

**SHARDA SHAIKSHANIK & SAMAJIK SANSTHAS**

**VIDHYDHAN COMMERCE &  
SCIENCE COLLEGE  
WALWADI, DHULE(200111)**



**A PROJECT REPORT ON**

**“Student Management system “**

**Submitted By**

**Mr.Pratik Nahanbhavu Patil**  
**PRN No:-201901540032756**

**GUIDE BY**

**PROF. MRS. LALITA NIKAM**

**IN PARTIAL FULLFILLMENT OF  
DEGREE OF BACHORAL OF COMPUTER  
APPLICATION (BCA)**



**Kavayitri Bahinabai Chaudhari**  
**North Maharashtra University, Jalgaon**





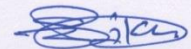
## Certificate


This is to certify that **Mr. Pratik Nhanbhau Patil** final year student of Bachelor in Bachelor of Computer Applications (**BCA**) has been successfully complete these project entitled "**Student Management System**" as the record of the work carried out by his is accepted in partial fulfillment of the requirement for degree of Bachelor in Bachelor of Computer Applications (**BCA**) in the Bachelor of Computer Applications (**BCA**) vidhyadhan Commerce & Science college Dhule under the guidance of **Prof. Mrs. Lalita Baburao Nikam** in the academic year 2021-2022.

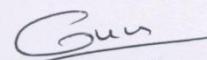
Place: - Dhule  
Date: -

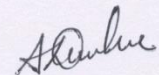
Prof.   
(Project Guide)



  
Head of Department  
H.O.D.  
Computer & Management  
Vidyadhan College, Valwadi

  
Internal Examiner

  
Principal  
Sharda Educ. & Social Sanstha's  
Vidyadhan Com. & Sci College  
Valwadi (Dhule)

  
External Examiner

# INDEX

# **"Student Management System"**

## **Index**

- **Introduction**
- **Requirement Analysis**
- **Module Description**
- **Design**
- **Database**
- **Testing**
- **Input Screen**
- **Output Screen**
- **Conclusion**
- **Bibliography**

# INTRODUCTION

## **Introduction**

Talking about the features of the Online Student Management System, the user can view announcements, recent posts, and departments. From the instructor account, the user can view the student's list and take actions on their grades by adding on each.

Talking about the admin panel, the admin has full access to the system. He/she can manage

students, payments, instructor, make announcements. From maintenance section of the admin panel, the admin can set semester, department, courses, curriculum, school year, year level and section. It also contains user management for the easy management of the system. He/she can also backup and restore the database, and view daily, weekly and monthly report.

Design of this project is pretty simple so that the user won't find any difficulties while working on it. Online Student Management System in PHP helps the student in managing timetables easily. To run this project you must have installed virtual server i.e XAMPP on your PC (for Windows). Online Student Management System in PHP with source code is free to download, Use for educational purposes only! For the project demo, have a look at the Video below. Managing an academic institution and bringing together various bodies associated with the institution to accomplish the mission is always a big challenge. Transforming these institutions with technology-enabled automation tools to facilitate the academic and administrative activities will make it easier to accomplish the mission. To manage everyday tasks of institutions, education must be modernized with cloud, mobile and digital technologies to enhance operational efficiency and also to manage the institution effectively.

Students/guardians no more have to stand for hours in the queue to get admission. Simplified registration and collection of fee through online forms, with the ability to send alerts and reminders via email, SMS alerts and push notifications makes the process easier.

Design of a course curriculum that can be adapted to the changing needs of the academic institution is essential. Having Course Management System institutions will be able to accomplish a lot with limited resources. Creating and tracking the course-work, assignments, and exam papers in a classroom environment to support the goal of graduating students.

Tracking of teachers progress and assessing the effectiveness of teachers work is crucial. The teachers performance can be evaluated based on the students feedback. Automated evaluation process help students to improve their learning skills, achievement and success.

The student discipline incidents are increasing because of communication gap between students and teachers. There is a need for platform to provide seamless communication between students, administrators, staff and teachers. Web based management system improves communication through notifications, email, SMS and push messages.

# REQUIREMENT ANALYSIS

## **Requirement Analysis**

### **Software Requirement**

- Project Name                      Student Management System in php



- Language Used PHP5.6, PHP7.x
- Database MySQL 5.x
- User Interface Design HTML, AJAX,JQUERY,JAVASCRIPT
- Web Browser Mozilla, Google Chrome, IE8, OPERA
- Software XAMPP / Wamp / Mamp/ Lamp (anyone)

## **Hardware Requirement**

### **◆ Hardware Requirement**

- P-III or higher Processor.
- 40 GB HDD
- 256 SDD
- 8 GB RAM (min)
- Color VGA 800x600 resolution monitor

## **Feasibility Study**

### **Technical Feasibility:**

The project work can be done with the current equipments & existing software and

available personal. The proposed system needs no new equipment installed at the users end. Technically this is very much safe.

### **Operational Feasibility:**

As system is very user friendly so user of the system can easily work and interact with the system operations of this system are similar to an existing system.

### **Economic Feasibility:**

There is no cost involve in successful execution of this software since it is self-contained, standalone program. It doesn't require any software to process on it. Additional hardware support other than the conventional hardware is not required. Due to ease of operation of the system no need of trained programmer is required.

### **Legal Feasibility**

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like zoning laws, data protection acts or social media laws. Let's say an organization wants to construct a new office building in a specific location. A feasibility study might reveal the organization's ideal location isn't zoned for that type of business. That organization has just saved considerable time and effort by learning that their project was not feasible right from the beginning.

### **Operational Feasibility**

This assessment involves undertaking a study to analyze and determine whether—and how well—the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

### **Scheduling Feasibility**

This assessment is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, an organization estimates how much time the project will take to complete.

### **PHP**

The term PHP is an acronym for *PHP: Hypertext Preprocessor*. PHP is a server-side scripting language designed specifically for web development. It is open-source which means it is free to download and use. It is very simple to learn and use. The files have the extension ".php".

Rasmus Lerdorf inspired the first version of PHP and participating in the later versions. It is an interpreted language and it does not require a compiler.

- PHP code is executed in the server.

- It can be integrated with many databases such as Oracle, Microsoft SQL Server, MySQL, PostgreSQL, Sybase, Informix.
- It is powerful to hold a content management system like WordPress and can be used to control user access.
- It supports main protocols like HTTP Basic, HTTP Digest, IMAP, FTP, and others.
- Websites like www.facebook.com, www.yahoo.com are also built on PHP.
- One of the main reasons behind this is that PHP can be easily embedded in HTML files and HTML codes can also be written in a PHP file.
- The thing that differentiates PHP from the client-side language like HTML is, PHP codes are executed on the server whereas HTML codes are directly rendered on the browser. PHP codes are first executed on the server and then the result is returned to the browser.
- The only information that the client or browser knows is the result returned after executing the PHP script on the server and not the actual PHP codes present in the PHP file. Also, PHP files can support other client-side scripting languages like CSS and JavaScript.

### **Common uses of PHP**

- PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You add, delete, modify elements within your database through PHP.
- Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website.
- It can encrypt data.

### **Characteristics of PHP**

Five important characteristics make PHP's practical nature possible –

- Simplicity
- Efficiency
- Security
- Flexibility
- Familiarity

**To further fortify your trust in PHP, here are a few applications of this amazing scripting language:**

1. It can be used to **create Web applications** like Social Networks(Facebook, Digg), Blogs(Wordpress, Joomla), eCommerce websites(OpenCart, Magento etc.) etc.
2. **Comman Line Scripting**. You can write PHP scripts to perform different operations on any machine, all you need is a PHP parser for this.

3. **Create Facebook applications** and easily integrate Facebook plugins in your website, using Facebook's PHP SDK. Check this [link](#) for more information.
4. **Sending Emails** or building email applications because PHP provides with a robust email sending function.
5. Wordpress is one of the most used blogging(CMS) platform in the World, and if you know PHP, you can try a hand in **Wordpress plugin development**.

### PHP is a cross-platform language

PHP can run on all major operating systems, including Linux, Windows, and macOS.

