

School Management System

A

Project Report

Submitted by

Marvaniya Smit Prafulbhai

Sagpariya Ujjaval Shaileshbhai

FOR FULFILMENT OF SIXTH SEMESTER IN

Bachelor of Computer Application

Under the Guidance Of

Mrs. Karishma Rupani



Department of Computer Science & I.T

SHREE H.N.SHUKLA GROUP OF COLLEGES

► Index ◀

Index	Title
1	Certificate of College
2	Student information
3	Preface
4	Project profile
5	Acknowledgement
6	Abstract
7	Introduction 1) Purpose: Goals and Objectives 2) Scope: (What it can do and what it cannot do) 3) Brief literature review and Search of project
8	System Analysis and design 1) Study of current system 2) problem and weakness of current system 3) Requirements of new system (User and System requirements)
9	Use Case Diagram
10	Data Modelling 1) Class Diagram 2) Modules 3) Data Dictionary
11	Functional and behavioural modelling 1) DFD 2) ER Diagram
12	Design
13	Sample Coding
14	Summary
15	Future Enhancement
16	Bibliography

► Certificate of College ◀



SHREE H.N.SHUKLA COLLEGE OF I.T. & MGMT.

[Affiliated to Saurashtra university]

SHREE H.N.SHUKLA COLLEGE I.T BUILDING
2-VAISHALI NAGAR,
NR. AMRAPALI UNDER BRIDGE,
RAIYA ROAD,RAJKOT.

SHREE H.N.SHUKLA MANAGEMENTBUILDING
3-VAISHALI NAGAR,
NR. AMRAPALI UNDER BRIDGE,
RAIYA ROAD, RAJKOT.

PROJECT CERTIFICATE

This is to certify that Mr./Ms.
.....the students ofhas
successfully carried out the project under our
supervision and guidance.

DATE :

PLACE :

GUIDED BY

PRINCIPAL

► Student Information ◀

Name	Marvaniya smit prafulbhai
Stream	BCA (sem-6)
Mobile no.	9714416761
E-Mail	smitppatel542@gmail.com



Name	Sagpariya Ujjaval Shaileshbhai
Stream	BCA (sem-6)
Mobile no.	9327943115
E-Mail	ujjavalasagpariya@gmail.com



► Preface ◀

- We are Presenting a Project on The “School management System”. In this Project we have tried to give all the important Content about School management System related things
- In this Project Content is on adding and maintaining Student and Teacher Information
- We are thankful to my guided teaches who give us more support and guided us to Complete Perfect Project on time During this Project make Process, Come to know many things.

► Project profile ◀

Project Title	School Management System
Frontend	ASP.NET
Backend	Microsoft Sql
Language	ASP.NET
Project Guide	Mrs. Karishma Rupani
Developed by	Marvaniya Smit Prafulbhai & Sagpariya Ujjaval Shaileshbhai
Submitted To	IT Department of Saurashtra University
University	Saurashtra University

► Acknowledgement ◀

- On the successful completion of our project we would especially like to thank Mrs. Karishma Rupani for giving us guidance throughout the project and for her constant support and help in the design and implementation of this project.
- We are also thankful to Mrs. Karishma Rupani for encouragement and exclusive guidance, without which we should, been lacking something.
- We are grateful to Mrs. Karishma Rupani (H.O.D.), Shree H.N.Shukla College of IT & Management, for all the facilities they provide me throughout are academic semester and for encouraging us to take the activity. we would also like to thank the rest of faculties and staff members of College for their encouragement and enhancing cooperation.

Thank You

Marvaniya smit &

Sagpariya Ujjaval

► Abstract ◀

- The School Management System (SMS) is a sophisticated software solution designed to automate administrative processes in educational institutions. It focuses on efficiently managing student, teacher, and administrative data, offering features such as real-time record updates and secure deletions. In Student Management, administrators can seamlessly add, update, and remove student records. Teacher Management allows for comprehensive teacher record management, while Administrative Management handles staff details.
- The SMS goes beyond basic functions, incorporating advanced features like user authentication and data security. the SMS aims to user-friendly solution tailored to diverse educational needs.

► Introduction ◀

- A School Management System (SMS) is a sophisticated software solution designed to automate administrative tasks in educational institutions. Its primary functions include managing student, teacher, and administrative data. Key features encompass the ability to insert, update, and delete records for each category.

- **Student Management:**

- **Insert (Add):** Administrators can add new student records.
- **Update:** Existing student data can be modified, allowing for real-time records.
- **Delete:** Administrators have the authority to remove student records.

