**GOAL GETTERS**

*A*

*Mini Project Report*

*Submitted in partial fulfilment of the*

*Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

IN

**INFORMATION TECHNOLOGY**

By

**S. Bhavana \_ 1602-20-737-069**

**Nithya\_1602-20-737-110**

**S. Alekya\_1602-20-737-065**

****

**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Ibrahimbagh, Hyderabad-31**

**2022**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**

****

**DECLARATION BY THE CANDIDATE**

We, **S. Bhavana, Nithya** and **S. Alekya,** bearing hall ticket numbers, **1602-20-737-069, 1602-20-737-110** and **1602-20-737-065**, hereby declare that the project report entitled **“Goal Getters”** is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology**

This is a record of bonafide work carried out by us and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

**S. Bhavana**

**1602-20-737-069**

**Nithya**

**1602-20-737-110**

**S. Alekya**

**1602-20-737-065**

(Faculty In-Charge) (Head,Dept of IT)

**Acknowledgement**

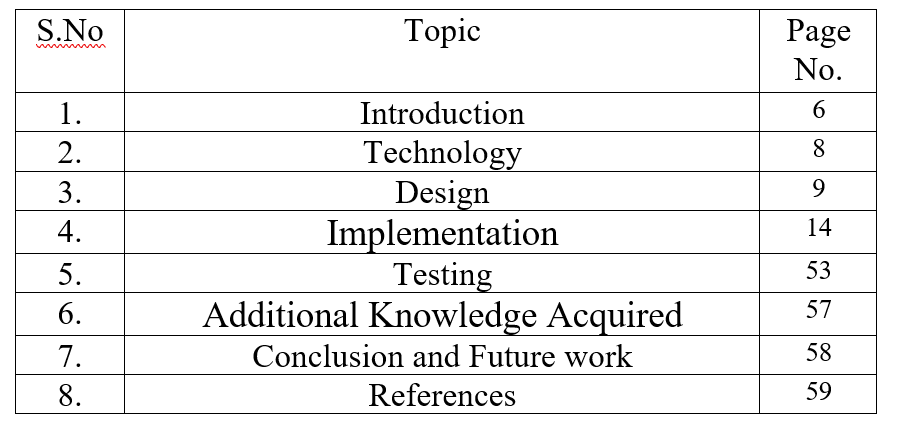
The enduring pages of the work are the cumulative sequence of extensive guidance and arduous work. We wish to acknowledge and express our personal gratitude to all those without whom this work could not have been reality.

We feel very delighted to get this rare opportunity to show our profound senses of reverences and indebtedness to our lectures for their keen and sustained interest, valuable advice, throughout the course of which led our mini project, to a successful completion. For this kind act of consideration, we are beholder to them in special manner and no one can fully convey our feelings of respect and regard for them.

**Abstract**

* Goal getters, the title itself describe the content as reaching of your goals based on some mcqs related to maths, physics and chemistry.
* Here we will ask the user some questions and provide some related options and you have to choose them correctly.
* We will provide time limit for each question.
* We are going to display the score based on their performances.
* Based on their score, we will recommend the books of the respective subjects.

**Table of contents**



**INTRODUCTION**

Have you ever come across situation where you went for a library and don’t know what book to be choose? or Have you ever come this situation where your family/friends/lecture recommend you a book but you don’t understand whether that book is equal to your knowledge or not? I am right?

So we came across a app where we first understand your knowledge by conducting a test in the format of mcqs and then based on result we will provide you some books and you can select your book in which your interested. This program is done using python code in which we imported modules like tkinter, random, sqlite3, time, sys.

Tkinter is used for creating outlook of the app like background, app size, background photo, background colour etc. we have imported random to get random questions on each login to the exam. Time function has been imported to set the time on each question so that the user get more concentrated in answering the questions.

To define a login page we have created a function called loginPage in which we have defined variables like user\_name and password. Login page help us to directly login to the site so that your data will be saved.

After user get login to the app user has to choose the subjects from the given list(maths, physics, chemistry) after selecting the subjects you will have 9 questions which is displayed one after the other, in this 9 questions first 3 are easy questions, next 3 are moderate type questions, and next 3 are high level questions and you will have a timer of 20 sec for each questions. Questions will be pity easy we need to just apply logic. After answering all the 9 questions your score and based on your score some books will get displayed.