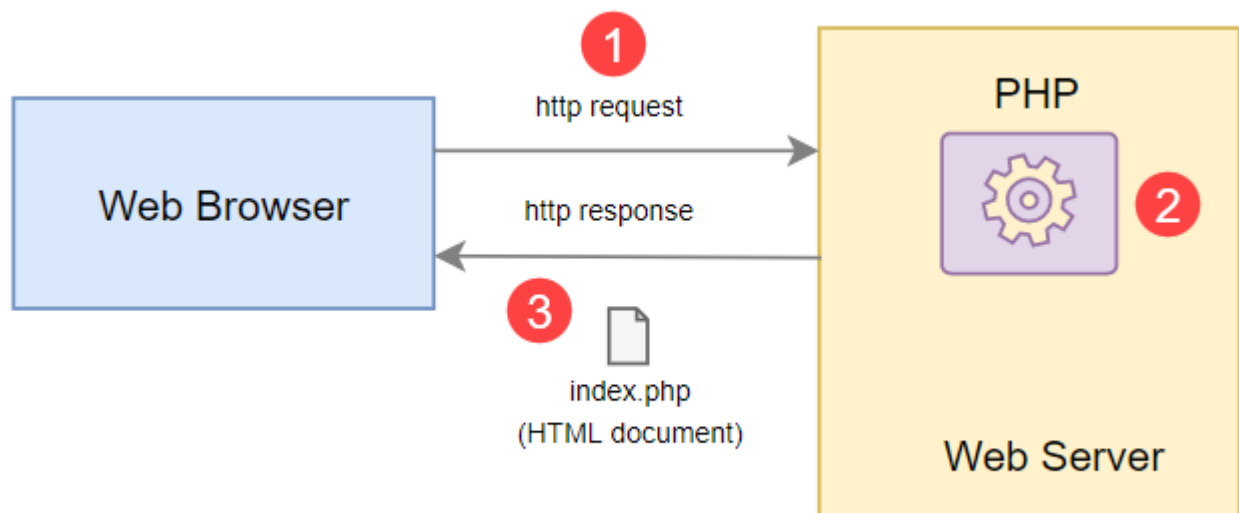
You can use PHP with all leading web servers such as Nginx, OpenBSD, and Apache. Some cloud environments also support PHP like Microsoft Azure and Amazon AWS.

PHP is quite flexible. It's not just limited to processing HTML. PHP has built-in support for generating PDF, GIF, JPEG, and PNG images.

One notable feature of PHP is that it supports many databases, including MySQL, PostgreSQL, MS SQL, db2, Oracle Database, and MongoDB.

### How PHP Works

The following illustrates how PHP works:



How PHP works:

- First, the web browser sends an HTTP request to the web server, e.g., index.php.
- Second, the PHP preprocessor that locates on the web server processes PHP code to generate the HTML document.
- Third, the web server sends the HTML document back to the web browser.

Advantages of PHP

Since PHP is designed for the web in the first place, it brings many advantages to web development:

- Simple – PHP is quite easy to learn and get started.
- Fast – PHP websites typically run very fast.
- Stable – PHP is stable since it has been in existence for a long time.
- Open-source and free – PHP is open source and free. It means that you don't have to pay a

license fee to use PHP to develop software products.

- Community support – PHP has an active online community that helps you whenever you face an issue.

## MySQL

MySQL tutorial provides basic and advanced concepts of MySQL. Our MySQL tutorial is designed for beginners and professionals.

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by **Oracle Company**.

Our MySQL tutorial includes all topics of MySQL database that provides for how to manage database and to manipulate data with the help of various SQL queries. These queries are: insert records, update records, delete records, select records, create tables, drop tables, etc. There are also given MySQL interview questions to help you better understand the MySQL database.

### What is Database?

It is very important to understand the database before learning MySQL. A database is an application that stores the organized collection of records. It can be accessed and managed by the user very easily. It allows us to organize data into tables, rows, columns, and indexes to find the relevant information very quickly. Each database contains distinct API for performing database operations such as creating, managing, accessing, and searching the data it stores. Today, many databases are available like MySQL, Sybase, Oracle, MongoDB, PostgreSQL, SQL Server, etc. In this section, we are going to focus on MySQL mainly.

### What is MySQL?

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

It is developed, marketed, and supported by **MySQL AB, a Swedish company**, and written in C programming language and C++ programming language. The official pronunciation of MySQL is not the My Sequel; it is **My Ess Que Ell**. However, you can pronounce it in your way. Many small and big companies use MySQL. MySQL supports many Operating Systems like Windows, Linux, MacOS, etc. with C, C++, and Java languages.

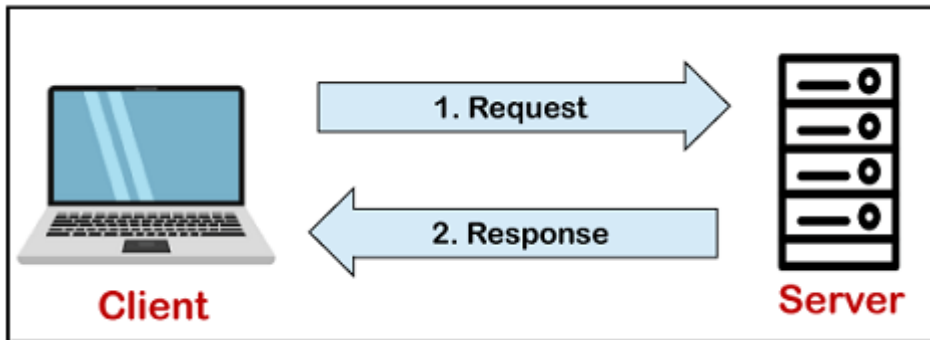
MySQL is a Relational Database Management System (RDBMS) software that provides many things, which are as follows:

- It allows us to implement database operations on tables, rows, columns, and indexes.
- It defines the database relationship in the form of tables (collection of rows and columns), also known as relations.
- It provides the Referential Integrity between rows or columns of various tables.
- It allows us to update the table indexes automatically.

- It uses many SQL queries and combines useful information from multiple tables for the end-users.

### How MySQL Works?

MySQL follows the working of Client-Server Architecture. This model is designed for the end-users called clients to access the resources from a central computer known as a server using network services. Here, the clients make requests through a graphical user interface (GUI), and the server will give the desired output as soon as the instructions are matched. The process of MySQL environment is the same as the client-server model.



The core of the MySQL database is the MySQL Server. This server is available as a separate program and responsible for handling all the database instructions, statements, or commands. The working of MySQL database with MySQL Server are as follows:

1. MySQL creates a database that allows you to build many tables to store and manipulate data and defining the relationship between each table.
2. Clients make requests through the GUI screen or command prompt by using specific SQL expressions on MySQL.
3. Finally, the server application will respond with the requested expressions and produce the desired result on the client-side.

A client can use any MySQL GUI. But, it is making sure that your GUI should be lighter and user-friendly to make your data management activities faster and easier. Some of the most widely used MySQL GUIs are MySQL Workbench, SequelPro, DBVisualizer, and the Navicat DB Admin Tool. Some GUIs are commercial, while some are free with limited functionality, and some are only compatible with MacOS. Thus, you can choose the GUI according to your needs.

### Reasons for popularity

MySQL is becoming so popular because of these following reasons:

- MySQL is an open-source database, so you don't have to pay a single penny to use it.
- MySQL is a very powerful program that can handle a large set of functionality of the most expensive and powerful database packages.
- MySQL is customizable because it is an open-source database, and the open-source GPL license facilitates programmers to modify the SQL software according to their own specific environment.
- MySQL is quicker than other databases, so it can work well even with the large data set.
- MySQL supports many operating systems with many languages like PHP, PERL, C, C++, JAVA, etc.

- MySQL uses a standard form of the well-known SQL data language.
- MySQL is very friendly with PHP, the most popular language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

### History of MySQL

The project of MySQL was started in 1979 when MySQL's inventor **Michael Widenius** developed an in-house database tool called **UNIREG** for managing databases. After that, UNIREG has been rewritten in several different languages and extended to handle big databases. After some time, Michael Widenius contacted **David Hughes**, the author of mSQL, to see if Hughes would be interested in connecting mSQL to UNIREG's B+ ISAM handler to provide indexing to mSQL. That's the way MySQL came into existence.

### HTML

**HTML** (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

HTML uses "markup" to annotate text, images, and other content for display in a Web browser.

HTML markup includes special "elements" such

as <head>, <title>, <body>, <header>, <footer>, <article>, <section>, <p>, <div>, <span>, <img>, <aside>, <audio>, <canvas>, <datalist>, <details>, <embed>, <nav>, <output>, <progress>, <video>, <ul>, <ol>, <li> and many others.

An HTML element is set off from other text in a document by "tags", which consist of the element name surrounded by "<" and ">". The name of an element inside a tag is case insensitive. That is, it can be written in uppercase, lowercase, or a mixture. For example, the <title> tag can be written as <Title>, <TITLE>, or in any other way. However, the convention and recommended practice is to write tags in lowercase.

