CHAPTER-1

INTRODUCTION

CHAPTER-1

INTRODUCTION

1.1 OVERVIEW OF THE PROJECT:

Our E-Fitness Membership System is a gym and health club membership management system. You can keep records on your members, their memberships, and have quick and easy communication between you and your members. E-Fitness Membership also includes a booking system, point of sale, banking, accounting, concessions and has a range of reports that help in the management of your club. Our E-Fitness Membership Software is a complete gym and recreation facility system program which looks after all of your members, memberships and activities. It is designed for gyms, recreation centers, and health clubs. This system structure is become very simple to understand because of Data Flow Diagram provided by us. Context level Diagram and Some chart are also available in this case study. The demo of using the software such as customer detail form, data base of software is also provided by us.

It provides fitness businesses the functionality to manage schedules, memberships, and facilities. The capabilities of E-Fitness Membership systems include storing member information in a database, managing financial records, scheduling classes and reserving facilities.

The E-Fitness membership system is basically a software that is designed to seamlessly integrate all the aspects of your business, allowing you to more easily and efficiently run your facility. This system will obviously be used by you, the owner, as well as your front desk staff.

Without E-Fitness membership system, all tasks will need to be carried separately. For example, you may manage billing through one piece of software, keep track of administrative tasks somewhere else, and use a spreadsheet for financial forecasting. You may have member management information online or offline in the office. Either way, there is no centralized spot where you can find all information about your business.

CHAPTER-2

SYSTEM ANALYSIS

CHAPTER-2

SYSTEM ANALYSIS

Preliminary System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action. A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal. Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

2.1 EXISTING SYSTEM:

- In the existing system the exams are done only manually but in proposed system we have to computerize the exams using this application.
- Lack of security of data.
- More man power.
- Time consuming.
- Consumes large volume of pare work.
- Needs manual calculations.
- No direct role for the higher officials

2.2 PROPOSED SYSTEM:

- The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work.
- Security of data.
- Ensure data accuracy's.
- Proper control of the higher officials.
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.

2.3 SYSTEM REQUIREMENT SPECIFICATION

2.3.1 SOFTWARE:

Every application needs the software in which it has to be executed and a hardware the application is going to perform its function. Some application cannot run on every platforms and some applications needs some specific requirement in the software or in hardware to get operated. Let's take an example of the applications which cannot be run on every platforms like windows, android, Linux, etc. Applications made in visual basic is only

supported for the windows, one cannot access this applications from the mobile phones, etc. So, here are some hardware and software specifications which are mandatory for the application to get operated.

2.3.2 HARDWARE:

- Visual Basic has been used for developing the User Layout for the system
- MySQL Script has been used for creating all the validations and client side scripting functionality
- CSS has been used for designing the web pages of the system.

HARDWARE INTERFACE:

Processor : Intel Pentium IV or more Ram : 512 MB

Cache : 1 MB

Hard Disk : 10 recommended

CHAPTER-3

SYSTEM DESIGN

CHAPTER-3

SYSTEM DESIGN

During analysis, the focus is on what needs to be done, independent of how it is done. During design, decisions are made about how the problem will be solved, first at high level, then at increasingly detailed levels.

System design is the first design stage in which the basic approach to solving the problem is selected. During system design, the overall structure and style are decided. The system architecture is the overall organization of the system into components called subsystems. The architecture provides the context in which more detailed decisions are made in later design stages. By making high level decisions that apply to the entire system, the system designer partitions the problem into subsystems so that further work can be done by several designers working independently on different subsystems.

3.1 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

There are various kinds of methods in software design: They are as follows:

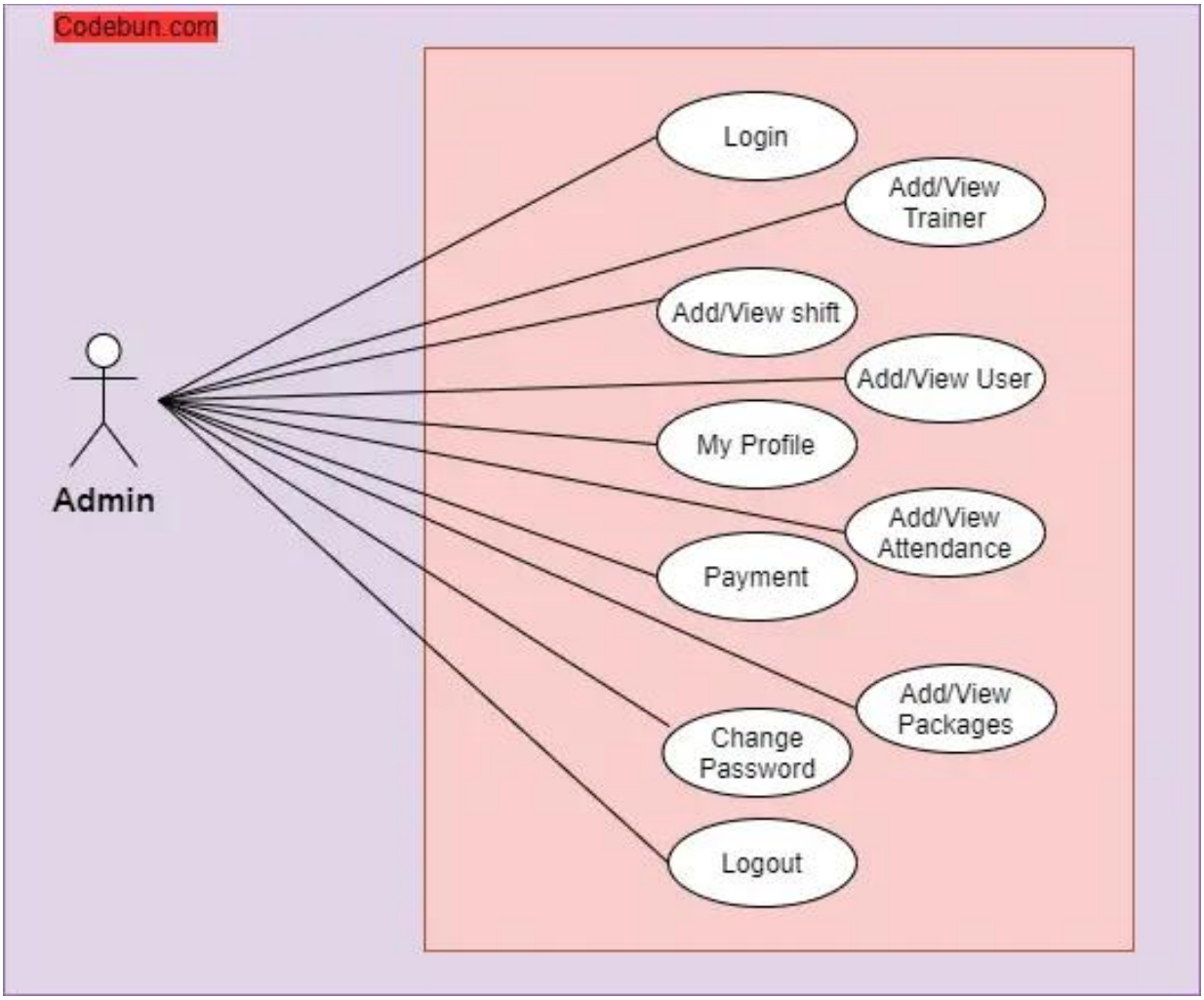
1. Use case Diagram
2. Sequence Diagram
3. Activity Diagram
4. Component diagram
5. Data flow diagram

3.1.1 USE CASE DAIGRAM

A UML use case diagram can create a broad, high-level view of the relationship between use cases, actors involved, and systems being performed.

