

VELAMMAL VIDHYASHRAM MAMBAKKAM



ACADEMIC YEAR : 2021-22

PROJECT REPORT ON
DANCE SCHOOL MANAGEMENT SYSTEM

ROLL NO : 20627269
NAME : P.Kavya
CLASS : XII
SUBJECT : COMPUTER SCIENCE
SUB CODE : 083

PROJECT GUIDE:

PGT (CS)

VELAMMAL VIDHYASHRAM MAMBAKKAM

CHENNAI

TAMILNADU



BONAFIDE CERTIFICATE

This is to certify that this **COMPUTER SCIENCE** Project on the topic

Dance School Management System has been successfully completed by :
P.Kavya of grade **XII** , Roll.no **20627269** at Velammal Vidhyashram,
Mambakkam, for the partial fulfillment of this project as a part of **All**
India Senior School Certificate Examination-CBSE, New Delhi for the
academic Year **2021 - 2022**.

Date:

Signature of Principal

Signature of the Guide

Signature of the
Internal Examiner

Signature of the
External Examiner

VELAMMAL VIDHYASHRAM MAMBAKKAM



CERTIFICATE

This is to certify that Cadet _____ P.Kavya_____ CBSE Roll No: _____20627269_____ has successfully completed the project Work entitled "**DANCE SCHOOL MANAGEMENT SYSTEM.**"in the subject Computer Science (083) laid down in the regulations of CBSE for the purpose of Practical [Examination in Class XII.

I certify this Project up to my expectation & as per guidelines issued by CBSE, NEW DELHI.

Internal Examiner:

Name: _____

Signature:

Date:

External Examiner:

Name: _____

Signature:

Date:

TABLE OF CONTENTS		
SLNO	DESCRIPTION	PAGE NO
01	ACKNOWLEDGEMENT	4
02	INTRODUCTION	5
03	OBJECTIVES OF THE PROJECT	5
04	PROPOSED SYSTEM	6
05	SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)	7
06	PHASES OF SYSTEM DEVELOPMENT LIFE CYCLE	8
07	DATABASE DESIGN	15
08	SOURCE CODE	16
09	OUTPUT	35
10	TESTING	40
11	HARDWARE AND SOFTWARE REQUIREMENTS	42
12	INSTALLATION PROCEDURE	43
13	BIBILOGRAPHY	43

ACKNOWLEDGMENT

It is with pleasure that I acknowledge my sincere gratitude to our teacher,.....who taught and undertook the responsibility of teaching the subject computer science. I have been greatly benefited from his classes.

I am especially indebted to our Principal who has always been a source of encouragement and support and without whose inspiration this project would not have been a successful I would like to place on record heartfelt thanks to him.

Finally, I would like to express my sincere appreciation for all the other students for my batch their friendship & the fine times that we all shared together.

INTRODUCTION

"DANCE SCHOOL MANAGEMENT SYSTEM" This project is useful for the people who run dance schools. The emerging of digital system made information available on finger tips. By automating the admissions one can view the details as and when required in no time. This project emphasises on creation of new student details, managing the existing student details in the school. By making digital system one can efficiently manage student details.

OBJECTIVES OF THE PROJECT

The objective of this project is to let the students apply the programming knowledge into a real- world situation/problem and exposed the students how programming skills helps in developing a good software.

1. Write programs utilizing modern software tools.
2. Apply object oriented programming principles effectively when developing small to medium sized projects.
3. Write effective procedural code to solve small to medium sized problems.
4. Students will demonstrate a breadth of knowledge in computer science, as exemplified in the areas of systems, theory and software development.
5. Students will demonstrate ability to conduct a research or applied Computer Science project, requiring writing and presentation skills which exemplify scholarly style in computer science.