The books are recommended based on your score like if your score is in between 0-3 then we will provide basic books in that subjects, and if your score is in between 3-6 then we will provide moderate standard books and if your score is in between 7-9 then we will provide high standard books.

After choosing your book you can exit from the page.

**TECHNOLOGY**

**Hardware Requirements**

* 512 MB RAM
* 2GB HDD
* CORE i5

**Software Requirements**

* Windows XP/ Windows 2000
* PYTHON INTERPRETER

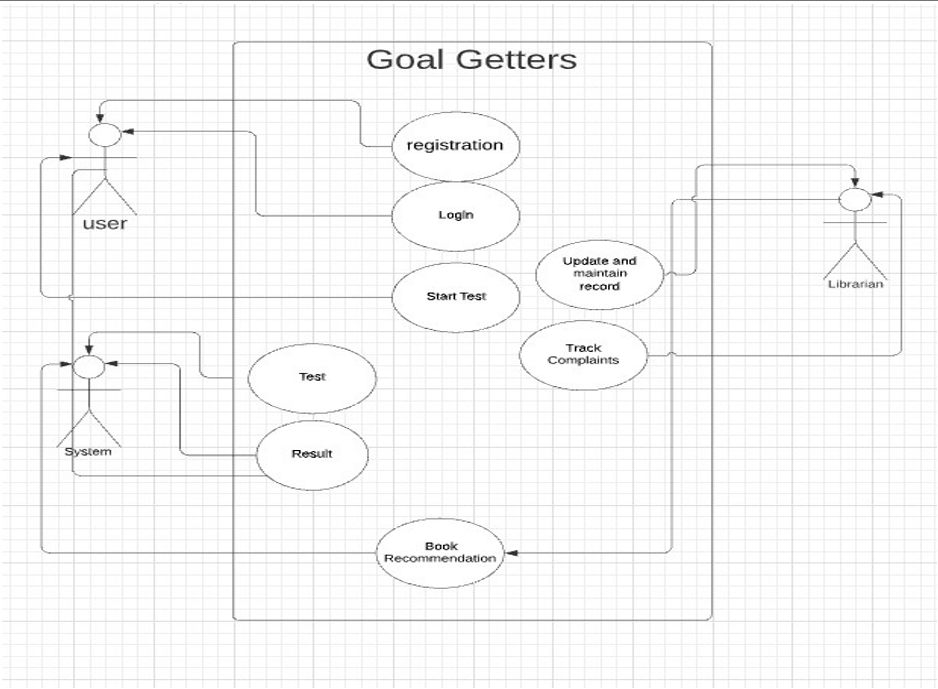
**Packages used**

* tkinter
* random