As you can see from the examples below, use cases are represented by oval shapes, and the lines then show at which point an actor/user participates and interacts with their corresponding use case.

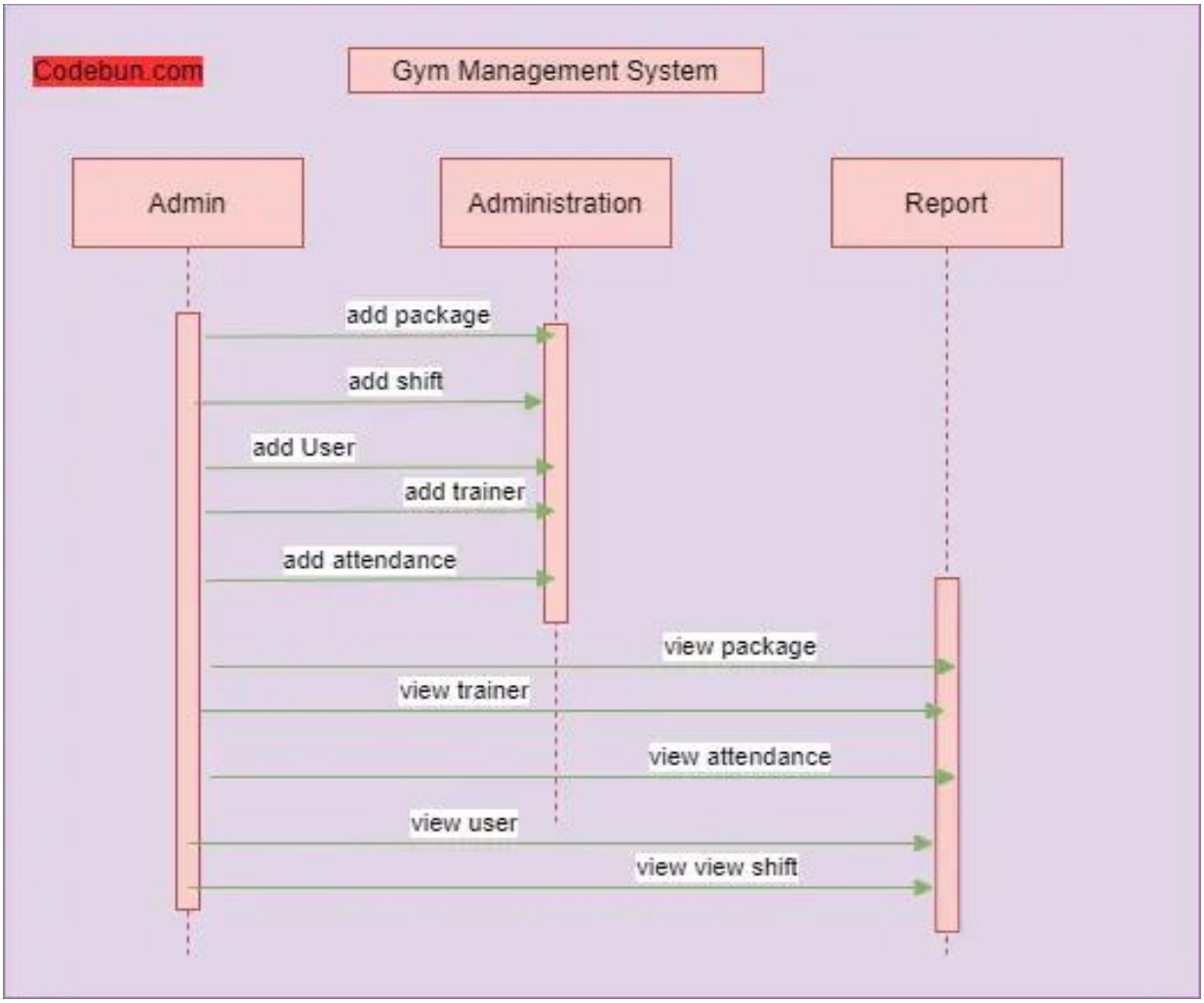
Since this is a single-user project, Admin is the sole user and every use case can be accessed by Admin only. Admin will be responsible for adding, removing, and updating any detail related to the gym.



3.1.2 SEQUENCE DIAGRAM

Sequence diagrams in UML are used to illustrate the sequence of messages between objects in an interaction. A sequence diagram consists of a group of objects that are represented by lifelines and the messages that they exchange over time during the interaction.

Admin can Add/Remove/Update any details related to the gym like gym, trainer, and shifts, update prices of the packages, and can add attendance and mainta in gym databases. Admin can view and manage gym reports.