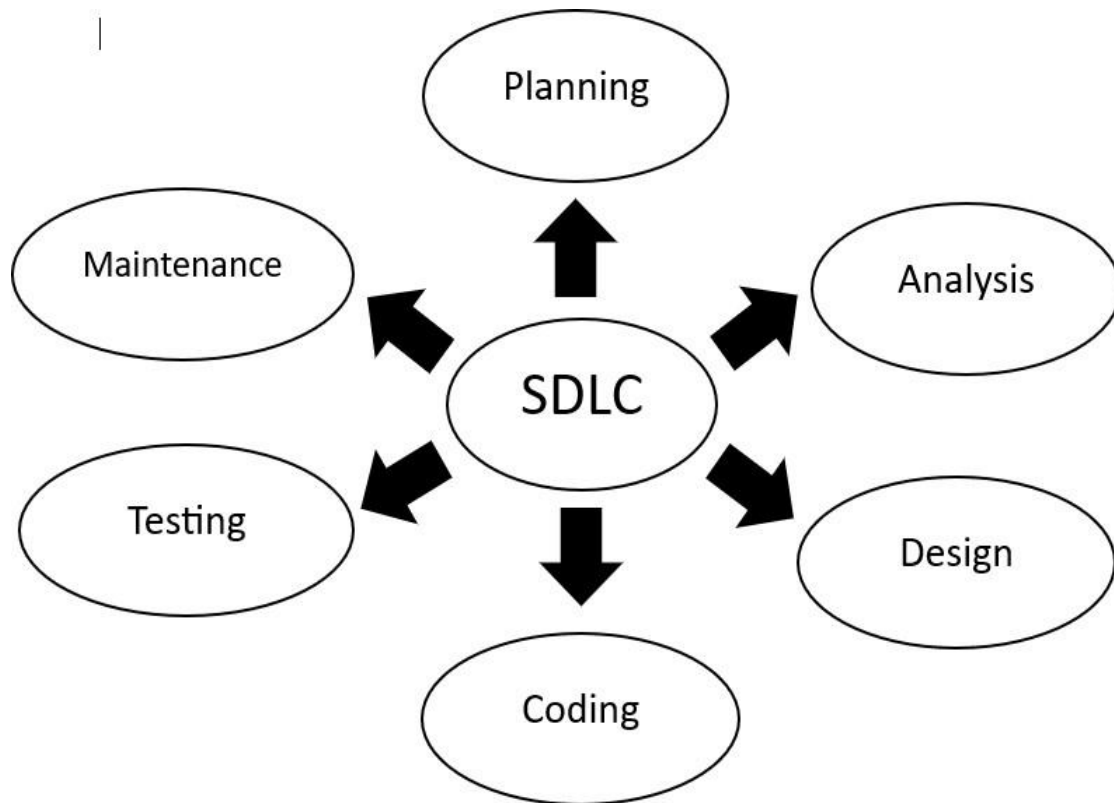
- **Teacher Management:**

- **Insert (Add):** New teacher records can be seamlessly added, including information such as names, contact details, educational qualifications, and subject or class assignments.
- **Update:** Teacher information can be updated dynamically, accommodating changes in contact details, professional qualifications, or subject assignments.
- **Delete:** The system provides an efficient process for removing teacher records when educators leave the institution or retire.

- **Admin Management:**

- **Insert (Add):** Administrative staff details, such as name and password, can be added to create a comprehensive database.
- **Update:** Admin records can be easily updated to reflect changes in name and password.
- **Delete:** Administrative staff records can be deleted when individuals are no longer associated with the institution.

► System Analysis and design ◀



Planning:-

- At the Stage of Planning I am deciding...

- ❖ Our goals
- ❖ Our Purposes

Analysis:-

- At the stage of Analysis I am doing...

- ❖ Collecting information which is related to my project.
- ❖ Analysis of this information.
- ❖ Making Documentation based on analysis.

Design:-

- At the stage of Design I am doing...
 - ❖ Creating Design of Project based on Documentation.
 - ❖ Designing of different web pages & Different forms.

Coding:-

- At the stage of Coding...
 - ❖ Write Different logic & coding of Project & Database.

