**Project Name - “AMOUSEMENT PARK”**

**Introduction**

Amusement Management System is a web based technology which manages people and provides ticket to the person who comes to visits and take ride in park with his/her family. This web application provides a way to effectively control record & track the people who visit to park.

Amusement Management system effectively manages and handles all the functioning of a park. The software system can store the data of people tickets that came to visit in the park. The system also maintains and calculates the price of ticket. The system needs an administrator to input the detail of ticket like how many are adult and how many are child and print the ticket and give it to person.

In this project we use PHP and MySQL database and it has only two module i.e. Admin & User

**Advantages:**

* It helps the admin to handle and manage ticket data.
* It helps the admin to handle and manage multiple listing
* Reduce time consumption.
* Reduce error scope.
* All system managements are automated.
* Centralized database management.
* Easy operations for operator of the system.
* No paper work requirement.

**Disadvantages:**

* The system can only handle from desktop
* The system does not include bank payment, dd, cheque status.

**Applications:**

* To be used in management of booking and payment.

**Feasibility study**

Whenever we design a new system, normally the management will ask for a feasibility report of the new system. The management wants to know the technicalities and cost involved in creation of new system.

- Technical feasibility

- Economic feasibility

- Physical feasibility

Technical feasibility:

Technical feasibility involves study to establish the technical capability of the system being created to accomplish all requirements to the user. The system should be capable of handling the proposed volume of data and provide users and operating environment to increase their efficiency.

For example, system should be capable of handling the proposed volume of data and provide users.

Economic feasibility:

Economic feasibility involves study to establish the cost benefit analysis. Money spent on the system must be recorded in the form of benefit from the system. The benefits are of two types:

**Tangible benefits:**

* + Saving man labor to do tedious tasks saves time.

**Intangible benefits:**

* + Improves the quality of organization.

Physical feasibility:

It involves study to establish the time responses of the new system being created. For e.g., if the new system takes more than one day to prepare crucial finance statement for the management, wherever it was required in an hour, the system fails to provide the same.

It should be clearly establish that the new system requirements in the form of time responses would be completely met with. It may call for increase in cost. If the required cost is sacrificed then the purpose of the new system may not be achieved even if it was found to be technically feasible.

**Scope of the Project**

The proposed system will affect or interface with the person who visits in the park and administrator.

The system works and fulfills all the functionalities as per the proposed system.

It will provide reduced response time against the queries made by different users.

This project is based on PHP language with MYSQL database which manages people and provides ticket to the person who comes to visits in park with his/her family.

All possible features such as verification, validation, security, user friendliness etc have been considered.

**In this project there is one module i.e.**

**Admin**

Admin:

1. **Dashboard**: In this section, admin can see total entry ticket, normal entry tickets and water ride tickets.
2. **Pricing:** In this Section, admin can update the price of entry ticket , normal ride and water ride ticket.
3. **Entry Ticket:** In this section, admin can add, edit and delete the entry tickets.
4. **Ride Ticket**: In this section, admin can add, edit and delete the normal and water ride tickets.

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

**Software & Hardware requirements**

* Any Version of browser after Mozilla Firefox 4.0, Internet Explorer 6.0,chrome

**Hardware requirements:**

* Any processor after Pentium 4.
* Any version of Windows XP or later.
* Processor speed: 2.0 GHz
* RAM : 1GB
* Hard disk: 40GB to 80 GB

**Software requirements:**

* Database : MySQL
* Server : Apache
* Frontend : HTML
* Scripting Language : JavaScript
* IDE : Sublime
* Technology : PHP

**System Design**

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer’s requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data

**2.5 FEASIBILITY STUDY**

A feasibility analysis usually involves a through assessment of the operational (need), financial and technical aspects of a proposal. Feasibility study is the test of the system proposal made to identify whether the user needs may be satisfied using the current software and hardware technologies, whether the system will be cost effective from a business point of view and whether it can be developed with the given budgetary constraints. A feasibility study should be relatively cheap and done at the earliest possible time. Depending on the study, the decision is made whether to go ahead with a more detailed analysis.

When a new project is proposed, it normally goes through feasibility assessment. Feasibility study is carried out to determine whether the proposed system is possible to develop with available resources and what should be the cost consideration. Facts considered in the feasibility analysis were

* Technical Feasibility
* Economic Feasibility
* Behavioral Feasibility

**2.5.1 Technical Feasibility**

Technical Feasibility deals with the hardware as well as software requirements. Technology is not a constraint to type system development. We have to find out whether the necessary technology, the proposed equipments have the capacity to hold the data, which is used in the project, should be checked to carryout this technical feasibility.

The technical feasibility issues usually raised during the feasibility stage of investigation includes these

* This software is running in windows 2000 Operating System, which can be easily installed.
* The hardware required is Pentium based server.
* The system can be expanded.

**2.5.2 Economical Feasibility**

This feasibility study present tangible and intangible benefits from the prefect by comparing the development and operational cost. The technique of cost benefit analysis is often used as a basis for assessing economic feasibility. This system needs some more initial investment than the existing system, but it can be justifiable that it will improve quality of service.

Thus feasibility study should center along the following points:

* Improvement resulting over the existing method in terms of accuracy, timeliness.
* Cost comparison
* Estimate on the life expectancy of the hardware
* Overall objective

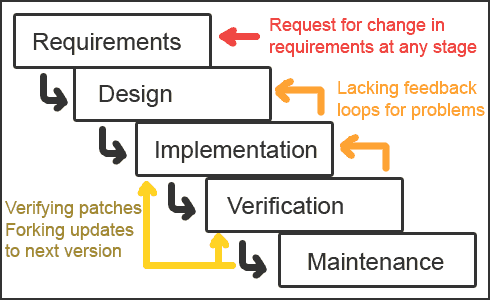
Our project is economically feasible. It does not require much cost to be involved in the overall process. The overall objectives are in easing out the requirement processes.

**2.5.3 Behavioral/ Operational Feasibility**

This analysis involves how it will work when it is installed and the assessment of political and managerial environment in which it is implemented. People are inherently resistant to change and computers have been known to facilitate change. The new proposed system is very much useful to the useful to the users and there for it will accept broad audience from around the world.

**2.6 PROJRCT PLANNING & SCHEDULING**

* 1. **SOFTWARE ENGINEERING PARADIGM APPLIED**



**Waterfall Model**

The waterfall model derivers its name due to the cascading effect from one phase to the other as is illustrated in above figure. In this model each phase well define starting and ending point, with identifiable deliveries to the next phase. Note that this model is sometime referred to as the linear sequential model or the software life cycle model. The water fall diagram is basically divided into following 5 models.

**Requirement**

**Design**

**Implementation**

**Verification**

**Maintenance**

* + **Requirement:-**

In the requirement phase the need to create the application is specified. What is the need of the system is defined. What information to be feeder to create the application will come under the requirement phase?

* + **Design:**

After the requirement phase the next phase is the Design phase where the application is designed according to the forms and other modules created. This phase is much important phase because it will structure the layout of your application.

* + **Implementation:**

Implementation is the process of having a system personnel phase check out and put new equipment into use, train users, install new application and construct any file of data need to use it.

* + **Verification:**

After the whole application is being the developed the main phase is the verification phase where the whole application tested and verified to check the whole application.

* + **Maintenance:**

After the successful verification of the application the main phase is the maintenance phase where the application needs to be maintained for its successful operation in future.

**3. SYSTEM SPECIFICATIONS**

**3.1 HARDWARE DESCRIPTION**

The selection of hardware is very important in the existence and proper working of any software. When selecting hardware, the size and requirements are also important.

Minimum Requirements:

Processor : Pentium II class, 450MHz

RAM : 128MB

Hard Disk Drive : 3GB

Video : 800X600, 256 colors

CD-ROM : Required

**The proposed System is developed on:**

Processor : INTEL Pentium 4

RAM : 512MB

Hard Disk Drive : 40GB

Key Board : Standard 101/102 or Digi Sync Family

Monitor : Display Panel (1024 X 764)

Display Adapter : Trident Super VGA

Network Adapter : SMC Ethernet Card Elite 16 Ultra

Mouse : Logitech Serial Mouse

**3.2 SOFTWARE DESCRIPTION**

Operating System : Windows XP and higher

Front- End : HTML,CSS,JAVA SCRIPT

Back- End : PHP,MYSQL

**5. OVERVIEW OF THE LANGUAGE USED**

***5.1 About Html:-***

# UNDERSTANDING HTML

* HTML was originated by Tim Berners-Lee.
* HTML developed a few years ago as a subset of SGML (Standard Generalized Mark-up Language), which is a higher-level mark-up language that has long been a favorite of the Department of Defense.
* Any HTML document is also valid for SGML.
* HTML is a Hyper Text Markup Language that is used to develop web pages.
* HTML is not a programming language like C, C++ and Java etc.
* It is a cross platform markup language that is design to be flexible enough to display text and other elements like graphical on a variety of views.
* The HTML documents consists of special Tags that are embedded in an ASCII document.
* Web browser like Internet Explorer, Netscape Navigator etc, interprets these Tags.

## Basic HTML

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<!DOCTYPE>](https://www.w3schools.com/tags/tag_doctype.asp) | Defines the document type |
| [<html>](https://www.w3schools.com/tags/tag_html.asp) | Defines an HTML document |
| [<head>](https://www.w3schools.com/tags/tag_head.asp)  Defines information about the document  [<title>](https://www.w3schools.com/tags/tag_title.asp)  Defines a title for the document | |
| [<body>](https://www.w3schools.com/tags/tag_body.asp) | Defines the document's body |
| [<h1> to <h6>](https://www.w3schools.com/tags/tag_hn.asp) | Defines HTML headings |
| [<p>](https://www.w3schools.com/tags/tag_p.asp) | Defines a paragraph |
| [<br>](https://www.w3schools.com/tags/tag_br.asp) | Inserts a single line break |
| [<hr>](https://www.w3schools.com/tags/tag_hr.asp) | Defines a thematic change in the content |
| [<!--...-->](https://www.w3schools.com/tags/tag_comment.asp) | Defines a comment |

## Formatting

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<abbr>](https://www.w3schools.com/tags/tag_abbr.asp) | Defines an abbreviation or an acronym |
| [<address>](https://www.w3schools.com/tags/tag_address.asp) | Defines contact information for the author/owner of a document/article |
| [<b>](https://www.w3schools.com/tags/tag_b.asp) | Defines bold text |
| [<bdi>](https://www.w3schools.com/tags/tag_bdi.asp) | Isolates a part of text that might be formatted in a different direction from other text outside it |
| [<bdo>](https://www.w3schools.com/tags/tag_bdo.asp) | Overrides the current text direction |
| [<blockquote>](https://www.w3schools.com/tags/tag_blockquote.asp) | Defines a section that is quoted from another source |
| [<cite>](https://www.w3schools.com/tags/tag_cite.asp) | Defines the title of a work |
| [<code>](https://www.w3schools.com/tags/tag_code.asp) | Defines a piece of computer code |
| [<del>](https://www.w3schools.com/tags/tag_del.asp) | Defines text that has been deleted from a document |
| [<dfn>](https://www.w3schools.com/tags/tag_dfn.asp) | Represents the defining instance of a term |
| [<em>](https://www.w3schools.com/tags/tag_em.asp) | Defines emphasized text |
| [<i>](https://www.w3schools.com/tags/tag_i.asp) | Defines a part of text in an alternate voice or mood |
| [<ins>](https://www.w3schools.com/tags/tag_ins.asp) | Defines a text that has been inserted into a document |
| [<kbd>](https://www.w3schools.com/tags/tag_kbd.asp) | Defines keyboard input |
| [<mark>](https://www.w3schools.com/tags/tag_mark.asp) | Defines marked/highlighted text |
| [<meter>](https://www.w3schools.com/tags/tag_meter.asp) | Defines a scalar measurement within a known range (a gauge) |
| [<pre>](https://www.w3schools.com/tags/tag_pre.asp) | Defines preformatted text |
| [<progress>](https://www.w3schools.com/tags/tag_progress.asp) | Represents the progress of a task |
| [<q>](https://www.w3schools.com/tags/tag_q.asp) | Defines a short quotation |
| [<rp>](https://www.w3schools.com/tags/tag_rp.asp) | Defines what to show in browsers that do not support ruby annotations |
| [<rt>](https://www.w3schools.com/tags/tag_rt.asp) | Defines an explanation/pronunciation of characters (for East Asian typography) |
| [<ruby>](https://www.w3schools.com/tags/tag_ruby.asp) | Defines a ruby annotation (for East Asian typography) |
| [<s>](https://www.w3schools.com/tags/tag_s.asp) | Defines text that is no longer correct |
| [<samp>](https://www.w3schools.com/tags/tag_samp.asp) | Defines sample output from a computer program |
| [<small>](https://www.w3schools.com/tags/tag_small.asp) | Defines smaller text |
| [<strong>](https://www.w3schools.com/tags/tag_strong.asp) | Defines important text |
| [<sub>](https://www.w3schools.com/tags/tag_sub.asp) | Defines subscripted text |
| [<sup>](https://www.w3schools.com/tags/tag_sup.asp) | Defines superscripted text |
| [<template>](https://www.w3schools.com/tags/tag_template.asp) | Defines a template |
| [<time>](https://www.w3schools.com/tags/tag_time.asp) | Defines a date/time |
| [<u>](https://www.w3schools.com/tags/tag_u.asp) | Defines text that should be stylistically different from normal text |
| [<var>](https://www.w3schools.com/tags/tag_var.asp) | Defines a variable |
| [<wbr>](https://www.w3schools.com/tags/tag_wbr.asp) | Defines a possible line-break |

