

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
JNANASANGAMA, BELAGAVI - 590018



**Mini Project Report
on
ONLINE DJ MANAGEMENT SYSTEM**

Submitted in partial fulfillment for the award of degree of

**Bachelor of Engineering
in
COMPUTER SCIENCE AND ENGINEERING**

Submitted by
NAGAJYOTHI MS
1BG18CS066



Vidyayāmruthamashnute

B.N. Institute of Technology

Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.
All UG branches - CSE, ECE, EEE, ISE & Mech.E accredited by NBA for academic years 2018-19 to 2020-21 & valid upto 30.06.2021
Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070, INDIA
Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www.bnmit.org

**Department of Computer Science and Engineering
2020-2021**

B.N.M. Institute of Technology

Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.

All UG branches - CSE, ECE, EEE, ISE & Mech.E accredited by NBA for academic years 2018-19 to 2020-21 & valid upto 30.06.2021

Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070,
INDIA Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www.bnmit.org

Department of Computer Science and Engineering



Vidyayāmruthamashnute

CERTIFICATE

Certified that the Mini Project entitled **Online DJ Management System** carried out by **Ms.Nagajyothi MS USN 1BG18CS066** a bonafide student of V Semester B.E., **B.N.M Institute of Technology** in partial fulfillment for the Bachelor of Engineering in COMPUTER SCIENCE AND ENGINEERING of the **Visvesvaraya Technological University**, Belagavi during the year 2020-2021. It is certified that all corrections / suggestions indicated for internal Assessment have been incorporated in the report. The Mini project report has been approved as it satisfies the academic requirements in respect of Database Management Systems Laboratory with Mini Project as prescribed for the said degree.

Dr. Sejal Santosh Nimborkar
Assistant Professor & Lab-Incharge
Department of CSE
BNMIT, Bengaluru

Dr. Sahana D. Gowda
Professor & HOD
Department of CSE
BNMIT, Bengaluru

Name & Signature

Examiner1:

Examiner2:

ACKNOWLEDGEMENT

The completion of this project on “Online DJ Management System” has been a good learning experience and I would like to thank all the teachers who have made it possible.

I would like to thank **Shri. Narayan Rao R Maanay**, Secretary, BNMIT, Bengaluru for providing excellent academic environment in the college.

I would like to sincerely thank **Prof. T J Rama Murthy**, Director, BNMIT, Bengaluru for having extended his support and encouragement during the course of the work. I extend my deep sense of sincere gratitude to **Dr. S. Y. Kulkarni** Additional director, BNMIT, Bengaluru, for providing us the facilities required for the project.

I would like to express my gratitude to **Prof. Eishwar N Maanay**, Dean, BNMIT, Bengaluru for his relentless support, guidance and assistance.

I would like to thank **Dr. Krishnamurthy G N**, Principal, BNMIT, Bengaluru for his constant encouragement.

I would like to thank, **Dr. Sahana D Gowda**, Professor and Head of the Department of Computer Science and Engineering who has shared her opinions and thoughts which helped me in giving my presentation successfully.

I extend my heartfelt gratitude to **Dr. Sejal Santhosh Nimborkar**, Associate Professor, Department of Computer Science and Engineering, BNMIT, Bengaluru, for completion of the project.

Finally, I would like to thank all the faculty members of Department of Computer Science and Engineering, BNMIT, Bengaluru, for their support. I would like to thank my family and friends for their unfailing moral support and encouragement.

Nagajyothi MS
1BG18CS066

ABSTRACT

The objective of this application is to develop a system that effectively manages all the data related to the various DJ events that take place in a function. The purpose is to maintain a centralized database of all DJ event related information ,DJ selection according to the customer and other functions as desired by the users to either find or book a suitable DJ according to their requirement or their desired taste of music or genre of music depending on the occasion i.e. concerts, competitions, marriages etc. The goal is to support various functions and processes necessary to manage the data efficiently by making use of HTML , CSS, javascript and PHP as the fundamentals tools of creation for the project.

TABLE OF CONTENTS

SL NO.	CHAPTERS	PAGE NO.
1.	INTRODUCTION 1.1. OVERVIEW OF DATABASE MANAGEMENT SYSTEM 1.2. INTRODUCTION TO FRONT-END 1.2.1 HTML 1.2.2 CSS 1.2.3 JAVASCRIPT 1.3. INTRODUCTION TO BACK-END 1.3.1 XAMPP 1.3.2 MYSQL 1.3.3 PHP	1-5
2.	REQUIREMENT ANALYSIS 2.1 PURPOSE OF THE PROJECT 2.1.1. OBJECTIVES 2.2 SCOPE OF THE PROJECT 2.3 FUNCTIONAL REQUIREMENTS 2.4 NON FUNCTIONAL REQUIREMENTS	6-8
3.	SOFTWARE REQUIREMENT SPECIFICATION 3.1 HARDWARE REQUIREMENTS 3.2 SOFTWARE REQUIREMENTS	9
4.	SYSTEM DESIGN 4.1 ER DIAGRAM 4.2 MAPPING OF ER DIAGRAM TO SCHEMA DIAGRAM 4.3 NORMALISATION	10-14
5.	IMPLEMENTATION 5.1 TABLE CREATION 5.2 POPULATED TABLES 5.3 SQL TRIGGERS 5.4 STORED PROCEDURES	15-19
6.	SNAPSHOTS	20-35
7.	CONCLUSION	36
8.	FUTURE ENHANCEMENTS	37

LIST OF FIGURES

FIG NO.	FIGURES	PAGE NO.
4.1	ER DIAGRAM	10
4.2	SCHEMA DIAGRAM	11
5.3	TRIGGER AND STORED PROCEDURES	18-19
6.2	ADMIN PAGES	19-31
6.3	USER PAGES	32-37

Chapter 1

INTRODUCTION

1.1 Overview of Database Management System

A Database is a collection of related data organized in a way that data can be easily accessed, managed and updated and various operations can be performed on it. By Data, we mean known facts that can be recorded and that have implicit meaning. A DBMS is a collection of programs that enables users to create and maintain a database. The DBMS is a general-purpose software system that facilitates the processes of defining, constructing, manipulating, and sharing databases among various users and applications. DBMS also provides protection and security to database. It maintains data consistency in case of multiple users. Here are some examples of popular DBMS, MySQL, Oracle, Sybase, Microsoft Access and IBM DB2 etc.

The database system can be divided into four components:

- The database system can be divided into System developer and End users.
- Database application: Database application may be Personal, Departmental, Enterprise and Internal
- DBMS: Software that allow users to define, create and manage database access,
Ex: MySQL, Oracle etc.
- Database: Collection of logical data.

Functions of database management system:

- Provides Recovery services
- Provides utility
- Provides data Independence
- Provides a clear and logical view of the process that manipulates data

1.2 INTRODUCTION TO FRONT-END

1.2.1 HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page. HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content.

1.2.2 CSS

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

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

complexity and repetition in the structural content as well as enabling the .CSS file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device. The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/CSS is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents. In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

1.2.3 JAVASCRIPT

JavaScript often abbreviated as JS, is a high-level language, just-in-time compiled, object-oriented programming language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript supports event-driven, functional, and imperative programming styles. It has APIs for working with text, arrays, dates, regular expressions, and the DOM, but the language itself does not include input output, such as networking, storage, or graphics facilities.

1.3 INTRODUCTION TO BACK-END

1.3.1 XAMP

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. XAMPP ease of deployment means a WAMP or LAMP stack can be installed quickly and simply on an operating system by

a developer, with the advantage that common add-in applications such as WordPress and Joomla! can also be installed with similar ease using Bitnami.

1.3.2 MYSQL

MySQL is multithreaded, multi user SQL database management System (DBMS). The basic program run as server providing multiuser access to a number of databases. The project's source code is available under terms of the GNU General Public License, as well as under a variety of property arguments. MySQL is a database. The data in a MySQL is stored in a Database objects called tables. A table is a collection of related data entries and it consists of columns and rows.

The databases are useful when storing information categorically.

MySQL is a central components of the LAMP open source web application software stack (and other “AMP” stacks). LAMP is an acronym for Linux, Apache, MySQL, Perl/PHP/ Python. Application that use the MySQL database include TYPO3, MODx, Joomla, WordPress, PHPBB, MyBB and Drupal. MySQL is also used in many high profile, large scale web sites, including Google (Though not for the searches).

1.3.3 PHP

PHP code may be executed with a command line interface (CLI), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in a web server or as a Common Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

The PHP language evolved without a written formal specification or standard until 2014, with the original implementation acting as the *de facto* standard implementations aimed to follow. Since 2014, work has gone on to create a formal PHP specification.