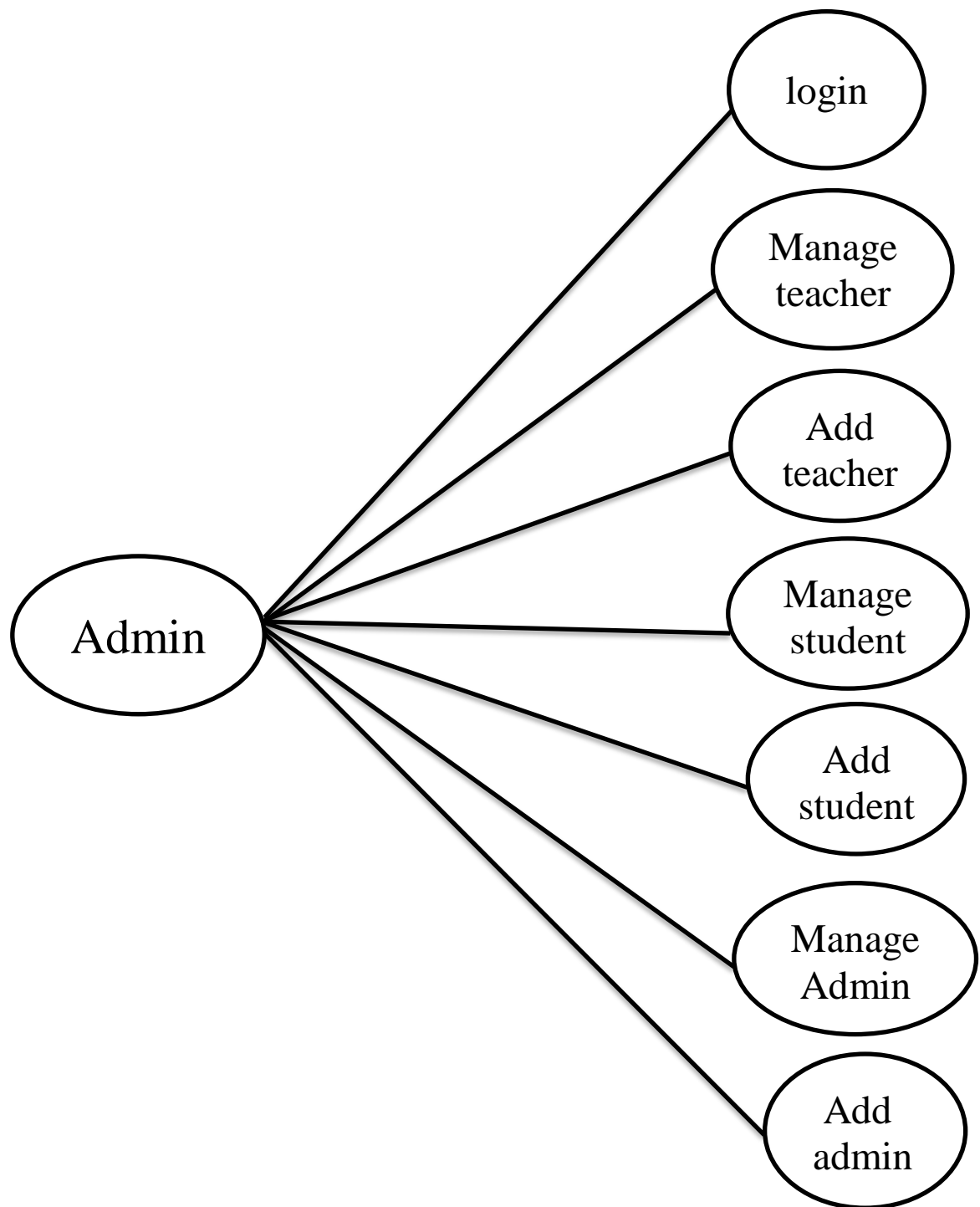
Testing:-

- At the stage of Testing...
 - ❖ Done Different types of Testing For Project.
 - Black Box testing
 - White Box testing
 - Gray Box testing

Maintenance:-

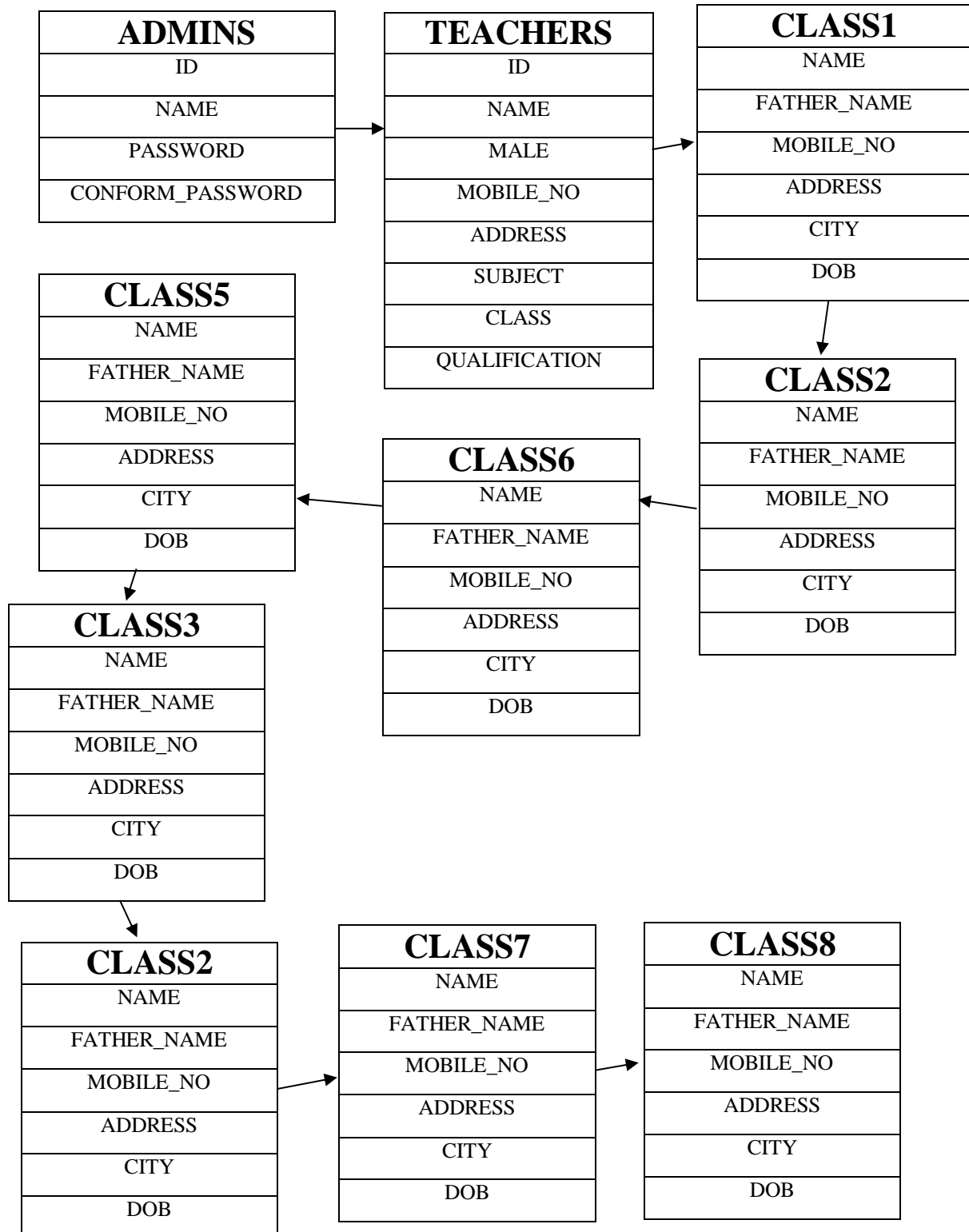
- At the stage of Maintenance...
 - ❖ Keep system updates & secure.

