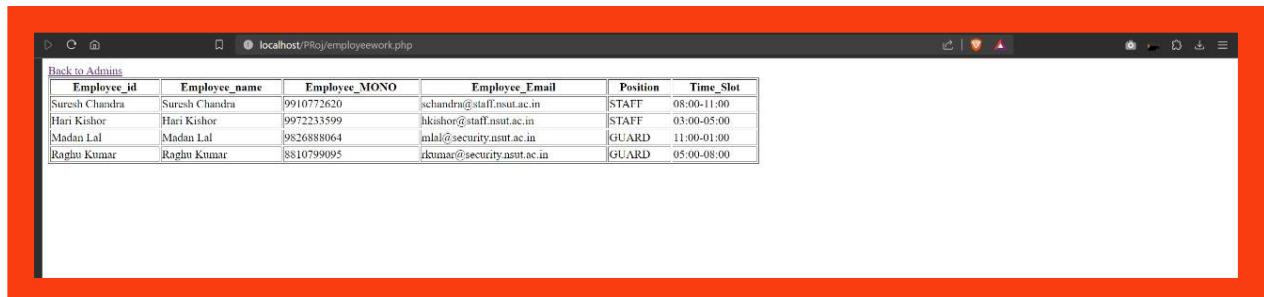
Roll_Number
 First_Name
 Student_MONO
 HrDS (H, D)
 Student_Branch
 Student_Role

(query – display all equipment not working)



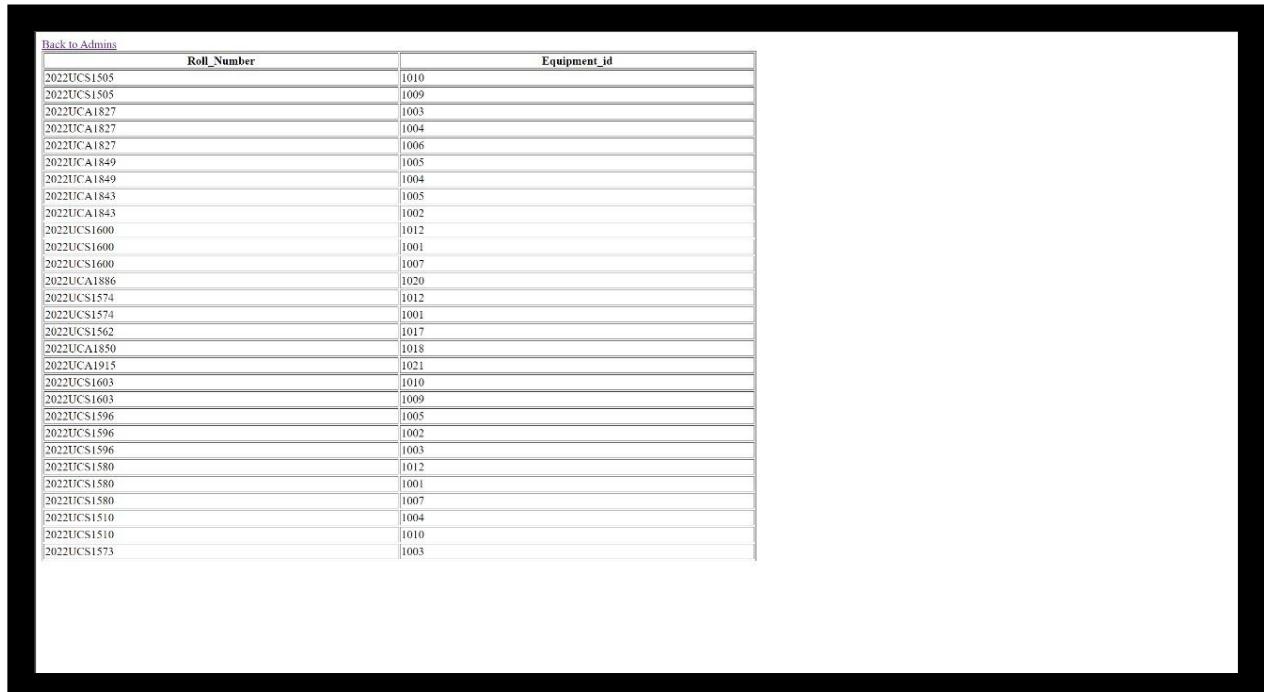
Equipment_id	Equipment_name	Equipment_Condition	TypeofEquipment	Manufacturer
1006	CHEST FLY	NOT-WORKING	GYM	HEALTHLINE
1008	HAMSTRING CURL	NOT-WORKING	GYM	FLEXNEST
1011	LAT PULLDOWN	NOT-WORKING	GYM	MS
1018	CARAMBOARD	NOT-WORKING	GAMES	VOLATILITY