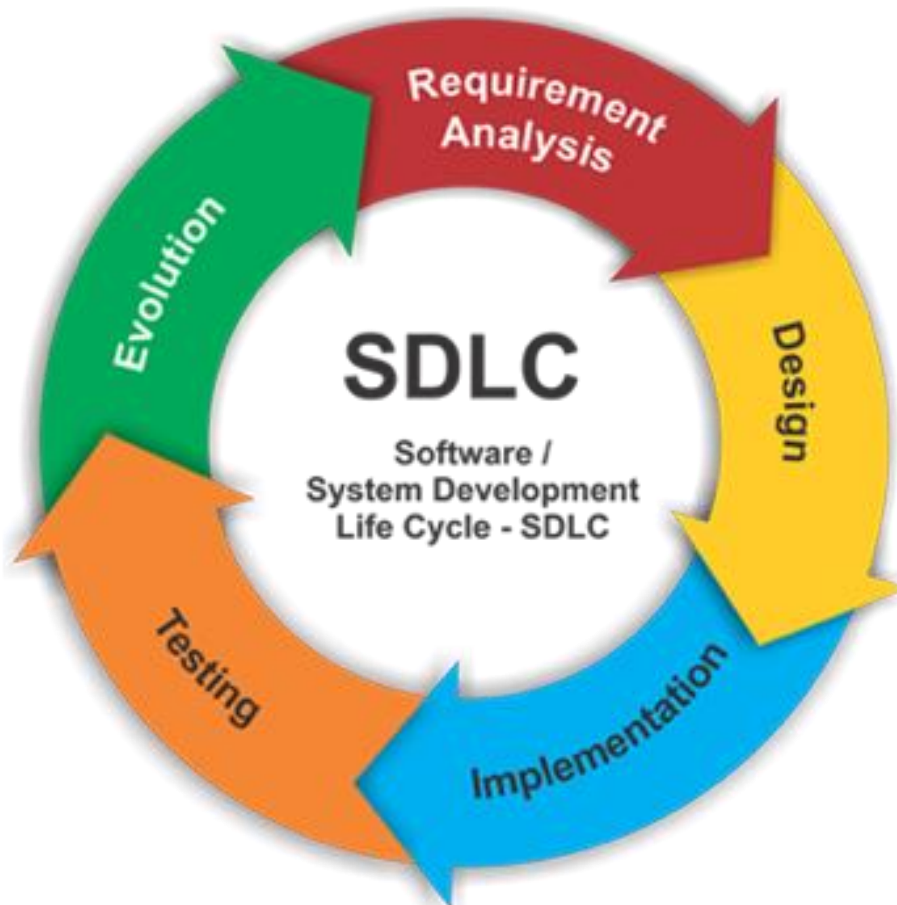
PROPOSED SYSTEM

Today one cannot afford to rely on the fallible human beings of be really wants to stand against today's merciless competition where not to wise saying **"to err is human"** no longer valid, it's outdated to rationalize your mistake. So, to keep pace with time, to bring about the best result without malfunctioning and greater efficiency so to replace the unending heaps of files with a much sophisticated hard disk of the computer.

One has to use the data management software. Software has been an ascent in atomization various organisations. Many software products working are now in markets, which have helped in making the organizations work easier and efficiently. Data management initially had to maintain a lot of ledgers and a lot of paperwork has to be done but now software production this organization has made their work faster and easier. Now only this software has to be loaded on the computer and work can be done.

This prevents a lot of time and money. The work becomes fully automated and any information regarding the organization can be obtained by clicking the button. Moreover, now it's an age of computers of and automating such an organization gives the better look.

SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)



The systems development life cycle is a project management technique that divides complex projects into smaller, more easily managed segments or phases. Segmenting projects allows managers to verify the successful completion of project phases before allocating resources to subsequent phases.

Software development projects typically include initiation, planning, design, development, testing, implementation, and maintenance phases. However, the phases may be divided differently depending on the organization involved.






For example, initial project activities might be designated as request, requirements-definition, and planning phases, or initiation, concept-development, and planning phases. End users of the system under development should be involved in reviewing the output of each phase to ensure the system is being built to deliver the needed functionality

PHASES OF SYSTEM DEVELOPMENT LIFE CYCLE

INITIATION PHASE

The Initiation Phase begins when a business sponsor identifies a need or an opportunity.

The purpose of the Initiation Phase is to:











-  Identify and validate an opportunity to improve business accomplishments of the organization or a deficiency related to a business need.
-  Identify significant assumptions and constraints on solutions to that need.
-  Recommend the exploration of alternative concepts and methods to satisfy the need including questioning the need for technology, i.e., will a change in the business process offer a solution?
-  Assure executive business and executive technical sponsorship. The Sponsor designates a Project Manager and the business need is documented in a Concept Proposal. The Concept Proposal includes information about the business process and the relationship to the Agency/Organization.
 -  Infrastructure and the Strategic Plan. A successful Concept Proposal results in a Project Management Charter which outlines the authority of the project manager to begin the project.

Careful oversight is required to ensure projects support strategic business objectives and resources are effectively implemented into an organization's enterprise architecture. The initiation phase begins when an opportunity to add, improve, or correct a system is identified and formally requested through the presentation of a business case. The business case should, at a minimum, describe a proposal's purpose, identify expected benefits, and explain how the proposed system supports one of the organization's business strategies. The business case should also identify alternative solutions and detail as many informational, functional, and network requirements as possible.

SYSTEM CONCEPT DEVELOPMENT PHASE

The System Concept Development Phase begins after a business need or opportunity is validated by the Agency/Organization Program Leadership and the Agency/Organization CIO.

The purpose of the System Concept Development Phase is to:

-  Determine the feasibility and appropriateness of the alternatives.
-  Identify system interfaces.
-  Identify basic functional and data requirements to satisfy the business need.
-  Establish system boundaries; identify goals, objectives, critical success factors, and performance measures.
-  Evaluate costs and benefits of alternative approaches to satisfy the basic functional requirements
-  Assess project risks
-  Identify and initiate risk mitigation actions, and Develop high-level technical architecture, process models, data models, and a concept of operations. This phase explores potential technical solutions within the context of the business need.
-  It may include several trade-off decisions such as the decision to use COTS software products as opposed to developing custom software or reusing software components, or the decision to use an incremental delivery versus a complete, onetime deployment.
-  Construction of executable prototypes is encouraged to evaluate technology to support the business process. The System Boundary Document serves as an important reference document to support the Information Technology Project Request (ITPR) process.
-  The ITPR must be approved by the State CIO before the project can move forward.

PLANNING PHASE

The planning phase is the most critical step in completing development, acquisition, and maintenance projects. Careful planning, particularly in the early stages of a project, is necessary to coordinate activities and manage project risks effectively. The depth and formality of project plans should be commensurate with the characteristics and risks of a given project. Project plans refine the information gathered during the initiation phase by further identifying the specific activities and resources required to complete a project.





A critical part of a project manager's job is to coordinate discussions between user, audit, security, design, development, and network personnel to identify and document as many functional, security, and network requirements as possible. During this phase, a plan is developed that documents the approach to be used and includes a discussion of methods, tools, tasks, resources, project schedules, and user input. Personnel assignments, costs, project schedule, and target dates are established.

A Project Management Plan is created with components related to acquisition planning, configuration management planning, quality assurance planning, concept of operations, system security, verification and validation, and systems engineering management planning.