**Forms and Input**

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<form>](https://www.w3schools.com/tags/tag_form.asp) | Defines an HTML form for user input |
| [<input>](https://www.w3schools.com/tags/tag_input.asp) | Defines an input control |
| [<textarea>](https://www.w3schools.com/tags/tag_textarea.asp) | Defines a multiline input control (text area) |
| [<button>](https://www.w3schools.com/tags/tag_button.asp) | Defines a clickable button |
| [<select>](https://www.w3schools.com/tags/tag_select.asp) | Defines a drop-down list |
| [<optgroup>](https://www.w3schools.com/tags/tag_optgroup.asp) | Defines a group of related options in a drop-down list |
| [<option>](https://www.w3schools.com/tags/tag_option.asp) | Defines an option in a drop-down list |
| [<label>](https://www.w3schools.com/tags/tag_label.asp) | Defines a label for an <input> element |
| [<fieldset>](https://www.w3schools.com/tags/tag_fieldset.asp) | Groups related elements in a form |
| [<legend>](https://www.w3schools.com/tags/tag_legend.asp) | Defines a caption for a <fieldset> element |
| [<datalist>](https://www.w3schools.com/tags/tag_datalist.asp) | Specifies a list of pre-defined options for input controls |
| [<output>](https://www.w3schools.com/tags/tag_output.asp) | Defines the result of a calculation |

## Frames

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<iframe>](https://www.w3schools.com/tags/tag_iframe.asp) | Defines an inline frame |

## Images

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<img>](https://www.w3schools.com/tags/tag_img.asp) | Defines an image |
| [<map>](https://www.w3schools.com/tags/tag_map.asp) | Defines a client-side image-map |
| [<area>](https://www.w3schools.com/tags/tag_area.asp) | Defines an area inside an image-map |
| [<canvas>](https://www.w3schools.com/tags/tag_canvas.asp) | Used to draw graphics, on the fly, via scripting (usually JavaScript) |
| [<figcaption>](https://www.w3schools.com/tags/tag_figcaption.asp) | Defines a caption for a <figure> element |
| [<figure>](https://www.w3schools.com/tags/tag_figure.asp) | Specifies self-contained content |
| [<picture>](https://www.w3schools.com/tags/tag_picture.asp) | Defines a container for multiple image resources |
| [<svg>](https://www.w3schools.com/tags/tag_svg.asp) | Defines a container for SVG graphics |

## Audio / Video

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<audio>](https://www.w3schools.com/tags/tag_audio.asp) | Defines sound content |
| [<source>](https://www.w3schools.com/tags/tag_source.asp) | Defines multiple media resources for media elements (<video>, <audio> and <picture>) |
| [<track>](https://www.w3schools.com/tags/tag_track.asp) | Defines text tracks for media elements (<video> and <audio>) |
| [<video>](https://www.w3schools.com/tags/tag_video.asp) | Defines a video or movie |

## Links

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<a>](https://www.w3schools.com/tags/tag_a.asp) | Defines a hyperlink |
| [<link>](https://www.w3schools.com/tags/tag_link.asp) | Defines the relationship between a document and an external resource (most used to link to style sheets) |
| [<nav>](https://www.w3schools.com/tags/tag_nav.asp) | Defines navigation links |

## Lists

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<ul>](https://www.w3schools.com/tags/tag_ul.asp) | Defines an unordered list |
| [<ol>](https://www.w3schools.com/tags/tag_ol.asp) | Defines an ordered list |
| [<li>](https://www.w3schools.com/tags/tag_li.asp) | Defines a list item |
| [<dl>](https://www.w3schools.com/tags/tag_dl.asp) | Defines a description list |
| [<dt>](https://www.w3schools.com/tags/tag_dt.asp) | Defines a term/name in a description list |
| [<dd>](https://www.w3schools.com/tags/tag_dd.asp) | Defines a description of a term/name in a description list |
| [<menu>](https://www.w3schools.com/tags/tag_menu.asp) | Defines a list/menu of commands |
| [<menuitem>](https://www.w3schools.com/tags/tag_menuitem.asp) | Defines a command/menu item that the user can invoke from a popup menu |

## Tables

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<table>](https://www.w3schools.com/tags/tag_table.asp) | Defines a table |
| [<caption>](https://www.w3schools.com/tags/tag_caption.asp) | Defines a table caption |
| [<th>](https://www.w3schools.com/tags/tag_th.asp) | Defines a header cell in a table |
| [<tr>](https://www.w3schools.com/tags/tag_tr.asp) | Defines a row in a table |
| [<td>](https://www.w3schools.com/tags/tag_td.asp) | Defines a cell in a table |
| [<thead>](https://www.w3schools.com/tags/tag_thead.asp) | Groups the header content in a table |
| [<tbody>](https://www.w3schools.com/tags/tag_tbody.asp) | Groups the body content in a table |
| [<tfoot>](https://www.w3schools.com/tags/tag_tfoot.asp) | Groups the footer content in a table |
| [<col>](https://www.w3schools.com/tags/tag_col.asp) | Specifies column properties for each column within a <colgroup> element |
| [<colgroup>](https://www.w3schools.com/tags/tag_colgroup.asp) | Specifies a group of one or more columns in a table for formatting |

## Styles and Semantics

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<style>](https://www.w3schools.com/tags/tag_style.asp) | Defines style information for a document |
| [<div>](https://www.w3schools.com/tags/tag_div.asp) | Defines a section in a document |
| [<span>](https://www.w3schools.com/tags/tag_span.asp) | Defines a section in a document |
| [<header>](https://www.w3schools.com/tags/tag_header.asp) | Defines a header for a document or section |
| [<footer>](https://www.w3schools.com/tags/tag_footer.asp) | Defines a footer for a document or section |
| [<main>](https://www.w3schools.com/tags/tag_main.asp) | Specifies the main content of a document |
| [<section>](https://www.w3schools.com/tags/tag_section.asp) | Defines a section in a document |
| [<article>](https://www.w3schools.com/tags/tag_article.asp) | Defines an article |
| [<aside>](https://www.w3schools.com/tags/tag_aside.asp) | Defines content aside from the page content |
| [<details>](https://www.w3schools.com/tags/tag_details.asp) | Defines additional details that the user can view or hide |
| [<dialog>](https://www.w3schools.com/tags/tag_dialog.asp) | Defines a dialog box or window |
| [<summary>](https://www.w3schools.com/tags/tag_summary.asp) | Defines a visible heading for a <details> element |
| [<data>](https://www.w3schools.com/tags/tag_data.asp) | Links the given content with a machine-readable translation |

## Meta Info

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<head>](https://www.w3schools.com/tags/tag_head.asp) | Defines information about the document |
| [<meta>](https://www.w3schools.com/tags/tag_meta.asp) | Defines metadata about an HTML document |
| [<base>](https://www.w3schools.com/tags/tag_base.asp) | Specifies the base URL/target for all relative URLs in a document |

## Programming

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<script>](https://www.w3schools.com/tags/tag_script.asp) | Defines a client-side script |
| [<noscript>](https://www.w3schools.com/tags/tag_noscript.asp) | Defines an alternate content for users that do not support client-side scripts |
| [<embed>](https://www.w3schools.com/tags/tag_embed.asp) | Defines a container for an external (non-HTML) application |
| [<object>](https://www.w3schools.com/tags/tag_object.asp) | Defines an embedded object |
| [<param>](https://www.w3schools.com/tags/tag_param.asp) | Defines a parameter for an object |

***5.2 About JavaScript:-***

* What is JavaScript?
* JavaScript was designed to add interactivity to HTML pages
* JavaScript is a scripting language (a scripting language is a lightweight programming language)
* A JavaScript consists of lines of executable computer code
* A JavaScript is usually embedded directly into HTML pages
* JavaScript is an interpreted language (means that scripts execute without preliminary compilation)
* Everyone can use JavaScript without purchasing a license
* Are Java and JavaScript the Same?
* NO! Java and JavaScript are two completely different languages in both concept and design!
* Java (developed by Sun Microsystems) is a powerful and much more complex programming language - in the same category as C and C++.
* What can a JavaScript Do?
* JavaScript gives HTML designers a programming tool - HTML authors are normally not programmers, but JavaScript is a scripting language with a very simple syntax! Almost anyone can put small "snippets" of code into their HTML pages

**JavaScript Syntax.**

* Unlike HTML, JavaScript is case sensitive.
* Dot Syntax is used to combine terms.

e.g., **document.write("Hello World")**

* Certain characters and terms are reserved.

JavaScript is simple text (ASCII).

**JavaScript Terminology**

* JavaScript programming uses specialized terminology.
* Understanding JavaScript terms is fundamental to understanding the script.
* Objects, Properties, Methods, Events, Functions, Values, Variables, Expressions, Operators.

***5.3 About Php:-***

* **Introduction to PHP:**
* The full form of PHP is “Hypertext Preprocessor”. Its original name was “Personal Home Page”
* Rasmus Lerdorf software engineer, Apache team member is the creator and original driving force behind PHP. The first part of PHP was developed for his personal use in late 1994.
* By the middle of 1997, PHP was being used on approximately 50,000 sites worldwide.
* PHP is server-side scripting language, which can be embedded in HTML or used as a stand-alone.
* PHP doesn’t do anything about what a page looks and sounds like. In fact, most of what PHP does is invisible to the end user.
* Someone looking at a PHP page will not necessarily be able to tell that it was not written purely in HTML, because usually the result of PHP is HTML.
* PHP is an official module of Apache HTTP Server.
* PHP is fully cross-platform, meaning it runs native on several flavors of Unix, as well as on Windows and now on Mac OS X.
* **Advantages of PHP**
* *Cost*: PHP costs you nothing. It is open source software and doesn’t need to purchase it for development.
* *Ease of Use*: PHP is easy to learn, compared to the others. A lot of Ready-made PHP scripts are freely available in market so, you can use them in your project or get some help from them.
* *HTML- Support:* PHP is embedded within HTML; In other words, PHP pages are ordinary HTML pages that escape into PHP mode only when necessary. When a client requests this page, the web server preprocesses it. This means it goes through the page from top to bottom, looking for sections of PHP, which it will try to resolve.
* *Cross-platform compatibility*: MySQL run native on every popular flavor of Unix and windows. A huge percentage PHP and of the world’s HTTP servers run on one of these two classes of operating system.
* *PHP is compatible with the three leading Web servers:* Apache HTTP Server for Unix and Windows, Microsoft Internet Information Server, and Netscape Enterprise Server.

It also works with several lesser-known servers, including Alex Blits’ fhttpd, Microsoft’s Personal Web Server, AOL Server and Omnicentrix’s Omni server application server.

* *Stability:* The word stable means two different things in this context:
  + - The server doesn’t need to be rebooted often
    - The software doesn’t change radically and incompatibly from release to release.

To our advantage, both of these apply to both MySQL and PHP.

* Speed: PHP is pleasingly zippy in its execution, especially when compiled as and Apache module on the Unix side. Although it takes a slight performance hit by being interpreted rather than compiled, this is far outweighed by the benefits PHP drives from its status as a Web server module.

***5.4 About MySql:-***

* MySQL Database Management System
* MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL AB.
* MySQL AB is a commercial company, founded by the MySQL developers. It is a second generation Open Source Company that unites Open Source values and methodology with a successful business model.
* The MySQL Web site (<http://www.mysql.com/>) provides the latest information about MySQL software and MySQL AB.
* The official way to pronounce “MySQL” is “My Ess Que Ell” (not “my sequel”), but we don't mind if you pronounce it as “my sequel” or in some other localized way.
* MySQL Features:
  + MySQL is a database management system.
  + MySQL is a relational database management system.
  + MySQL software is Open Source.
  + The MySQL Database Server is very fast, reliable, and easy to use.
  + MySQL Server works in client/server or embedded systems.
  + A large amount of contributed MySQL software is available.

1. **SYSTEM DESIGN**

**DEFINATION**

The most creative and challenging face of the system development is System Design. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Design goes through the logical and physical stages of development.

In designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. The first step is to determine how the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of the proposed output. The operational phases are handled through program construction and testing.

Design of a system can be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Thus system design is a solution to “how to” approach to the creation of a new system. Thus important phase provides the understanding and the procedural details necessary for implementing the system recommended in the feasibility study. The design step provides a data design, architectural design, and a procedural design.

**6.1 OUTPUT DESIGN**

In the output design, the emphasis is on producing a hard copy of the information requested or displaying the output on the CRT screen in a predetermined format. Two of the most output media today are printers and the screen. Most users now access their reports from a hard copy or screen display. Computer’s output is the most important and direct source of information to the user, efficient, logical, output design should improve the systems relations with the user and help in decision-making.

As the outputs are the most important source of information to the user, better design should improve the system’s relation and also should help in decision-making. The output device’s capability, print capability, print capability, response time requirements etc should also be considered form design elaborates the way output is presented and layout available for capturing information. It’s very helpful to produce the clear, accurate and speedy information for end users.

**6.2 INPUT DESIGN**

In the input design, user-oriented inputs are converted into a computer based system format. It also includes determining the record media, method of input, speed of capture and entry on to the screen. Online data entry accepts commands and data through a keyboard. The major approach to input design is the menu and the prompt design. In each alternative, the user’s options are predefined. The data flow diagram indicates logical data flow, data stores, source and destination. Input data are collected and organized into a group of similar data. Once identified input media are selected for processing.

In this software, importance is given to develop Graphical User Interface (GUI), which is an important factor in developing efficient and user-friendly software. For inputting user data, attractive forms are designed. User can also select desired options from the menu, which provides all possible facilities.

Also the important input format is designed in such a way that accidental errors are avoided. The user has to input only just the minimum data required, which also helps in avoiding the errors that the users may make. Accurate designing of the input format is very important in developing efficient software. The goal or input design is to make entry as easy, logical and free from errors.

**6.3 LOGICAL DESIGN**

Logical data design is about the logically implied data. Each and every data in the form can be designed in such a manner to understand the meaning. Logical data designing should give a clear understanding and idea about the related data used to construct a form.