(query – display employee data)



Employee_id	Employee_name	Employee_MONO	Employee_Email	Position	Time_Slot
Suresh Chandra	Suresh Chandra	9910772620	schandra@staff.nsut.ac.in	STAFF	08:00-11:00
Hari Kishor	Hari Kishor	9972233599	hkishor@staff.nsut.ac.in	STAFF	03:00-05:00
Madan Lal	Madan Lal	9826888064	mlal@security.nsut.ac.in	GUARD	11:00-01:00
Raghu Kumar	Raghu Kumar	8810799095	rkumar@security.nsut.ac.in	GUARD	05:00-08:00

(query – display which student uses which equipment)



Roll_Number	Equipment_id
2022UCS1505	1010
2022UCS1505	1009
2022UCA1827	1003
2022UCA1827	1004
2022UCA1827	1006
2022UCA1849	1005
2022UCA1849	1004
2022UCA1843	1005
2022UCA1843	1002
2022UCS1600	1012
2022UCS1600	1001
2022UCS1600	1007
2022UCA1886	1020
2022UCS1574	1012
2022UCS1574	1001
2022UCS1562	1017
2022UCA1850	1018
2022UCA1915	1021
2022UCS1603	1010
2022UCS1603	1009
2022UCS1596	1005
2022UCS1596	1002
2022UCS1596	1003
2022UICSI580	1012
2022UICSI580	1001
2022UICSI580	1007
2022UICSI510	1004
2022UCS1510	1010
2022UCS1573	1003

(query – display which equipment will train the following muscle group – back, chest, shoulder, legs)

A screenshot of a web browser window. The address bar shows 'localhost/Proj/musclegroup.html'. A dropdown menu is open under the heading 'Muscle Groups'. The options listed are 'Back', 'Chest', 'Leg', and 'Shoulder'. Each option has a small icon next to it.

Equipment_name	Muscle_Name
LAT PULLDOWN	BACK
SEATED CABLE ROW	BACK

Equipment_name	Muscle_Name
LEG EXTENTION	LEG
HAMSTRING CURL	LEG
SQUAT RACK	LEG
SMITH-MACHINE	LEG