REQUIREMENTS ANALYSIS PHASE

This phase formally defines the detailed functional user requirements using high-level requirements identified in the Initiation, System Concept, and Planning phases. It also delineates the requirements in terms of data, system performance, security, and maintainability requirements for the system. The requirements are defined in this phase to a level of detail sufficient for systems design to proceed. They need to be measurable, testable, and relate to the business need or opportunity identified in the Initiation Phase. The requirements that will be used to determine acceptance of the system are captured in the Test and Evaluation Master Plan.

The purposes of this phase are to:

-  Further define and refine the functional and data requirements and document them in the Requirements Document,
-  Complete business process reengineering of the functions to be supported (i.e., verify what information drives the business process, what information is generated, who generates it, where does the information go, and who processes it),
-  Develop detailed data and process models (system inputs, outputs, and the process.
-  Develop the test and evaluation requirements that will be used to determine acceptable system performance.

DESIGN PHASE

The design phase involves converting the informational, functional, and network requirements identified during the initiation and planning phases into unified design specifications that developers use to script programs during the development phase. Program designs are constructed in various ways. Using a top-down approach, designers first identify and link major program components and interfaces, then expand design layouts as they identify and link smaller subsystems and connections. Using a bottom-up approach, designers first identify and link minor program components and interfaces, then expand design layouts as they identify and link larger systems and connections. Contemporary design techniques often use prototyping tools that build mock-up designs of items such as application screens, database layouts, and system architectures. End users, designers, developers, database managers, and network administrators should review and refine the prototyped designs in an iterative process until they agree on an acceptable design. Audit, security, and quality assurance personnel should be involved in the review and approval process. During this phase, the system is designed to satisfy the functional requirements identified in the previous phase. Since problems in the design phase could be very expensive to solve in the later stage of the software development, a variety of elements are considered in the design to mitigate risk. These include:

-  Identifying potential risks and defining mitigating design features.

- ✚ Performing a security risk assessment.
- ✚ Developing a conversion plan to migrate current data to the new system.
- ✚ Determining the operating environment.
- ✚ Defining major subsystems and their inputs and outputs.
- ✚ Allocating processes to resources.
- ✚ Preparing detailed logic specifications for each software module. The result is a draft System Design Document which captures the preliminary design for the system.
- ✚ Everything requiring user input or approval is documented and reviewed by the user. Once these documents have been approved by the Agency CIO and Business Sponsor, the final System Design Document is created to serve as the Critical/Detailed Design for the system.
- ✚ This document receives a rigorous review by Agency technical and functional representatives to ensure that it satisfies the business requirements. Concurrent with the development of the system design, the Agency Project Manager begins development of the Implementation Plan, Operations and Maintenance Manual, and the Training Plan.

DEVELOPMENT PHASE

The development phase involves converting design specifications into executable programs. Effective development standards include requirements that programmers and other project participants discuss design specifications before programming begins. The procedures help ensure programmers clearly understand program designs and functional requirements. Programmers use various techniques to develop computer programs. The large transaction oriented programs associated with financial institutions have traditionally been developed using procedural programming techniques. Procedural programming involves the line-by-line scripting of logical instructions that are combined to form a program. Effective completion of the previous stages is a key factor in the success of the Development phase. The Development phase consists of:

- ✚ Translating the detailed requirements and design into system components.
- ✚ Testing individual elements (units) for usability.
- ✚ Preparing for integration and testing of the IT system.

INTEGRATION AND TEST PHASE

- ✚ Subsystem integration, system, security, and user acceptance testing is conducted during the integration and test phase. The user, with those responsible for quality assurance, validates that the functional requirements, as defined in the functional requirements document, are satisfied by the developed or modified system. OIT Security staff assess the system security and issue a security certification and accreditation prior to installation/implementation.

Multiple levels of testing are performed, including:

- ✚ Testing at the development facility by the contractor and possibly supported by end users
- ✚ Testing as a deployed system with end users working together with contract personnel
- ✚ Operational testing by the end user alone performing all functions. Requirements are traced throughout testing, a final Independent Verification & Validation evaluation is performed and all documentation is reviewed and accepted prior to acceptance of the system.

IMPLEMENTATION PHASE





This phase is initiated after the system has been tested and accepted by the user. In this phase, the system is installed to support the intended business functions. System performance is compared to performance objectives established during the

planning phase. Implementation includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes. This phase continues until the system is operating in production in accordance with the defined user requirements.

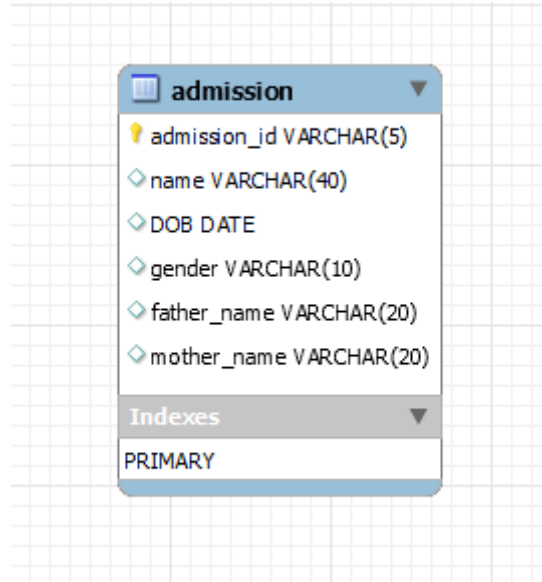
OPERATIONS AND MAINTENANCE PHASE

The system operation is on-going. The system is monitored for continued performance in accordance with user requirements and needed system modifications are incorporated. Operations continue as long as the system can be effectively adapted to respond to the organization's needs. When modifications or changes are identified, the system may re-enter the planning phase.