► Use Case Diagram ◀

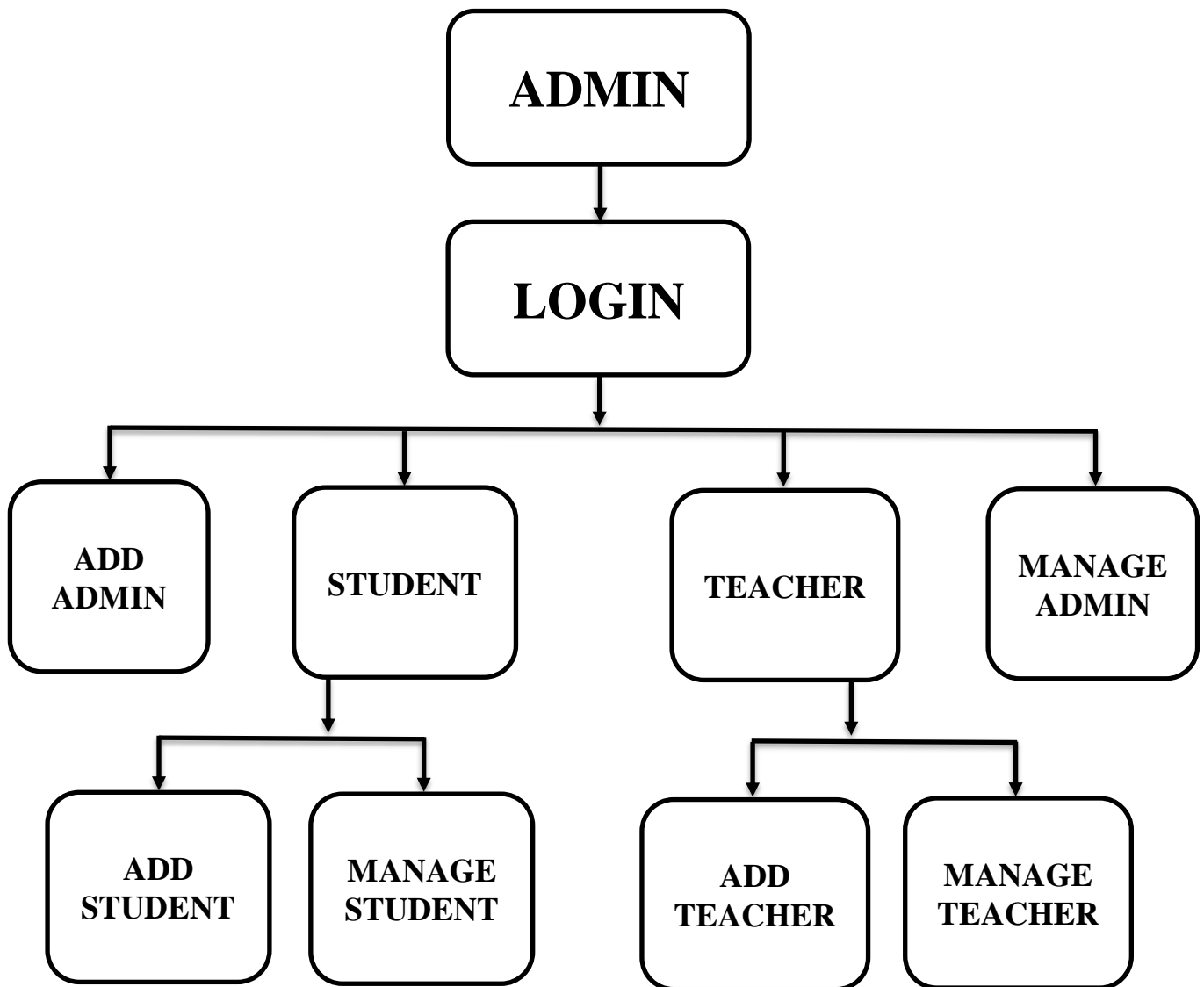


► Data Modelling ◀

1) Class Diagram



2) Modules



3) Data Dictionary

:ADMINS:

Description: This Table is use to Store Admin information.

Column name	Data type
ID	Int
NAME	Varchar(50)
PASSWORD	Varchar(50)
CONFORM_PASSWORD	Varchar(50)

:TEACHERS:

Description: This Table is use to Teacher information.