Chapter 2

REQUIREMENT ANALYSIS

2.1 Scope of the Project

The objective of this application is to develop a system that effectively manages all the data related to the various DJ events that take place in a function. The purpose is to maintain a centralized database of all DJ event related information. The goal is to support various functions and processes necessary to manage the data efficiently.

2.2 Existing System

This existing system is not providing secure registration and profile management of all the users properly. This system is not providing on-line Help. This system doesn't provide tracking of users activities and their progress. This manual system gives us very less security for saving data and some data may be lost due to mismanagement. This system is not providing event management through internet. This system is not providing proper events information. The system is giving manual information through the event management executer.

2.3 Functional Requirements

The system after careful analysis has been identified to be presented with the following modules:

Online DJ Management System Module

In ODJMS project we use PHP and MySQL database. It has two modules.

- 1.Admin Module
- 2.User Module

Admin Module

1. Dashboard: In this section, admin can see all detail in brief like the total services, Total unread queries, Total read queries, Total new booking, Total Approved booking, Total Cancelled Booking and Total Event Type
2. DJ Services: In this section, admin can manage services (add/delete).
3. Type of Event: In this section, admin can manage event type (add/delete).
4. Pages: In this section, the admin can manage about us and contact us pages.
5. Booking: In this section, admin can view new, approved, cancelled bookings and also give a remark.
6. Contact us Queries: In this section, admin can view and maintain the Queries.
7. Reports: In this section, admin can view booking in a particular period.
Search: In this section, admin can search booking details and user queries with the help of name, mobile number and booking id

Admin can also update his profile, change password and recover password.

User: user can view the website and check out the information about ODJMS services and they can also inquiry and book

2.4 Non-Functional Requirements

Performance Requirements:

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as given below:

The system should be able to interface with the existing system should be accurate.

The system should be better than the existing system.

Reliability:

In this system reliability means the mail which is send by the source must reach the target user with any modification and accurate.

Security:

The web server and database server should be protected from hacking, virus etc

Portability:

The application will be developed using standard open source software like PHP, Apache web server, MySQL database, Internet Explorer Browser etc these software will work both on Windows and Linux o/s. Hence portability problems will not arise.

Availability:

This software will be available always.

Maintainability:

In this system the presentation layer is clearly separated from the service layer. So any modification in future will be done with less effort. The database will be running at the server. Users access these forms by using the user-ids and the passwords.

Chapter 3

SOFTWARE REQUIREMENT SPECIFICATION

The requirements can be broken into 2 major categories namely hardware and software requirements. The formal specification the minimal hardware facilities expected in a system in which the project has to be run. The later specifies the essential software needed to build and run the project.

3.1 Hardware Requirements

- Minimum hardware specification
 - Disk:** 100 GB
 - RAM:** 1GB OR more
 - Microprocessor:** Intel®Pentium®CPU N3710,1.60GHz*4

3.2 Software Requirements

- Minimum software specification
 - XAMPP Server
 - Technology Implemented:** Apache Server.
 - Language Used:** PHP.
 - Database:** MySQL.
 - User Interface Design:** HTML, CSS,JAVASCRIPT.
 - Web Browser:** Mozilla, Google chrome.

Technology Implemented:

- Apache Server. • Mysql Server.

CHAPTER 4

SYSTEM DESIGN

4.1 ER DIAGRAM

An entity-relationship model (ER model) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.

ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure that can be implemented in a database, typically a relational database.

The main components of ER model are: entity set and relationship set.

Here are the geometric shapes and their meaning in an ER Diagram

Rectangle : Represents Entity sets.

Ellipses : Attributes.

Diamonds: Relationship set.

Lines : They link attributes to Entity Sets and this to Relationship Set.

Fig no: 4.1 is the ER diagram of “online dj Management System” with their entities

4.1ER-DIAGRAM

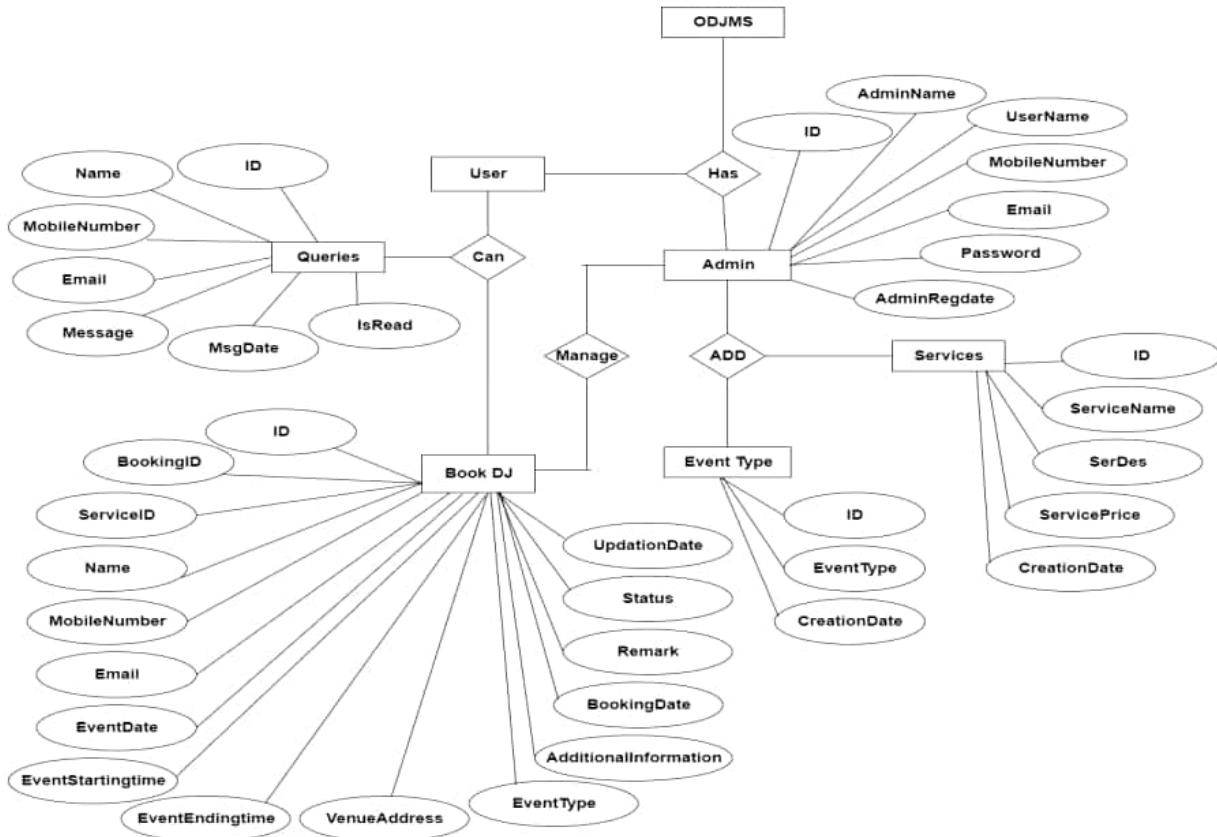


Fig. 4.1.: ER Diagram

4.2 MAPPING OF ER TO SCHEMA DIAGRAM

For each binary 1:1 relationship type are in the ER schema identify the relation S and T That correspond to the entity type participating in are. There are three are possible approaches The foreign key approach, The merged relationship approach, The cross reference or relationship relation approach,

Foreign key approach: Chooses one of the relation S and include as a foreign key in S the primary key of T. It is better to choose an entity type with total participation in R in the role of S include all the simple attribute of the 1:1 relationship type R as a attribute of S. In this database, relationship type animal_types is mapped by choosing the primary key