### AJAX

Ajax stands for **A**synchronous **J**avascript **A**nd **X**ml. Ajax is just a means of loading data from the server and selectively updating parts of a web page without reloading the whole page.

Basically, what Ajax does is make use of the browser's built-in XMLHttpRequest (XHR) object to send and receive information to and from a web server asynchronously, in the background, without blocking the page or interfering with the user's experience.

Ajax has become so popular that you hardly find an application that doesn't use Ajax to some extent. The example of some large-scale Ajax-driven online applications is: Gmail, Google Maps, Google Docs, YouTube, Facebook, Flickr, and so many other applications.

### JavaScript

**JavaScript** is a lightweight, interpreted **programming language**. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. **JavaScript** is very easy to implement because it is integrated with HTML. It is open and cross-platform

**Javascript** is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning Javascript:

- Javascript is the most popular **programming language** in the world and that makes it a programmer's great choice. Once you learnt Javascript, it helps you developing great front-end as well as back-end softwares using different Javascript based frameworks like jQuery, Node.JS etc.
- Javascript is everywhere, it comes installed on every modern web browser and so to learn Javascript you really do not need any special environment setup. For example Chrome, Mozilla Firefox , Safari and every browser you know as of today, supports Javascript.
- Javascript helps you create really beautiful and crazy fast websites. You can develop your website with a console like look and feel and give your users the best Graphical User Experience.
- JavaScript usage has now extended to mobile app development, desktop app development, and game development. This opens many opportunities for you as Javascript Programmer.
- Due to high demand, there is tons of job growth and high pay for those who know JavaScript. You can navigate over to different job sites to see what having JavaScript skills looks like in the job market.
- Great thing about Javascript is that you will find tons of frameworks and Libraries already developed which can be used directly in your software development to reduce your time to market.

There could be 1000s of good reasons to learn Javascript Programming. But one thing for sure, to learn any **programming language**, not only Javascript, you just need to code, and code and finally code until you become expert.



# MODULE DESCRIPTION

## **Module Description**

The two main users involved in this system are

- 1.** User(i.e. Students)
- 2.** Admin

**Admin:**

1. **Dashboard:** In this section, admin can see all detail in brief like Total Classes, Total Students, Total Class Notices and Total Public Notices.
2. **Class:** In this section, admin can manage class (Add/Update/Delete).
3. **Students:** In this section, admin can manage the students (Add/Update/Delete).
4. **Notices:** In this section, the admin can manage notices (Add/Update/Delete).
5. **Public Notices:** In this section, the admin can manage public notices.
6. **Pages:** In this section admin, can manage about us and contact us page of administration
7. **Search:** In this section admin, can search students by their student id.
8. **Reports:** In this section admin, can view how much students has been register in particular period.
9. Admin can also update his profile, change the password and recover the password.

#### **User (Students):**

1. **Dashboard:** It is welcome page for students.
2. **View Notices:** In this section, user can view notices which are announced by administrator.
3. Student can also view his profile, change the password and recover the password.

#### **User (Non-Register):**

1. **.Home:** It is welcome page for user.
2. **About:** User can view about us page.
3. **Contact:** User can view contact us page.

# DESIGN

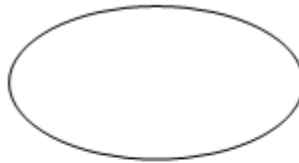
## **Design**

A DFD represents flow of data through a system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms the input into desired output. A DFD shows movement of data through the different transformations or processes in the system.

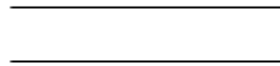
Dataflow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to restock how any system is developed can be determined through a dataflow diagram. The appropriate register saved in database and maintained by appropriate authorities.

### **Data Flow Diagram Notation:**

Function/Process



File/Database



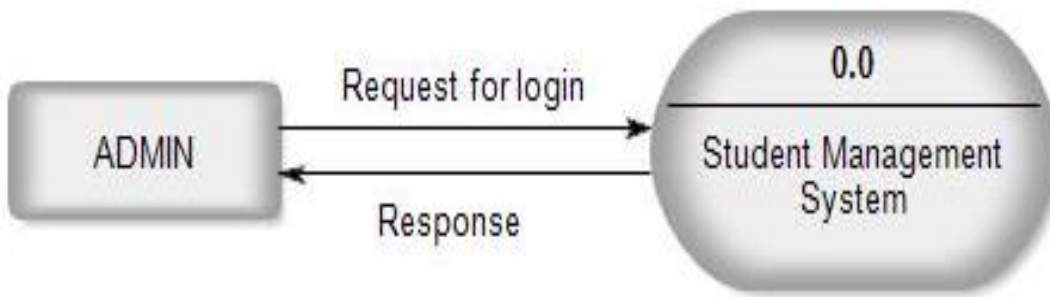
Input/output



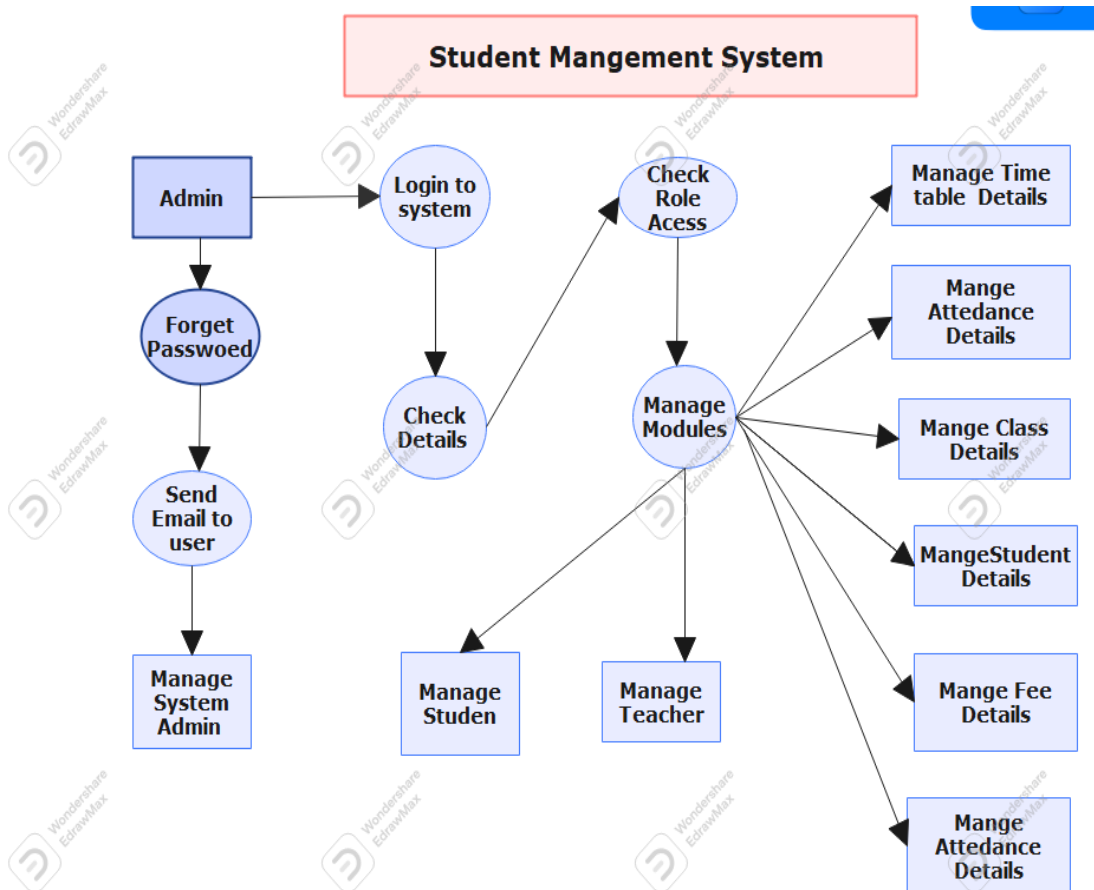
Flow



**First Level Data Flow Diagram**



## Second Level Data flow diagram



## ER-Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the

logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.

At first look, an ER diagram looks very similar to the flowchart. However, ER Diagram includes many specialized symbols, and its meanings make this model unique. The purpose of ER Diagram is to represent the entity framework infrastructure.

ER diagrams are related to data structure diagrams (DSDs), which focus on the relationships of elements within entities instead of relationships between entities themselves. ER diagrams also are often used in conjunction with data flow diagrams (DFDs), which map out the flow of information for processes or systems.

### Common Entity Relationship Diagram Symbols

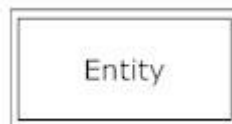
An ER diagram is a means of visualizing how the information a system produces is related. There are five main components of an ERD:

- **Entities**, which are represented by rectangles. An entity is an object or concept about



which you want to store information.