**6.4 DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. It’s a structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

**Unified Modelling Language Diagrams (UML):**

* + The unified modelling language allows the software engineer to express an analysis model using the modelling notation that is governed by a set of syntactic semantic and pragmatic rules.
  + A UML system is represented using five different views that describe the system from distinctly different perspective. Each view is defined by a set of diagram, which is as follows.

**User Model View**

* + 1. This view represents the system from the users perspective.
    2. The analysis representation describes a usage scenario from the end-users perspective**.**

**Structural model view**

◆In this model the data and functionality are arrived from inside the system.

◆ This model view models the static structures.

**Behavioural Model View**

◆ It represents the dynamic of behavioural as parts of the system, depicting the interactions of collection between various structural elements described in the user model and structural model view.

**Implementation Model View**

* + In this the structural and behavioural as parts of the system are represented as they are to be built.

**Environmental Model View**

In this the structural and behavioural aspects of the environment in which the system is to be implemented are represented.

UML is specifically constructed through two different domains they are

* + UML Analysis modelling, which focuses on the user model and structural model views of the system?
  + UML design modelling, which focuses on the behavioural modelling, implementation modelling and environmental model views**.**

### Use Case Diagrams Admin

#### ENTITY-RELATIONSHIP Diagrams

E-R (Entity-Relationship) Diagram is used to represents the relationship between entities in the table.

## The symbols used in E-R diagrams are:

SYMBOL PURPOSE

Represents Entity sets.

Represent attributes.

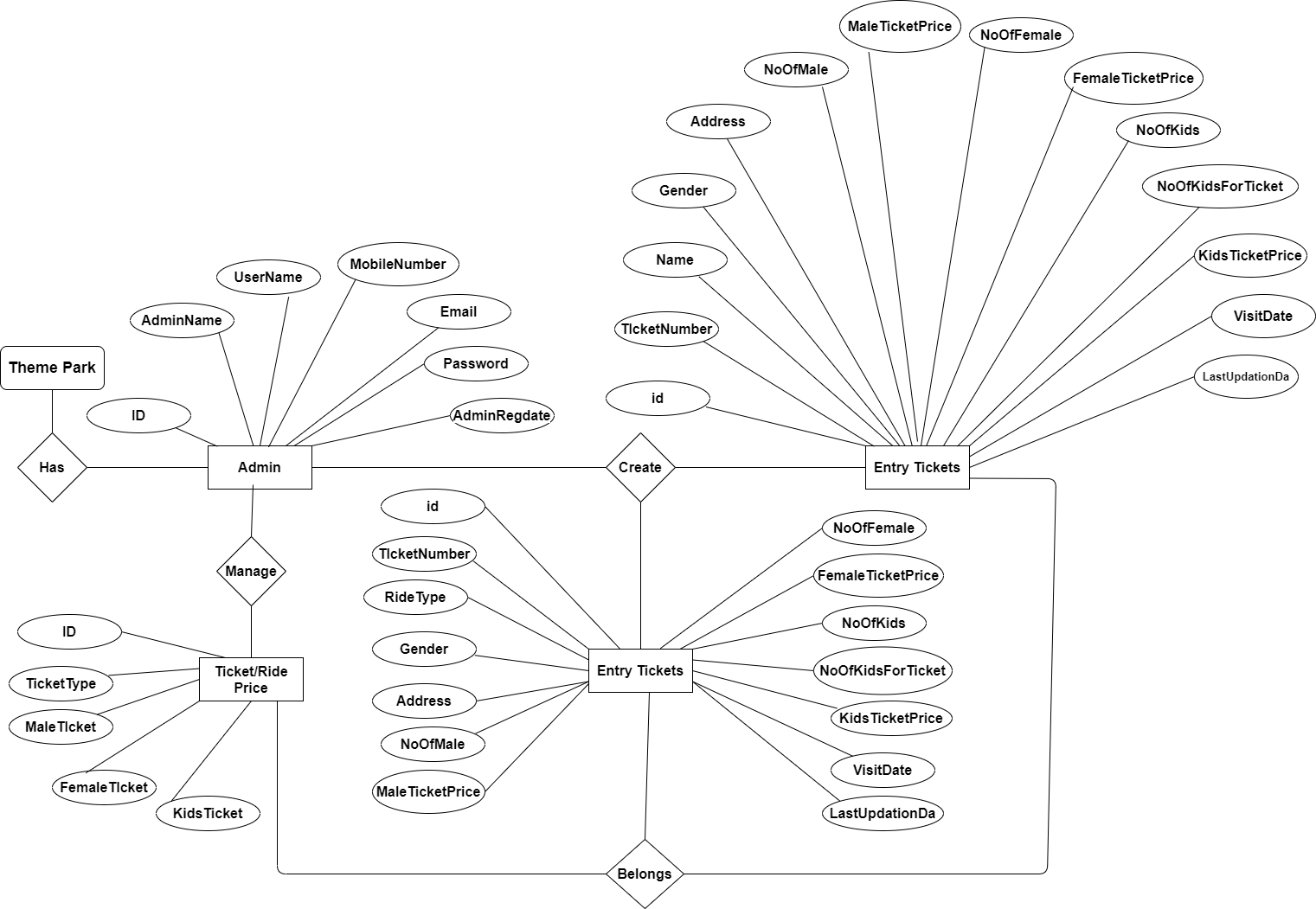
Represent Relationship Sets.

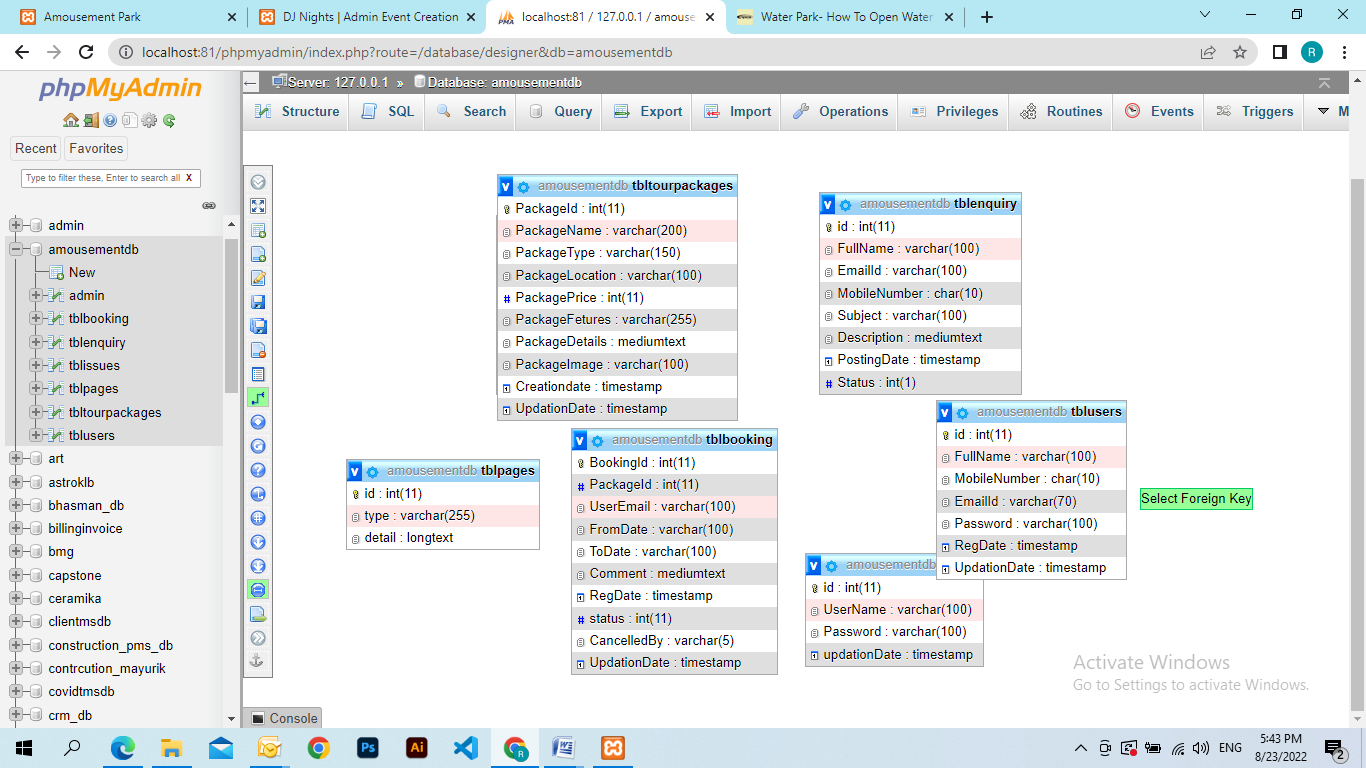
Line represents flow

Structured analysis is a set of tools and techniques that the analyst.

To develop a new kind of a system:

The traditional approach focuses on the cost benefit and feasibility analysis, Project management, and hardware and software selection a personal considerations.

****

****

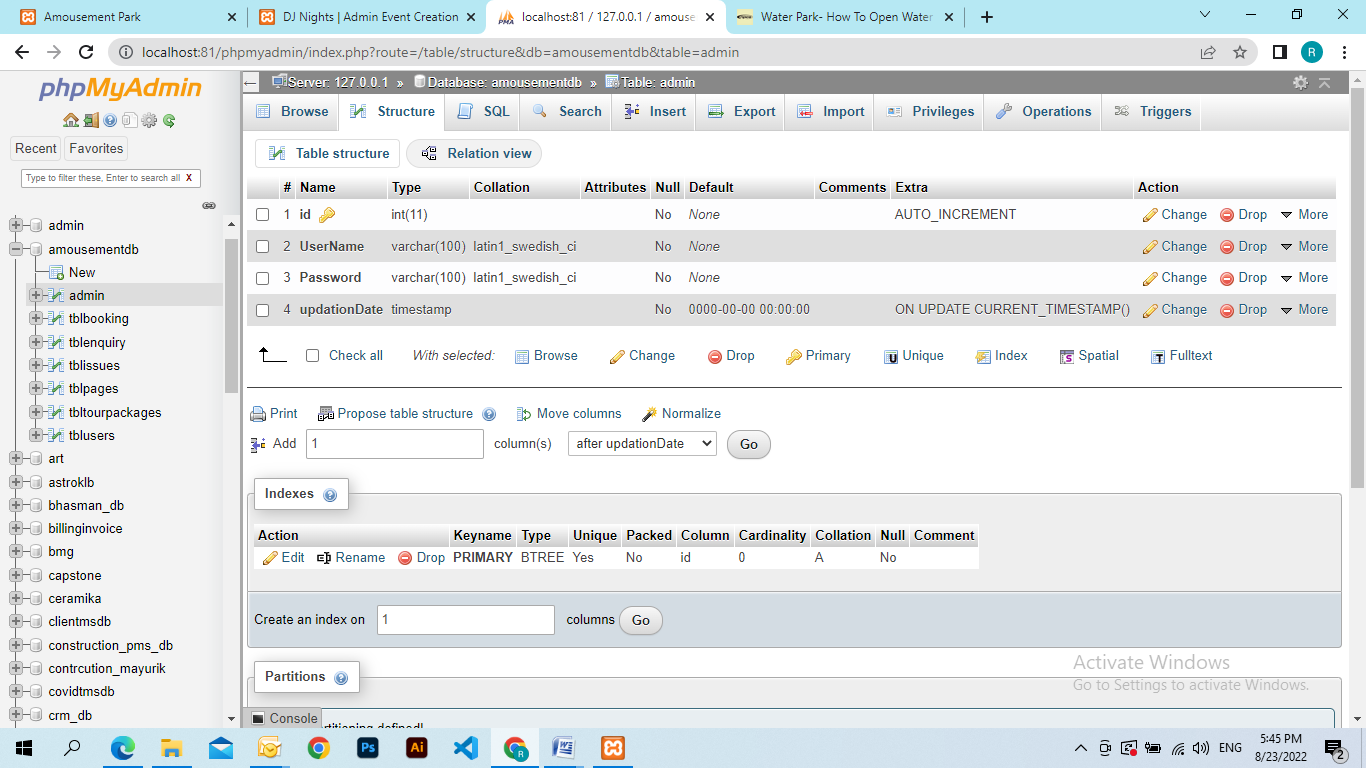
**DATABASE DESIGN**

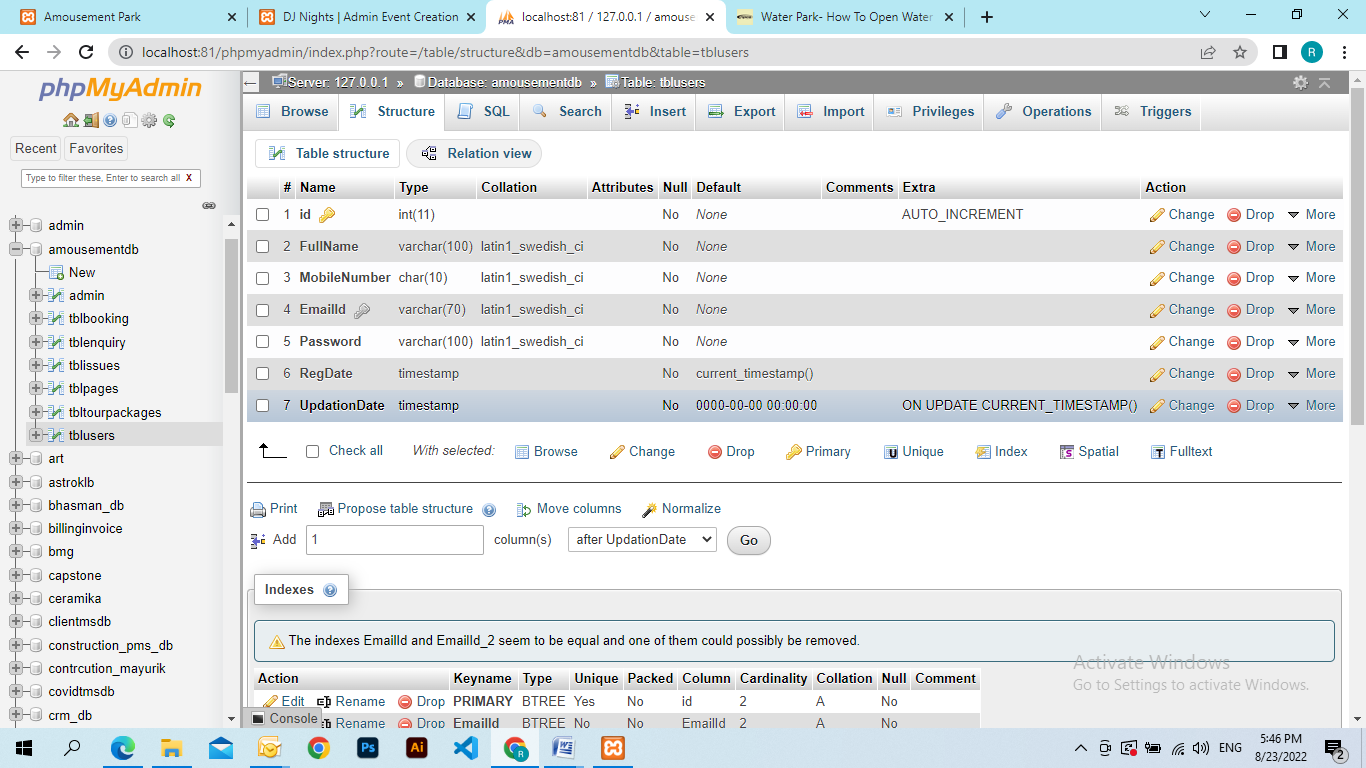
The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