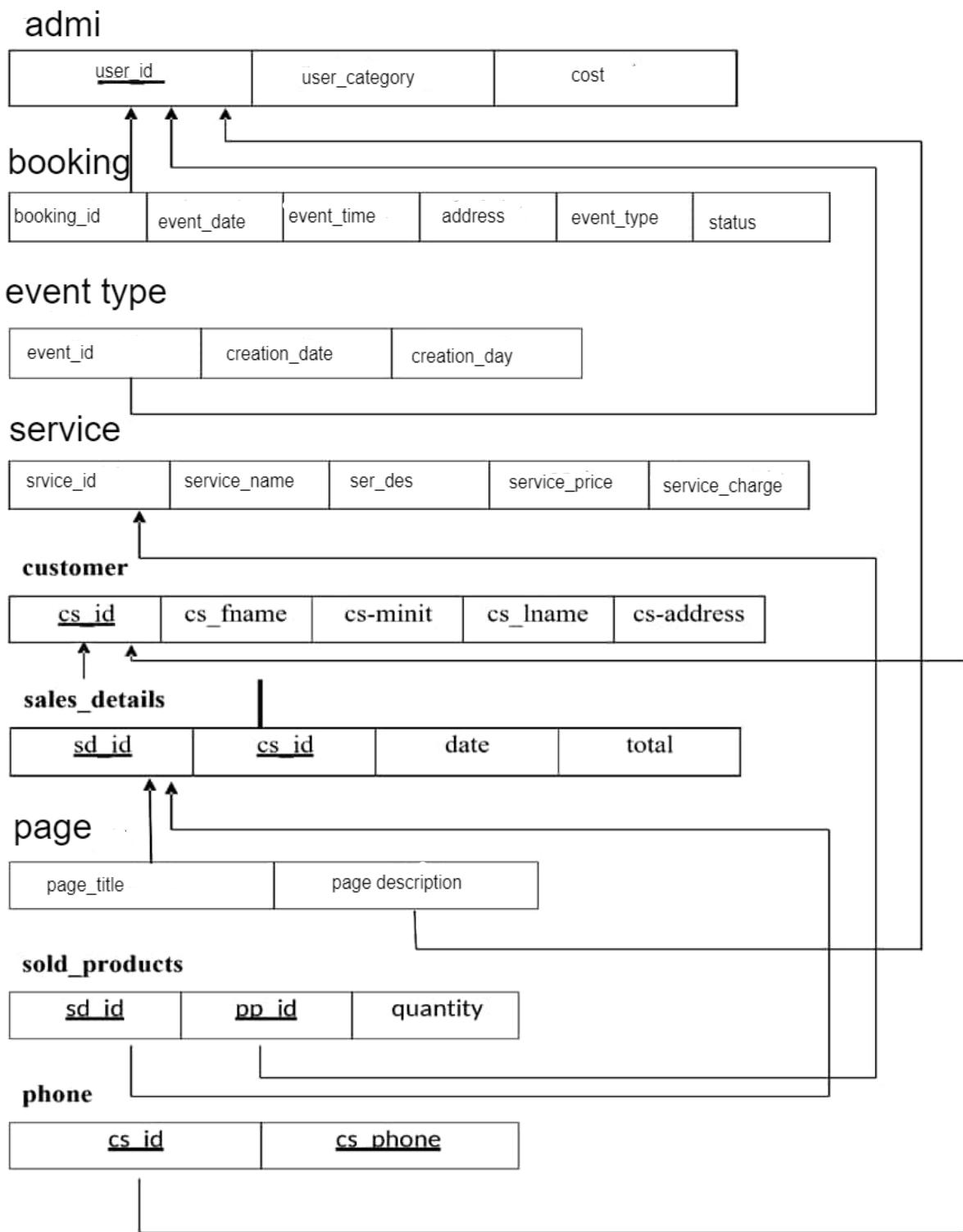


Fig. no 4.3 Schema diagram of online dj management system.

4.3 NORMALISATION

Normalization is a process of organizing the data in database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly.

There are three main types of normal forms:

- a) First normal form(1NF)
- b) Second normal form(2NF)
- c) Third normal form(3NF)
 - 1. First normal form (1NF)
 - a) As per the rule of first normal form, an attribute (column) of a table cannot hold multiple values.
 - b) It should hold only atomic values.
 - 2. Second normal form (2NF)
 - A table is said to be in 2NF if both the following conditions hold:
 - a) Table is in 1NF (First normal form)
 - b) No non-prime attribute is dependent on the proper subset of any candidate key of table.
 - c) An attribute that is not part of any candidate key is known as non-prime attribute
 - 3. Third Normal form (3NF)
 - A table design is said to be in 3NF if both the following conditions hold:
 - a) Table must be in 2NF
 - b) Transitive functional dependency of non-prime attribute on any super key should be removed.
 - c) An attribute that is not part of any candidate key is known as non-prime attribute. In other words 3NF can be explained like this: A table is in 3NF if it is in 2NF and for each functional dependency $X \rightarrow Y$ at least one of the following conditions hold:
 - X is a super key of table
 - Y is a prime attribute of table

This table holds only the atomic values company id and the company name and no multiple values are stored in this table so it can be considered as the 1NF.

2. Second normal form (2NF)

A table is said to be in 2NF if both the following conditions hold:

- a) Table is in 1NF (First normal form)
- b) No non-prime attribute is dependent on the proper subset of any candidate key of table.
- c) An attribute that is not part of any candidate key is known as non-prime attribute

3. Third Normal form (3NF)

A table design is said to be in 3NF if both the following conditions hold:

- a) Table must be in 2NF
- b) Transitive functional dependency of non-prime attribute on any super key should be removed.
- c) An attribute that is not part of any candidate key is known as non-prime attribute. In other words 3NF can be explained like this: A table is in 3NF if it is in 2NF and for each functional dependency $X \rightarrow Y$ at least one of the following conditions hold:

X is a super key of table

Y is a prime attribute of table

An attribute that is a part of one of the candidate keys is known as prime attribute. The relations are already in the normalized form in the schema diagram without any redundancy.

Chapter 5

IMPLEMENTATION

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MySQL Access database has been chosen for developing the relevant databases.

Online DJ Management System (ODJMS) contains 6 MySQL tables:

tbladmin : This table store the admin login details

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID 	int(10)			No	None		AUTO_INCREMENT
2	AdminName	varchar(120)	utf8mb4_general_ci		Yes	NULL		
3	UserName	varchar(120)	utf8mb4_general_ci		Yes	NULL		
4	MobileNumber	bigint(10)			Yes	NULL		
5	Email	varchar(200)	utf8mb4_general_ci		Yes	NULL		
6	Password	varchar(120)	utf8mb4_general_ci		Yes	NULL		
7	AdminRegdate	timestamp			Yes	current_timestamp()		

tblbooking: This table store the booking details.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID 	int(10)			No	None		AUTO_INCREMENT
2	BookingID 	int(10)			Yes	NULL		
3	ServiceID 	int(10)			Yes	NULL		
4	Name	varchar(200)	utf8mb4_general_ci		Yes	NULL		
5	MobileNumber	bignint(10)			Yes	NULL		
6	Email	varchar(200)	utf8mb4_general_ci		Yes	NULL		
7	EventDate	varchar(200)	utf8mb4_general_ci		Yes	NULL		
8	EventStartingtime	varchar(200)	utf8mb4_general_ci		Yes	NULL		
9	EventEndingtime	varchar(200)	utf8mb4_general_ci		Yes	NULL		
10	VenueAddress	mediumtext	utf8mb4_general_ci		Yes			
11	EventType 	varchar(200)	utf8mb4_general_ci		Yes	NULL		
12	AdditionalInformation	mediumtext	utf8mb4_general_ci		Yes			
13	BookingDate	timestamp			Yes	current_timestamp()		
14	Remark	varchar(200)	utf8mb4_general_ci		Yes	NULL		
15	Status	varchar(200)	utf8mb4_general_ci		Yes	NULL		
16	UpdationDate	timestamp			Yes	NULL		ON UPDATE CURRENT_TIMESTAMP()

tbleventType: This table store the type of event details.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID 	int(10)			No	None		AUTO_INCREMENT
2	EventType 	varchar(200)	utf8mb4_general_ci		Yes	NULL		
3	CreationDate	timestamp			Yes	current_timestamp()		

tblpage: This table about us and contact us detail.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID 	int(10)			No	None		AUTO_INCREMENT
2	PageType	varchar(100)	utf8mb4_general_ci		Yes	NULL		
3	PageTitle	mediumtext	utf8mb4_general_ci		Yes			
4	PageDescription	mediumtext	utf8mb4_general_ci		Yes			
5	Email	varchar(200)	utf8mb4_general_ci		Yes	NULL		
6	MobileNumber	bignint(10)			Yes	NULL		
7	UpdationDate	timestamp			Yes	NULL		ON UPDATE CURRENT_TIMESTAMP()

tblservice: This table store the details of DJ services.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID 🎵	int(10)			No	None		AUTO_INCREMENT
2	ServiceName	varchar(200)	utf8mb4_general_ci		Yes	NULL		
3	SerDes	varchar(250)	utf8mb4_general_ci		No	None		
4	ServicePrice	varchar(200)	utf8mb4_general_ci		Yes	NULL		
5	CreationDate	timestamp			Yes	current_timestamp()		