A weak entity is an entity that must be defined by a foreign key relationship with another entity as it cannot be uniquely identified



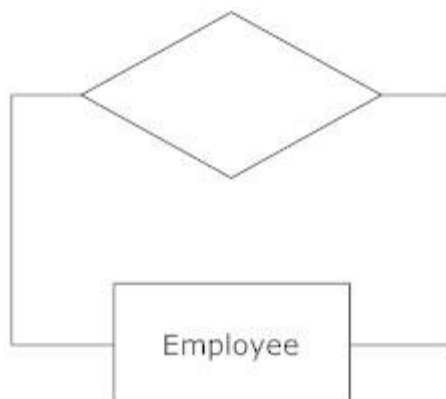
by its own attributes alone.

- **Actions**, which are represented by diamond shapes, show how two entities share



information in the database.

In some cases, entities can be self-linked. For example, employees can supervise other employees.



- **Attributes**, which are represented by ovals. A key attribute is the unique, distinguishing characteristic of the entity. For example, an employee's social security number might be the employee's key attribute.



A multivalued attribute can have more than one value. For example, an



employee entity can have multiple skill values.

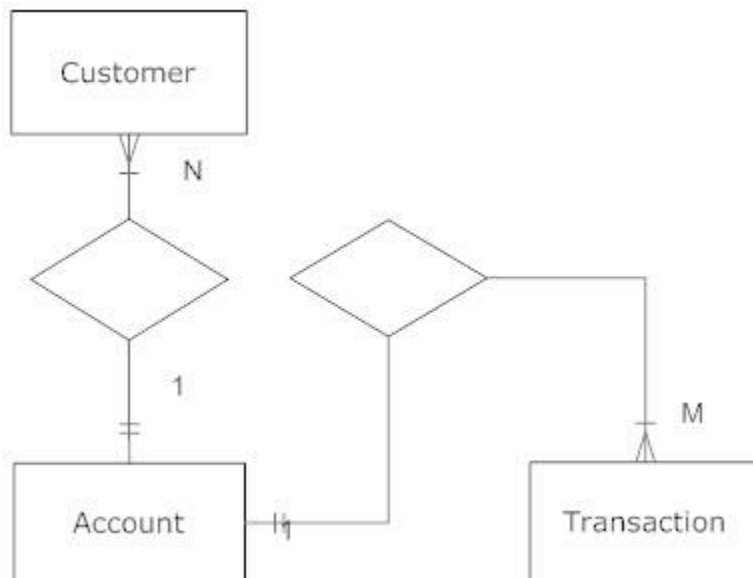
A derived attribute is

based on another attribute. For example, an employee's monthly salary is based on the



employee's annual salary.

- **Connecting lines**, solid lines that connect attributes to show the relationships of entities in the diagram.
- **Cardinality** specifies how many instances of an entity relate to one instance of another entity. Ordinality is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinality describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinality specifies the absolute minimum number of relationships.

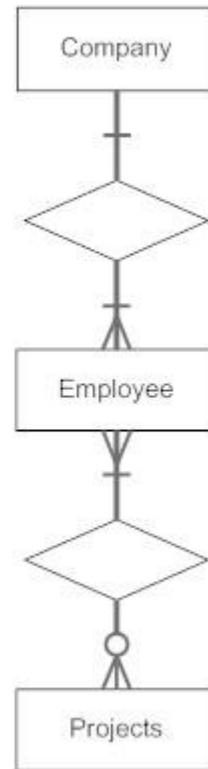
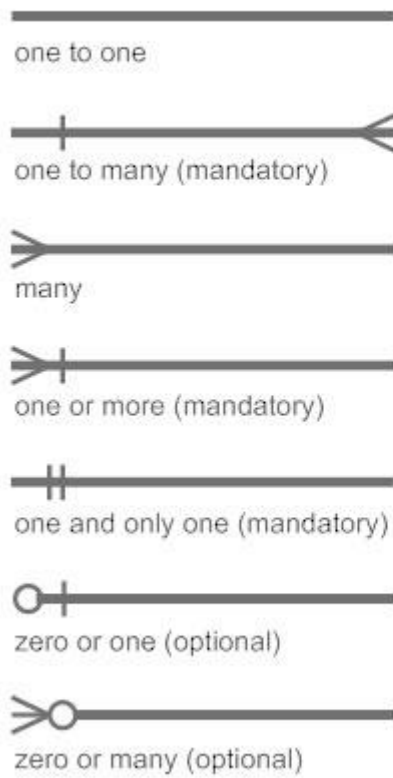


There are many notation styles

that express cardinality.

### Information Engineering Style

## Information Engineering Style



- **Chen Style**



## Chen Style

Ordinality - describes the minimum (optional vs mandatory)

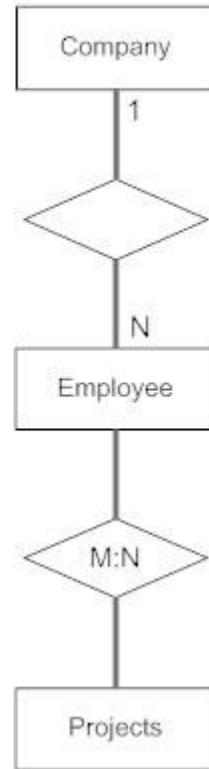
→ M:N ←

Cardinality - describes the maximum

1:N (n=0,1,2,3...)  
one to zero or more

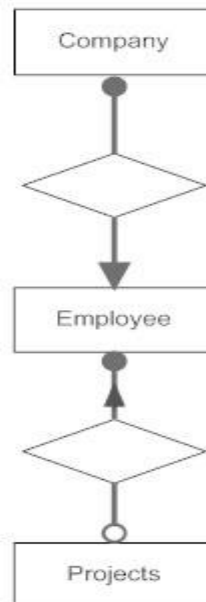
M:N (m and n=0,1,2,3...)  
zero or more to zero or more  
(many to many)

1:1  
one to one



- Bachman Style**

### Bachman Style



- **Martin Style**

### Martin Style

1 - one, and only one (mandatory)

\* - many (zero or more - optional)

1...\* - one or more (mandatory)

0...1 - zero or one (optional)

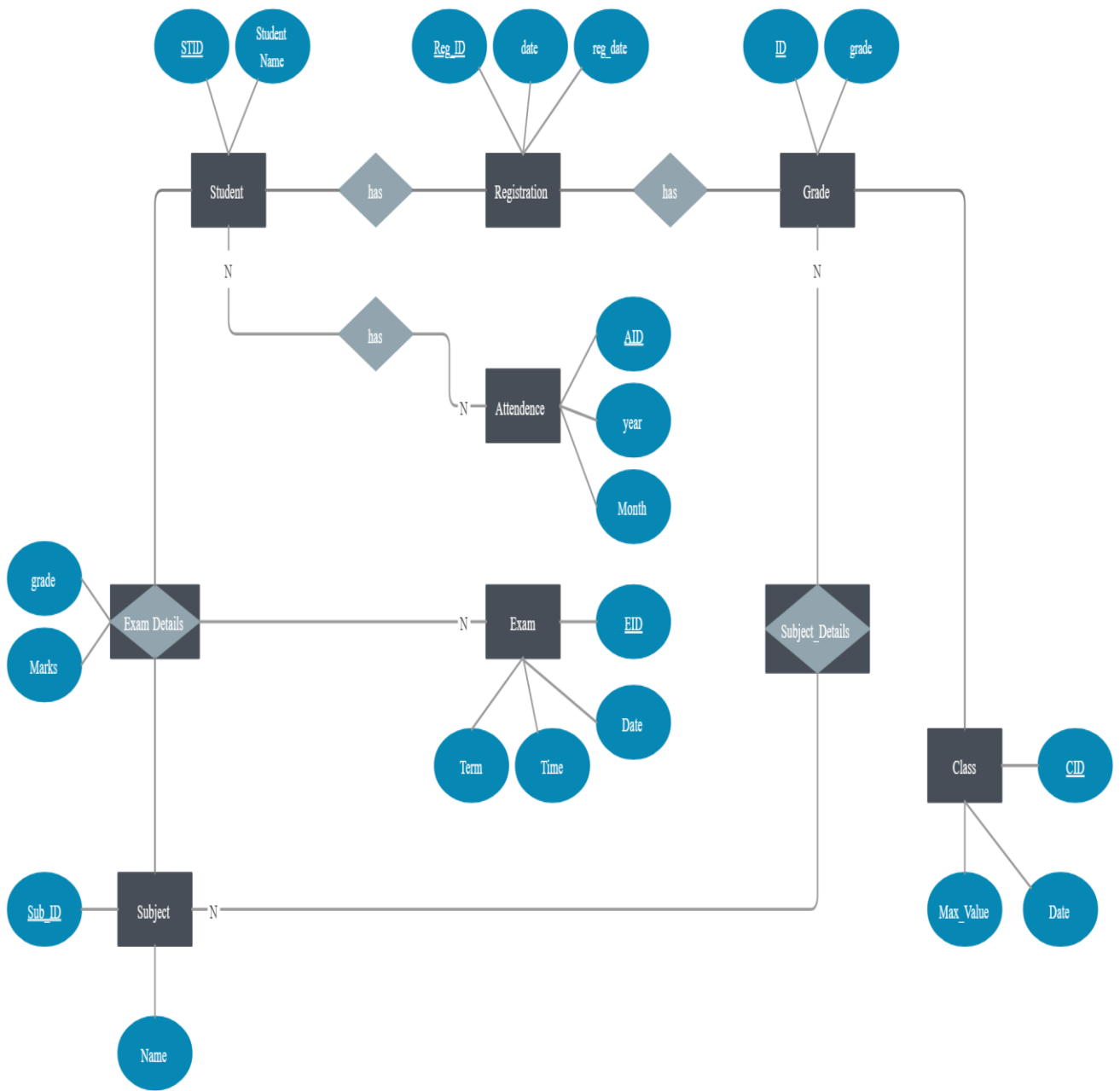
(0,1) - zero or one (optional)