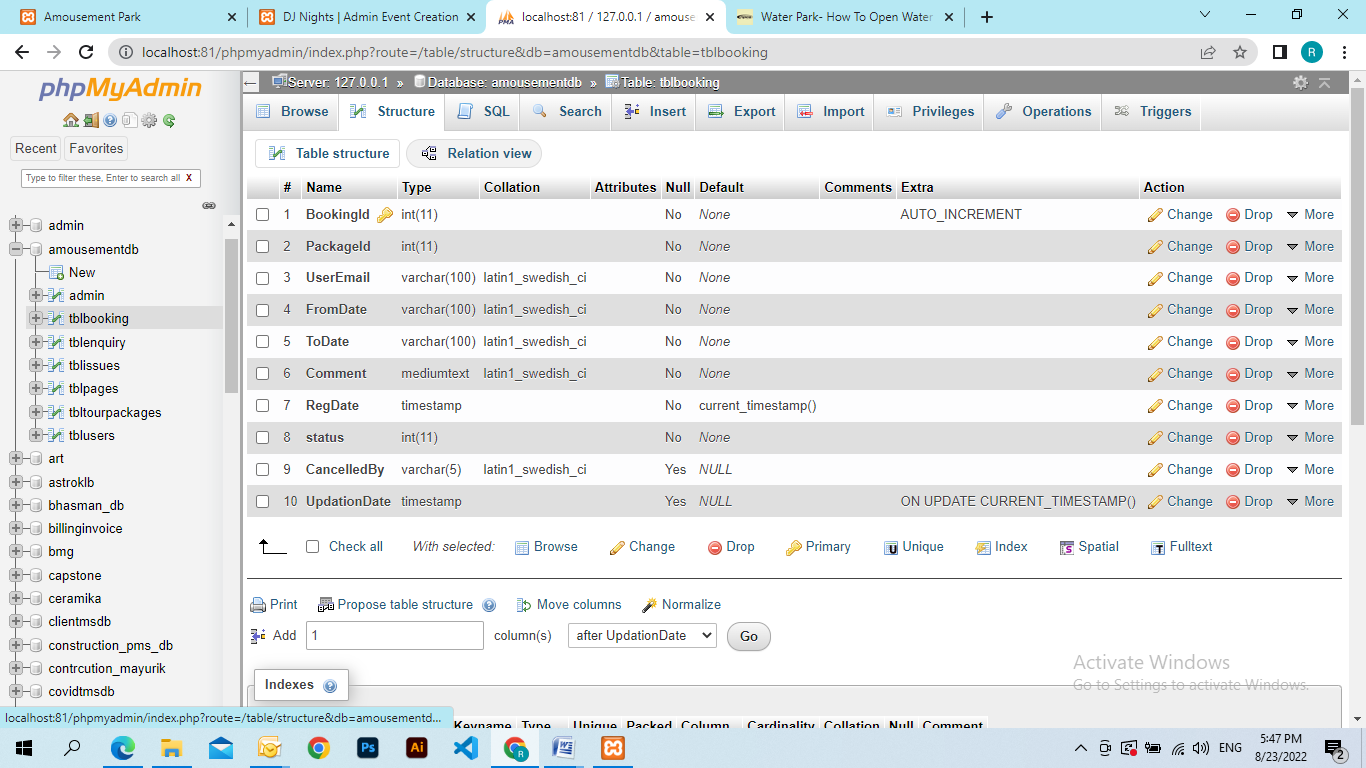
A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items 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 MS Access database has been chosen for developing the relevant databases.

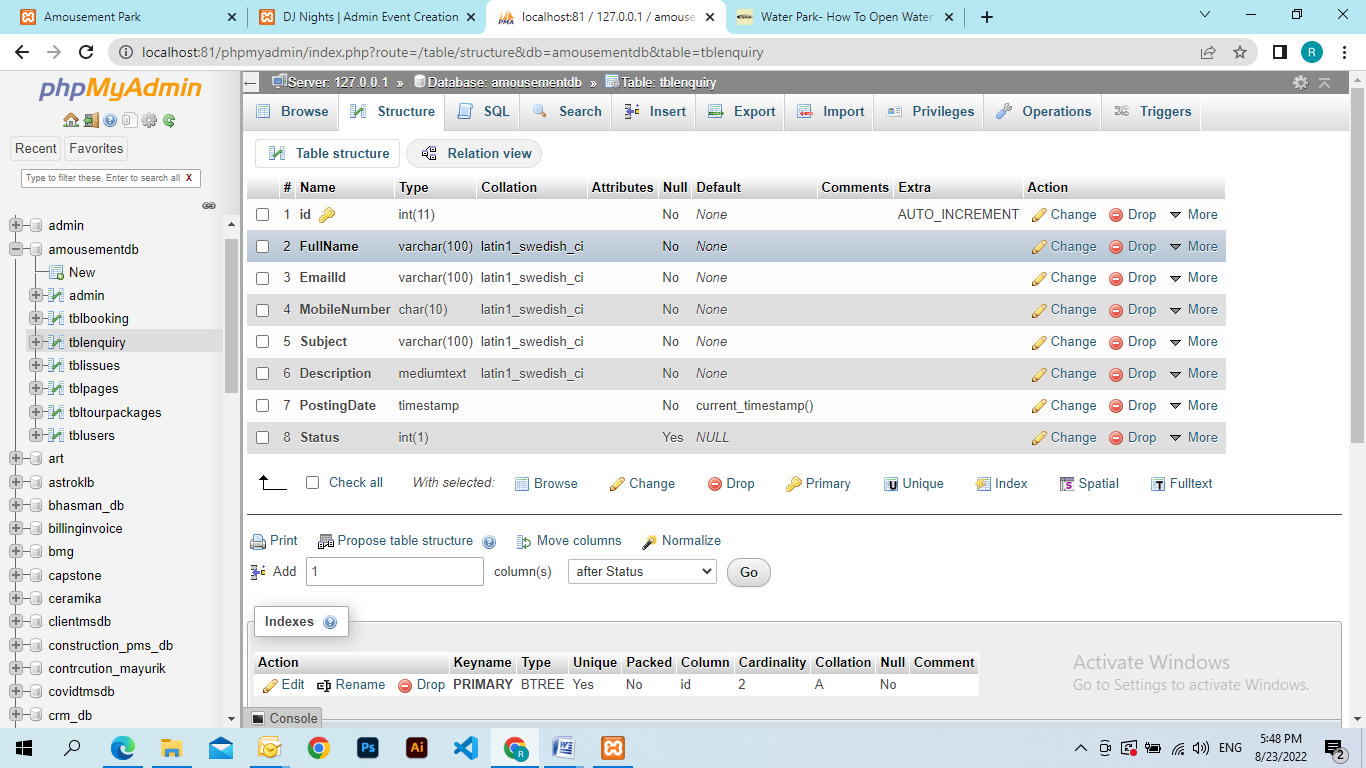
**Amousement Management System (PTMS) contains 4 MySQL tables :**

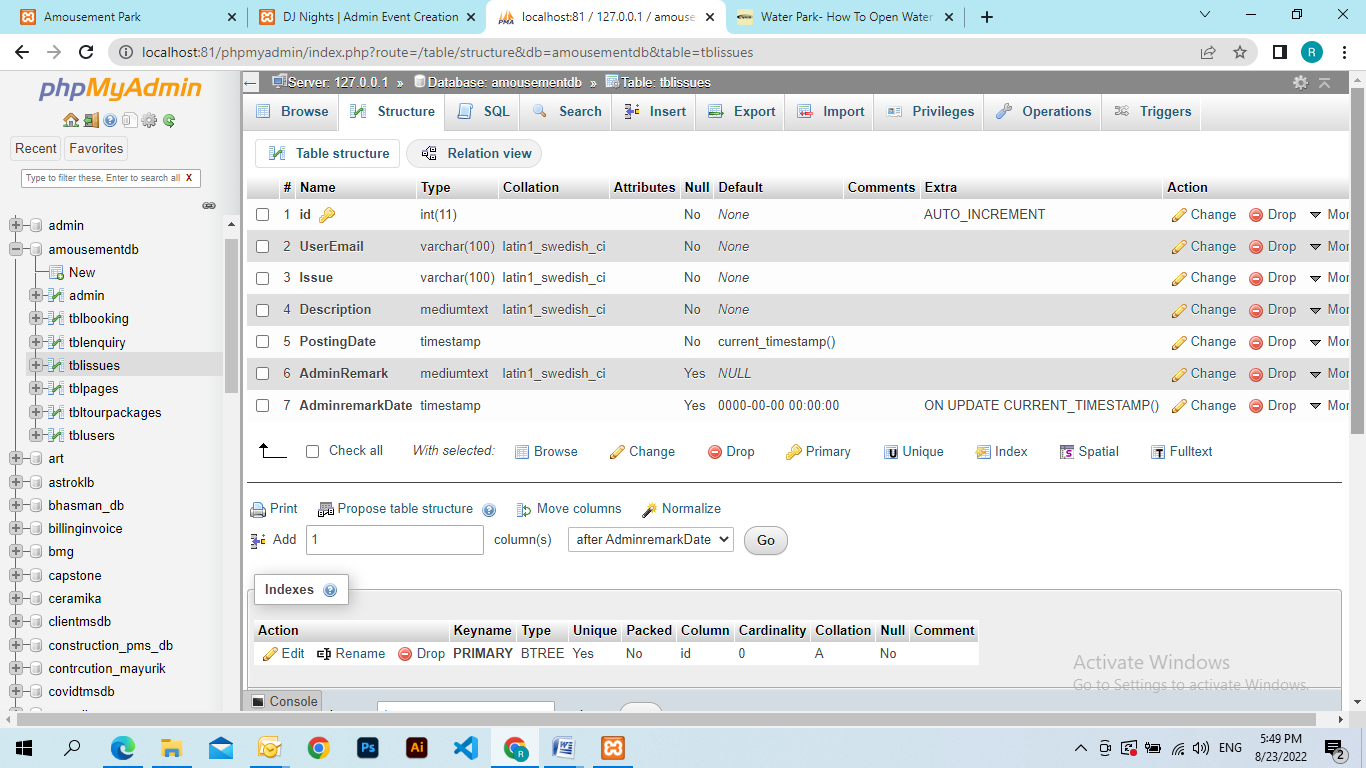
**tbladmin table Structure :** This table store the admin login and personal Details.