Column name	Data type
ID	Int
NAME	Varchar(50)
MALE	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
SUBJECT	Varchar(50)
CLASS	Varchar(50)
QUALIFICATION	Varchar(50)

:CLASS1:

Description: This Table is use to Class1 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

:CLASS2:

Description: This Table is use to Class2 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

:CLASS3:

Description: This Table is use to Class3 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

:CLASS4:

Description: This Table is use to Class4 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

:CLASS5:

Description: This Table is use to Class5 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

:CLASS6:

Description: This Table is use to Class6 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

:CLASS7:

Description: This Table is use to Class7 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

:CLASS8:

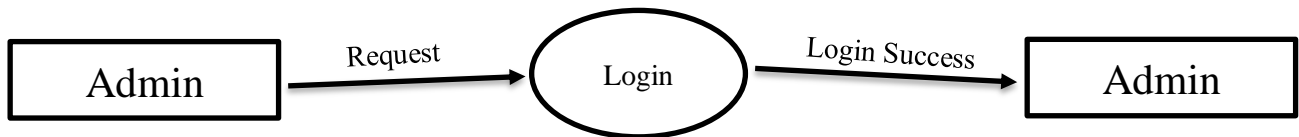
Description: This Table is use to Class8 student information.

Column name	Data type
NAME	Varchar(50)
FATHER_NAME	Varchar(50)
MOBILE_NO	Varchar(50)
ADDRESS	Varchar(MAX)
CITY	Varchar(50)
DOB	datetime

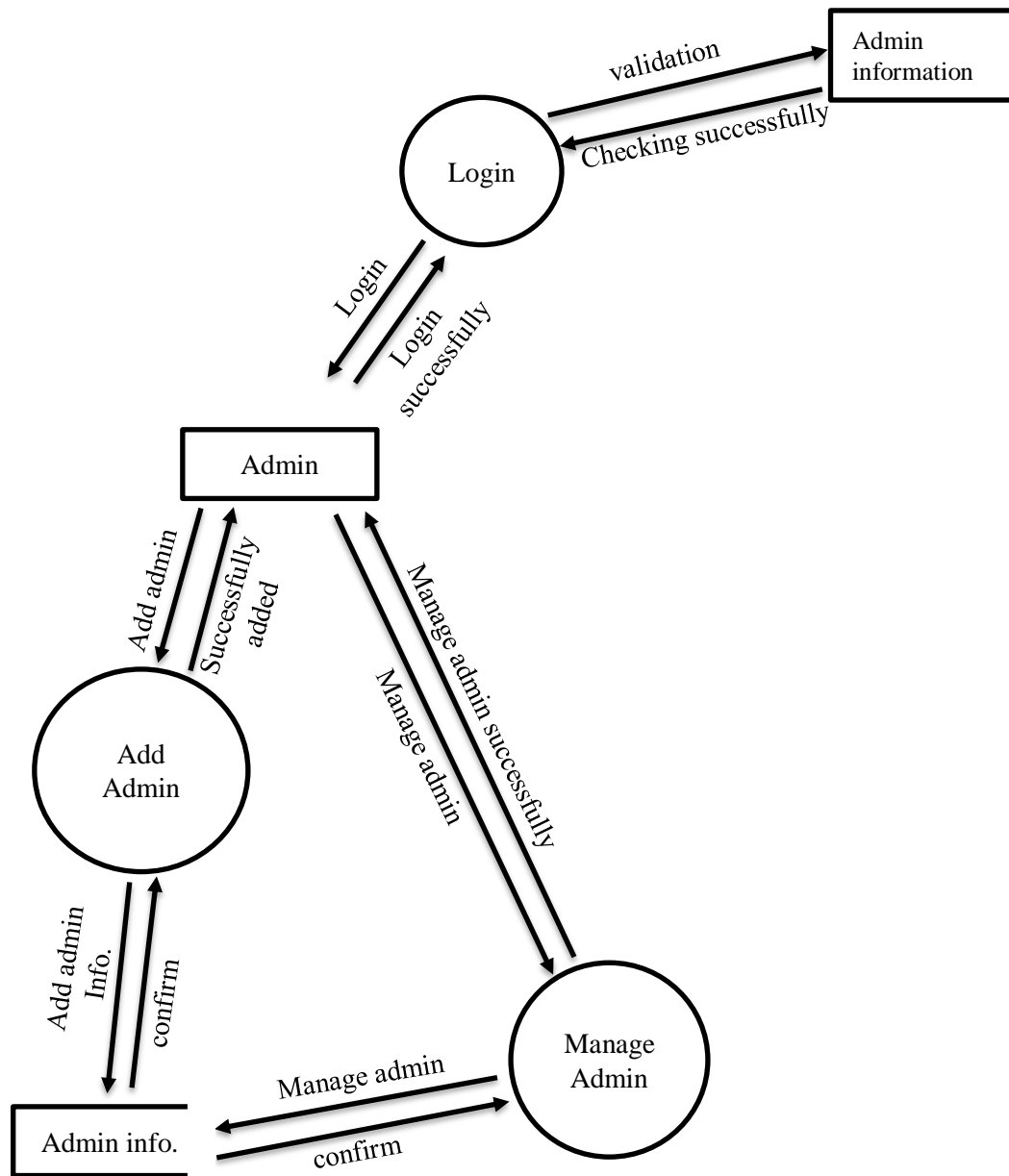
► Functional and behavioural modelling ◀

(DFD)