(1,n) - one or more (mandatory)

(0,n) - zero or more (optional)

(1,1) - one and only one (mandatory)





# DATABASE

# Database

## What is Data?

Data is a collection of a distinct small unit of information. It can be used in a variety of forms like text, numbers, media, bytes, etc. it can be stored in pieces of paper or electronic memory, etc.

Word 'Data' is originated from the word 'datum' that means 'single piece of information.' It is plural of the word datum.

In computing, Data is information that can be translated into a form for efficient movement and processing. Data is interchangeable.

What is Database?

A **database** is an organized collection of data, so that it can be easily accessed and managed.

You can organize data into tables, rows, columns, and index it to make it easier to find relevant information.

**Database handlers** create a database in such a way that only one set of software program provides access of data to all the users.

The **main purpose** of the database is to operate a large amount of information by storing, retrieving, and managing data.

There are many **dynamic websites** on the World Wide Web nowadays which are handled through databases. For example, a model that checks the availability of rooms in a hotel. It is an example of a dynamic website that uses a database.

There are many **databases available** like MySQL, Sybase, Oracle, MongoDB, Informix, PostgreSQL, SQL Server, etc.

Modern databases are managed by the database management system (DBMS).

**SQL** or Structured Query Language is used to operate on the data stored in a database. SQL depends on relational algebra and tuple relational calculus.

A cylindrical structure is used to display the image of a database.

## Evolution of Databases

The database has completed more than 50 years of journey of its evolution from flat-file system to

relational and objects relational systems. It has gone through several generations.

## The Evolution

### File-Based

1968 was the year when File-Based database were introduced. In file-based databases, data was maintained in a flat file. Though files have many advantages, there are several limitations.

One of the major advantages is that the file system has various access methods, e.g., sequential, indexed, and random.

It requires extensive programming in a third-generation language such as COBOL, BASIC.

### Hierarchical Data Model

1968-1980 was the era of the Hierarchical Database. Prominent hierarchical database model was IBM's first DBMS. It was called IMS (Information Management System).

In this model, files are related in a parent/child manner.

Below diagram represents Hierarchical Data Model. Small circle represents objects.

Like file system, this model also had some limitations like complex implementation, lack structural independence, can't easily handle a many-many relationship, etc.

### Network data model

**Charles Bachman** developed the first DBMS at Honeywell called Integrated Data Store (IDS). It was developed in the early 1960s, but it was standardized in 1971 by the CODASYL group (Conference on Data Systems Languages).

In this model, files are related as owners and members, like to the common network model.

### **Network data model identified the following components:**

- Network schema (Database organization)
- Sub-schema (views of database per user)
- Data management language (procedural)

This model also had some limitations like system complexity and difficult to design and maintain.

### Relational Database

**1970 - Present:** It is the era of Relational Database and Database Management. In 1970, the relational model was proposed by E.F. Codd.

Relational database model has two main terminologies called instance and schema.

The instance is a table with rows or columns

Schema specifies the structure like name of the relation, type of each column and name.

This model uses some mathematical concept like set theory and predicate logic.

The first internet database application had been created in 1995.

During the era of the relational database, many more models had introduced like object-oriented model, object-relational model, etc.

Cloud database

Cloud database facilitates you to store, manage, and retrieve their structured, unstructured data via a cloud platform. This data is accessible over the Internet. Cloud databases are also called a database as service (DBaaS) because they are offered as a managed service.

### **What is SQL?**

SQL is a short-form of the structured query language, and it is pronounced as S-Q-L or sometimes as See-Quell.

This database language is mainly designed for maintaining the data in relational database management systems. It is a special tool used by data professionals for handling structured data (data which is stored in the form of tables). It is also designed for stream processing in RDSMS.

You can easily create and manipulate the database, access and modify the table rows and columns, etc. This query language became the standard of ANSI in the year of 1986 and ISO in the year of 1987.

If you want to get a job in the field of data science, then it is the most important query language to learn. Big enterprises like Facebook, Instagram, and LinkedIn, use SQL for storing the data in the back-end.

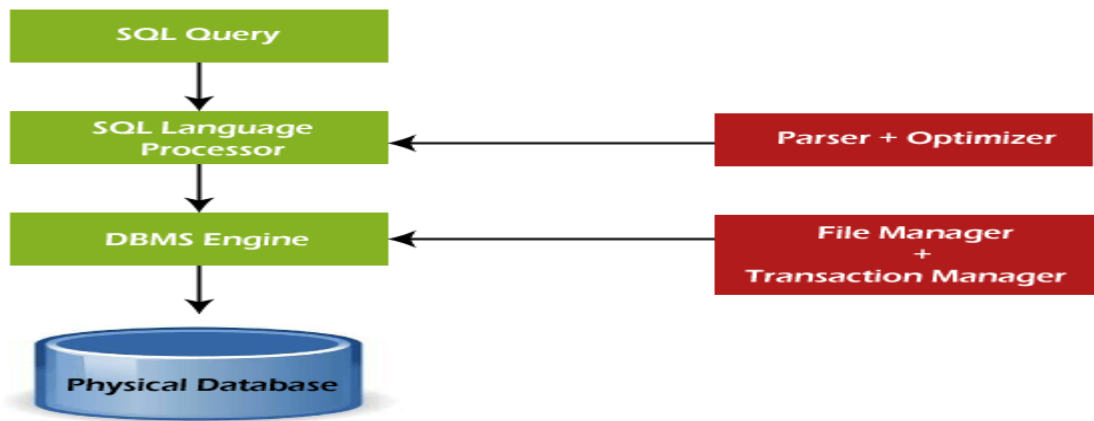
Process of SQL

When we are executing the command of SQL on any Relational database management system, then the system automatically finds the best routine to carry out our request, and the SQL engine determines how to interpret that particular command.

Structured Query Language contains the following four components in its process:

- Query Dispatcher
- Optimization Engines
- Classic Query Engine
- SQL Query Engine, etc.

A classic query engine allows data professionals and users to maintain non-SQL queries. The architecture of SQL is shown in the following diagram:



### Some SQL Commands

The SQL commands help in creating and managing the database. The most common SQL commands which are highly used are mentioned below:

1. CREATE command
2. UPDATE command
3. DELETE command
4. SELECT command
5. DROP command
6. INSERT command

#### CREATE Command

This command helps in creating the new database, new table, table view, and other objects of the database.

#### UPDATE Command

This command helps in updating or changing the stored data in the database.

#### DELETE Command

This command helps in removing or erasing the saved records from the database tables. It erases single or multiple tuples from the tables of the database.

#### SELECT Command

This command helps in accessing the single or multiple rows from one or multiple tables of the database. We can also use this command with the WHERE clause.

#### DROP Command

This command helps in deleting the entire table, table view, and other objects from the database.

#### INSERT Command

This command helps in inserting the data or records into the database tables. We can easily insert the records in single as well as multiple rows of the table.