tblbookings: This table store the event booking details.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 🎵	int(11)			No	None		AUTO_INCREMENT
2	BookingId	bigint(12)			Yes	NULL		
3	UserId	int(11)			Yes	NULL		
4	EventId	int(11)			Yes	NULL		
5	NumberOfMembers	int(11)			Yes	NULL		
6	UserRemark	mediumtext	latin1_swedish_ci		Yes	NULL		
7	AdminRemark	mediumtext	latin1_swedish_ci		Yes	NULL		
8	UserCancelRemark	mediumtext	latin1_swedish_ci		Yes	NULL		
9	BookingDate	timestamp			Yes	current_timestamp()		
10	BookingStatus	varchar(100)	latin1_swedish_ci		Yes	NULL		
11	LastUpdationDate	timestamp			Yes	NULL		ON UPDATE CURRENT_TIMESTAMP()

tbluser: This table store the details user queries

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID 🎵	int(10)			No	None		AUTO_INCREMENT
2	Name	varchar(200)	utf8mb4_general_ci		Yes	NULL		
3	MobileNumber	bigint(10)			Yes	NULL		
4	Email	varchar(200)	utf8mb4_general_ci		Yes	NULL		
5	Message	mediumtext	utf8mb4_general_ci		Yes			
6	MsgDate	timestamp			Yes	current_timestamp()		
7	IsRead	int(5)			Yes	NULL		

5.3 CREATION Of TRIGGERS

A trigger is a special kind of a store procedure that executes in response to certain action on the table like insertion, deletion or updation of data.

Here in this database, trigger avoids the updation of sold pet values in dj event entity .

```
create or replace trigger
check_sold before update on events
for each
row BEGIN
DECLARE
checking int;
set checking=(select count(*) from sold_events
where
event_id=old.event_id); if (checking > 0) then
signal sqlstate '45000' set message_text = 'cannot update sold event';
end if;
END
```

5.4 CREATION OF STORED PROCEDURES

A stored procedure is a set of structured query language(SQL) statements with an assigned name, which are stored in a relational database management system as a group, so it can be reused and shared by multiple programs. Stored procedures can access or modify data in a database

Here in this database , there are two stored procedures

1. calculations_for_djevents : it calculates the cost of events sold to a particular sale and updates that in sales_details entity by adding the cost with the old total value of that sale.

2.calculations_for_djservices: it calculates the cost of product sold to a particular sale and updates that in sales_details entity by adding the cost with the old total value of that sale.

1. calculations_for_djevents

```
create procedure calculations_for_EVENTS(in pid varchar(9),in sid
varchar(9))    BEGIN
DECLARE
    epid ,csid int DEFAULT 0; set epid=(select cost
from pets where pet_id=pid); set csid=(select
total from sales_details where sd_id=sid); set
csid=csid+epid; update sales_details set total=csid
where sd_id=sid; end
```

2. calculations_for_djservices

```
create procedure calculations_for_djservices(in ppid  varchar(9),in sid
varchar(9),
in qnty int(11))
BEGIN
DECLARE
    eppid ,csid int DEFAULT 0;
    set eppid=(select cost from pet_products where pp_id=ppid);
    set csid=(select total from sales_details where sd_id=sid);
    set csid=csid+qnty*eppid; update sales_details set total=csid where sd_id=sid;
end
```