The purpose of this phase is to:

-  Operate, maintain, and enhance the system.
-  Certify that the system can process sensitive information.
-  Conduct periodic assessments of the system to ensure the functional requirements continue to be satisfied.
-  Determine when the system needs to be modernized, replaced, or retired.

DATABASE DESIGN



SOURCE CODE:

HOME PAGE

```
from tkinter import *
import Admission_page
import view_detail
from PIL import ImageTk, Image
#=====Base=====
root=Tk()
root.title("Dance Class Management System")
root.geometry("1920x1080+0+0")
root.config(bg="#F8F1E9")

img_resize=Image.open(r'D:\The Actual
Project\image_bg.jpg').resize((1440,720))
pic=ImageTk.PhotoImage(img_resize)
label=Label(root,image=pic)
label.place(x=0)

#=====Heading=====
=====
```

```
Title= Label(root,text=" Dance School Management System",fg="white",bg="#AA6F73",font=("Times New Roman",18,"bold"),bd=7,relief="ridge")
Title.place(x=0,y=0,width=1355,height=70)
```

```
def ad_mision():
    Admission_page.display()
```

```
def view_detail_fun():
    view_detail.display()
```

```
#===== Buttons=====
admission_b=Button(root,text="Admission",font=("Times New Roman",15,"bold"),command=ad_mision,bg="#BF656F",fg="white",padx=50,pady=50,relief="ridge")
admission_b.place(x=350,y=150,height=50,width=200)

view_b=Button(root,text="View Student Details",font=("Times New Roman",15,"bold"),command=view_detail_fun,bg="#BF656F",fg="white",padx=50,pady=50,relief="ridge")
view_b.place(x=850,y=150,height=50,width=200)
```

```
root.mainloop()
```

ADMISSION PAGE

```
from tkinter import *
```

```
from tkinter import ttk
```

```
import mysql.connector as mysql
```

```
import tkinter.messagebox as MessageBox
```

```
from PIL import ImageTk, Image
```

```
#=====
```

```
#=====Base=====
```

```
def display():
```

```
    root_1=Tk()
```

```
    root_1.title("Dance Class Management System")
```

```
    root_1.geometry("500x1080+0+0")
```

```
    root_1.config(bg="#F8F1E9")
```

```
#=====Heading=====
```

```
hdng=Label(root_1,text="Admission",bg="#AA6F73",font=("Time  
s New Roman",16,"bold"),bd=7,relief="ridge")
```

```
    hdng.place(x=0,y=0,width=500,height=70)
```

#=====Entry Labels=====

#=====Admission_no=====

```
adm_no_lbl=Label(root_1,text="Admission \n No
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
adm_no_lbl.place(x=80,y=100)
```

```
adm_no=Entry(root_1,bd=5)
```

```
adm_no.place(x=180,y=110)
```

#==name==

```
name_lbl=Label(root_1,text="Name
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
name_lbl.place(x=100,y=150)
```

```
name=Entry(root_1,bd=5)
```

```
name.place(x=180,y=150)
```

#==DOB==

```
DOB_lbl=Label(root_1,text="DOB
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
DOB_lbl.place(x=100,y=200)
```

```
DOB_y=Entry(root_1,bd=5)
```

```
DOB_y.place(x=180,y=200)
```

```
#==Gender==
```

```
gender_lbl=Label(root_1,text="Gender  
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
```

```
gender_lbl.place(x=100,y=255)
```

```
gender_entry=ttk.Combobox(root_1,font=("Arial",12,"bold"))
```

```
gender_entry['values']=("Female","Male","Neither")
```

```
gender_entry.place(x=180,y=255)
```

```
#==Father name==
```

```
f_name_lbl=Label(root_1,text="Father's \n Name  
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
```

```
f_name_lbl.place(x=100,y=305)
```

```
f_name=Entry(root_1,bd=5)
```

```
f_name.place(x=185,y=315)
```

```
#==Mother name==
```

```
m_name_lbl=Label(root_1,text="Mother's \n Name
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
m_name_lbl.place(x=100,y=375)
```

```
m_name=Entry(root_1,bd=5)
m_name.place(x=185,y=380)
```

```
continue_b=Button(root_1,text="Add",font=("Times New
Roman",16,"bold"),bg="#BF656F",fg="white",command=add)
continue_b.place(x=200,y=520)
```

```
root_1.mainloop()
```

```
#=====
```

```
def add():
```

```
con=mysql.connect(host="localhost",user="root",password="kavi
kavya123",database="dncedb")
cursor=con.cursor()
```

```

        cursor.execute(f"insert into admission
values('{admission_no.get()}', '{name.get()}', '{DOB_y.get()}', \

'{gender_entry.get()}', '{f_name.get()}', '{m_name.get()}')")
        cursor.execute("commit")
        con.close()

```

VIEW DETAIL

```

from tkinter import *
import Search_detail
import Updat_detail
import delete_detail
from PIL import ImageTk, Image
#=====Base=====

def search():
    Search_detail.display()

def update():
    Updat_detail.display()

def delete():
    delete_detail.display()

```

```

def display():
    root_2=Tk()
    root_2.title("Dance Class Management System")
    root_2.geometry("1920x1080+0+0")
    root_2.config(bg="#F8F1E9")

    #=====Heading=====
    hdng=Label(root_2,text="Student
Details",fg="white",bg="#AA6F73",font=("Times New
Roman",16,"bold"),bd=7,relief="ridge")
    hdng.place(x=0,y=0,width=1355,height=70)

    #=====Button=====
    search_b=Button(root_2,text="Search and View Student
detail",font=("Times New
Roman",15,"bold"),command=search,bg="#BF656F",fg="white",p
adx=50,pady=50,relief="ridge")
    search_b.place(x=520,y=150,height=70,width=300)