****



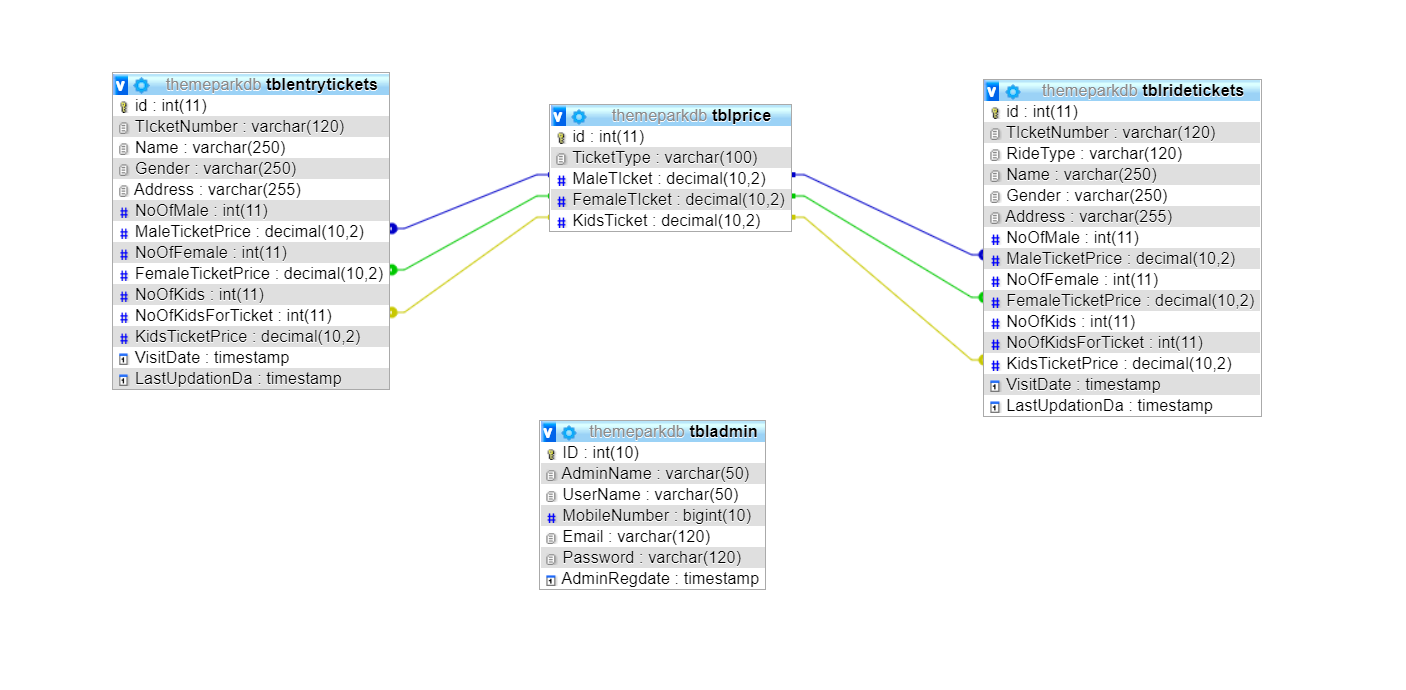
****

****

****

**Class Diagram:**

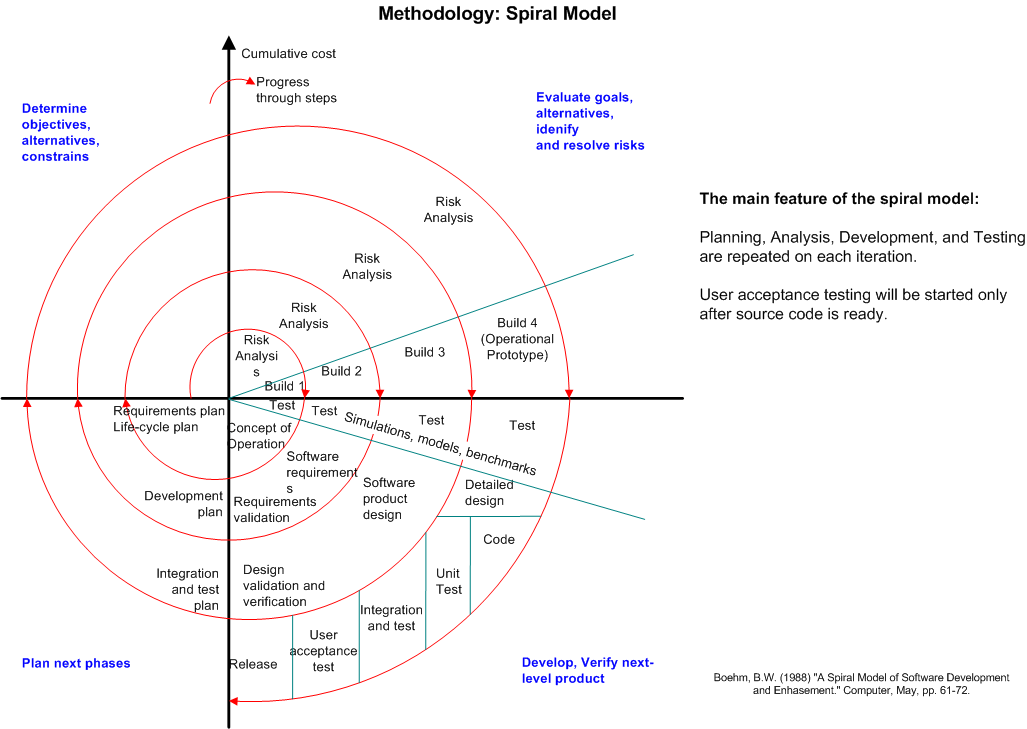
The class diagram shows a set of classes, interfaces, collaborations and their relationships.



# 6.7 Process Model

**Spiral Model**

DEFINITION - The spiral model, also known as the spiral lifecycle model, is a systems development method (SDM) used in information technology (IT). This model of development combines the features of the prototyping model and the [waterfall model](http://en.wikipedia.org/wiki/Waterfall_model). The spiral model is intended for large, expensive, and complicated projects.

[](http://myprojects.kostigoff.net/methodology/development_models/development_models.htm#spiral)

The steps in the spiral model can be generalized as follows:

1. A preliminary design is created for the new system.
2. A first [prototype](http://en.wikipedia.org/wiki/Prototype) of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
3. A second prototype is evolved by a fourfold procedure:

(I) evaluating the first prototype in terms of its strengths, weaknesses,

And risks;

(ii) Defining the requirements of the second prototype;

(iii) Planning and designing the second prototype;

(iv) Constructing and testing the second prototype.

1. At the customer's option, the entire project can be aborted if the risk is deemed too great. Risk factors might involve development cost overruns, operating-cost miscalculation, or any other factor that could, in the customer's judgment, result in a less-than-satisfactory final product.
2. The existing prototype is evaluated in the same manner as was the previous prototype, and, if necessary, another prototype is developed from it according to the fourfold procedure outlined above.
3. The preceding steps are iterated until the customer is satisfied that the refined prototype represents the final product desired.
4. The final system is constructed, based on the refined prototype.
5. The final system is thoroughly evaluated and tested. Routine maintenance is carried out on a continuing basis to prevent large-scale failures and to minimize downtime.

## Advantages

* Estimates (i.e. budget, schedule, etc.) get more realistic as work progresses, because important issues are discovered earlier.
* It is more able to cope with the (nearly inevitable) changes that software development generally entails.
* Software engineers (who can get restless with protracted design processes) can get their hands in and start working on a project earlier.

1. **CODING**

First phase of implementation is coding. Coding can be done in two ways. One by automatic program code and other by programmer’s manually written code. A code generator is a suite of programs that matches the input to an appropriate code template and from these produces modules of code.

The code is made simple in such a way that another programmer can easily understand and work on that in future. The crucial phase in the system development life cycle is the successful implementation of the new system design. The process of converting as new or revised system into an operational one is known as system implementation.

This includes all those activities that take place to convert from an old system to a new system. The system can be implemented only after a through testing is done and if it is found to work according to the specifications. The most crucial stage in achieving a new successful system and giving confident on the new system for the users is that it will work effectively and efficiently. If involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over.

Index.php

<?php

session\_start();

error\_reporting(0);

include('includes/config.php');

?>

<!DOCTYPE HTML>

<html>

<head>

<title>Amousement Park | BANGALORE</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<script type="applijewelleryion/x-javascript"> addEventListener("load", function() { setTimeout(hideURLbar, 0); }, false); function hideURLbar(){ window.scrollTo(0,1); } </script>

<link href="css/bootstrap.css" rel='stylesheet' type='text/css' />

<link href="css/style.css" rel='stylesheet' type='text/css' />

<link href='//fonts.googleapis.com/css?family=Open+Sans:400,700,600' rel='stylesheet' type='text/css'>

<link href='//fonts.googleapis.com/css?family=Roboto+Condensed:400,700,300' rel='stylesheet' type='text/css'>

<link href='//fonts.googleapis.com/css?family=Oswald' rel='stylesheet' type='text/css'>

<link href="css/font-awesome.css" rel="stylesheet">

<!-- Custom Theme files -->

<script src="js/jquery-1.12.0.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<!--animate-->

<link href="css/animate.css" rel="stylesheet" type="text/css" media="all">

<script src="js/wow.min.js"></script>

<script>

new WOW().init();

</script>

<!--//end-animate-->

</head>

<body>

<?php include('includes/header.php');?>

<div class="banner">

<div class="container">

<h1 class="wow zoomIn animated animated" data-wow-delay=".5s" style="visibility: visible; animation-delay: 0.5s; animation-name: zoomIn;"> Amousement Park - Bangalore </h1>

</div>

</div>

<!--- rupes ---->

<div class="container">

<div class="rupes">

<div class="col-md-4 rupes-left wow fadeInDown animated animated" data-wow-delay=".5s" style="visibility: visible; animation-delay: 0.5s; animation-name: fadeInDown;">

<div class="rup-left">

<a href="offers.html"><i class="fa fa-rupes"></i></a>

</div>

<div class="rup-rgt">

<h3>UP TO RS. 50 OFF</h3>

<h4><a href="offers.html">LIVE SMART</a></h4>

</div>

<div class="clearfix"></div>

</div>

<div class="col-md-4 rupes-left wow fadeInDown animated animated" data-wow-delay=".5s" style="visibility: visible; animation-delay: 0.5s; animation-name: fadeInDown;">

<div class="rup-left">

<a href="offers.html"><i class="fa fa-h-square"></i></a>

</div>

<div class="rup-rgt">

<h3>UP TO 70% OFF</h3>

<h4><a href="offers.html">ON HOTELS </a></h4>

</div>

<div class="clearfix"></div>

</div>

<div class="col-md-4 rupes-left wow fadeInDown animated animated" data-wow-delay=".5s" style="visibility: visible; animation-delay: 0.5s; animation-name: fadeInDown;">

<div class="rup-left">

<a href="offers.html"><i class="fa fa-mobile"></i></a>

</div>

<div class="rup-rgt">

<h3>FLAT USD. 50 OFF</h3>

<h4><a href="offers.html">US APP OFFER</a></h4>

</div>

<div class="clearfix"></div>

</div>

</div>

</div>

<!--- /rupes ---->

<div class = "content"><!--body content holder-->

<div class = "container">

<div class = "col-md-12"><!--body content title holder with 12 grid columns-->

<h1 class="weha">What we have</h1><!--body content title-->

</div>

</div>

<div class="container">

<div class="col-md-12">

<hr>

</div>

</div>

<div class="row"><!--wedding content-->

<section>

<div class="container">

<div class="col-md-6"><!--image holder with 6 grid columns-->

<img src="images/party1.jpg" class="img-responsive">

</div>

<div class="subcontent col-md-6"><!--Text holder with 6 column grid-->

<h1>Water Park</h1><!--title-->

<p><!--content-->

The most important day in a couple's life.

Guaranteeing personalized solutions and flawless execution, our venues provide the perfect

location for your special day.

</p>

</div><!--subcontent div-->

</div><!--container div-->

</section>

</div><!--row div-->

<div class="container">

<div class="col-md-12">

<hr>

</div>

</div>

<div class="row">

<section>

<div class="container">

<div class="col-md-6"><!--image holder with 6 grid columns-->

<img src="images/party2.jpg" class="img-responsive">

</div>

<div class="subcontent col-md-6"><!--Text holder with 6 column grid-->

<h1>Birthday Party</h1><!--title-->

<p><!--content-->

Whether an all-day or the ultimate extravaganza that

lasts well into the wee hours, our Urban Events team is here to make sure all your birthday

party wishes come true so you can kick back, drink up and enjoy your special day!

</p>

</div><!--subcontent div-->

</div><!--container div-->

</section>

</div><!--row div-->

<div class="container">

<div class="col-md-12">

<hr>

</div>

</div>

<div class="row">

<section>

<div class="container">

<div class="col-md-6"><!--image holder with 6 grid columns-->

<img src="images/party3.jpg" class="img-responsive">

</div>

<div class="subcontent col-md-6"><!--Text holder with 6 column grid-->

<h1>Fashion</h1><!--title-->

<p><!--content-->

Fast becoming to go-to location for fashion events, PR gatherings and product launches,

The Urban Purveyor Group venues provide you with choice and quality in premium locations

for all your event needs.

</p>

</div><!--subcontent div-->

</div><!--container div-->

</section>

</div><!--row div-->

<div class="container">

<div class="col-md-12">

<hr>

</div>

</div>

<div class="row">

<section>

<div class="container">

<div class="col-md-6"><!--image holder with 6 grid columns-->

<img src="images/party4.jpg" class="img-responsive">

</div>

<div class="subcontent col-md-6"><!--Text holder with 6 column grid-->

<h1>Meeting</h1><!--title-->

<p><!--content-->

From formal, to not-so-formal, our flexible event

spaces can cater to your every need for meetings and conferences large or small, and our

dedicated event team can assist with all aspects of your event planning.