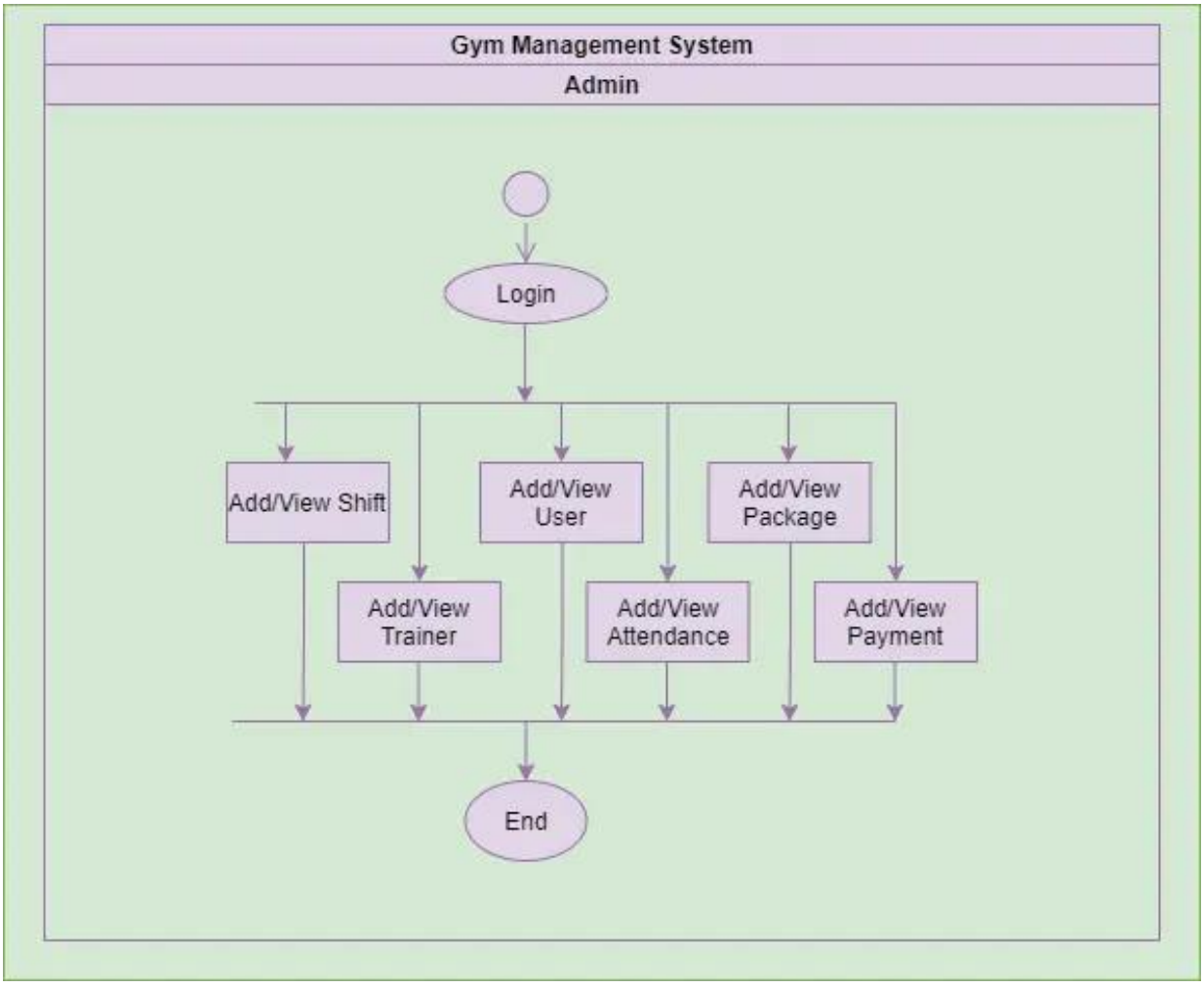


3.1.3 ACTIVITY DIAGRAM

Activity diagrams in UML display the functionalities of various activities and flow in management processes and software systems. The flow in the activity diagram can be sequential, branched, or concurrent.

Admin will have the main control over the system. Admin will be responsible for maintaining the system and also, keeping a check on trainers, and users, updating features and facilities of the centre, etc.

Admin can Add/Remove trainers, manage attendance, user details, availability, etc. Admin can Add/Remove the package with the help of this module. Admin can track the attendance activity and also, can shift timing details.

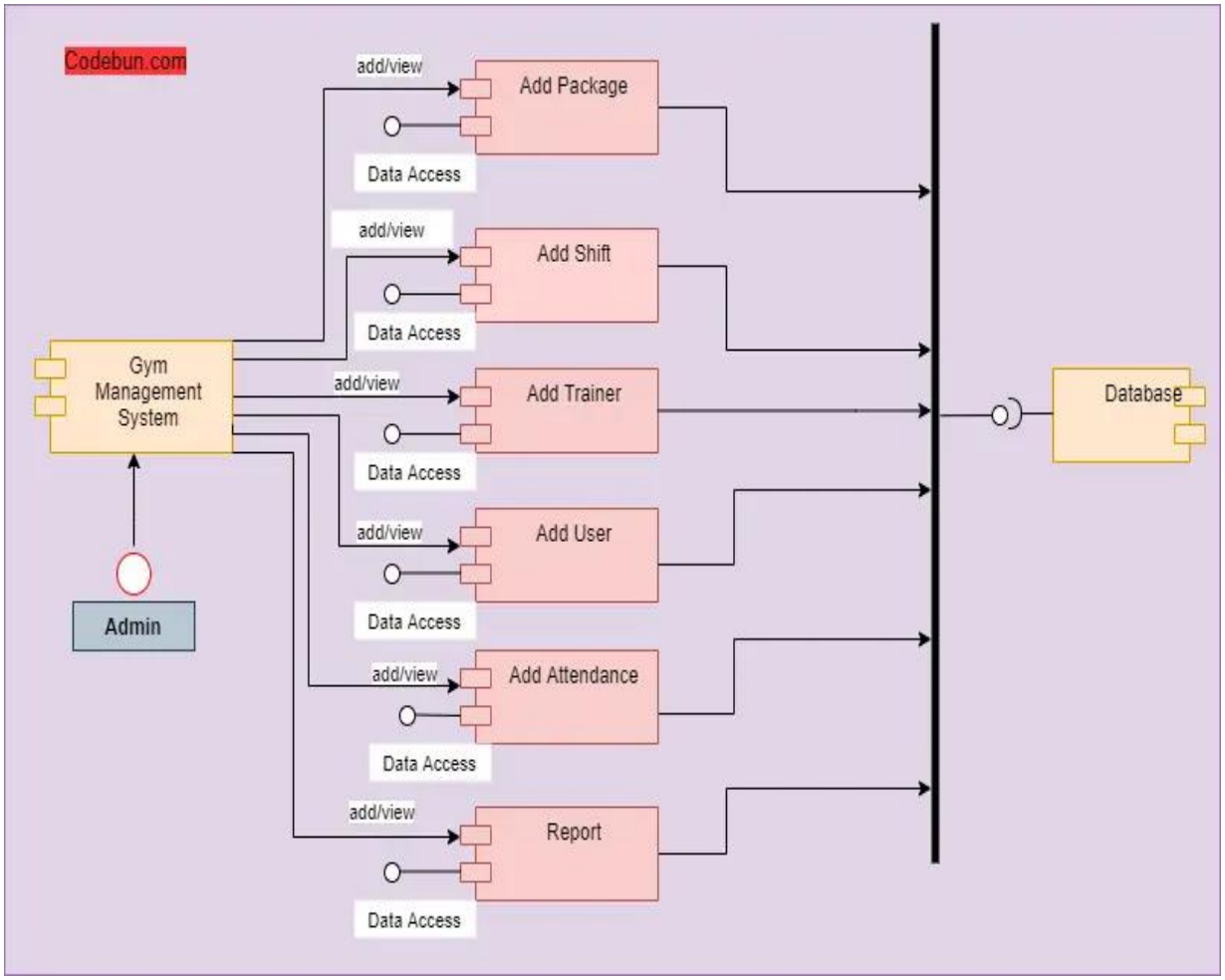


3.1.4 COMPONENT DIAGRAM

The component diagram below shows the structural relations between components in a Gym Management system. The connected components by lines represent relationships within the systems. In the diagram, it can be seen that there are components namely product, order, customer, and account.

It shows how the customer component connects to the other components while using the system. Everything from the account details to clothes booking to payment flow can be seen in the component diagram.

Admin will be responsible for maintaining the system and also, keeping a check on the products and management of the system, can add/remove/update package, trainers, shift, attendance, and check availability of trainers. Admin can add users. Admin can view user lists



CHAPTER-4

SYSTEM MODULE

CHAPTER-4

SYSTEM MODULE

4.1 USER MODULE:

The Gym Tracking System first activates the login form. Here the user enters the User name and password and our system starts the authentication process in which the username and password are matched with the existing username and password in the database. If the password matches then it is allowed to the main page else it warns the user for Invalid User name and password. After successful authentication the system activates menus. The activity log also prepared for failures and security.