```



```

s_update_b=Button(root_2,text="Update Student
detail",font=("Times New
Roman",15,"bold"),command=update,bg="#BF656F",fg="white",p
adx=50,pady=50,relief="ridge")
s_update_b.place(x=520,y=300,height=70,width=300)

s_delete_b=Button(root_2,text="Delete Student
detail",font=("Times New
Roman",15,"bold"),command=delete,bg="#BF656F",fg="white",p
adx=50,pady=50,relief="ridge")
s_delete_b.place(x=520,y=450,height=70,width=300)

mainloop()

```

UPDATE DETAIL

```

from tkinter import *
from tkinter import ttk
import mysql.connector as mysql
import tkinter.messagebox as MessageBox

#=====Base=====

def display():
    base_3=Tk()
    base_3.title("Dance Class Management System")

```

```

base_3.geometry("800x1060+0+0")
base_3.config(bg="#F8F1E9")

#=====Heading=====

hdng=Label(base_3,text="Update Student
Details",fg='white',bg="#AA6F73",font=("Times New
Roman",16,"bold"),bd=4,relief="ridge")
hdng.place(x=0,y=0,width=800,height=70)

stud_id_lbl=Label(base_3,text="Student ID
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
stud_id_lbl.place(x=100,y=150)

stud_id=Entry(base_3,bd=5)
stud_id.place(x=210,y=150)

cat_lbl=Label(base_3,text="Category of Updation
:",bg="#F8F1E9",font=("Times New Roman",16,"bold"))
cat_lbl.place(x=80,y=250)

cat_entry=ttk.Combobox(base_3,font=("Arial",12,"bold"))

cat_entry['values']=("admission_id","name","DOB","gender","fathe
r_name","mother_name")

```

```

cat_entry.place(x=290,y=260)

updt_entry=Entry(base_3,bd=5)
updt_entry.place(x=600,y=260)

def update():

con=mysql.connect(host="localhost",user="root",password="kavi
kavya123",database="dncedb")
    cursor=con.cursor()
    cursor.execute(f"update admission set
{cat_entry.get()}=('{updt_entry.get()}') where
admission_id=('{stud_id.get()}')")
    cursor.execute("commit")
    con.close()

updt_b=Button(base_3,text="Update",bg="#AA6F73",fg="white",f
ont=("Times New Roman",16,"bold"),command=update)
updt_b.place(x=180,y=500)

base_3.mainloop()

```

SEARCH DETAIL

```

from tkinter import *
import mysql.connector as mysql
import tkinter.messagebox as MessageBox
from tkinter import ttk

#=====Base=====
def display():
    base_1=Tk()
    base_1.title("Dance Class Management System")
    base_1.geometry("930x1060+0+0")
    base_1.config(bg="#F8F1E9")

#=====Heading=====
def view():
    con=mysql.connect(host="localhost",user="root",password="kavi
    kavya123",database="dncedb")
    cursor=con.cursor()
    cursor.execute(f"select * from admission")
    result_set=cursor.fetchall()
    if len(result_set)!=0:
        data_table.delete(*data_table.get_children())
    for row in result_set:
        data_table.insert('',END,values=row)

```

else:

MessageBox.showinfo("Status","No record Found")

cursor.execute("commit")

con.close()

def search_tbl():

con=mysql.connect(host="localhost",user="root",password="kavi
kavya123",database="dncedb")

cursor=con.cursor()

cursor.execute(f"select * from admission where
{search_by.get()}='{val1.get()}' ")

result_set=cursor.fetchall()

if len(result_set)!=0:

data_table.delete(*data_table.get_children())

for row in result_set:

data_table.insert(' ',END,values=row)

else:

MessageBox.showinfo("Status","No record Found")

if search_by.get() and val1.get() == "":

MessageBox.showinfo("Status","Please select!")

cursor.execute("commit")

```
con.close()
```

```
data_frame=Frame(base_1,relief="groove",bg="#F8F1E9",bd=5)
```

```
data_frame.place(x=35,y=80,width=850,height=550)
```

```
search_frame=Frame(data_frame,relief="groove",bg="#AA6F73",  
bd=5)
```

```
search_frame.pack(side="top",fill="x")
```

```
search_by=ttk.Combobox(search_frame,font=("Arial",12,"bold"))
```

```
search_by['values']=("Admission_id","name","DOB","Gender","Fat  
her_name","Mother_name")
```

```
val1=Entry(search_frame)
```

```
val1.grid(row=0,column=1,padx=2,pady=2)
```

```
search_by.grid(row=0,column=0,padx=2,pady=2)
```

```
#-----add scroll bar-----  
-----
```

```
y_scroll=ttk.Scrollbar(data_frame,orient=VERTICAL)
x_scroll=ttk.Scrollbar(data_frame,orient=HORIZONTAL)
data_table=ttk.Treeview(data_frame,columns=("Admission_id","name","DOB","Gender","Father_name","Mother_name"),yscrollcommand=y_scroll.set,xscrollcommand=x_scroll.set)
y_scroll.config(command=data_table.yview)
x_scroll.config(command=data_table.xview)
y_scroll.pack(side=RIGHT,fill="y")
x_scroll.pack(side=BOTTOM,fill="x")
```

```
data_table.heading("Admission_id",text="Admission_id")
data_table.heading("name",text="name")
data_table.heading("DOB",text="DOB")
data_table.heading("Gender",text="Gender")
data_table.heading("Father_name",text="Father_name")
data_table.heading("Mother_name",text="Mother_name")
```

```
data_table.pack(fill="both",expand=True)
data_table["show"]='headings'
```

```
data_table.column("Admission_id",width=100)
data_table.column("name",width=100)
data_table.column("DOB",width=100)
```

```
data_table.column("Gender",width=100)
data_table.column("Father_name",width=100)
data_table.column("Mother_name",width=100)
```

```
data_table.pack(fill="both",expand=True)
data_table["show"]='headings'
```

```
view_b=Button(search_frame,text="View All",font=("Times New
Roman",15,"bold"),command=view,bg="#BF656F",fg="white",pad
x=50,pady=50,relief="ridge")
view_b.place(x=410,y=0,height=29,width=80)
```

```
hdng=Label(base_1,text="Search Student
Details",fg="white",bg="#AA6F73",font=("Times New
Roman",16,"bold"),bd=4,relief="ridge")
hdng.place(x=0,y=0,width=930,height=70)
```

```
search_b=Button(search_frame,text="Search",font=("Times New
Roman",15,"bold"),command=search_tbl,bg="#BF656F",fg="whit
e",padx=50,pady=50,relief="ridge")
search_b.place(x=610,y=0,height=29,width=75)
```



```
mainloop()
```