</p>

</div><!--subcontent div-->

</div><!--container div-->

</section>

</div><!--row div-->

</div><!--body content div-->

</div>

<!---holiday---->

<div class="container">

<div class="holiday">

<h3>Package List</h3>

<?php $sql = "SELECT \* from tbltourpackages order by rand() limit 4";

$query = $dbh->prepare($sql);

$query->execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query->rowCount() > 0)

{

foreach($results as $result)

{ ?>

<div class="rom-btm">

<div class="col-md-3 room-left wow fadeInLeft animated" data-wow-delay=".5s">

<img src="admin/pacakgeimages/<?php echo htmlentities($result->PackageImage);?>" class="img-responsive" alt="">

</div>

<div class="col-md-6 room-midle wow fadeInUp animated" data-wow-delay=".5s">

<h4>Package Name: <?php echo htmlentities($result->PackageName);?></h4>

<h6>Package Type : <?php echo htmlentities($result->PackageType);?></h6>

<p><b>Package Location :</b> <?php echo htmlentities($result->PackageLocation);?></p>

<p><b>Features</b> <?php echo htmlentities($result->PackageFetures);?></p>

</div>

<div class="col-md-3 room-right wow fadeInRight animated" data-wow-delay=".5s">

<h5>RS <?php echo htmlentities($result->PackagePrice);?></h5>

<a href="package-details.php?pkgid=<?php echo htmlentities($result->PackageId);?>" class="view">Details</a>

</div>

<div class="clearfix"></div>

</div>

<?php }} ?>

<div><a href="package-list.php" class="view">View More Packages</a></div>

</div>

<div class="clearfix"></div>

</div>

<!--- routes ---->

<div class="routes">

<div class="container">

<div class="col-md-4 routes-left wow fadeInRight animated" data-wow-delay=".5s">

<div class="rou-left">

<a href="#"><i class="glyphicon glyphicon-list-alt"></i></a>

</div>

<div class="rou-rgt wow fadeInDown animated" data-wow-delay=".5s">

<h3>80000</h3>

<p>Enquiries</p>

</div>

<div class="clearfix"></div>

</div>

<div class="col-md-4 routes-left">

<div class="rou-left">

<a href="#"><i class="fa fa-user"></i></a>

</div>

<div class="rou-rgt">

<h3>1900</h3>

<p>Regestered users</p>

</div>

<div class="clearfix"></div>

</div>

<div class="col-md-4 routes-left wow fadeInRight animated" data-wow-delay=".5s">

<div class="rou-left">

<a href="#"><i class="fa fa-ticket"></i></a>

</div>

<div class="rou-rgt">

<h3>7,00,00,000+</h3>

<p>Booking</p>

</div>

<div class="clearfix"></div>

</div>

<div class="clearfix"></div>

</div>

</div>

<?php include('includes/footer.php');?>

<!-- signup -->

<?php include('includes/signup.php');?>

<!-- //signu -->

<!-- signin -->

<?php include('students/index.php');?>

<!-- //signin -->

<!-- write us -->

<?php include('includes/write-us.php');?>

<!-- //write us -->

</body>

</html>

**SYSTEM TESTING**

**SOFTWARE TESTING TECHNIQUES:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, designing and coding.

**TESTING OBJECTIVES:**

1. Testing is process of executing a program with the intent of finding an error.
2. A good test case design is one that has a probability of finding an as yet undiscovered error.
3. A successful test is one that uncovers an as yet undiscovered error.

These above objectives imply a dramatic change in view port.

Testing cannot show the absence of defects, it can only show that software errors are present.

There are three types of testing strategies

1. Unit test
2. Integration test

3. Performance test

**Unit Testing:**

Unit testing focuses verification efforts on the smallest unit of software design module. The unit test is always white box oriented. The tests that occur as part of unit testing are testing the module interface, examining the local data structures, testing the boundary conditions, execution all the independent paths and testing error-handling paths.

**Integration Testing:**

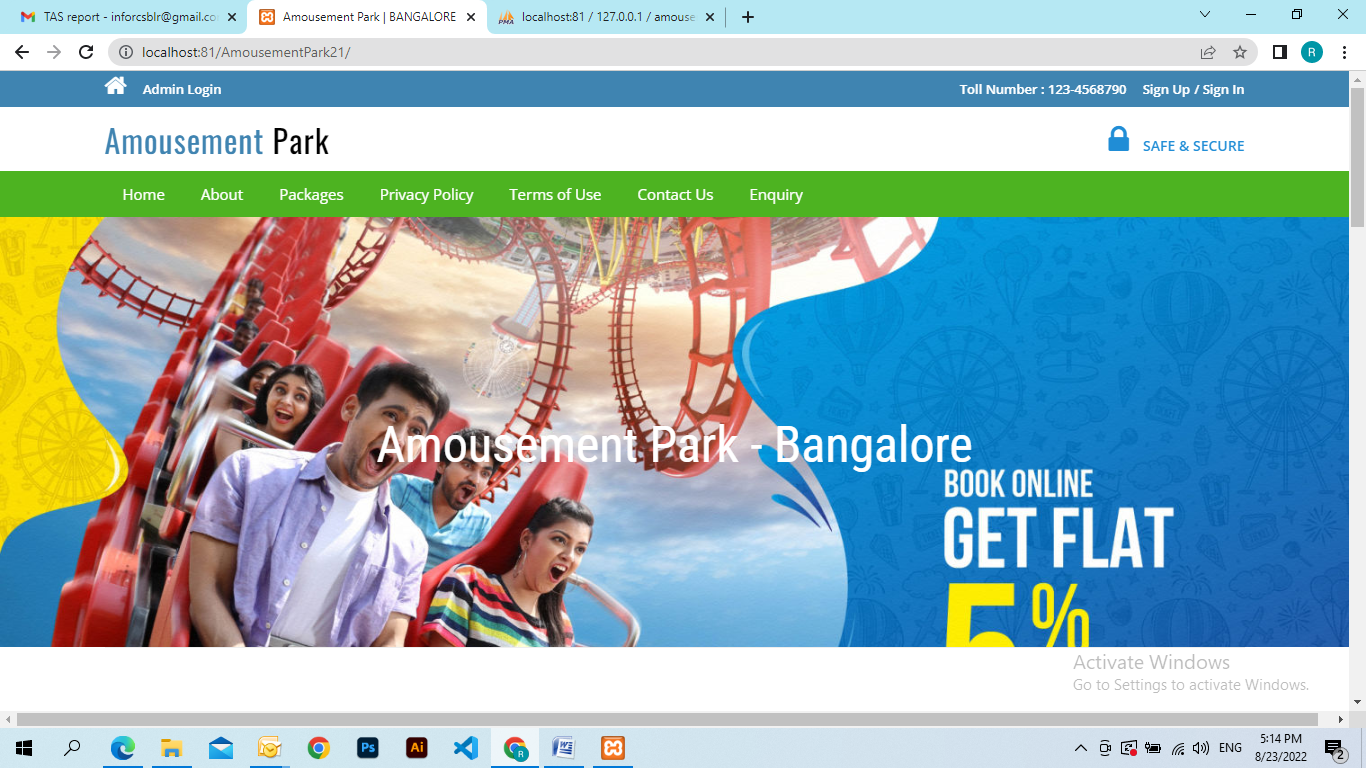
Integration testing is a systematic technique or construction the program structure while at the same time conducting tests to uncover errors associated with interfacing. Scope of testing summarizes the specific functional, performance, and internal design characteristics that are to be tested. It employs top-down testing and bottom-up testing methods for this case.

**Performance Testing:**

Timing for both read and update transactions should be gathered to determine whether system functions are being performed in an acceptable timeframe.