## Table admin

The screenshot shows the phpMyAdmin interface for the 'tbladmin' table. The table structure is displayed in a table format with columns for #, Name, Type, Collation, Attributes, Null, Default, Comments, Extra, and Action.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 ID	int(10)			No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	2 AdminName	varchar(120)	latin1_swedish_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	3 UserName	varchar(120)	latin1_swedish_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	4 MobileNumber	bigint(10)			Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	5 Email	varchar(200)	latin1_swedish_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	6 Password	varchar(200)	latin1_swedish_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	7 AdminRegdate	timestamp			Yes	current_timestamp()			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

## Table class

The screenshot shows the phpMyAdmin interface for the 'tblclass' table. The table structure is displayed in a table format with columns for #, Name, Type, Collation, Attributes, Null, Default, Comments, Extra, and Action.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 ID	int(5)			No	None		AUTO_INCREMENT	<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	2 ClassName	varchar(50)	latin1_swedish_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	3 Section	varchar(20)	latin1_swedish_ci		Yes	NULL			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
<input type="checkbox"/>	4 CreationDate	timestamp			Yes	current_timestamp()			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

At the bottom of the interface, there are several utility buttons: [Check all](#), [With selected:](#), [Browse](#), [Change](#), [Drop](#), [Primary](#), [Unique](#), [Index](#), [Spatial](#), [Fulltext](#), and [Add to central columns](#).

## Table notice

localhost / 127.0.0.1 / studentms

localhost/phpmyadmin/index.php?route=/table/structure&db=studentmsdb&table=tblnotice

Server: 127.0.0.1 » Database: studentmsdb » Table: tblnotice

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 ID	int(5)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2 NoticeTitle	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	3 ClassId	int(10)		Swedish, case-insensitive					Change Drop More
<input type="checkbox"/>	4 NoticeMsg	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	5 CreationDate	timestamp			Yes	current_timestamp()			Change Drop More

## Table page

localhost / 127.0.0.1 / studentms

localhost/phpmyadmin/index.php?route=/table/structure&db=studentmsdb&table=tblpage

Server: 127.0.0.1 » Database: studentmsdb » Table: tblpage

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 ID	int(10)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2 PageType	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	3 PageTitle	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	4 PageDescription	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	5 Email	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	6 MobileNumber	bigint(10)			Yes	NULL			Change Drop More
<input type="checkbox"/>	7 UpdationDate	date			Yes	NULL			Change Drop More

## Table public notice

localhost / 127.0.0.1 / studentms

localhost/phpmyadmin/index.php?route=/table/structure&db=studentmsdb&table=tblpublicnotice

Server: 127.0.0.1 » Database: studentmsdb » Table: tblpublicnotice

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 ID	int(5)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2 NoticeTitle	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	3 NoticeMessage	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	4 CreationDate	timestamp			Yes	current_timestamp()			Change Drop More

Check all    With selected:    Browse    Change    Drop    Primary    Unique    Index    Spatial    Fulltext    Add to central columns  
 Remove from central columns

## Student table

localhost / 127.0.0.1 / studentms: x

localhost/phpmyadmin/index.php?route=/table/structure&db=studentmsdb&table=tblstudent

Server: 127.0.0.1 » Database: studentmsdb » Table: tblstudent

Table structure | Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 ID	int(10)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2 StudentName	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	3 StudentEmail	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	4 StudentClass	varchar(100)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	5 Gender	varchar(50)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	6 DOB	date			Yes	NULL			Change Drop More
<input checked="" type="checkbox"/>	7 StuID	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	8 FatherName	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	9 MotherName	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	10 ContactNumber	bigint(10)			Yes	NULL			Change Drop More
<input type="checkbox"/>	11 AltenateNumber	bigint(10)			Yes	NULL			Change Drop More
<input type="checkbox"/>	12 Address	mediumtext	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	13 UserName	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	14 Password	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	15 Image	varchar(200)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	16 DateofAdmission	timestamp			Yes	current_timestamp()			Change Drop More

Console

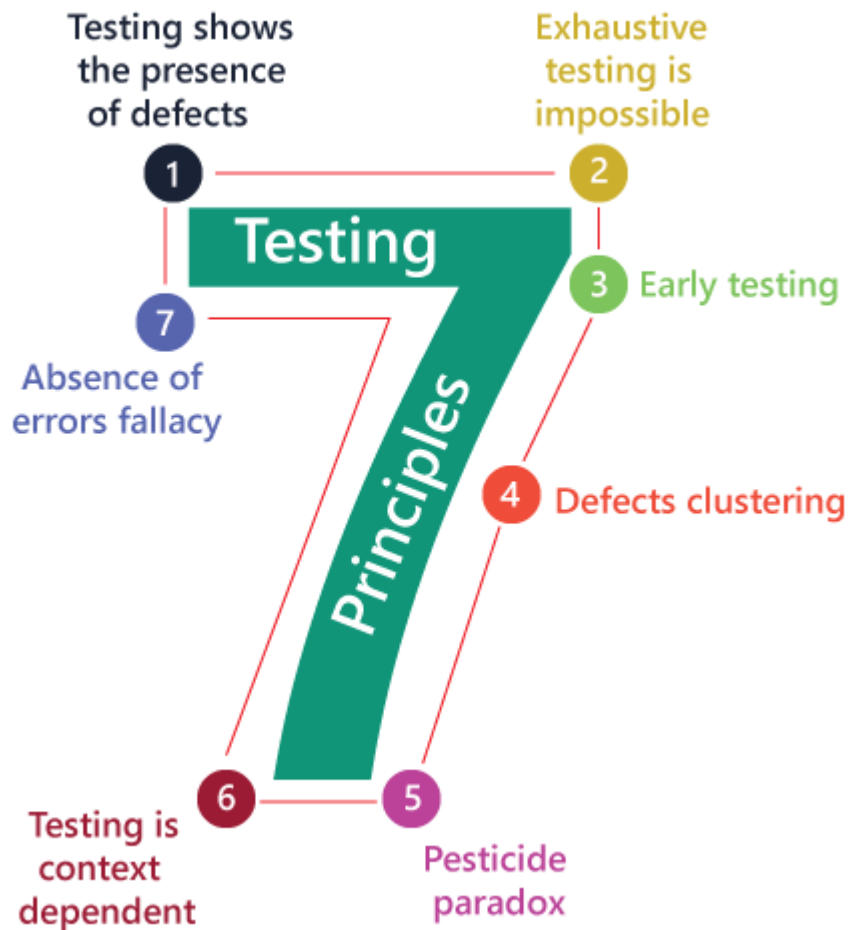
# Designs

# Testing

Software testing is a procedure of implementing software or the application to identify the defects or bugs. For testing an application or software, we need to follow some principles to make our product defects free, and that also helps the test engineers to test the software with their effort and time. Here, in this section, we are going to learn about the seven essential principles of software testing.

Let us see the seven different testing principles, one by one:

- Testing shows the presence of defects
- Exhaustive Testing is not possible
- Early Testing
- Defect Clustering
- Pesticide Paradox
- Testing is context-dependent
- Absence of errors fallacy



### Testing shows the presence of defects

The test engineer will test the application to make sure that the application is bug or defects free. While doing testing, we can only identify that the application or software has any errors. The primary purpose of doing testing is to identify the numbers of unknown bugs with the help of various methods and testing techniques because the entire test should be traceable to the customer requirement, which means that to find any defects that might cause the product failure to meet the client's needs.

By doing testing on any application, we can decrease the number of bugs, which does not mean that the application is defect-free because sometimes the software seems to be bug-free while performing multiple types of testing on it. But at the time of deployment in the production server, if the end-user encounters those bugs which are not found in the testing process.

### Exhaustive Testing is not possible

Sometimes it seems to be very hard to test all the modules and their features with effective and non-effective combinations of the inputs data throughout the actual testing process.

Hence, instead of performing the exhaustive testing as it takes boundless determinations and most of the hard work is unsuccessful. So we can complete this type of variations according to the

importance of the modules because the product timelines will not permit us to perform such type of testing scenarios.

### **Early Testing**

Here early testing means that all the testing activities should start in the early stages of the software development life cycle's **requirement analysis stage** to identify the defects because if we find the bugs at an early stage, it will be fixed in the initial stage itself, which may cost us very less as compared to those which are identified in the future phase of the testing process.

To perform testing, we will require the requirement specification documents; therefore, if the requirements are defined incorrectly, then it can be fixed directly rather than fixing them in another stage, which could be the development phase.

### **Defect clustering**

The defect clustering defined that throughout the testing process, we can detect the numbers of bugs which are correlated to a small number of modules. We have various reasons for this, such as the modules could be complicated; the coding part may be complex, and so on.