4.2 PRODUCTS MODULE:

4.1.1 LIST OF PRODUCTS: After successful authentication the user is provided with the list existing products. Here the user can view the details of products and can modify the existing products. This project even provides the facility of adding new projects.

4.2.2 PRODUCT VERSIONS: All the products are maintained in several versions. As it is not possible to complete the whole project in a single version Features required for the product are categorized into several version with deadlines. And the versions are completed according to their dead line dates. Here the user can add new versions to a product or can modify the existing details of version.

4.2.3 PRODUCT USERS: In order to complete the project each product is allotted with Resources or users. First all the employees with their names and qualifications are stored in the database. Each user is allotted to the product based on their rating, Qualification and designation. For each user Effective date is stored which specifies the total period a user is valid for that product.

4.3 DETAILS MODULE:

4.3.1 GYM DETAILS: In this module the user is provided with the facility for adding Gyms or updating the existing Gyms. As the number of Gyms for a product can be very large this system is provided with efficient filtering. The user can filter the Gyms based on the priority, database, operating system and status. After the user applies filter the list of Gyms are displayed from the database.

4.3.2 GYM HISTORY: Here the Gym history is maintained. All the solutions given for the Gym resolution by various users are stored. As the Gym needs several techniques or methods for resolution it is important to store the history of the Gym.

4.3.3 GYM ASSIGNEE: This displays the list of users for whom the Gym is assigned for resolution. As the Gym need to be resolved for completing the product several user are assigned to find a solution for the Gym. The user can add this Gym to a new user or he can modify the existing user details.

4.3.4 GYM ATTACHMENTS: This gives a list of attachments for a particular Gym. The Gym can be of any type it can be a database Gym or a GUI Gym. So while you add a Gym you need to provide with the details of Gym. So the file attachments can be a document, database file or an image file. All then attachments are stored in a location along with the size and type of the file. Here the user can add anew attachment or can change the details of existing files.

4.4 TRACKING MODULE:

4.4.1 TRACK HIERARCHY: All the Gyms saved in the database will have a particular hierarchy. There might be Gyms which can be related to the earlier Gyms saved in the database so our system is provided with a hierarchy. And user can add child nodes in this hierarchy or he can modify the existing values of the nodes. This hierarchy is based on the parent child relationship between the Gyms.

4.4.2 TRACK RESOLUTION: This displays a list of all solutions provided by the users allotted to a Gym. This stores the action type and the necessary resolution provided by the user.

4.4.3 TRACK RESOURCES: This displays list of resources allotted to the project. As the Gyms need to be resolved resources are provided for the Gyms. These Resources will be the resources allotted to the project. The resources are allotted based on the rating of the employee.

4.5 SEARCH MODULE:

Our system provides with the feature of advanced search technique. Generally Number of Gyms for a project increased tremendously so if we want to know about a particular Gym It takes much amount of time. With the search screen provided one can filter the Gym's base on priority, product, severity, database and type of operating system. He can also list the Gyms between particular time based on the start date and end date. After Searching it displays a list of Gyms. From this list the user can modify the existing Gyms or can add a new Gym.

4.6 ADMIN:

4.6.1 USERS: All the users of this system are displayed in this module. One can add new user or can update the details of an existing user. Here the password provided by the user is encrypted before saving them to the database for proper security. This module saves the details like address, phone and email.

4.6.2 CONFIGURATION: All the Values that we are using in this system are configurable. Values like status, priority and others can be added dynamically on the screen. Suppose if we limit these fields by hot coding them and if the user wants to add a new value again he has to come to the developer of the product. So In order to avoid this it is

provided with the feature of adding values from the screen. And the user can change the status to In Active whenever he wants.

4.6.3 LOG VIEW: In order for the efficient Tracking of the system logs are maintained. As the logs will be in vast it will be a problem for user for checking the database. The Log View module can be searched based on the user and Records between a start date and end date.

4.7 LOGOUT MODULE:

In this once the user clicks on Log out First the session variable is killed and then the system is redirected to the login page.