**Output Screen of Project**

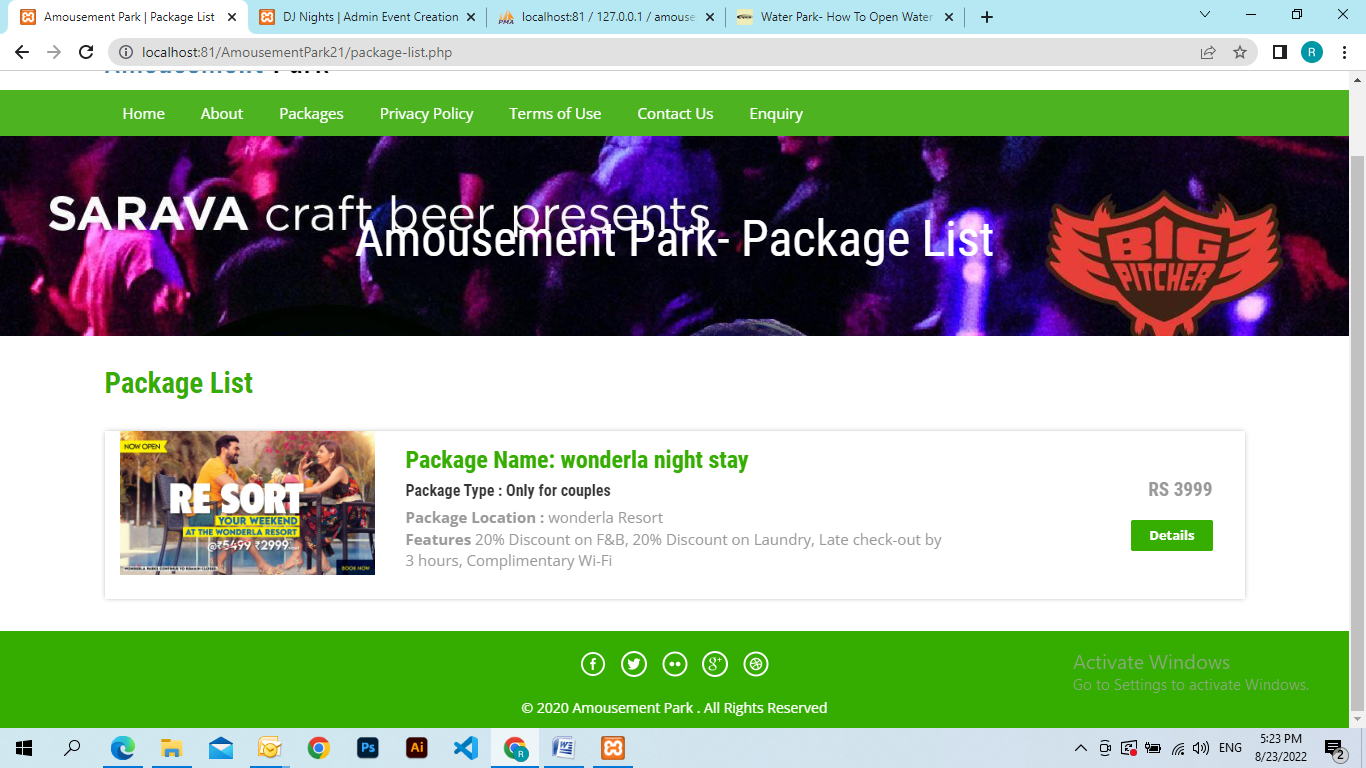
**Home Page**

****

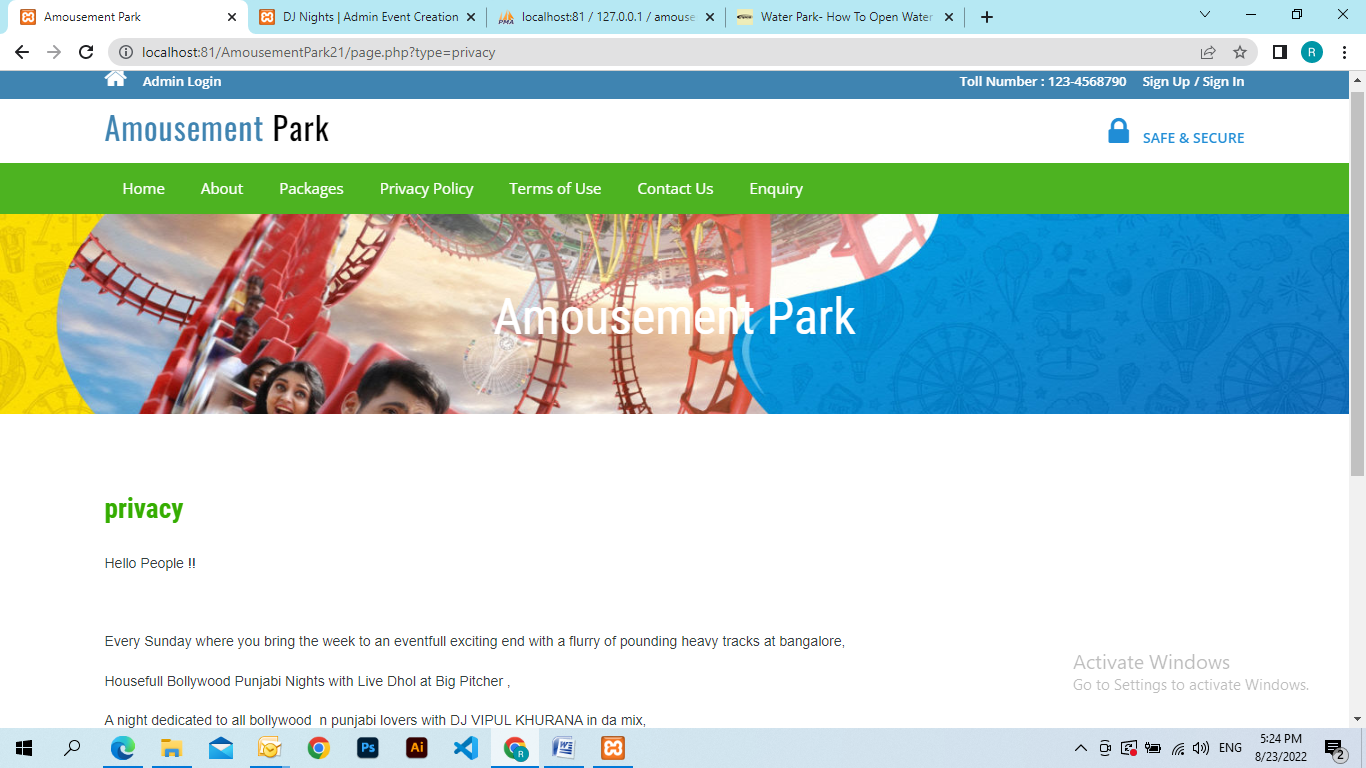
**About us**

****

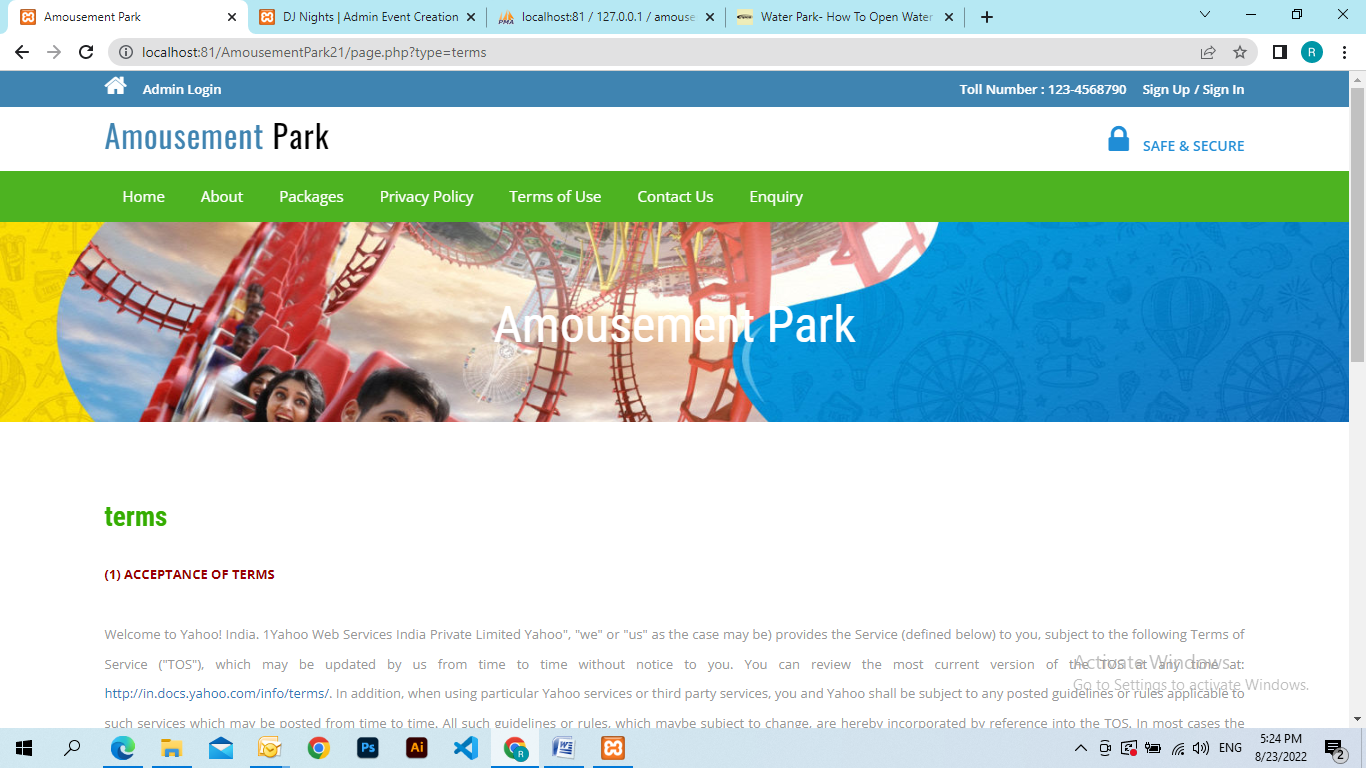
**Packages**

****

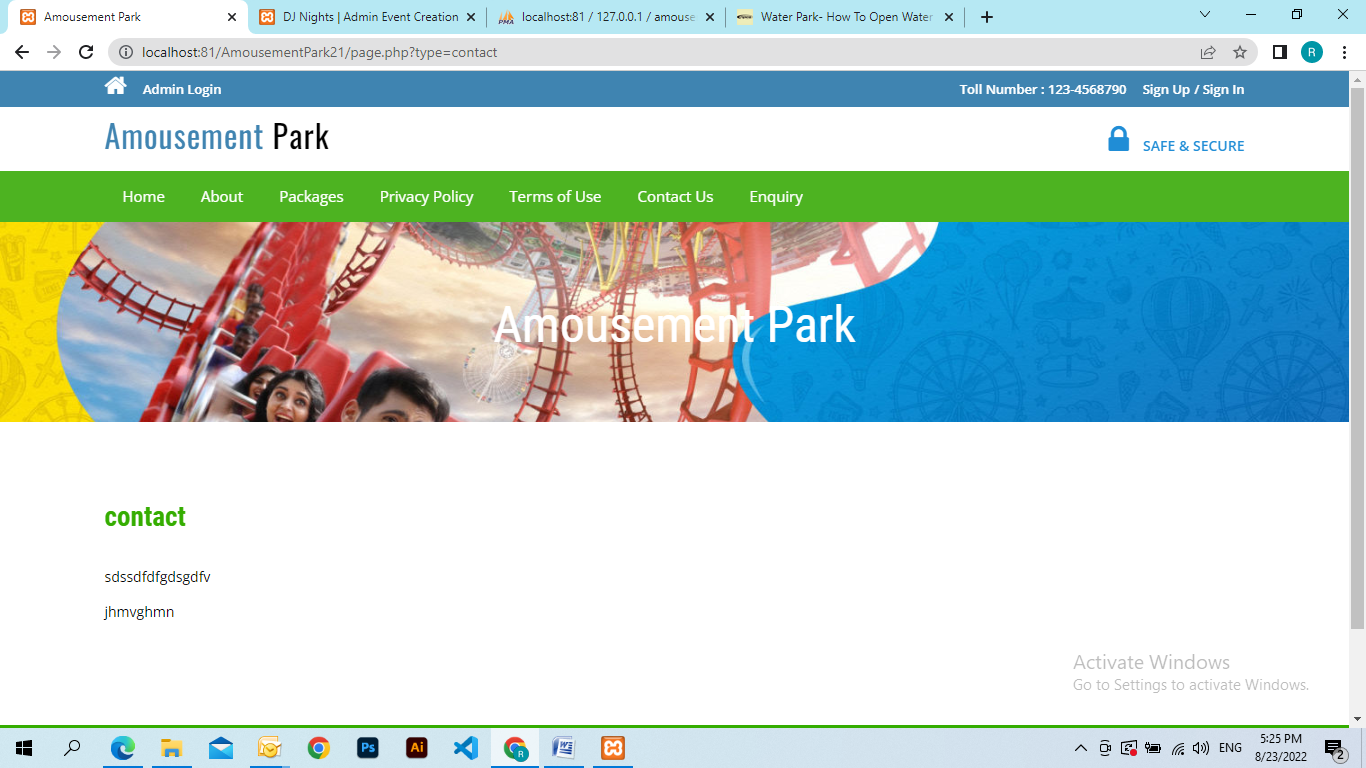
**Privacy**

****

**Tern of Uses**

****

**Contact**

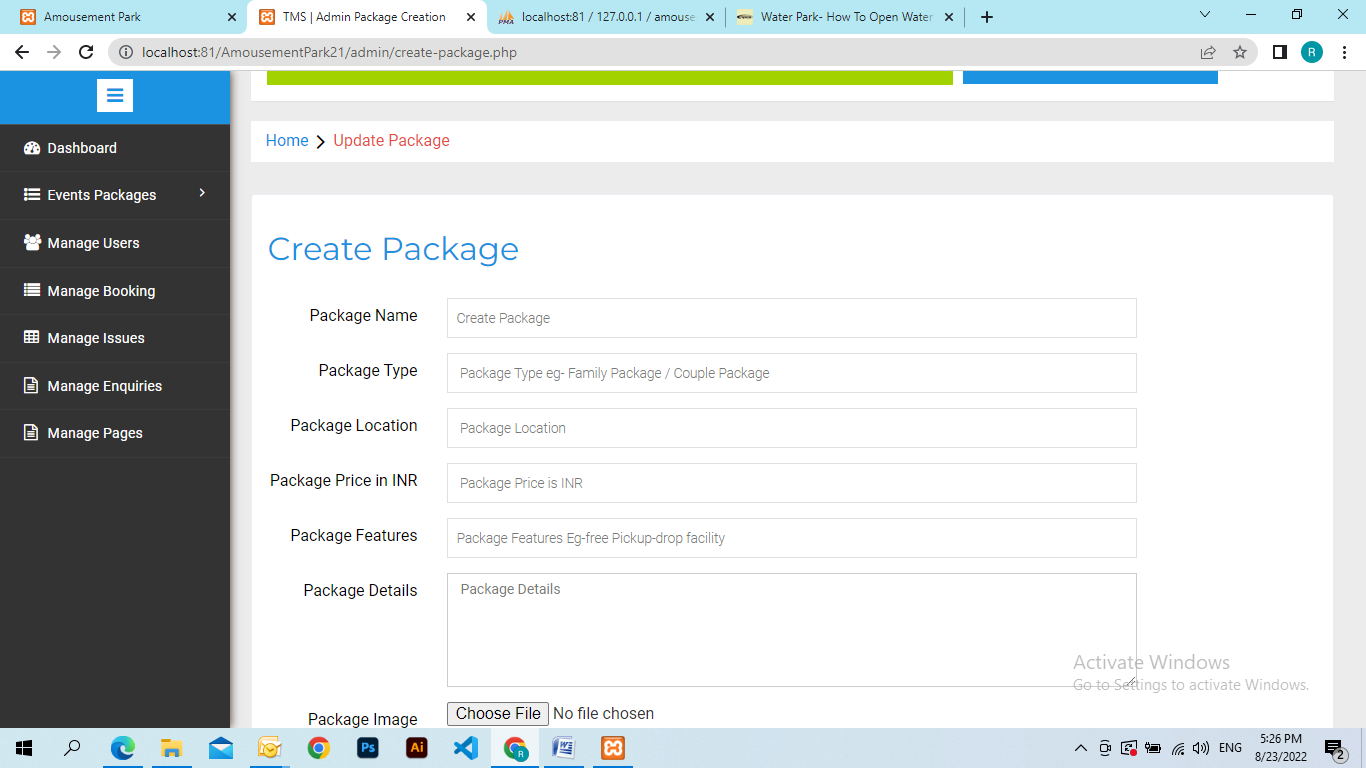
****

**Admin**

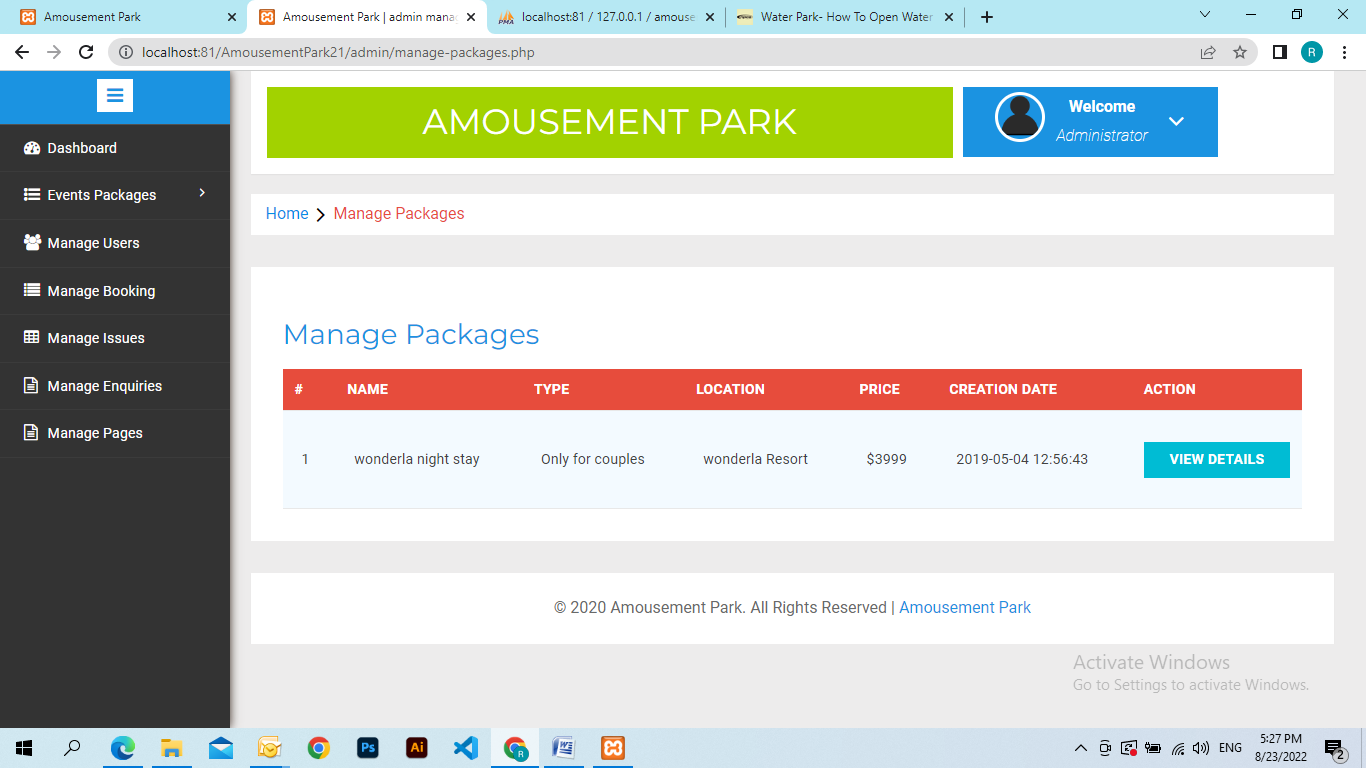
**Dashboard**

****

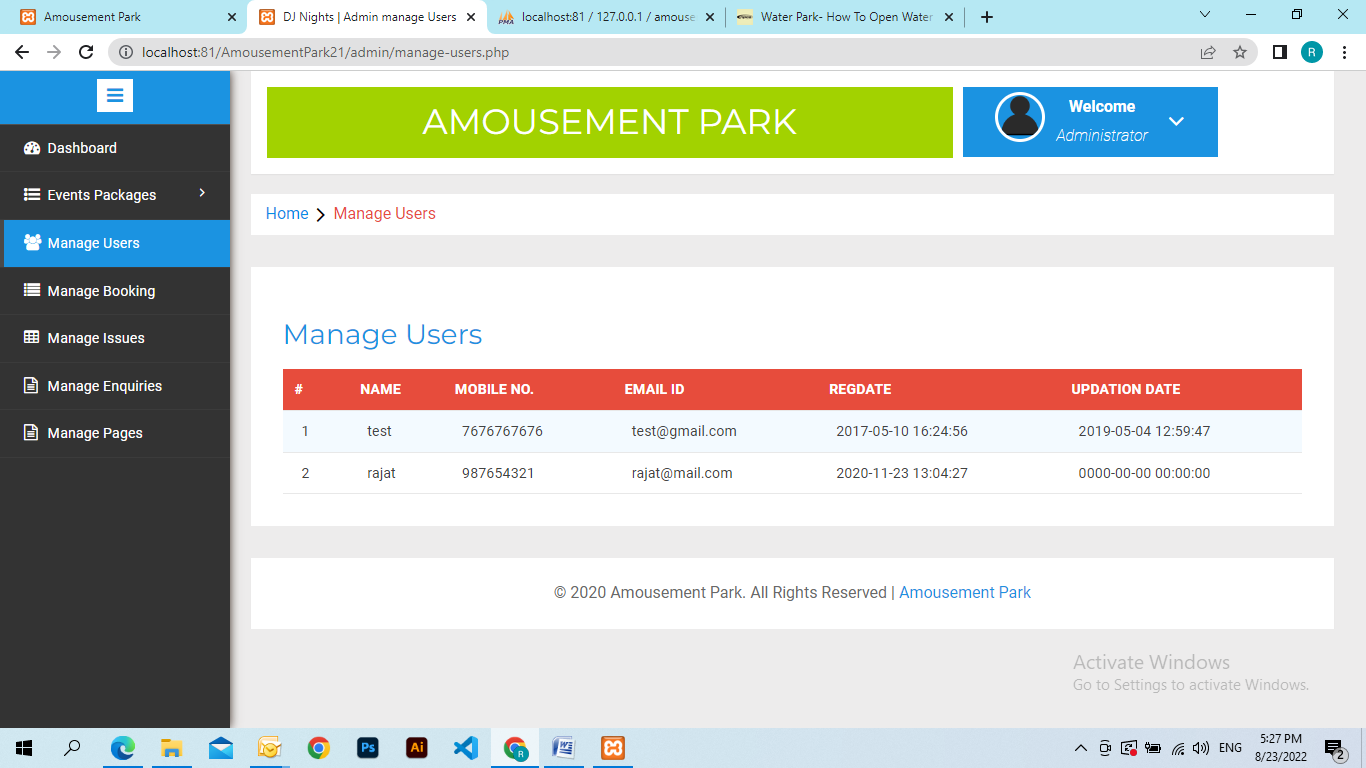
**Create Package**

****

**Manage Package**

****

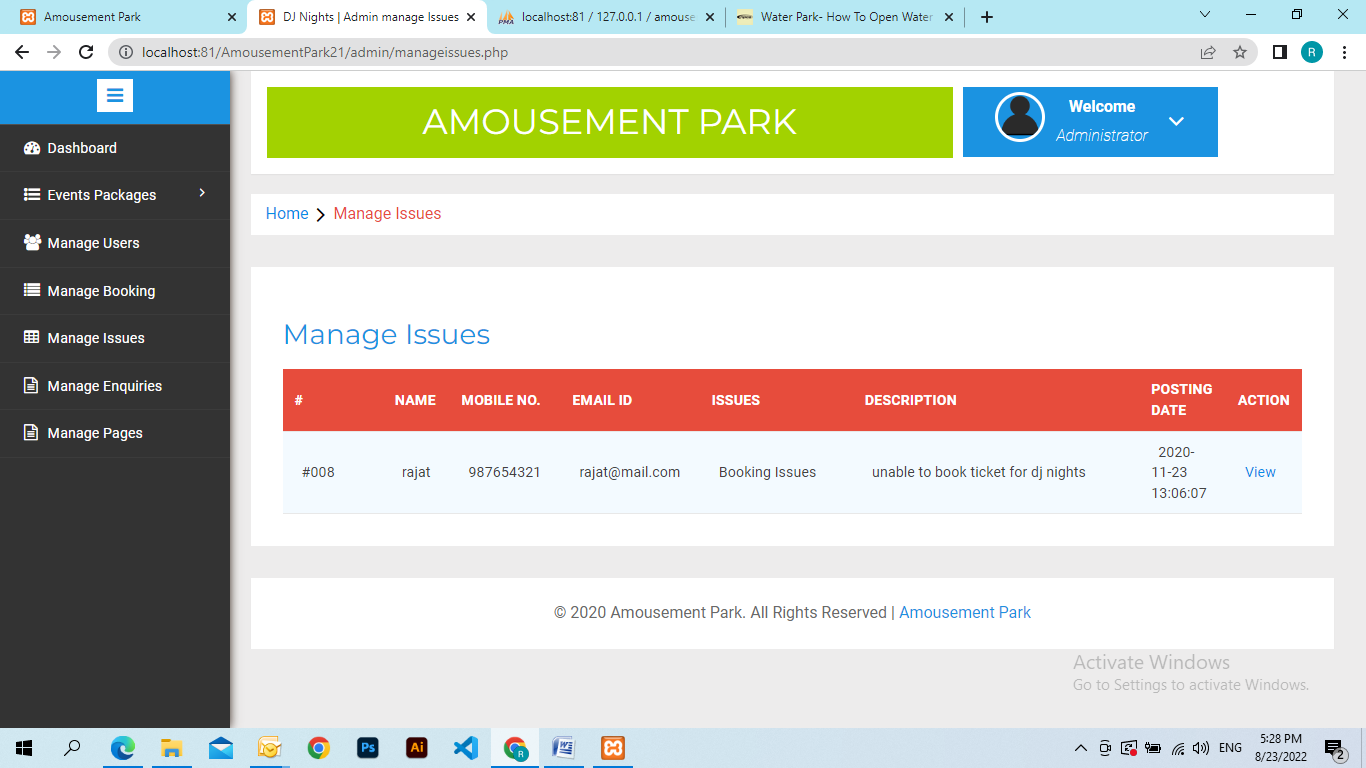
**Manage User**

****

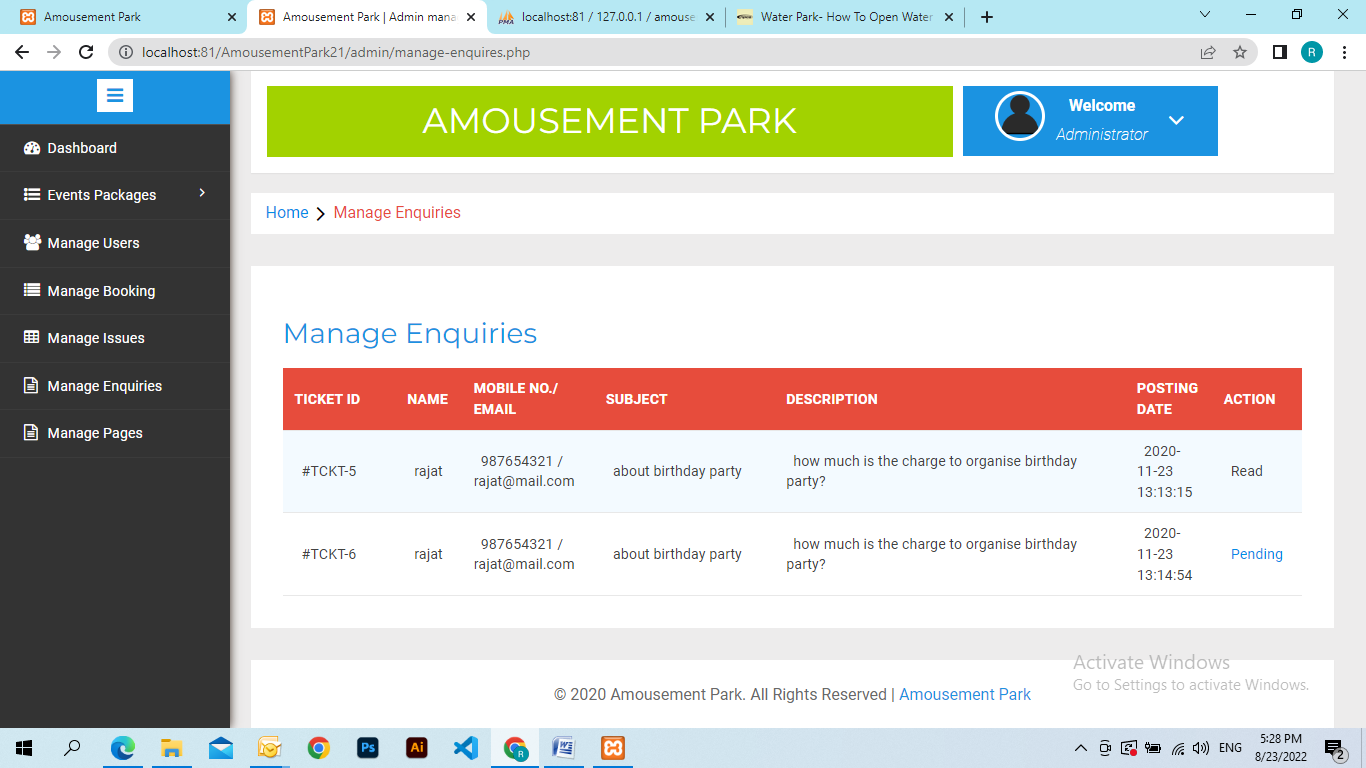
**Manage Booking**

****

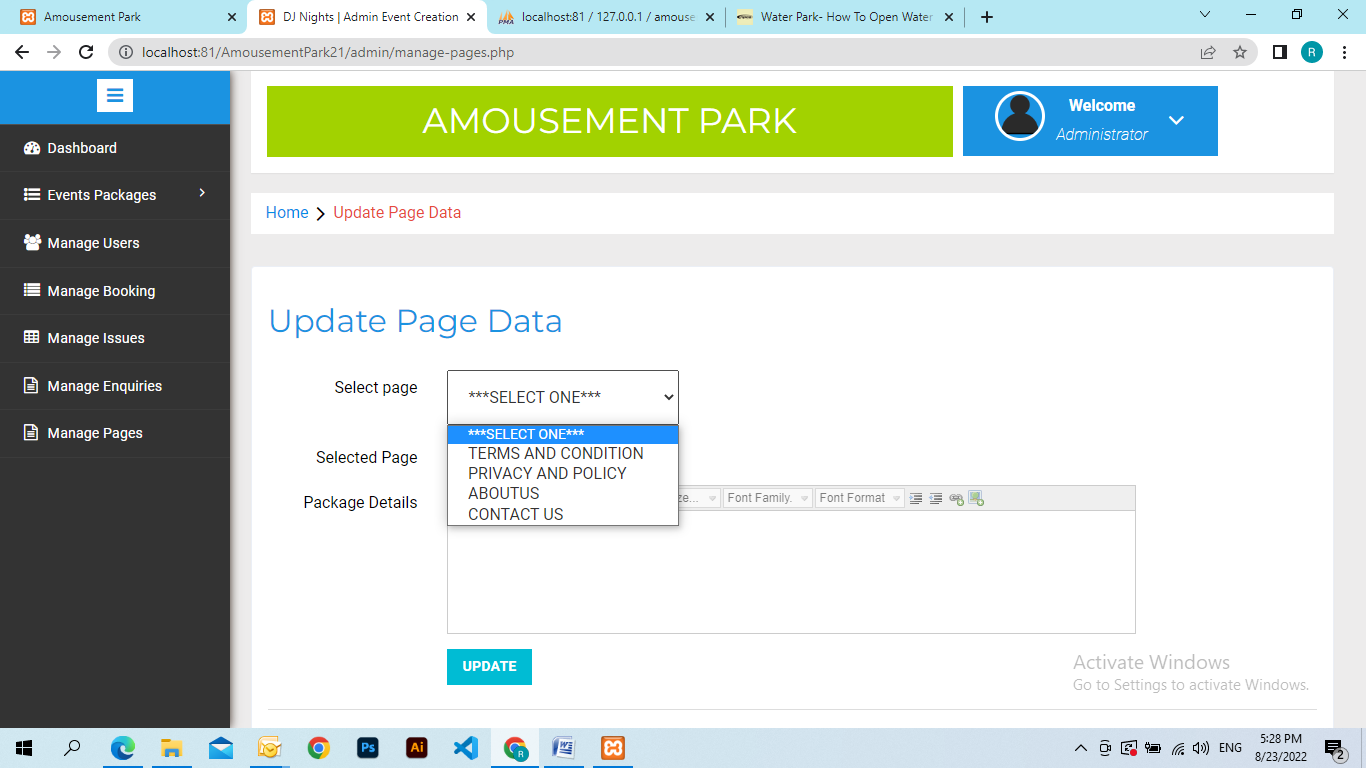
**Manage Issue**

****

**Manage Enquires**

****

**Manage Pages**

****

**Conclusion**

The project titled as **Amousepark Management System** was deeply studied and analyzed to design the code and implement. It was done under the guidance of the experienced project guide. All the current requirements and possibilities have been taken care during the project time.

**Amousement Management System** is a web based application which manages and handles the people ticket who visited in the park.

**Bibliography**

**For PHP**

* <https://www.w3schools.com/php/default.asp>
* <https://www.sitepoint.com/php/>
* <https://www.php.net/>

**For MySQL**

* <https://www.mysql.com/>
* [http://www.mysqltutorial.org](http://www.mysqltutorial.org/)

**For XAMPP**

* <https://www.apachefriends.org/download.html>

**Bibliography**

**For PHP**

* <https://www.w3schools.com/php/default.asp>
* <https://www.sitepoint.com/php/>
* <https://www.php.net/>

**For MySQL**

* <https://www.mysql.com/>
* [http://www.mysqltutorial.org](http://www.mysqltutorial.org/)

**For XAMPP**

* <https://www.apachefriends.org/download.html>