Chapter 6

SNAPSHOTS

Admin Module Screens

Sign In



Fig 6.1 Sign In

Dashboard

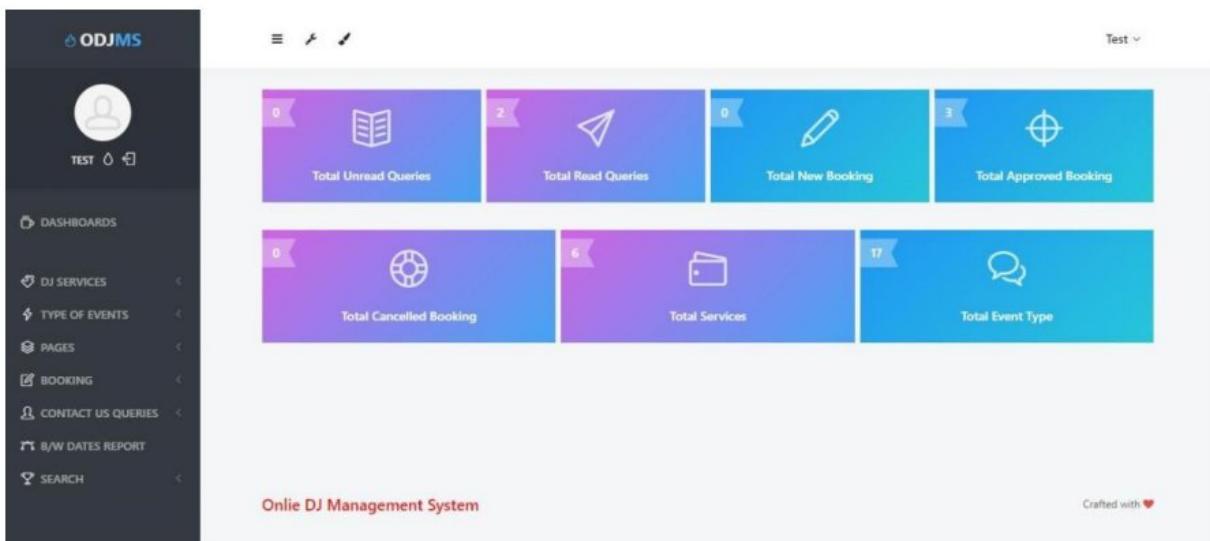


Fig 6.2 Dashboard

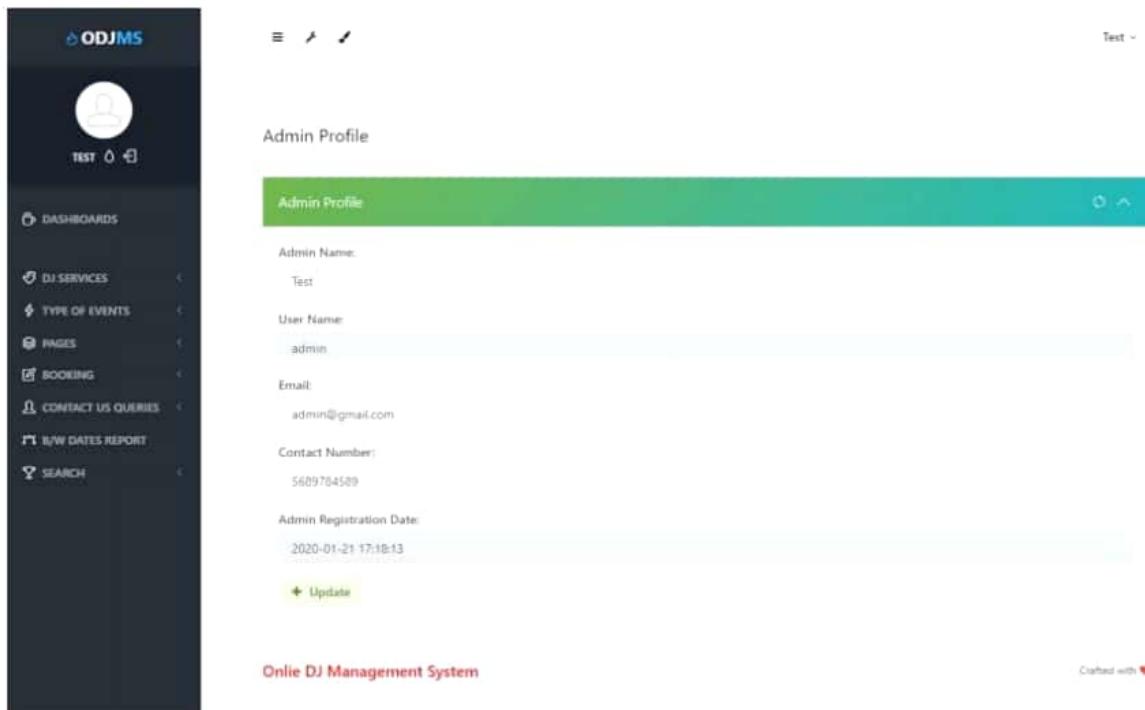


Fig 6.3 Dashboard

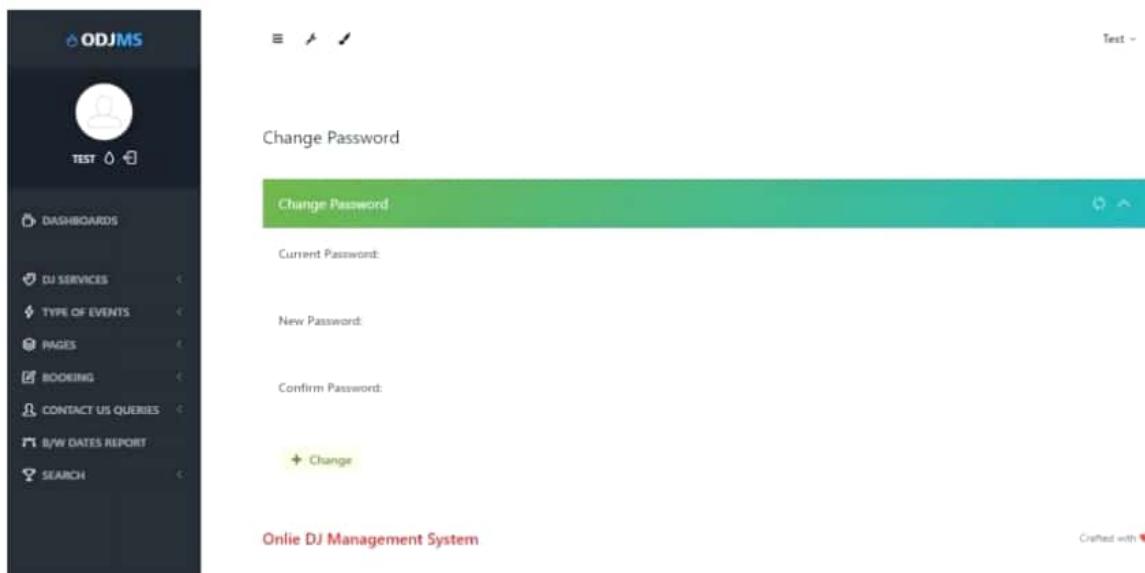


Fig 6.4 Change Password

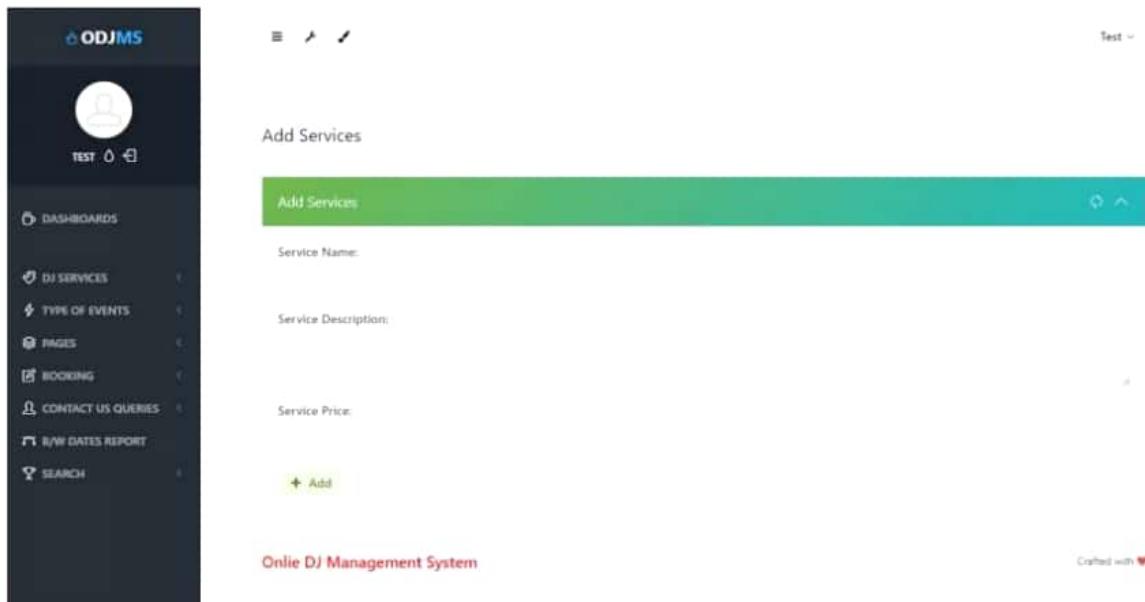


Fig 6.5 Add Services

Manage Services

Manage Services				
Show: 8 entries		Search:		
	Service Name	Service Price	Creation Date	Action
1	Wedding DJ	\$600	2020-01-24 12:47:43	
2	Party DJ	\$700	2020-01-24 12:48:32	
3	Ceremony Music	\$650	2020-01-24 12:48:41	
4	Photo Booth Hire	\$500	2020-01-24 12:49:31	
5	Karaoke Add-on	\$450	2020-01-24 12:50:36	
6	Uplighters	\$200	2020-01-24 12:51:14	

Fig 6.6 Manage Services

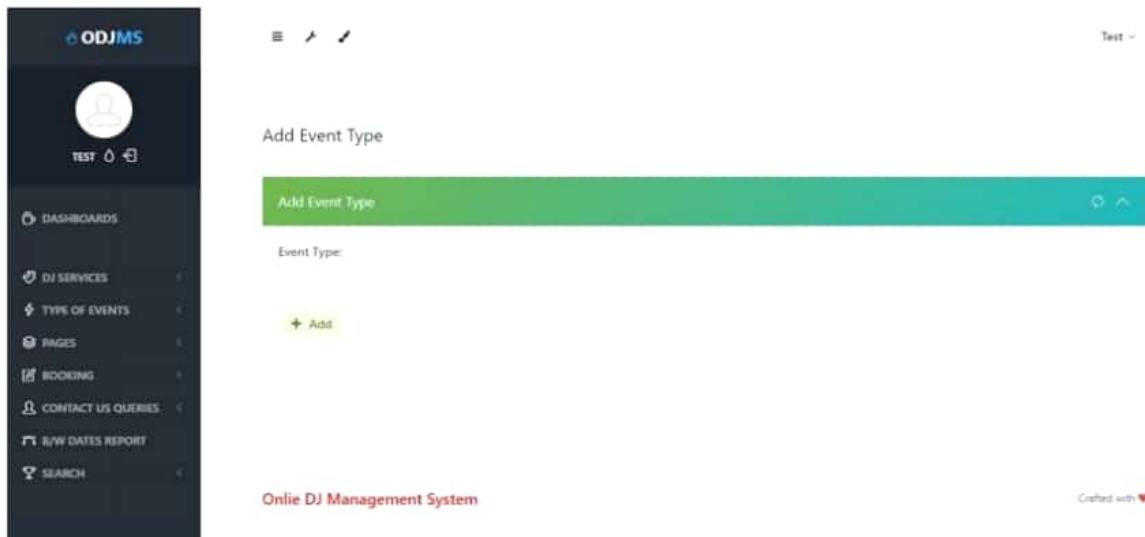


Fig 6.7 Add Event Type

Manage Event Type

Manage Event Type				
Show:	Event Name	Creation Date	Creation Date	Action
8	1 Anniversary	2020-01-22 12:31:09	2020-01-22 12:31:09	Edit
	2 Birthday Party	2020-01-22 12:32:34	2020-01-22 12:32:34	Edit
	3 Charity	2020-01-22 12:32:43	2020-01-22 12:32:43	Edit
	4 Cocktail	2020-01-22 12:33:00	2020-01-22 12:33:00	Edit
	5 College	2020-01-22 12:33:11	2020-01-22 12:33:11	Edit
	6 Community	2020-01-22 12:33:24	2020-01-22 12:33:24	Edit
	7 Concert	2020-01-22 12:33:35	2020-01-22 12:33:35	Edit
	8 Engagement	2020-01-22 12:33:51	2020-01-22 12:33:51	Edit