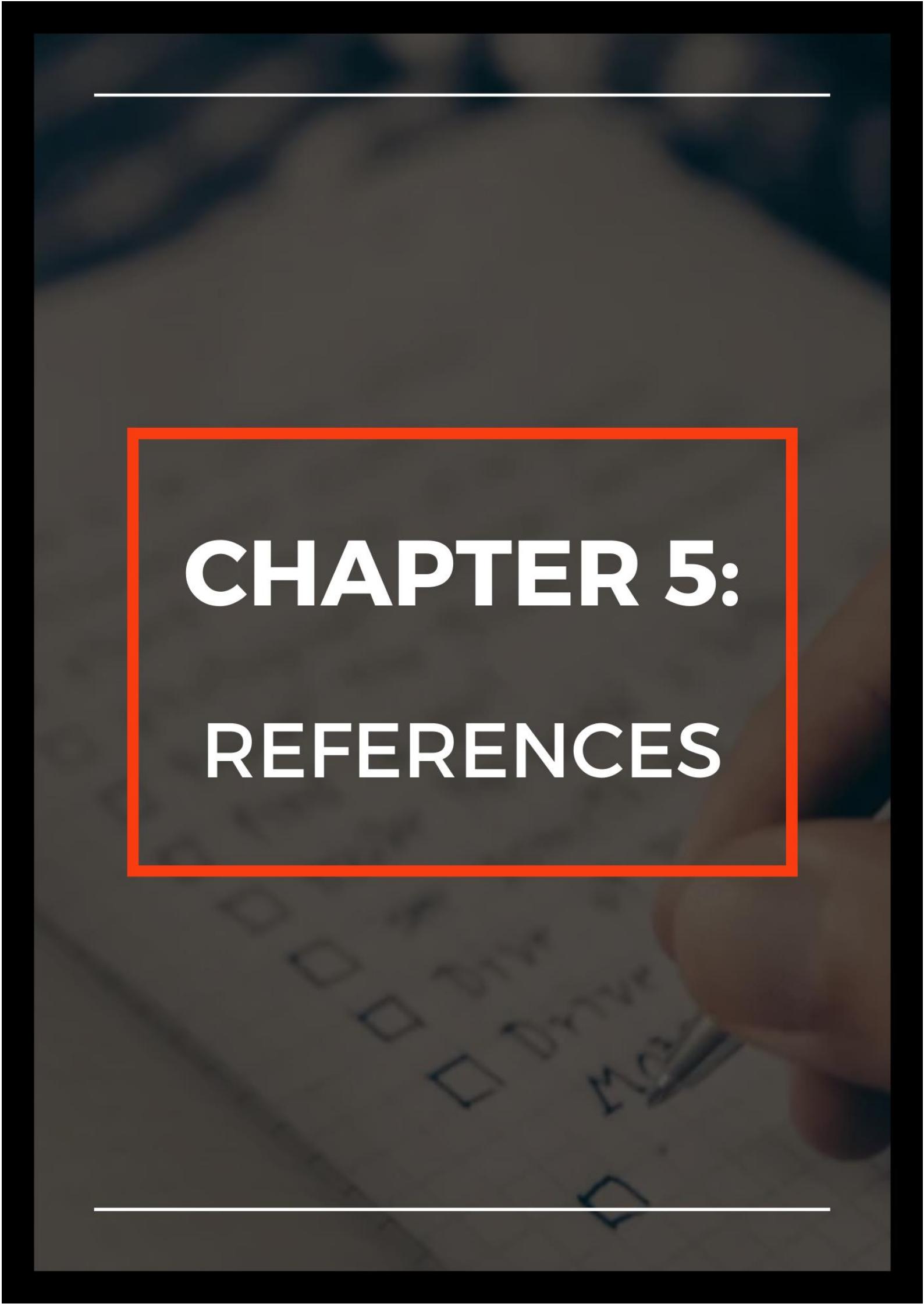
Equipment_name	Muscle_Name
SHOULDER PRESS	SHOULDER

Equipment_name	Muscle_Name
CHEST PRESS	CHEST
CHEST FLY	CHEST
FLAT BENCH	CHEST
INCLINE BENCH	CHEST

(query- display all students which have exceeded 90min gym time)

A screenshot of a web browser window. The address bar shows 'localhost/Proj/late.php'. There is a link 'Back to Admins' at the top left. Below it is a table with columns: Roll_Number, First_Name, Last_Name, Check_in, Check_out, and Duration. One row is visible: Roll_Number 2022UCS1562, First_Name Shreya, Last_Name Kohli, Check_in 03:00:00, Check_out 04:45:00, and Duration 01:45:00.

Roll_Number	First_Name	Last_Name	Check_in	Check_out	Duration
2022UCS1562	Shreya	Kohli	03:00:00	04:45:00	01:45:00



CHAPTER 5:

REFERENCES



REFERENCES

1. W3SCHOOLS
2. CODING WARRIORS
3. PROGRAMMINGWITHMOSH
4. GEEKSFORGEEKS
5. W3SCHOOLS
6. ERDPLUS
7. DATABASE SYSTEM CONCEPTS
(BOOK BY ABRAHAM SILBERSCHATZ, HENRY F. KORTH, AND S. SUDARSHAN)

THE END

SUBMITTED BY:

BHAVYA TIWARI - 2022UCS1574

PIYUSH LAHORI - 2022UCS1580

HUZAIFA MULLA - 2022UCS1510

SHOBIT MALHOTRA - 2022UCS1505