These types of software or the application will follow the **Pareto Principle**, which states that we can identify that approx. Eighty percent of the complication is present in 20 percent of the modules. With the help of this, we can find the uncertain modules, but this method has its difficulties if the same tests are performing regularly, hence the same test will not able to identify the new defects.

### **Pesticide paradox**

This principle defined that if we are executing the same set of test cases again and again over a particular time, then these kinds of the test will not be able to find the new bugs in the software or the application. To get over these pesticide paradoxes, it is very significant to review all the test cases frequently. And the new and different tests are necessary to be written for the implementation of multiple parts of the application or the software, which helps us to find more bugs.

### **Testing is context-dependent**

Testing is a context-dependent principle states that we have multiple fields such as e-commerce websites, commercial websites, and so on are available in the market. There is a definite way to test the commercial site as well as the e-commerce websites because every application has its own needs, features, and functionality. To check this type of application, we will take the help of various kinds of testing, different technique, approaches, and multiple methods. Therefore, the testing depends on the context of the application.

### **Absence of errors fallacy**

Once the application is completely tested and there are no bugs identified before the release, so we can say that the application is 99 percent bug-free. But there is the chance when the application is tested beside the incorrect requirements, identified the flaws, and fixed them on a given period would not help as testing is done on the wrong specification, which does not apply to the client's requirements. The absence of error fallacy means identifying and fixing the bugs would not help if the application is impractical and not able to accomplish the client's requirements and needs.

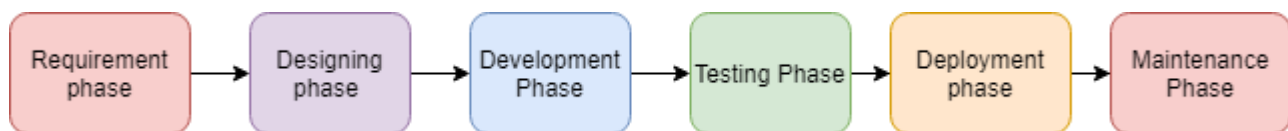
## **Software Development Life Cycle (SDLC)**

SDLC is a process that creates a structure of development of software. There are different phases within SDLC, and each phase has its various activities. It makes the development team able to design, create, and deliver a high-quality product.

SDLC describes various phases of software development and the order of execution of phases. Each phase requires deliverable from the previous phase in a life cycle of software development. Requirements are translated into design, design into development and development into testing; after testing, it is given to the client.

### **Let's see all the phases in detail:**

Different phases of the software development cycle



- Requirement Phase
- Design Phase
- Build /Development Phase
- Testing Phase
- Deployment/ Deliver Phase
- Maintenance

### **1. Requirement Phase**

This is the most crucial phase of the software development life cycle for the developing team as well as for the project manager. During this phase, the client states requirements, specifications, expectations, and any other special requirement related to the product or software. All these are gathered by the business manager or project manager or analyst of the service providing company.

The requirement includes how the product will be used and who will use the product to determine the load of operations. All information gathered from this phase is critical to developing the product as per the customer requirements.



## **2. Design Phase**

The design phase includes a detailed analysis of new software according to the requirement phase. This is the high priority phase in the development life cycle of a system because the logical designing of the system is converted into physical designing. The output of the requirement phase is a collection of things that are required, and the design phase gives the way to accomplish these requirements. The decision of all required essential tools such as programming language like Java, .NET, PHP, a database like Oracle, MySQL, a combination of hardware and software to provide a platform on which software can run without any problem is taken in this phase.

There are several techniques and tools, such as data flow diagrams, flowcharts, decision tables, and decision trees, Data dictionary, and the structured dictionary are used for describing the system design.

## **3. Build /Development Phase**

After the successful completion of the requirement and design phase, the next step is to implement the design into the development of a software system. In this phase, work is divided into small units, and coding starts by the team of developers according to the design discussed in the previous phase and according to the requirements of the client discussed in requirement phase to produce the desired result.

Front-end developers develop easy and attractive GUI and necessary interfaces to interact with back-end operations and back-end developers do back-end coding according to the required operations. All is done according to the procedure and guidelines demonstrated by the project manager. Since this is the coding phase, it takes the longest time and more focused approach for the developer in the software development life cycle.

## **4. Testing Phase**

Testing is the last step of completing a software system. In this phase, after getting the developed GUI and back-end combination, it is tested against the requirements stated in the requirement phase. Testing determines whether the software is actually giving the result as per the requirements addressed in the requirement phase or not. The Development team makes a test plan to start the test. This test plan includes all types of essential testing such as integration testing, unit testing, acceptance testing, and system testing. Non-functional testing is also done in this phase.

If there are any defects in the software or it is not working as per expectations, then the testing team gives information to the development team in detail about the issue. If it is a valid defect or worth to sort out, it will be fixed, and the development team replaces it with the new one, and it also needs to be verified.

## **5. Deployment/ Deliver Phase**

When software testing is completed with a satisfying result, and there are no remaining issues in the working of the software, it is delivered to the customer for their use. As soon as customers

receive the product, they are recommended first to do the beta testing. In beta testing, customer can require any changes which are not present in the software but mentioned in the requirement document or any other GUI changes to make it more user-friendly. Besides this, if any type of defect is encountered while a customer using the software; it will be informed to the development team of that particular software to sort out the problem. If it is a severe issue, then the development team solves it in a short time; otherwise, if it is less severe, then it will wait for the next version. After the solution of all types of bugs and changes, the software finally deployed to the end-user.

## **6. Maintenance**

The maintenance phase is the last and long-lasting phase of SDLC because it is the process which continues until the software's life cycle comes to an end. When a customer starts using software, then actual problems start to occur, and at that time there's a need to solve these problems. This phase also includes making changes in hardware and software to maintain its operational effectiveness like to improve its performance, enhance security features and according to customer's requirements with upcoming time. This process to take care of product time to time is called maintenance.

### **White Box Testing**

The box testing approach of software testing consists of black box testing and white box testing. We are discussing here white box testing which also known as glass box is **testing, structural testing, clear box testing, open box testing and transparent box testing**. It tests internal coding and infrastructure of a software focus on checking of predefined inputs against expected and desired outputs. It is based on inner workings of an application and revolves around internal structure testing. In this type of testing programming skills are required to design test cases. The primary goal of white box testing is to focus on the flow of inputs and outputs through the software and strengthening the security of the software.

The term 'white box' is used because of the internal perspective of the system. The clear box or white box or transparent box name denote the ability to see through the software's outer shell into its inner workings.

Developers do white box testing. In this, the developer will test every line of the code of the program. The developers perform the White-box testing and then send the application or the software to the testing team, where they will perform the black box testing and verify the application along with the requirements and identify the bugs and sends it to the developer.

The developer fixes the bugs and does one round of white box testing and sends it to the testing team. Here, fixing the bugs implies that the bug is deleted, and the particular feature is working fine on the application.

### **Black box testing**

Black box testing is a technique of software testing which examines the functionality of software without peering into its internal structure or coding. The primary source of black box testing is a

specification of requirements that is stated by the customer.

In this method, tester selects a function and gives input value to examine its functionality, and checks whether the function is giving expected output or not. If the function produces correct output, then it is passed in testing, otherwise failed. The test team reports the result to the development team and then tests the next function. After completing testing of all functions if there are severe problems, then it is given back to the development team for correction.



### **Generic steps of black box testing**

- The black box test is based on the specification of requirements, so it is examined in the beginning.
- In the second step, the tester creates a positive test scenario and an adverse test scenario by selecting valid and invalid input values to check that the software is processing them correctly or incorrectly.
- In the third step, the tester develops various test cases such as decision table, all pairs test, equivalent division, error estimation, cause-effect graph, etc.
- The fourth phase includes the execution of all test cases.
- In the fifth step, the tester compares the expected output against the actual output.
- In the sixth and final step, if there is any flaw in the software, then it is cured and tested again.

# INPUT SCREEN

## Font Screen

The screenshot shows a web browser window with the URL `localhost/Student%20Management%20System%20PHP/studentms/index.php`. The page features a pink header with the text "SMS" and navigation links for "Home", "About", "Contact", "Admin", and "Student". The main content area has a background image of a book with colorful letters and a pencil. The text "Student Management System" is prominently displayed, followed by "Registered Students can Login Here" and a blue "Student Login" button with a right-pointing arrow. The Windows taskbar at the bottom shows the system tray with a temperature of 90°F, the word "Smoke", and the date/time 6:24 PM on 5/6/2022.