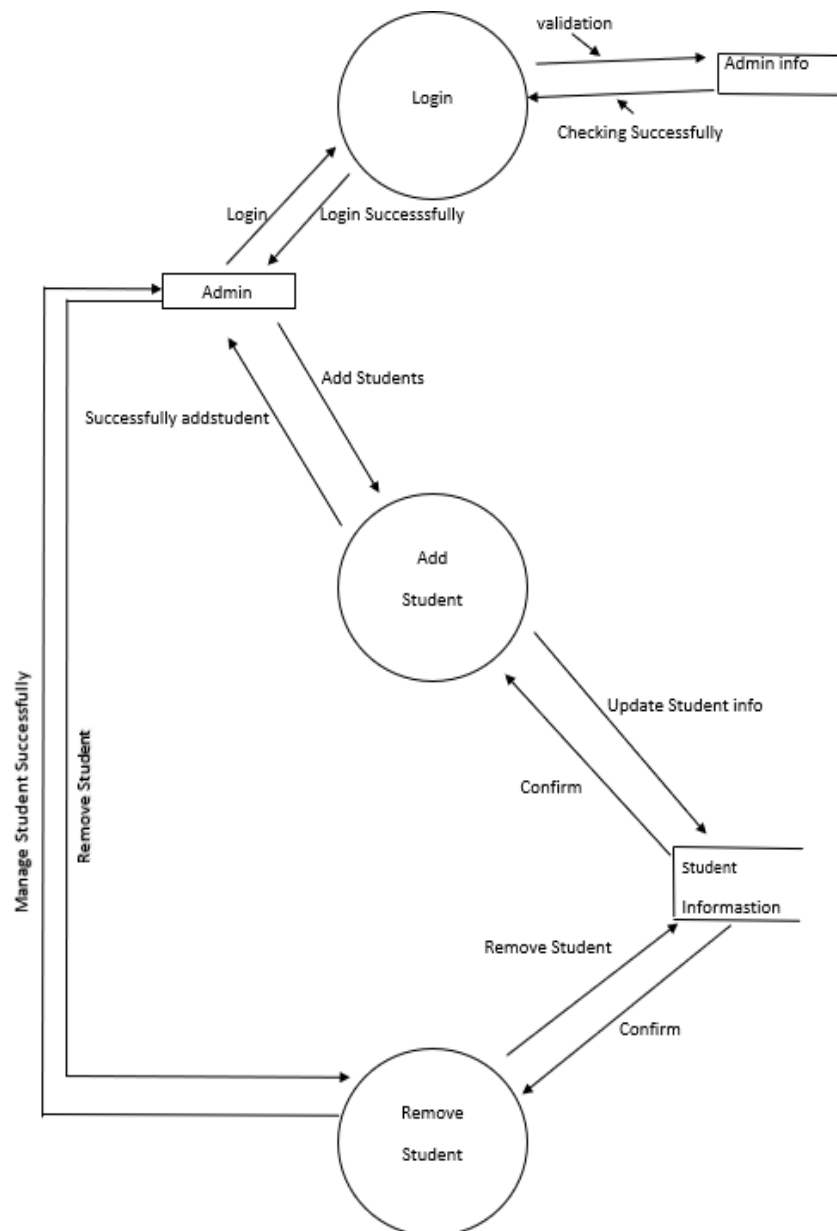
[0 Level DFD]



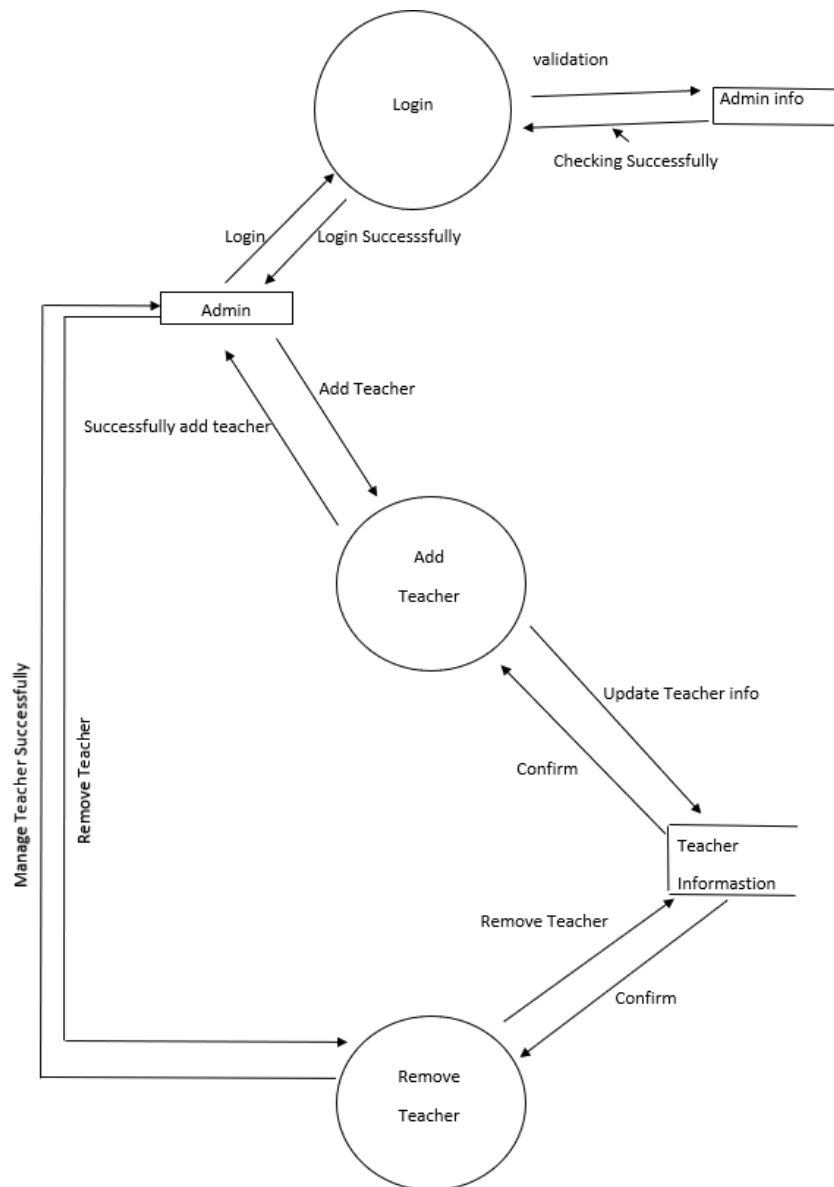
[1 Level (Admin)]