Fig 6.7 Manage Event Type

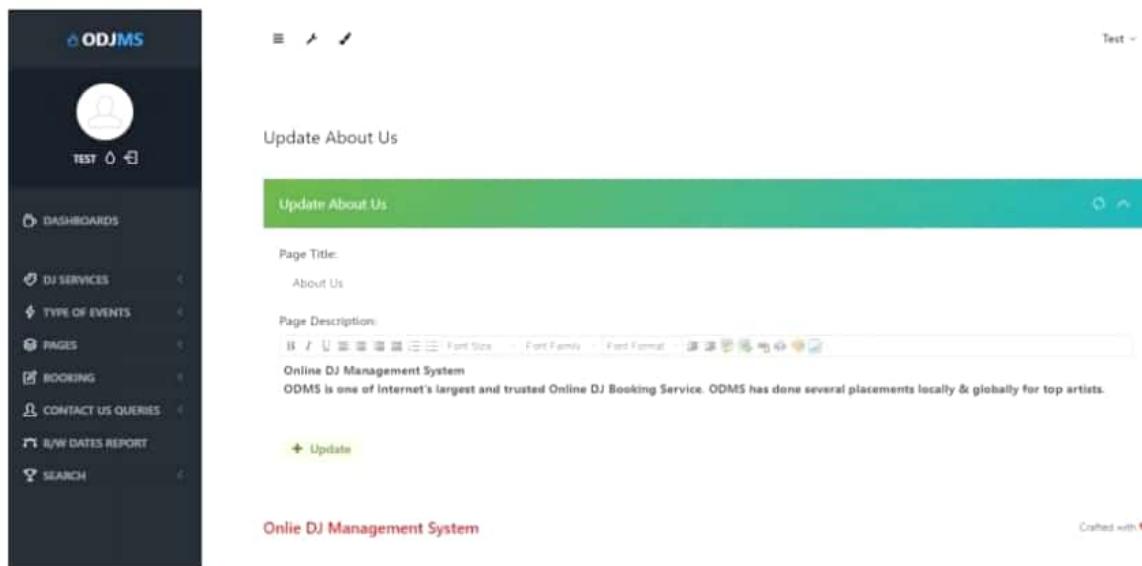


Fig 6.8 Update about Us

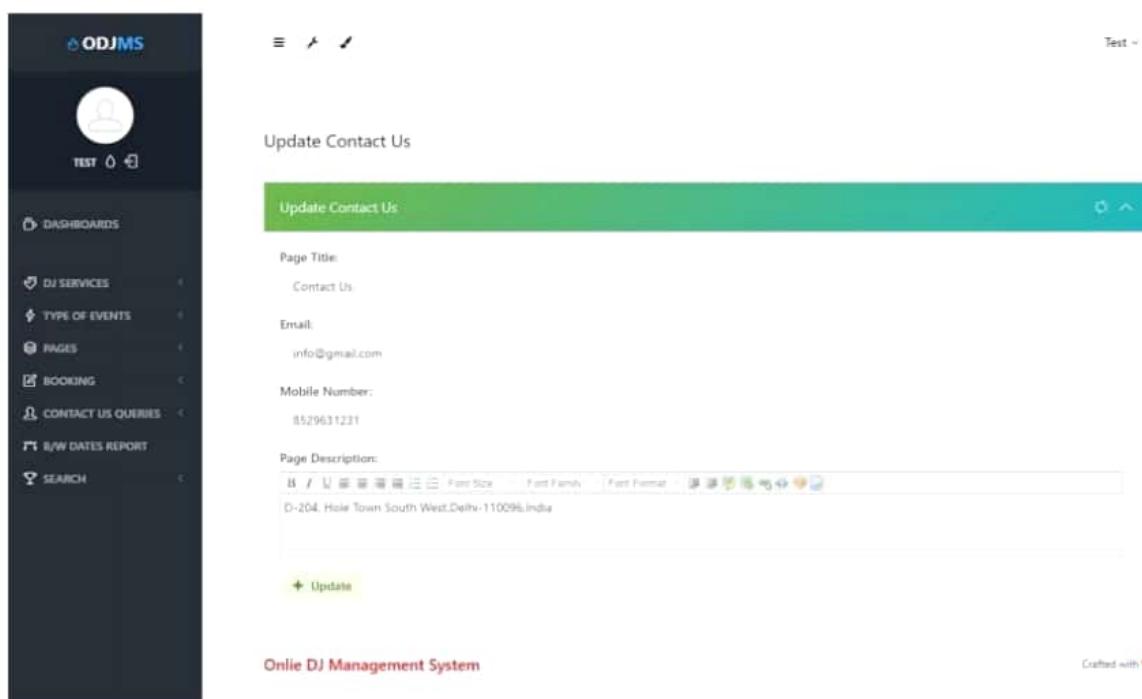


Fig 6.9 Update Contact Us

Booking ID	CUSTOMER NAME	MOBILE NUMBER	EMAIL	BOOKING DATE	STATUS	ACTION
1 206423586	Gunjan Singh	5555643433	gun@gmail.com	2020-01-29 15:07:40	Not Updated Yet	
2 365319422	Test	8097867576	test@gmail.com	2020-01-29 11:09:29	Not Updated Yet	
3 534626649	Raj Kumar	7674343543	raji@gmail.com	2020-01-29 15:15:03	Not Updated Yet	

Fig 6.10 New Booking

Booking Number	365319422	Client Name	Test
Mobile Number	8097867576	Email	test@gmail.com
Event Date	2020-02-12	Event Starting Time	7 p.m.
Event Ending Time	10 p.m.	Venue Address	R-789 KW Raj Nagar Ghaziabad
Event Type	Get Together	AdditionalInformation	xslijhj nbzijhgagwejmb gdjswgdsctxzmnbt
Service Name	Karaoke Add-on	Service Description	Karaoke is a great alternative to a disco. It's perfect for staff parties and children's parties.
Service Price	\$450	Apply Date	2020-01-29 11:09:29
Order Final Status	Not Responde yet	Admin Remark	Not Updated Yet

Fig 6.11 ViewBooking

Online DJ Management System

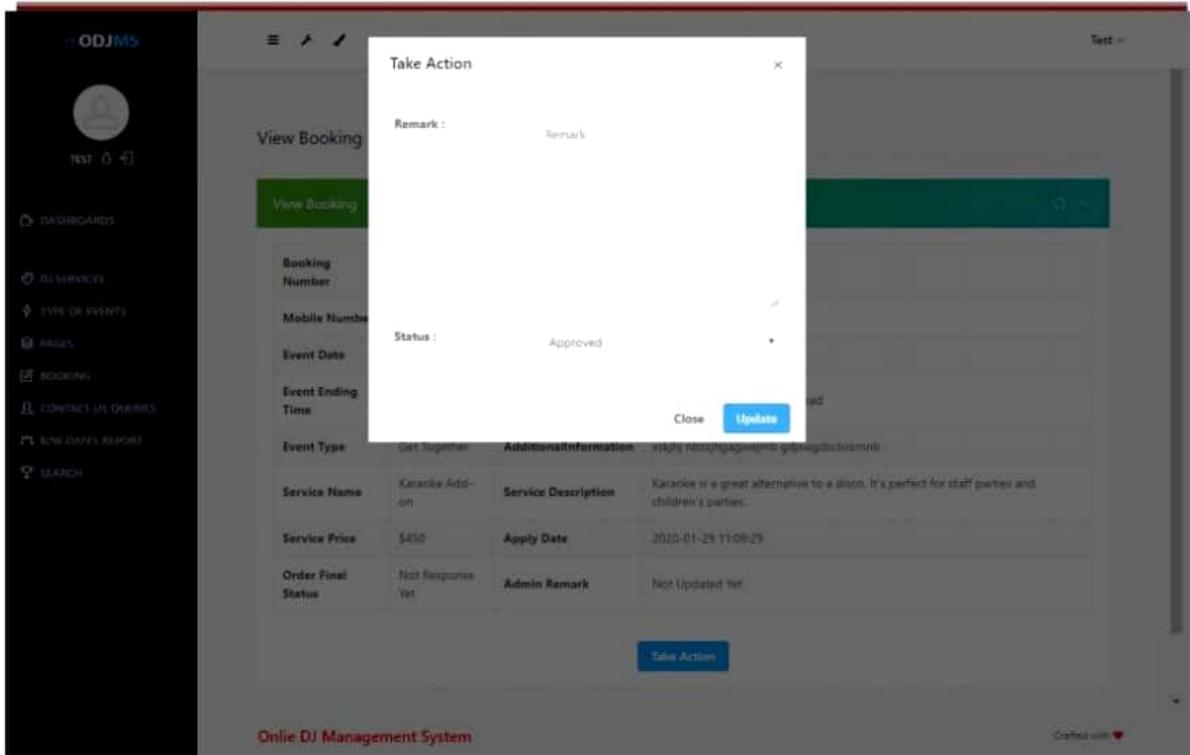


Fig 6.12 ViewBooking

Approved Booking						
Approved Booking						
Show: 8 entries		Search:				
BOOKING ID	CUSTOMER NAME	MOBILE NUMBER	EMAIL	BOOKING DATE	STATUS	ACTION
1 233064613	Ashutosh	8798787977	ashu@gmail.com	2020-01-24 18:16:29	Approved	Edit
2 750016128	Jonie	4654644648	jon@gmail.com	2020-01-28 15:34:27	Approved	Edit
3 215398258	Kishore Jha	7846466478	kis@gmail.com	2020-01-28 15:37:10	Approved	Edit

Fig 6.13 Approved Booking

Online DJ Management System

The screenshot shows a web-based application interface for managing DJ bookings. On the left is a dark sidebar with a user icon and the text "TEST". Below it are several menu items: DASHBOARDS, DJ SERVICES, TYPE OF EVENTS, PAGES, BOOKING (which is highlighted), CONTACT US QUERIES, R/W DATES REPORT, and SEARCH.