DELETE DETAIL

```
from tkinter import *
```

```
import mysql.connector as mysql
```

```
import tkinter.messagebox as MessageBox
```

```
from tkinter import ttk
```

```
#=====
```

```
#=====Base=====
```

```
def display():
```

```
    base_4=Tk()
```

```
    base_4.title("Dance Class Management System")
```

```
    base_4.geometry("500x500+0+0")
```

```
    base_4.config(bg="#F8F1E9")
```

```
    hdng=Label(base_4,text="Delete Student  
Details",fg="white",bg="#AA6F73",font=("Times New  
Roman",16,"bold"),bd=4,relief="ridge")
```

```
    hdng.place(x=0,y=0,width=500,height=70)
```

```
stud_id_lbl=Label(base_4,text="Student  
ID",bg="#F8F1E9",font=("Times New Roman",16,"bold"))  
stud_id_lbl.place(x=100,y=150)
```

```
stud_id=Entry(base_4,bd=5)  
stud_id.place(x=210,y=150)
```

```
def delete():  
  
con=mysql.connect(host="localhost",user="root",password="kavi  
kavya123",database="dncedb")  
    cursor=con.cursor()  
    cursor.execute(f"delete from admission where  
admission_id=('{stud_id.get()}')")  
    cursor.execute("commit")  
    con.close()
```

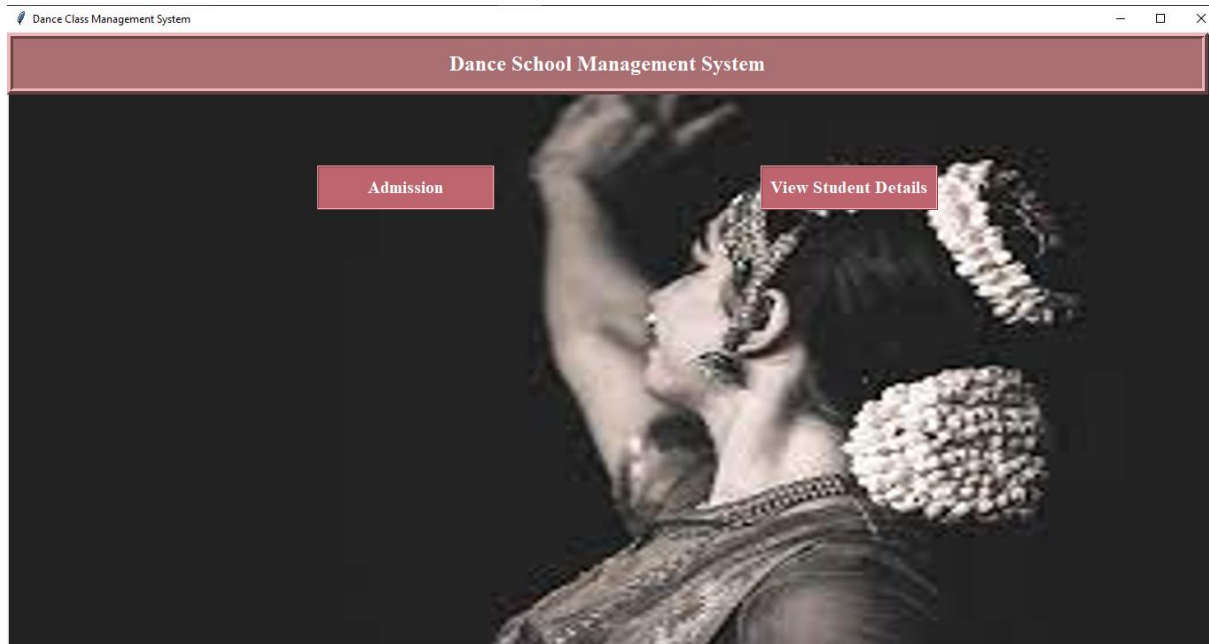
```
delete_b=Button(base_4,text="Delete",font=("Times New  
Roman",15,"bold"),command=delete,bg="#BF656F",fg="white",p  
adx=50,pady=50,relief="ridge")  
delete_b.place(x=25,y=250,height=50,width=200)
```

```
cancel_b=Button(base_4,text="Cancel",command=base_4.destroy,font=
("Times New Roman",15,"bold"),bg="#BF656F",fg="white",padx=50,pady=50,relief="ridge")
cancel_b.place(x=275,y=250,height=50,width=200)


mainloop()
```