CHAPTER-5

SAMPLE SOURCE CODE

CHAPTER-5

SAMPLE SOURCE CODE

```
Private Sub vkCommand1_Click()

Form3.Show

Unload Me

End Sub


Private Sub vkCommand2_Click()

Form4.Show

Unload Me

End Sub


Private Sub vkCommand3_Click()

Form6.Show

Unload Me

End Sub
```

```
Private Sub vkCommand4_Click()

Form7.Show

Unload Me

End Sub


Private Sub vkCommand5_Click()

Form8.Show

Unload Me

End Sub


Private Sub vkCommand6_Click()

Form5.Show

Unload Me

End Sub


Private Sub vkCommand7_Click()

End

End Sub
```

```
Private Sub vkCommand8_Click()
Form9.Show
Unload Me
End Sub
Private Sub vkCommand1_Click()
Form3.Show
Unload Me
End Sub

Private Sub vkCommand2_Click()
Form4.Show
Unload Me
End Sub

Private Sub vkCommand3_Click()
Form6.Show
Unload Me
```

```
End Sub


Private Sub vkCommand4_Click()

Form7.Show

Unload Me

End Sub


Private Sub vkCommand5_Click()

Form8.Show

Unload Me

End Sub


Private Sub vkCommand6_Click()

Form5.Show

Unload Me

End Sub


Private Sub vkCommand7_Click()

End
```

```
End Sub

Private Sub vkCommand8_Click()
Form9.Show
Unload Me
End SubPrivate Sub vkCommand1_Click()
Form3.Show
Unload Me
End Sub

Private Sub vkCommand2_Click()
Form4.Show
Unload Me
End Sub

Private Sub vkCommand3_Click()
Form6.Show
Unload Me
```

```
End Sub

Private Sub vkCommand4_Click()
Form7.Show
Unload Me
End Sub

Private Sub vkCommand5_Click()
Form8.Show
Unload Me
End Sub

Private Sub vkCommand6_Click()
Form5.Show
Unload Me
End Sub

Private Sub vkCommand7_Click()
```

```
End

End Sub


Private Sub vkCommand8_Click()

Form9.Show

Unload Me

End Sub

Private Sub vkCommand1_Click()

Form3.Show

Unload Me

End Sub


Private Sub vkCommand2_Click()

Form4.Show

Unload Me

End Sub


Private Sub vkCommand3_Click()
```

```
Form6.Show

Unload Me

End Sub


Private Sub vkCommand4_Click()

Form7.Show

Unload Me

End Sub


Private Sub vkCommand5_Click()

Form8.Show

Unload Me

End Sub


Private Sub vkCommand6_Click()

Form5.Show

Unload Me

End Sub
```

```
Private Sub vkCommand7_Click()

End

End Sub


Private Sub vkCommand8_Click()

Form9.Show

Unload Me

End Sub

Private Sub vkCommand1_Click()

Form3.Show

Unload Me

End Sub


Private Sub vkCommand2_Click()

Form4.Show

Unload Me

End Sub
```

```
Private Sub vkCommand3_Click()
Form6.Show
Unload Me
End Sub

Private Sub vkCommand4_Click()
Form7.Show
Unload Me
End Sub

Private Sub vkCommand5_Click()
Form8.Show
Unload Me
End Sub

Private Sub vkCommand6_Click()
Form5.Show
Unload Me
```

```
End Sub

Private Sub vkCommand7_Click()
End
End Sub

Private Sub vkCommand8_Click()
Form9.Show
Unload Me
End Sub

Private Sub vkCommand1_Click()
Form3.Show
Unload Me
End Sub

Private Sub vkCommand2_Click()
Form4.Show
Unload Me
```

```
End Sub

Private Sub vkCommand3_Click()
Form6.Show
Unload Me
End Sub

Private Sub vkCommand4_Click()
Form7.Show
Unload Me
End Sub

Private Sub vkCommand5_Click()
Form8.Show
Unload Me
End Sub

Private Sub vkCommand6_Click()
```

```
Form5.Show

Unload Me

End Sub


Private Sub vkCommand7_Click()

End

End Sub


Private Sub vkCommand8_Click()

Form9.Show

Unload Me

End Sub

Private Sub vkCommand1_Click()

Form3.Show

Unload Me

End Sub


Private Sub vkCommand2_Click()
```

```
Form4.Show

Unload Me

End Sub


Private Sub vkCommand3_Click()

Form6.Show

Unload Me

End Sub


Private Sub vkCommand4_Click()

Form7.Show

Unload Me

End Sub


Private Sub vkCommand5_Click()

Form8.Show

Unload Me

End Sub
```

```
Private Sub vkCommand6_Click()
Form5.Show
Unload Me
End Sub

Private Sub vkCommand7_Click()
End
End Sub

Private Sub vkCommand8_Click()
Form9.Show
Unload Me
End Sub
Private Sub vkCommand1_Click()
Form3.Show
Unload Me
End Sub
```

```
Private Sub vkCommand2_Click()

Form4.Show

Unload Me

End Sub


Private Sub vkCommand3_Click()

Form6.Show

Unload Me

End Sub


Private Sub vkCommand4_Click()

Form7.Show

Unload Me

End Sub


Private Sub vkCommand5_Click()

Form8.Show

Unload Me
```

```
End Sub

Private Sub vkCommand6_Click()
Form5.Show
Unload Me
End Sub

Private Sub vkCommand7_Click()
End
End Sub

Private Sub vkCommand8_Click()
Form9.Show
Unload Me
End Sub
Private Sub vkCommand1_Click()
Form3.Show
Unload Me
```

```
End Sub

Private Sub vkCommand2_Click()
Form4.Show
Unload Me
End Sub

Private Sub vkCommand3_Click()
Form6.Show
Unload Me
End Sub

Private Sub vkCommand4_Click()
Form7.Show
Unload Me
End Sub

Private Sub vkCommand5_Click()
```

```
Form8.Show

Unload Me

End Sub


Private Sub vkCommand6_Click()

Form5.Show

Unload Me

End Sub


Private Sub vkCommand7_Click()

End

End Sub


Private Sub vkCommand8_Click()

Form9.Show

Unload Me

End Sub

Private Sub vkCommand1_Click()
```

```
If vkTextBox1.Text = "" Or vkTextBox2.Text = "" Or vkTextBox3.Text = "" Or
vkTextBox4.Text = "" Or vkTextBox5.Text = "" Or vkTextBox6.Text = "" Or
vkTextBox7.Text = "" Then MsgBox "Please Enter the All Details", vbOKOnly +
vbCritical, "Information": Exit Sub


Conn.Execute ("Insert into
memberscard(MobileNO,MName,Address,Age,NoMonth,Fees,JDate) values('" &
vkTextBox1.Text & "','" & vkTextBox2.Text & "','" & vkTextBox3.Text & "','" &
vkTextBox4.Text & "','" & vkTextBox5.Text & "','" & vkTextBox6.Text & "','" &
vkTextBox7.Text & "') ON DUPLICATE KEY UPDATE
MobileNo=values(MobileNo),MName=values(MName),Address=values(Address),Age=va
lues(Age),NoMonth=values(NoMonth),Fees=values(Fees),JDate=values(JDate)")


MsgBox "Successfully Saved", vbOKOnly + vbInformation, "Information"


ClearAll


End Sub



Private Sub vkCommand2_Click()


If vkTextBox1.Text = "" Or vkTextBox2.Text = "" Or vkTextBox3.Text = "" Or
vkTextBox4.Text = "" Or vkTextBox5.Text = "" Or vkTextBox6.Text = "" Or
vkTextBox7.Text = "" Then MsgBox "Please Enter the All Details", vbOKOnly +
vbCritical, "Information": Exit Sub


Conn.Execute ("Insert into
memberscard(MobileNO,MName,Address,Age,NoMonth,Fees,JDate) values('" &
vkTextBox1.Text & "','" & vkTextBox2.Text & "','" & vkTextBox3.Text & "','" &
vkTextBox4.Text & "','" & vkTextBox5.Text & "','" & vkTextBox6.Text & "','" &
vkTextBox7.Text & "') ON DUPLICATE KEY UPDATE
MobileNo=values(MobileNo),MName=values(MName),Address=values(Address),Age=va
lues(Age),NoMonth=values(NoMonth),Fees=values(Fees),JDate=values(JDate)")


MsgBox "Successfully Updated", vbOKOnly + vbInformation, "Information"


ClearAll


End Sub
```

```
Private Sub vkCommand3_Click()

If vkTextBox1.Text = "" Or vkTextBox2.Text = "" Or vkTextBox3.Text = "" Or
vkTextBox4.Text = "" Or vkTextBox5.Text = "" Or vkTextBox6.Text = "" Or
vkTextBox7.Text = "" Then MsgBox "Please Enter the All Details", vbOKOnly +
vbCritical, "Information": Exit Sub

Conn.Execute ("Delete from memberscard where MobileNO='" & vkTextBox1.Text & "'")

MsgBox "Successfully Rejected", vbOKOnly + vbInformation, "Information"

ClearAll

End Sub


Private Sub vkCommand4_Click()

If Rs.State = 1 Then Rs.Close

Rs.Open "Select MobileNO,MName,Address,Age,NoMonth,Fees,JDate
from memberscard where MobileNo='" & vkTextBox1.Text & "'", Conn

If Rs.EOF = False Then

vkTextBox1.Text = Rs("MobileNo")

vkTextBox2.Text = Rs("MName")

vkTextBox3.Text = Rs("Address")

vkTextBox4.Text = Rs("Age")
```

```
vkTextBox5.Text  = Rs("NoMonth")

vkTextBox6.Text  = Rs("Fees")

vkTextBox7.Text  = Rs("JDate")

Else

MsgBox "Record Not Found", vbOKOnly + vbCritical, "Information"

ClearAll

End If

End Sub


Private Sub vkCommand5_Click()

Form2.Show

Unload Me

End Sub


Public Sub ClearAll()

vkTextBox1.Text  = ""

vkTextBox2.Text  = ""
```

```
vkTextBox3.Text = ""

vkTextBox4.Text = ""

vkTextBox5.Text = ""

vkTextBox6.Text = ""

vkTextBox7.Text = ""

End Sub



Private Sub vkCommand1_Click()

If vkTextBox1.Text = "" Or vkTextBox2.Text = "" Or vkTextBox3.Text = "" Or
vkTextBox4.Text = "" Then MsgBox "Please Enter the All Details", vbOKOnly +
vbCritical, "Information": Exit Sub

Conn.Execute ("Insert into mattendance(MobileNO,MName,ADate,ATime) values('" &
vkTextBox1.Text & "','" & vkTextBox2.Text & "','" & vkTextBox3.Text & "','" &
vkTextBox4.Text & "') ON DUPLICATE KEY UPDATE
MobileNo=values(MobileNo),MName=values(MName),ADate=values(ADate),ATime=val
ues(ATime)")

MsgBox "Successfully Saved", vbOKOnly + vbInformation, "Information"

ClearAll

End Sub

Public Sub ClearAll()

vkTextBox1.Text = ""
```

```
vkTextBox2.Text = ""

vkTextBox3.Text = ""

vkTextBox4.Text = ""

End Sub



Private Sub vkCommand2_Click()

If vkTextBox1.Text = "" Or vkTextBox3.Text = "" Then MsgBox "Please Enter the All Details", vbOKOnly + vbCritical, "Information": Exit Sub

Conn.Execute ("Delete from mattendance where MobileNO='" & vkTextBox1.Text & "' and ADate='" & vkTextBox3.Text & "'")

MsgBox "Successfully Removed", vbOKOnly + vbInformation, "Information"

ClearAll

End Sub



Private Sub vkCommand5_Click()

Form2.Show

Unload Me

End Sub

Private Sub vkCommand1_Click()
```

```
If vkTextBox1.Text = "" Or vkTextBox2.Text = "" Or vkTextBox3.Text = "" Or
vkTextBox4.Text = "" Or vkTextBox5.Text = "" Then MsgBox "Please Enter the All
Details", vbOKOnly + vbCritical, "Information": Exit Sub


Conn.Execute ("Insert into payment(MobileNO,MName,TFees,PaidAmt,PDate) values('"
& vkTextBox1.Text & "','" & vkTextBox2.Text & "','" & vkTextBox3.Text & "','" &
vkTextBox4.Text & "','" & vkTextBox5.Text & "') ON DUPLICATE KEY UPDATE
MobileNo=values(MobileNo),MName=values(MName),TFees=values(TFees),PaidAmt=v
alues(PaidAmt),PDate=values(PDate)")


MsgBox "Successfully Saved", vbOKOnly + vbInformation, "Information"


ClearAll


End Sub


Public Sub ClearAll()


vkTextBox1.Text = ""


vkTextBox2.Text = ""


vkTextBox3.Text = ""


vkTextBox4.Text = ""


vkTextBox5.Text = ""


End Sub



Private Sub vkCommand2_Click()


If vkTextBox1.Text = "" Or vkTextBox5.Text = "" Then MsgBox "Please Enter the
MobileNo and PaidDate Details", vbOKOnly + vbCritical, "Information": Exit Sub
```

```
Conn.Execute ("Delete from payment where MobileNO='" & vkTextBox1.Text & "' and
PDate='" & vkTextBox5.Text & "'")

MsgBox "Successfully Removed", vbOKOnly + vbInformation, "Information"

ClearAll

End Sub

Private Sub vkCommand5_Click()
```