The main content area has a green header bar with the text "View Booking". Below this is a table displaying booking details:

Booking Number	233064613	Client Name	Ashutosh
Mobile Number	8798787977	Email	ashu@gmail.com
Event Date	2020-01-31	Event Starting Time	2 p.m.
Event Ending Time	10 p.m.	Venue Address	ABC park, Jawahar Nahr New Delhi
Event Type	Engagement	Additional Information	jgjytugymjhgh
Service Name	Wedding DJ	Service Description	(we install the DJ equipment before your ceremony or after your wedding breakfast)
Service Price	\$800	Apply Date	2020-01-24 18:16:29
Order Final Status	Your Booking has been approved	Admin Remark	Approved

At the bottom of the main content area, there is a footer bar with the text "Onlie DJ Management System" and "Crafted with ❤".

Fig 6.14 View Approved Booking

The screenshot shows a web-based application interface for managing invoices. On the left is a dark sidebar with a user icon and the text "TEST". Below it are several menu items: DASHBOARDS, DJ SERVICES, TYPE OF EVENTS, PAGES, BOOKING (which is highlighted), CONTACT US QUERIES, R/W DATES REPORT, and SEARCH.

The main content area has a green header bar with the text "View Invoice". Below this is a table displaying invoice details:

Booking Number: 233064613			
Name of Client	Ashutosh	Mobile Number	8798787977
Email	ashu@gmail.com	Event Date	2020-01-31
Service Name	Wedding DJ	Service Price	800
Grand Total			800

At the bottom of the main content area, there are "Close" and "Print" buttons, and a footer bar with the text "Onlie DJ Management System" and "Crafted with ❤".

Fig 6.15 View Invoice

Cancelled Booking

Online DJ Management System

The screenshot shows the 'Cancelled Booking' section of the system. On the left is a dark sidebar with various menu items like 'DASHBOARDS', 'BOOKING', etc. The main area has a header 'Cancelled Booking'. Below it is a table with columns: BOOKING ID, CUSTOMER NAME, MOBILE NUMBER, EMAIL, BOOKING DATE, STATUS, and ACTION. One entry is shown: Booking ID 365319422, Customer Name Test, Mobile Number 8097867576, Email test@gmail.com, Booking Date 2020-01-29 11:09:29, Status Cancelled, and Action button. Navigation buttons at the bottom include First, Previous, Next, and Last.

BOOKING ID	CUSTOMER NAME	MOBILE NUMBER	EMAIL	BOOKING DATE	STATUS	ACTION
365319422	Test	8097867576	test@gmail.com	2020-01-29 11:09:29	Cancelled	

View Cancelled Booking

The screenshot shows the 'View Booking' page for a specific booking. The sidebar and header are identical to the previous screen. The main content is a table with the following details:

Booking Number	365319422	Client Name	Test
Mobile Number	8097867576	Email	test@gmail.com
Event Date	2020-02-12	Event Starting Time	7 p.m
Event Ending Time	10 p.m	Venue Address	R-789 KW Raj Nagar Ghaziabad
Event Type	Get Together	Additional Information	eskjh nkczjhgagewmb gijfwegdctbxmnb
Service Name	Karaoke Add-on	Service Description	Karaoke is a great alternative to a disco. It's perfect for staff parties and children's parties.
Service Price	\$450	Apply Date	2020-01-29 11:09:29
Order Final Status	Your Booking has been cancelled	Admin Remark	Cancelled

Online DJ Management System

All Booking

Total Booking

Total Booking

	BOOKING ID	CUSTOMER NAME	MOBILE NUMBER	EMAIL	BOOKING DATE	STATUS	ACTION
1	233064613	Ashutosh	8798787977	ashu@gmail.com	2020-01-24 10:16:29	Approved	Edit
2	750016128	Jone	4654644648	joni@gmail.com	2020-01-28 11:34:37	Pending	Edit
3	215398258	Kishore Jha	7846466478	kisti@gmail.com	2020-01-28 11:47:40	Approved	Edit
4	206423586	Gunjan Singh	5555643433	gun@gmail.com	2020-01-28 11:07:40	Not Updated Yet	Edit
5	365319422	Test	8097867576	test@gmail.com	2020-01-29 11:09:29	Canceled	Edit
6	534626649	Raj Kumar	7674343543	raj@gmail.com	2020-01-29 11:11:01	Not Updated Yet	Edit

Showing 1 to 6 of 6 entries

First Previous [3](#) Next Last

Onlie DJ Management System

Crafted with ❤️

Unread Queries

Unread Queries

Unread Queries

	NAME	MOBILE NUMBER	EMAIL	SEND MESSAGE DATE	ACTION
1	Test	7654659878	test@gmail.com	2020-01-29 11:31:00	Edit

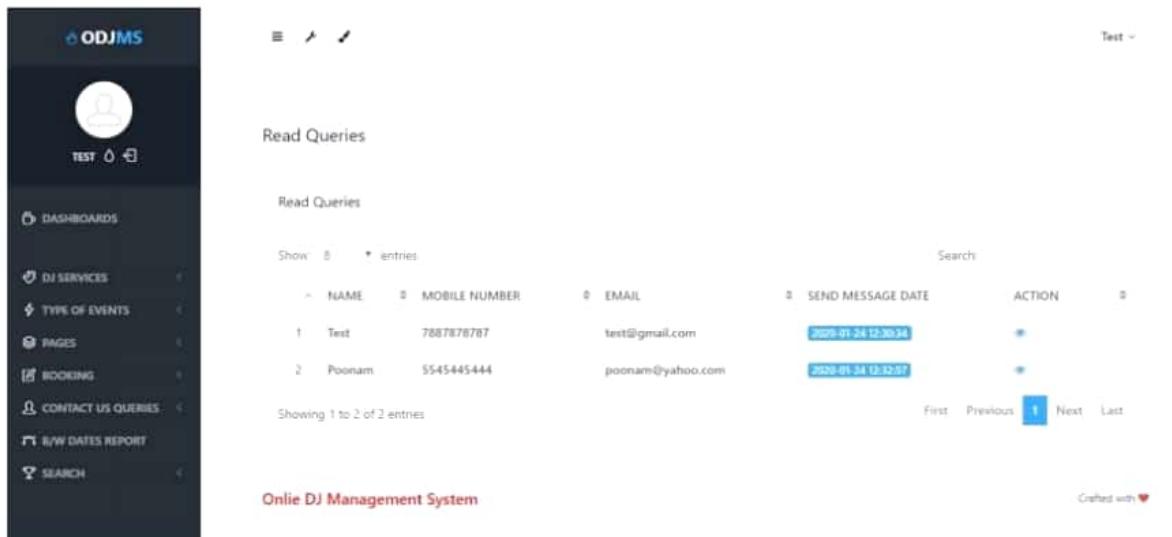
Showing 1 to 1 of 1 entries

First Previous [3](#) Next Last

Onlie DJ Management System

Crafted with ❤️

Read Queries



The screenshot shows the 'Read Queries' section of the application. On the left, there is a sidebar with various navigation options: DASHBOARDS, DJ SERVICES, TYPE OF EVENTS, PAGES, BOOKING, CONTACT US QUERIES, R/W DATES REPORT, and SEARCH. The 'CONTACT US QUERIES' option is currently selected. The main area displays a table of messages:

	NAME	MOBILE NUMBER	EMAIL	SEND MESSAGE DATE	ACTION
1	Test	7087878787	test@gmail.com	2020-01-24 12:30:34	*
2	Poonam	5545445444	poonam@yahoo.com	2020-01-24 12:32:57	*

Below the table, it says 'Showing 1 to 2 of 2 entries'. At the bottom right, there are links for 'First', 'Previous', 'Next', and 'Last'. The footer of the page reads 'Onlie DJ Management System' and 'Crafted with ❤'.