## Login Screen

The screenshot shows a web browser window with the URL `localhost/Student%20Management%20System%20PHP/studentms/admin/login.php`. The login form is centered on a light blue background. It includes a green square logo, the text "Hello! let's get started", and "Sign in to continue.". There are two input fields: one containing "admin" and another with the placeholder "enter your password". A green "Login" button is positioned below the fields. Below the button are two options: a checked checkbox for "Keep me signed in" and a link for "Forgot password?". At the bottom of the form is a blue "Back Home" button. The Windows taskbar at the bottom shows the system tray with a temperature of 90°F, the word "Smoke", and the date/time 6:25 PM on 5/6/2022.

## Add Class

localhost x localhost / 127.0.0.1 / studer x black box testing javatpoint x Software Testing | Black Box x Student Management System x +

localhost/Student%20Management%20System%20PHP/studentsms/admin/add-class.php

SMS Pratik Welcome dashboard! Pratik

pratikpati3095@gmail.com

DASHBOARD

Dashboard

Class

> Add Class

> Manage Class

Students

Notice

Public Notice

Pages

Reports

Search

90°F Smoke

Pratik Welcome dashboard!

Add Class

Dashboard / Add Class

Add Class

Class Name

SYBCA

Section

D

Add

K7 User Agent

K7AntiVirus Premium

Successfully completed the update!

6:27 PM 5/6/2022

## Manage Class

localhost x localhost / 127.0.0.1 / studer x black box testing javatpoint x Software Testing | Black Box x Student Management System x +

localhost/Student%20Management%20System%20PHP/studentsms/admin/add-students.php

SMS Pratik Welcome dashboard! Pratik

Dashboard

Class

Students

> Add Students

> Manage Students

Notice

Public Notice

Pages

Reports

Search

90°F Smoke

Pratik Welcome dashboard!

Add Students

Student Name

Mayur Nhanbhau Patil

Student Email

May 2022

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

Clear Today

mm/dd/yyyy

Student ID

6:34 PM 5/6/2022

# Add Notice

Pratik Welcome dashboard!

Pratik

pratikpatil3095@gmail.com

DASHBOARD

- Dashboard
- Class
- Students
- Notice
  - Add Notice
  - Manage Notice
- Public Notice
- Pages
- Reports
- Search

Add Notice

Dashboard / Add Notice

Notice Title

holiday

Notice For

FYBSC A

Notice Message

Tomorrow is holiday

Add

90°F Smoke

6:37 PM 5/6/2022

# Add Notice

Pratik Welcome dashboard!

Pratik

pratikpatil3095@gmail.com

DASHBOARD

- Dashboard
- Class
- Students
- Notice
- Public Notice
  - Add Public Notice
  - Manage Public Notice
- Pages
- Reports
- Search

Add Notice

Dashboard / Add Notice

Notice Title

holiday

Notice Message

tomorrow is holiday

Add

90°F Smoke

6:38 PM 5/6/2022

# OUTPUT SCREEN

**Add Class**



localhost / 127.0.0.1 / student / Student Management System

localhost/Student%20Management%20System%20PHP/studentms/admin/manage-class.php

SMS

Pratik Welcome dashboard!

Pratik

Manage Class View all Classes

S.No	Class Name	Section	Creation Date	Action
1	FYBCA	A	2022-04-14 00:08:50	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
2	SYBCA	A	2022-04-14 00:09:03	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
3	TYBCA	A	2022-04-14 00:09:13	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
4	FYBSC	A	2022-04-14 00:09:26	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
5	SYBSC	A	2022-04-14 00:09:37	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
6	TYBSC	A	2022-04-14 00:09:50	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
7	T.Y.B.Sc	B	2022-04-25 17:07:23	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>

[First](#) [Prev](#) [Next](#) [Last](#)

Student Management System

90°F Smoke

6:40 PM 5/6/2022

## Manage Class

localhost / 127.0.0.1 / student / Student Management System

localhost/Student%20Management%20System%20PHP/studentms/admin/manage-students.php

SMS

Pratik Welcome dashboard!

Pratik   
 pratikpatil3095@gmail.com

Manage Students View all Students

Dashboard / Manage Students

S.No	Student ID	Student Class	Student Name	Student Email	Admissin Date	Action
1	mayur1436	FYBCA A	Mayur Nhanbhau Patil	mpatil142652@gmail.com	2022-04-15 22:20:09	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>

[First](#) [Prev](#) [Next](#) [Last](#)

90°F Smoke

6:41 PM 5/6/2022

## Manage Notice

localhost x localhost / 127.0.0.1 / studen x black box testing javatpoint x Software Testing | Black Box x Student Management System x +

localhost/Student%20Management%20System%20PHP/studentms/admin/manage-notice.php

SMS

Pratik Welcome dashboard! Pratik

pratikpatil3095@gmail.com

DASHBOARD

Dashboard

Class

Students

Notice

> Add Notice

> Manage Notice

Public Notice

Pages

Reports

Search

Pratik Welcome dashboard!

Manage Notice [Dashboard / Manage Notice](#)

Manage Notice [View all Notice](#)

S.No	Notice Title	Class	Section	Notice Date	Action
1	Exam	FYBCA	A	2022-04-25 17:08:46	<a href="#">👁</a> <a href="#">  </a> <a href="#">🗑</a>

[First>](#) [Prev>](#) [Next>](#) [Last](#)

90°F Smoke

ENG IN 6:42 PM 5/6/2022

## Manage Public Notice

localhost x localhost / 127.0.0.1 / studen x black box testing javatpoint x Software Testing | Black Box x Student Management System x +

localhost/Student%20Management%20System%20PHP/studentms/admin/manage-public-notice.php

SMS

Pratik Welcome dashboard! Pratik

pratikpatil3095@gmail.com

DASHBOARD

Dashboard

Class

Students

Notice

> Add Public Notice

> Manage Public Notice

Pages

Reports

Search

Pratik Welcome dashboard!

Manage Public Notice [Dashboard / Manage Public Notice](#)

Manage Public Notice [View all Public Notice](#)

S.No	Notice Title	Notice Date	Action
1	School will re-open	2022-01-20 14:41:57	<a href="#">👁</a> <a href="#">  </a> <a href="#">🗑</a>
2	Test Public Notice	2022-02-03 00:34:10	<a href="#">👁</a> <a href="#">  </a> <a href="#">🗑</a>

[First>](#) [Prev>](#) [Next>](#) [Last](#)

90°F Smoke

ENG IN 6:43 PM 5/6/2022

# Reports

Pratik Welcome dashboard! Pratik

Between Dates Reports Dashboard / Between Dates Reports

Report from 2022-04-01 to 2022-05-06

S.No	Student ID	Student Class	Student Name	Student Email	Admissin Date	Action
1	mayur1436	FYBCA A	Mayur Nhanbhau Patil	mpatil142652@gmail.com	2022-04-15 22:20:09	

[First>](#) [Prev>](#) [Next>](#) [Last](#)

Student Management System

90°F Smoke ENG IN 6:44 PM 5/6/2022

# CONCLUSION

## **Conclusion**

Student Management System can be used by educational institutions to maintain their student records easily. Achieving this objective is difficult using the manual system as the information is scattered, can be redundant, and collecting relevant information may be very time-consuming. All these problems are solved by this project.

This system helps in maintaining the information of pupils of the organization. It can be easily accessed by the manager and kept safe for a long period of time without any changes.

Student management systems make faculty jobs more accessible by giving them an easy place to find and sort information. This system allows teachers and student managers to follow with their student engagement.

The idea is to create a scenario that makes the lives of administration and teachers easier. So try a bunch of different systems until you find the right one for you.

The main goal of academic institution is to provide a good quality of teaching and to provide good service to the students. As the results of the study showed that the developed Students' Profiling System met the need of the respondents and its intended users. It only shows that the majority of the respondents and its intended users find that the manual managing and updating all college students' profile needs to be enhanced.

Hence, the researchers concluded that the implementation of Student Information System will provide an effective way of managing and keeping of all the students' confidential record such as identifiable information and offenses in Guidance Office.

# BIBLIOGRAPHY

## **Bibliography**

- **FOR .NET INSTALLATION**

[www.support.microsoft.com](http://www.support.microsoft.com)

- **FOR DEPLOYMENT AND PACKING ON SERVER**

[www.developer.com](http://www.developer.com)

[www.15seconds.com](http://www.15seconds.com)

- **FOR SQL**

[www.msdn.microsoft.com](http://www.msdn.microsoft.com)

- **FOR PHP**

[www.w3school.com](http://www.w3school.com)

[www.w3source .com](http://www.w3source.com)

[www.google.com](http://www.google.com)

[www.javatpoint.com](http://www.javatpoint.com)

[www.tutorial.com](http://www.tutorial.com)