OUTPUT:

Home Page:



On click of “Admission”:

 Dance Class Management System

Admission

Admission No :

Name :

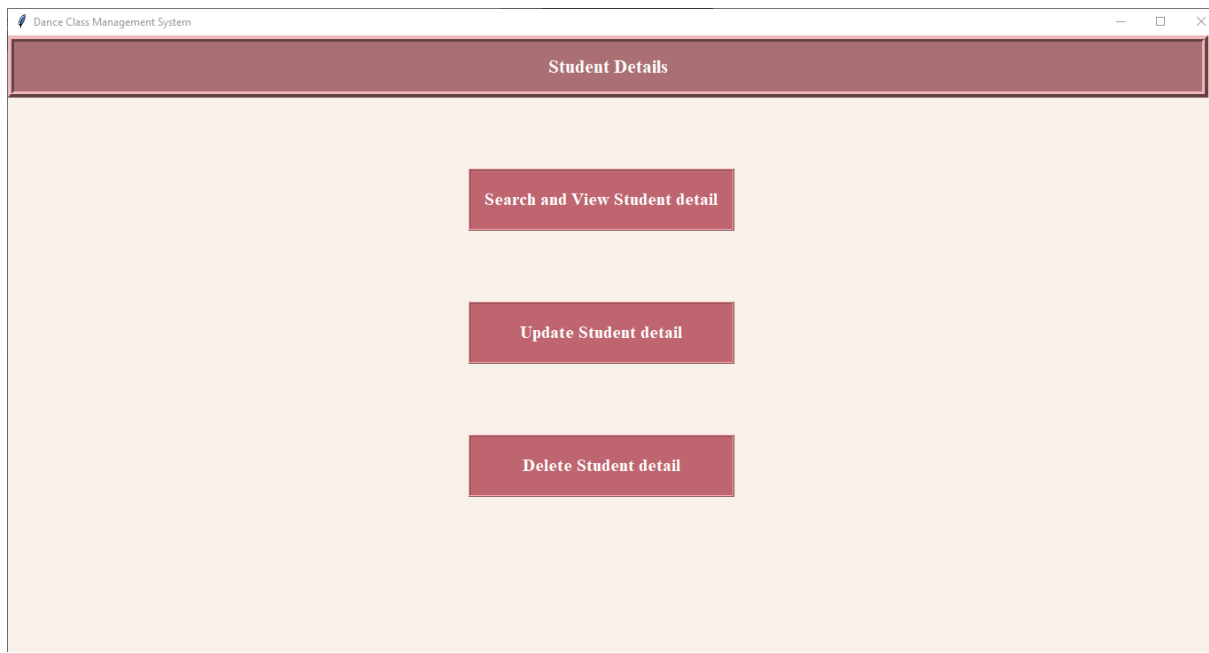
DOB :

Gender :

Father's Name :

Mother's Name :

On click of “View Student Details”:



Dance Class Management System

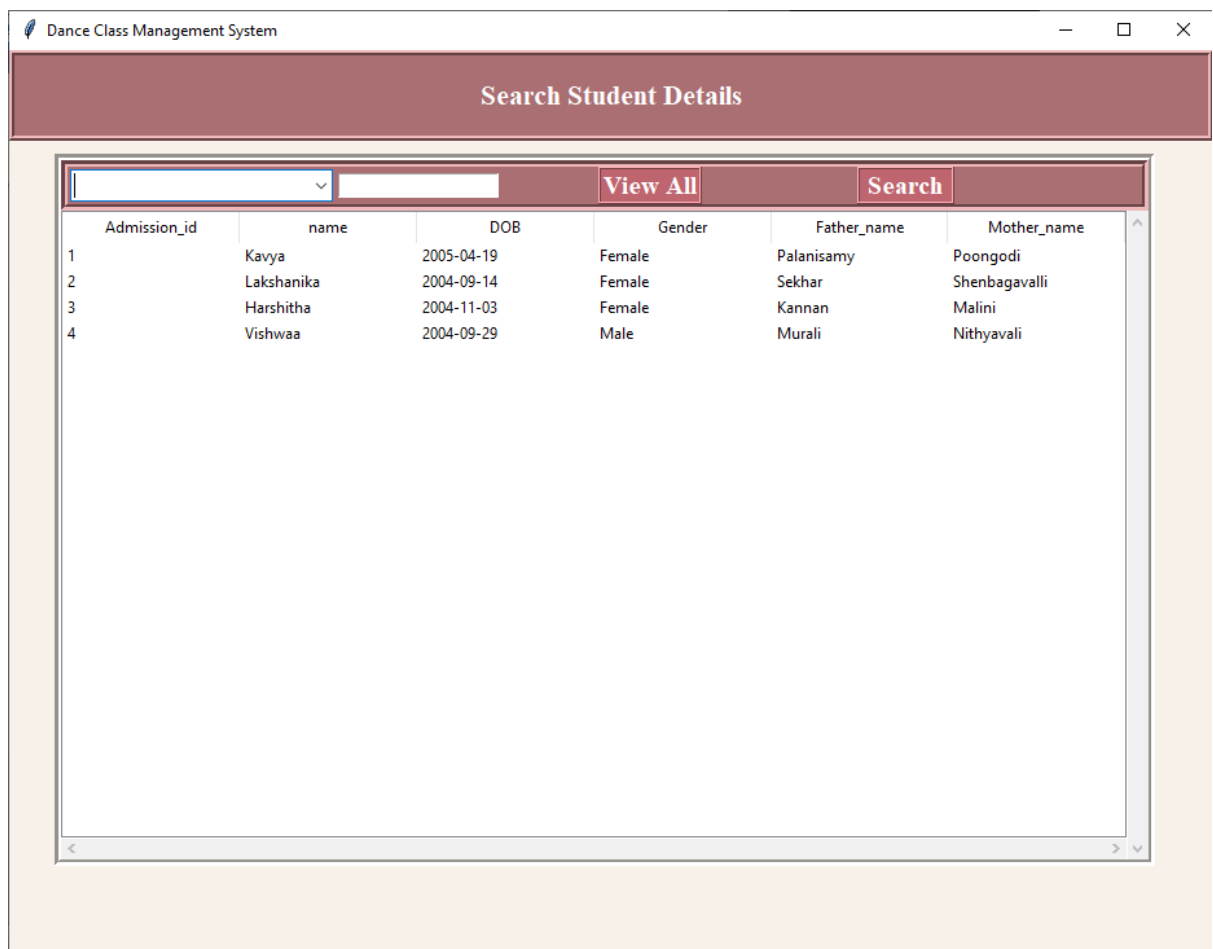
Student Details

Search and View Student detail

Update Student detail

Delete Student detail

i) On click of “Search and View Student detail”:



Dance Class Management System

Search Student Details

View All **Search**

Admission_id	name	DOB	Gender	Father_name	Mother_name
1	Kavya	2005-04-19	Female	Palanisamy	Poongodi
2	Lakshanika	2004-09-14	Female	Sekhar	Shenbagavalli
3	Harshitha	2004-11-03	Female	Kannan	Malini
4	Vishwaa	2004-09-29	Male	Murali	Nithyavali

ii) **On click of Update Student detail:**

The screenshot shows a web application window titled "Dance Class Management System". Inside the window, there is a form titled "Update Student Details". The form has a light beige background and a dark red header bar. The form contains the following fields and controls:

- Student ID :** A text input field containing the value "5".
- Category of Updation :** A dropdown menu with "DOB" selected.
- Date:** A date input field containing the value "10-02-2006".
- Update:** A dark red button with white text.

iii) **On click of Delete Student detail:**

The screenshot shows a window titled "Dance Class Management System" with standard window controls (minimize, maximize, close). Inside the window is a dialog box titled "Delete Student Details". The dialog box has a light beige background and a dark red header bar. Below the header, the text "Student ID" is displayed next to a text input field containing the number "5". At the bottom of the dialog box, there are two dark red buttons with white text: "Delete" and "Cancel".

TESTING

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test[1] , with respect to the context in which it is intended to operate. Software Testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks at implementation of the software. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

It can also be stated as the process of validating and verifying that a software program/application/product meets the business and technical requirements that guided its design and development, so that it works as expected and can be implemented with the same characteristics. Software Testing, depending on the testing method employed, can be implemented at any time in the development process, however the most test effort is employed after the requirements have been defined and coding process has been completed.

TESTING METHODS

Software testing methods are traditionally divided into black box testing and white box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

BLACK BOX TESTING

Black box testing treats the software as a "black box," without any knowledge of internal implementation. Black box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing.

SPECIFICATION-BASED TESTING

Specification-based testing aims to test the functionality of software according to the applicable requirements.[16] Thus, the tester inputs data into, and only sees the

output from, the test object. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behaviour), either "is" or "is not" the same as the expected value specified in the test case. Specification-based testing is necessary, but it is insufficient to guard against certain risks

ADVANTAGES AND DISADVANTAGES

The black box tester has no "bonds" with the code, and a tester's perception is very simple: a code must have bugs. Using the principle, "Ask and you shall receive," black box testers find bugs where programmers don't. But, on the other hand, black box testing has been said to be "like a walk in a dark labyrinth without a flashlight," because the tester doesn't know how the software being tested was actually constructed.



That's why there are situations when (1) a black box tester writes many test cases to check something that can be tested by only one test case, and/or (2) some parts of the back end are not tested at all. Therefore, black box testing has the advantage of "an unaffiliated opinion," on the one hand, and the disadvantage of "blind exploring," on the other.

WHITE BOX TESTING




White box testing, by contrast to black box testing, is when the tester has access to the internal data structures and algorithms (and the code that implement these)

Types of white box testing:-

The following types of white box testing exist:

-  api testing - Testing of the application using Public and Private APIs.
-  Code coverage - creating tests to satisfy some criteria of code coverage.



For example, the test designer can create tests to cause all statements in the program to be executed at least once.

-  fault injection methods.
-  mutation testing methods.
-  static testing - White box testing includes all static testing.

CODE COMPLETENESS EVALUATION

White box testing methods can also be used to evaluate the completeness of a test suite that was created with black box testing methods. This allows the software team to examine parts of a system that are rarely tested and ensures that the most important function points have been tested.

Two common forms of code coverage are:

-  Function Coverage: Which reports on functions executed and
-  Statement Coverage: Which reports on the number of lines executed to complete the test.

They both return coverage metric, measured as a percentage

Hardware Used:






Laptop with

- ✓ Operating system (OS) - Windows 10
- ✓ System type - 64-bit operating system, x64-based processor
- ✓ Processor – Intel
- ✓ Installed RAM - 8.00 GB

Software Used:

- ✓ Python 3.8.2
- ✓ Windows Operating System (OS)

INSTALLATION PROCEDURE

-  Install python and my sql from the web.
-  Open mysql and create the database **dncedb**
-  Then run the **home_page.py** file.
-  Next, a GUI based application opens.
-  It is user friendly, and easy to use.

BIBLIOGRAPHY

1. *Computer science With Python - Class XI* *By : SumitaArora*
2. https://www.tutorialspoint.com/python/python_gui_programming.htm