View Queries



The screenshot shows the 'View Queries' section of the application. The sidebar is identical to the previous screenshot. The main area displays a table of a single message:

Name	Mobile Number	Message
Test	7087878787	Hello

The footer of the page reads 'Onlie DJ Management System' and 'Crafted with ❤'.

Between Dates Report

Between Dates Report Details

BOOKING ID	CUTOMER NAME	MOBILE NUMBER	EMAIL	BOOKING DATE	STATUS	ACTION
1	Ashutosh	8798787977	ashu@gmail.com	2020-01-24 18:16:21	Approved	
2	Jone	4654644648	jon@gmail.com	2020-01-28 11:34:37	Approved	
3	Kishore Jha	7846466478	kis@gmail.com	2020-01-28 11:37:00	Approved	
4	Gunjan Singh	5555643433	gun@gmail.com	2020-01-29 11:07:40	Not Updated Yet	
5	Test	8097867576	test@gmail.com	2020-01-29 11:08:29	Created	
6	Raj Kumar	7674343543	raj@gmail.com	2020-01-29 11:11:01	Not Updated Yet	

User Queries Search

The screenshot shows the 'Search User Queries' section of the system. On the left, there's a sidebar with navigation links: DASHBOARDS, DJ SERVICES, TYPE OF EVENTS, PAGES, BOOKING, CONTACT US QUERIES, R/W DATES REPORT, and SEARCH. The main area has a search bar labeled 'Search User Queries' and a sub-search bar 'Search by Name/Mobile No.' Below these are two search results tables.

	NAME	MOBILE NUMBER	EMAIL	SEND MESSAGE DATE	ACTION
1	Poonam	5545445444	poonam@yahoo.com	2020-01-24 10:03:51	

Showing 1 to 1 of 1 entries

First Previous Next Last

Onlie DJ Management System Crafted with ❤

Search Booking

The screenshot shows the 'Search Booking' section of the system. The sidebar and layout are identical to the previous screenshot. The main area has a search bar labeled 'Search Booking' and a sub-search bar 'Search by Booking No./Name/Mobile No.'. Below these are two search results tables.

	BOOKING ID	CUSTOMER NAME	MOBILE NUMBER	EMAIL	BOOKING DATE	STATUS	ACTION
1	750016128	Jone	4654644648	jon@gmail.com	2020-01-28 11:14:37	Approved	

Showing 1 to 1 of 1 entries

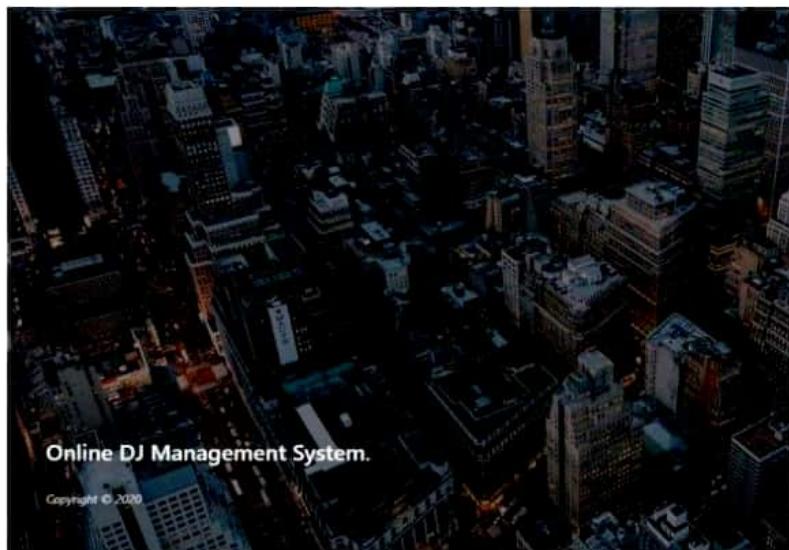
First Previous Next Last

Onlie DJ Management System Crafted with ❤

Forgot Password

USER MODEL

Home page



About Us

«Online DJ Management Systems» is one of Internet's largest and trusted Online DJ Booking Service. ODMS has done several placements locally & globally for top artists.

LATEST PHOTOS

Recent photos
Sedemque posuere enim vel ante societatem, vel nunc tempus.

Ante tempus
Sedemque posuere enim vel ante societatem, vel nunc tempus.

Aliquam rutrum
Sedemque posuere enim vel ante societatem, vel nunc tempus.

Sedemque posuere enim vel ante societatem, vel nunc tempus.

Aliquam rutrum
Sedemque posuere enim vel ante societatem, vel nunc tempus.

Copyright © 2020 Online DJ Management System

Contact Us

Services Pag

e

Book Services

ODJMS

Home / Book Services

Book Your Events now

Name:

Email:

Mobile Number:

Event Date: dd-mm-yyyy

Event Starting Time: Select Starting Time

Event Finish Time: Select Finish Time

Venue Address:

Type of Event: Choose Event Type

Additional Information:

Book

Copyright © 2020 Online DJ Management System

Chapter 7

CONCLUSION

The “**Online DJ Management System**” was successfully designed and is tested for accuracy and quality. During this project we have accomplished all the objectives and this project meets the needs of the organization. The developed will be used in searching, retrieving and generating information for the concerned requests.

GOALS ACHIVIED

- Reduced entry work
- Easy retrieval of information
- Reduced errors due to human intervention
- User friendly screens to enter the data
- Portable and flexible for further enhancement Web enabled.
- Fast finding of information request.

FUTURE ENHANCEMENT

This project can be used in the real-world after adding some more useful modules in the project for which the respected online dj management system is providing services

Nowadays online dj management system is providing many other facilities, this project can also be improved with the improvement in those.

Additional features that can be added to this project in future are: .

- Different online payment gateways can be added to felicitate online transaction for a fine.
- SMS based alert system can be added after the reservation is done.. It can tell when is the reserved date and time.
- Get a value insight of business.

Utmost care and back-up procedures must be established to ensure 100% successful implementation of the computerized online dj management system. In case of system failure, the organization should be able to process the transaction with another organization and in case of failure; it should be able to complete the process manually.

5.3 SQL Triggers & Stored Procedures

Triggers

A database trigger is procedural code that is automatically executed in response to certain events on a table or view in a database. The trigger is mostly used for maintaining the integrity of the information on the database.

Details	
Trigger name	check_sold
Table	pets
Time	BEFORE
Event	UPDATE
Definition	<pre> 1 BEGIN 2 DECLARE 3 checking int; 4 set checking=(select count(*) from sold_pets 5 where pet_id=old.pet_id); 6 if (checking > 0) then 7 signal sqlstate '45000' set message_text 8 = 'cannot update sold pet'; 9 end if; 10 END //</pre>
Definer	root@localhost

Figure 4.19 : Triggers

Stored Procedure

A stored procedure is a set of Structured Query Language (SQL) statements with an assigned name, which are stored in a relational database management system as a group, so it can be reused and shared by multiple programs.

Code for Stored Procedure is as follows:

Details

Routine name: calculations_for_pets

Type: PROCEDURE

Parameters:

Direction	Name	Type	Length/Values	Options
IN	pid	VAI	9	Char Drop
IN	sid	VAI	9	Char Drop

Add parameter

```

1 BEGIN
2 DECLARE
3   cpid ,csid int DEFAULT 0;
4 set cpid=(select cost from pets where pet_id=pid);
5 set csid=(select total from sales_details where sd_id=sid);
6 set csid=cpid;
7 update sales_details set total=csid where sd_id=sid;
8 end

```

Definition

Is deterministic:

Adjust privileges:

Definer: `root`@`localhost`

Security type: DEFINER

SQL data access: NO SQL

Comment:

Figure 4.20 : Stored Procedure

Details

Routine name: calculations_for_product

Type: PROCEDURE

Parameters:

Direction	Name	Type	Length/Values	Options
IN	ppid	VAI	9	Char Drop
IN	sid	VAI	9	Char Drop
IN	qty	INT	11	Drop

Add parameter

```

1 BEGIN
2 DECLARE
3   cppid ,csid int DEFAULT 0;
4 set cppid=(select cost from pet_products where pp_id=ppid);
5 set csid=(select total from sales_details where sd_id=sid);
6 set csid=csid*qty*cппid;
7 update sales_details set total=csid where sd_id=sid;
8 end

```

Definition

Is deterministic:

Adjust privileges:

Definer: `root`@`localhost`

Security type: DEFINER

SQL data access: NO SQL

Comment:

Figure 4.21 : Stored Procedure