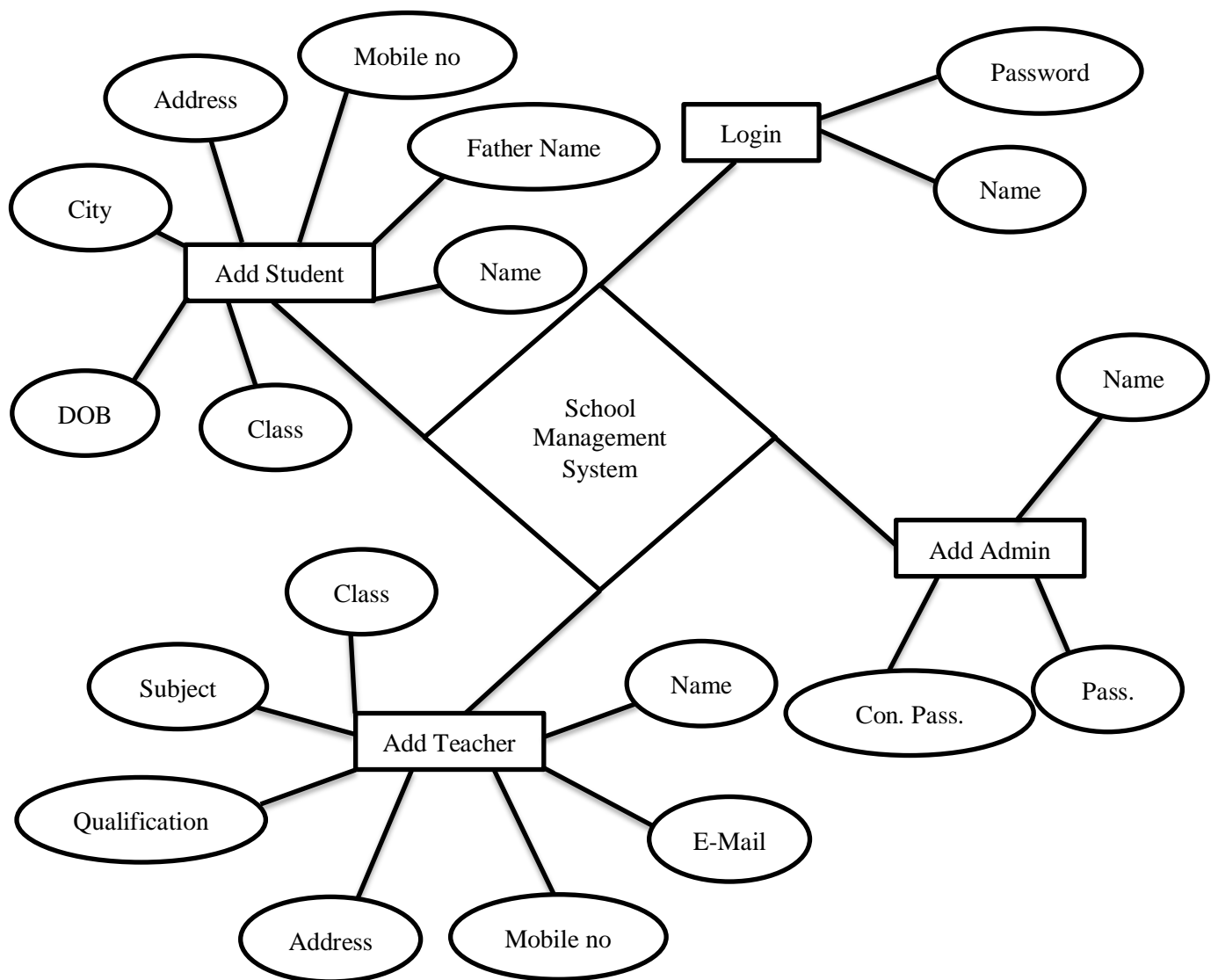
[1 Level (Student)]



[1 Level (Teacher)]



► ER Diagram ◀



► Design ◀

Page 1) Login page

Description: This Page is use to Login in to the Website.

The login page features a header with a yellow background and green abstract shapes. The text 'apna school' is displayed in a large, dark green font. Below the header, there is a light yellow rounded rectangle containing the login form. The form has two input fields: one for 'NAME' and one for 'PASSWORD'. A 'LOGIN' button is positioned below the password field.

apna school


NAME

PASSWORD

LOGIN

Page 2) Add admin

Description: This Page is use to Add new admin.



- › add admin
- › manage admin
- › teachers
- › student
- › logout

NAME

PASSWORD


CONFORM PASSWORD

CLEAR

INSERT

Page 3) ManageAdmin

Description: This Page is use to ManageAdmin Data.