CHAPTER-6

SYSTEM TESTING

CHAPTER-6

SYSTEM TESTING

6.1 SECURITY TESTING OF THE PROJECT:

Testing is vital for the success of any software. no system design is ever perfect. Testing is also carried in two phases. first phase is during the software engineering that is during the module creation. second phase is after the completion of software. this is system testing which verifies that the whole set of programs hanged together.

6.2 WHITE BOX TESTING:

In this technique, the close examination of the logical parts through the software are tested by cases that exercise species sets of conditions or loops. all logical parts of the software checked once. errors that can be corrected using this technique are typographical errors, logical expressions which should be executed once may be getting executed more than once and error resulting by using wrong controls and loops. When the box testing tests all the independent part within a module a logical decisions on their true and the false side are exercised , all loops and bounds within their operational bounds were exercised and internal data structure to ensure their validity were exercised once.

6.3 BLACK BOX TESTING:

This method enables the software engineer to device sets of input techniques that fully exercise all functional requirements for a program. black box testing tests the input, the output and the external data. it checks whether the input data is correct and whether we are getting the desired output.

6.4 ALPHA TESTING:

Acceptance testing is also sometimes called alpha testing. Be spoke systems are developed for a single customer. The alpha testing proceeds until the system developer and the customer agree that the provided system is an acceptable implementation of the system requirements.

6.5 BETA TESTING:

On the other hand, when a system is to be marked as a software product, another process called beta testing is often conducted. During beta testing, a system is delivered among a number of potential users who agree to use it. The customers then report problems to the developers. This provides the product for real use and detects errors which may not have been anticipated by the system developers.

6.6 UNIT TESTING:

Each module is considered independently. it focuses on each unit of software asimplemented in the source code. it is white box testing.

6.7 INTEGRATION TESTING:

Integration testing aims at constructing the program structure while at the same constructing tests to uncover errors associated with interfacing the modules. modules are integrated by using the top down approach.

6.8 VALIDATION TESTING:

Validation testing was performed to ensure that all the functional and performance requirements are met.