- › add admin
- › manage admin
- › teachers
- › student
- › logout

		Id	NAME	PASSWORD	CONFORM_PASSWORD
Delete	Edit	2	smit	64	64
Delete	Edit	12	uu	U	U
Delete	Edit	13	vardan	vp	vp

Page 4) Manage teacher

Description: This Page is use to Add new teachers.



- › manage teacher
- › add teacher
- › back

	Id	NAME	MAIL	MOBILE_NO	ADDRESS	SUBJECT	CLASS	QUALIFICATION
Edit Delete	13	BCHB	E@GMAIL.COM	12	WE	WE	CLASS 1	WE

Page 5) Add teacher

Description: This Page is use to Add new teachers.

apna school

> manage teacher

> add teacher

> back

NAME

E-MAIL

MOBILE NO.

ADDRESS

QUALIFICATION

SUBJECT

CLASS YOU TEACH


CLASS 1

CLEAR

INSERT

Page 6) Add student

Description: This Page is use to Add new students.



- » manage student
- » add student
- » back

NAME

FATHER NAME

MOBILE NO.

ADDRESS

CITY

D.O.B.

dd-mm-yyyy

CLASS


CLASS1

CLEAR

INSERT

Page 6) Manage Students

Description: This Page is use to Manage Student data.

	<ul style="list-style-type: none">› manage student› add student› back
<div>>CLASS 1<</div> <div>>CLASS 2<</div> <div>>CLASS 3<</div> <div>>CLASS 4<</div> <div>>CLASS 5<</div> <div>>CLASS 6<</div> <div>>CLASS 7<</div> <div>>CLASS 8<</div>	

► Sample Coding ◀

(ADDSTUDENTS.aspx.cs)

```
using System;
using System.Data.SqlClient;
using System.Data;
using System.Configuration;
public partial class ADDSTUDENTS : System.Web.UI.Page
{
    public string Data { get; private set; }
    SqlConnection con;
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button4_Click(object sender, EventArgs e)
    {
        try
        {
            string ClassName = CLASS.SelectedItem.Text;
            con = new
            SqlConnection(ConfigurationManager.ConnectionStrings["myConnectionString"].ConnectionString);
            con.Open();
            SqlCommand cmd1 = new SqlCommand("insert into " + ClassName + " values('" + NAME.Text + "','" +
            FATHERNAME.Text + "','" + MOBILENO.Text + "','" + ADDRESS.Text + "','" + CITY.Text + "','" + DOB.Text +
            "')",con);
            cmd1.ExecuteNonQuery();
            con.Close();

            NAME.Text = "";
            FATHERNAME.Text = "";
            MOBILENO.Text = "";
            ADDRESS.Text = "";
            CITY.Text = "";
            DOB.Text = "";
            ClientScript.RegisterStartupScript(this.GetType(), "alert", "alert('INSERTED " + NAME.Text + "');", true);
        }
        catch (Exception ex)
        {
            Response.Write("<script>alurt('" + ex.Message + "');</script>");
        }
    }
    protected void Button3_Click(object sender, EventArgs e)
    {
        NAME.Text = "";
        FATHERNAME.Text = "";
        MOBILENO.Text = "";
        ADDRESS.Text = "";
        CITY.Text = "";
        DOB.Text = "";
    }
}
```

(LOGIN.aspx.cs)

```
using System;
using System.Data.SqlClient;
using System.Data;

public partial class LOGIN : System.Web.UI.Page
{
    public string Data { get; private set; }
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        try
        {
            string x = "Data Source=(localdb)\\MSSQLLocalDB;Initial Catalog=SCHOOL;Integrated
Security=True;Pooling=False";
            SqlConnection con = new SqlConnection(x);
            if (con.State == ConnectionState.Closed)
            {
                con.Open();
            }
            SqlCommand cmd = new SqlCommand("SELECT * from ADMINs where Name='" + NAME.Text.Trim() + "
AND Password='" + PASSWORD.Text.Trim() + "';", con);
            SqlDataReader dr = cmd.ExecuteReader();
            if (dr.HasRows)
            {
                while (dr.Read())
                {
                    Response.Write("<script>alurt('" + dr.GetValue(1).ToString() + "');</script>");
                    Response.Redirect("ADDADMIN.aspx");
                }
            }
            else
            {
                ClientScript.RegisterStartupScript(this.GetType(), "alert", "alert('Pless try again');", true);
            }
        }
        catch (Exception ex)
        {
            Response.Write("<script>alurt('" + ex.Message + "');</script>");
        }
    }
}
```

► Summary ◀

- The School Management System (SMS) project is a sophisticated software solution designed to automate and streamline administrative tasks within educational institutions. The system focuses on efficient management of student, teacher, and administrative data, allowing administrators to insert, update, and delete records in each category.
- The Key features include seamless addition of new student and teacher records, dynamic updates to reflect real-time changes, and an efficient process for record removal.
- The system also encompasses Admin Management, enabling the addition, updating, and deletion of administrative staff details. With a commitment to user-friendly features and real-time updates, the SMS project aims to enhance administrative efficiency, providing a comprehensive solution tailored to the diverse needs of educational institutions.

► **Future Enhancement** ◀

- Strengthen data security measures
- Prioritise accessibility and mobile-friendliness
- Online payment

► Bibliography ◀

1. www.youtube.com
2. www.openai.com