6.9 SYSTEM TESTING

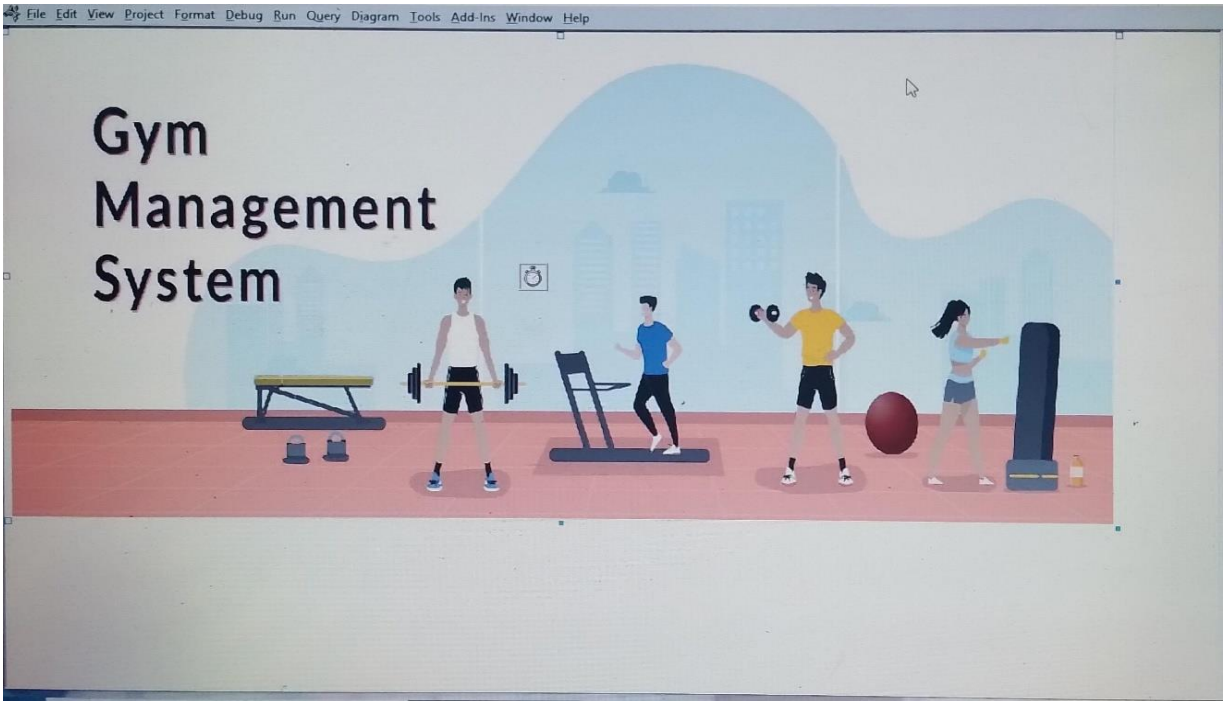
It is executing programs to check logical changes made in it with intention of finding errors. a system is tested for online response, volume of transaction, recovery from failure etc. System testing is done to ensure that the system satisfies all

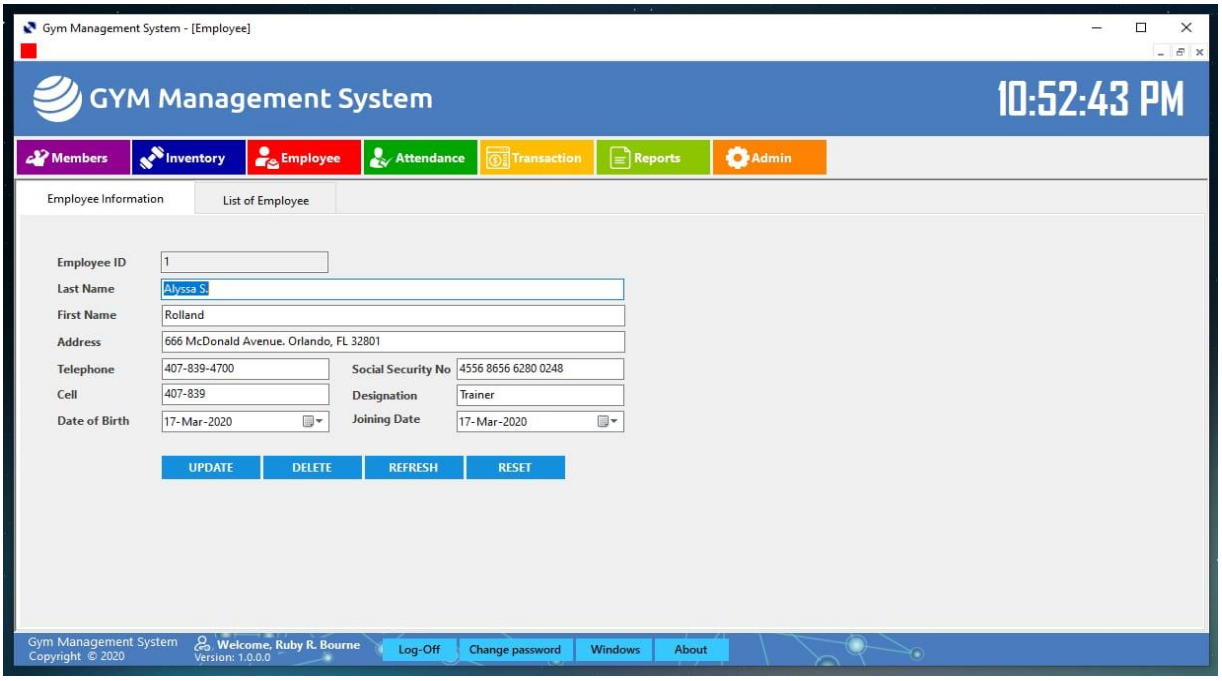
CHAPTER-7

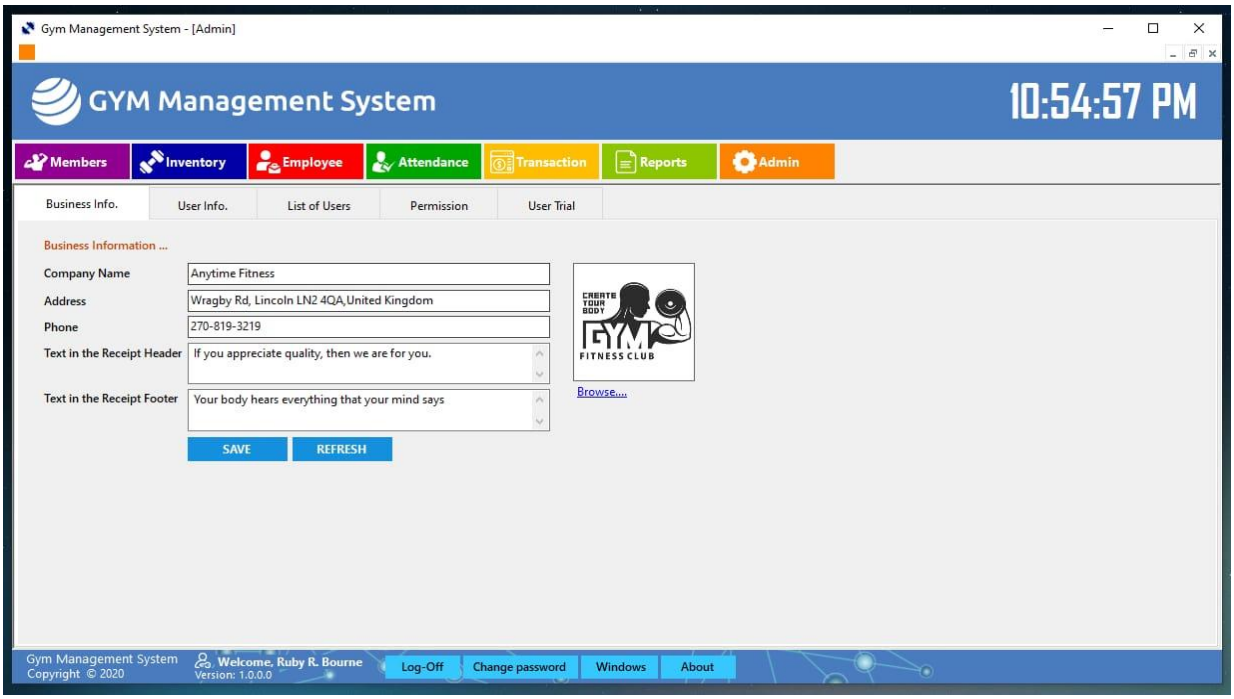
SAMPLE SCREEN LAYOUT

CHAPTER-7

SAMPLE SCREEN LAYOUT




Gym Management System - [Employee]
GYM Management System
10:52:43 PM
Members
Inventory
Employee
Attendance
Transaction
Reports
Admin
Employee Information
List of Employee
Employee ID
1
Last Name
Alyssa S.
First Name
Rolland
Address
666 McDonald Avenue. Orlando, FL 32801
Telephone
407-839-4700
Social Security No
4556 8656 6280 0248
Cell
407-839
Designation
Trainer
Date of Birth
17-Mar-2020
Joining Date
17-Mar-2020
UPDATE
DELETE
REFRESH
RESET
Gym Management System
Copyright © 2020
Welcome, Ruby R. Bourne
Version: 1.0.0.0
Log-Off
Change password
Windows
About


Gym Management System - [Admin]
GYM Management System
10:54:57 PM
Members
Inventory
Employee
Attendance
Transaction
Reports
Admin
Business Info.
User Info.
List of Users
Permission
User Trial
Business Information ...
Company Name
Anytime Fitness
Address
Wragby Rd, Lincoln LN2 4QA,United Kingdom
Phone
270-819-3219
Text in the Receipt Header
If you appreciate quality, then we are for you.
Text in the Receipt Footer
Your body hears everything that your mind says
SAVE
REFRESH
Browse...
Gym Management System
Copyright © 2020
Welcome, Ruby R. Bourne
Version: 1.0.0.0
Log-Off
Change password
Windows
About

CHAPTER-9

CONCLUSION

CHAPTER-9

CONCLUSION

conclusion, the gym management system project is a software application that helps gym owners manage their operations more efficiently. The system provides features such as member management, workout scheduling, equipment tracking, billing and payment processing, and more.

The system is designed to be user-friendly, customizable, and scalable. It can be used by small, medium, or large-sized gyms, and can be customized to meet the specific needs of each gym. The system is also designed to be secure, ensuring that member data is protected at all times.

Overall, the gym management system project is an excellent solution for gym owners who want to streamline their operations and improve their bottom line. With its user-friendly interface, customizable features, and robust functionality, it is sure to be a valuable asset to any gym that chooses to implement it

CHAPTER-10

FUTURE ENHANCEMENT

CHAPTER-10

FUTURE ENHANCEMENT

The health and fitness industry has become extremely diverse in the range of services and facilities it offers; varying from large scale leisure centres and gymnasiums, to individual personal trainers who travel from one client to the next in their cars., Some services specialize in offering structured classes, others are informal; some cater for a particular demographic, and others for anyone.

10.1 CLASSES FOR MATURE OLDER PEOPLE:

This group of people can make up a great proportion of your clientèle. They generally do not like loud music, so keep it soft or at a medium level. Most also prefer low-key surroundings so avoid bright lights, mirrors, or posters of super-fit young people for example. Older participants are often more concerned about joint movement and flexibility than weight loss and can relate better to an older instructor who shows respect and individual attention to their requirements. It is very common that this type of population attends for social reasons - to meet others and relax.

10.2 FOR PROFESSIONAL ATHLETES OR COMPETITIVE SPORTS PEOPLE

This can be a hard group to cater for because they can vary considerably in the type of sports they are involved in, and elite athletes may require very specialised fitness components. The triathlete, swimmer, runner or aerobics-orientated person will want a hard, advanced class to challenge their fitness.

The type of sport and the fitness components to be improved will indicate the type of training employed. You may find elite athletes require personal training, one-on-one, to best achieve their goals.

Personal training has become more and more popular over the past couple of decades, but it is expensive, and so tends to be limited to those who can afford it. However, it is a great way to increase motivation and to improve knowledge and technique with workouts. Personal trainers usually have up-to- date knowledge on

the latest techniques and research done in the fitness industry. For a hefty fee, some companies offer a live-in personal training service for several months whereby the trainer lives with the exerciser and takes control of what they eat and how and when they exercise. Other services might involve a trainer accompanying an individual at their place of employment to oversee their diet and exercise

10.3 FOR BEGINNERS:

These type of clients can generally be catered for a little easier than other groups. Generally, their cardiovascular fitness needs to be increased, and their basic fitness components of strength, flexibility and muscular endurance can be improved as well.

Attending classes with easy-to-follow fitness routines will allow beginners to keep up and enhance their skills almost immediately. A basic low-impact class with a caring and understanding instructor will be a good start to any beginners fitness

CHAPTER-11

BIBLIOGRAPHY

CHAPTER-11

BIBLIOGRAPHY

REFERENCE:

1. www.wikipedia.org
2. www.codeproject.com
3. Visual Basic 6.0 Resources Center http://madness.microsoft.com/hi-in/vbrun/ default (en-us).aspx
4. Visual Basic 6 Black Book By Steven Holzner
5. management system.

IEEE PAPERS:

1. Minghui Liu, Erxiang Chen, "Research and Design on Library Management System Based on Struts and Hibernate Framework," WASE International Conference on Information Engineering, ICIE, 2009.
    a. V. Dinesh Kumar, K Bhargav Ram Rayal, M. Saraswathi, "Smart Gym Management System," IJSRET, Trends Volume 6, Issue 3, May-June-2020.
2. Kavita Gupta, "Biometric Authentication" International Journal of Engineering Research Technology (IJERT), VIMPACT 2017.
3. Mr Akshay Sambare, Dipali Bondre, Sachin Thorat, Miss Archana Vishe, Prof. Ankit Sanghavi "Gym Monitoring Framework for
4. Fitness Management System," International
5. Journal of Advanced Research in Computer and Communication Engineering Vol. 6, Issue 3, March 2017.
6. Jing Yang, Bei Jing, Leixiao Li, Yan Zhao 'Combined Framework of Struts and Hibernate' Combined Framework of Struts and
7. Hibernate," IEEE-2011.
8. Muhammad Adul Shakoor, Muhammad Abbas, Muhammad Irfan Mehdi, Sajjad Hussain,
9. Ashraf Ali "DATABASE AND TRANSACTIONS MANAGEMENT SYSTEM FOR A
10. SMART GYM," Sci-Int, 2018.

11. Kasliwal Mahima, Raundal Pooja, Wagh Niyati, G. M. Lodha, “Gym Management System”. HBRP -Publication, 2019.
12. E. O. Badmus, O. P. Odekunle, and D. O. Oyewobi, “Smart Fingerprint Biometric and RFID Time-Based Attendance Management System”. Vol 5, Issue 4, EJECE, July 2021.
13. Donald Brown, Chad Michael Davis, and Scott Stan, “Struts 2 in Action”, ISBN: 193398807X, May, 2008.
14. James Holmes, “Struts 2: The Complete Reference”, ISBN-10: 0071489908, October 24, 2008.
15. Christian Bauer and Gavin King, “Hibernate in Action”, ISBN: 193239415X, 2004.
16. Jeff Linwood and Dave Minter, “Beginning Hibernate”, ISBN: 187647534X, Apr, 2007.
17. Jim Keogh, “j2ee: the complete reference”,