**PROPOSED WORK**

**DESIGN –**

USE CASE DIAGRAM –



**Use case descriptions**

Use Case ID: UC01

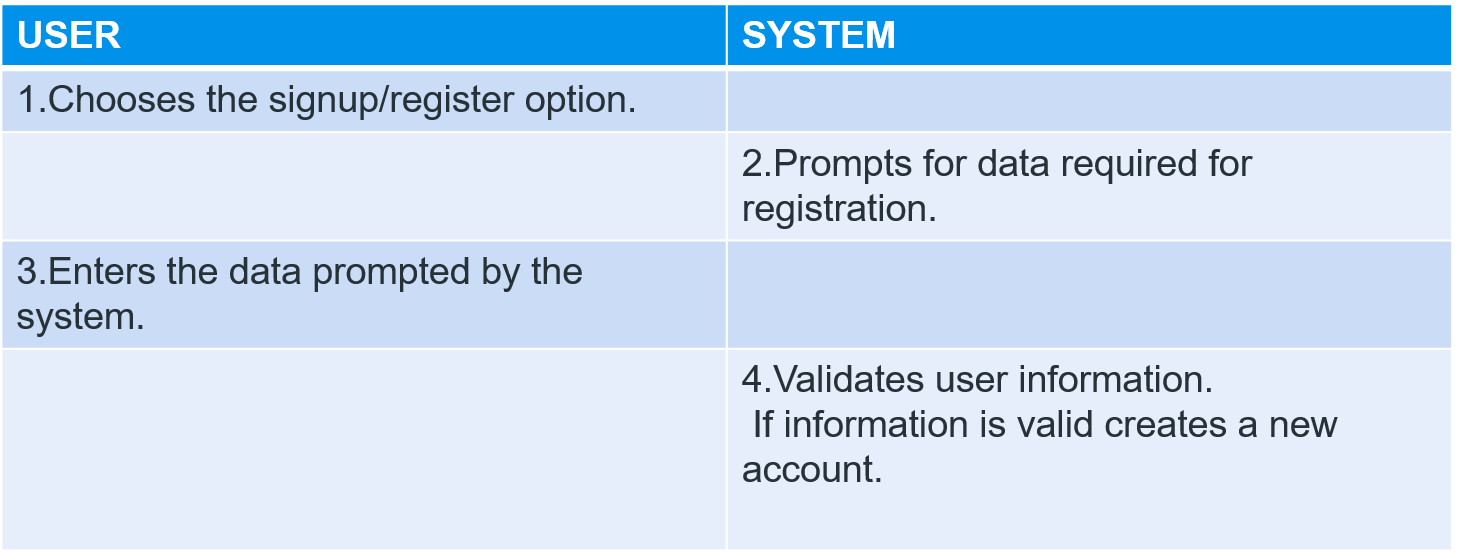
Name: Register

Actors: User

Description: Allows new user to register for an account

Pre-conditions: None

Post-conditions: An account is created for the user



Use Case ID: UC02

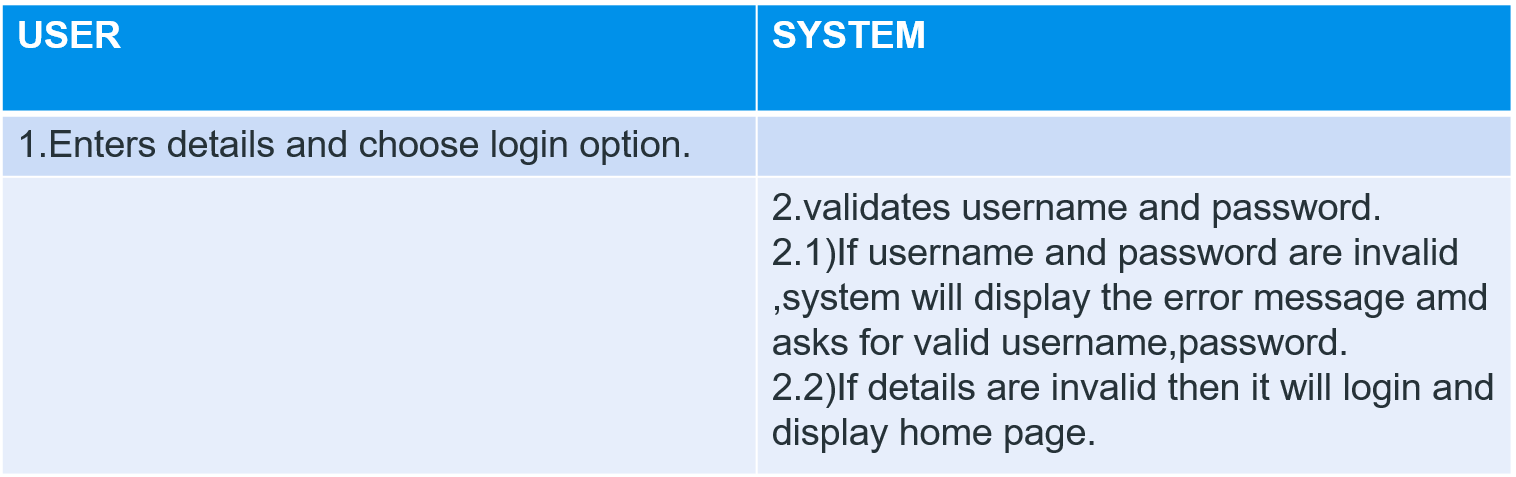
Name: Login

Actors: User

Description: Allows registered users to Login.

Pre-conditions: user should be registered with the system.

Post-conditions: User logs in and all the options are displayed on the screen.



Use case id:UC03

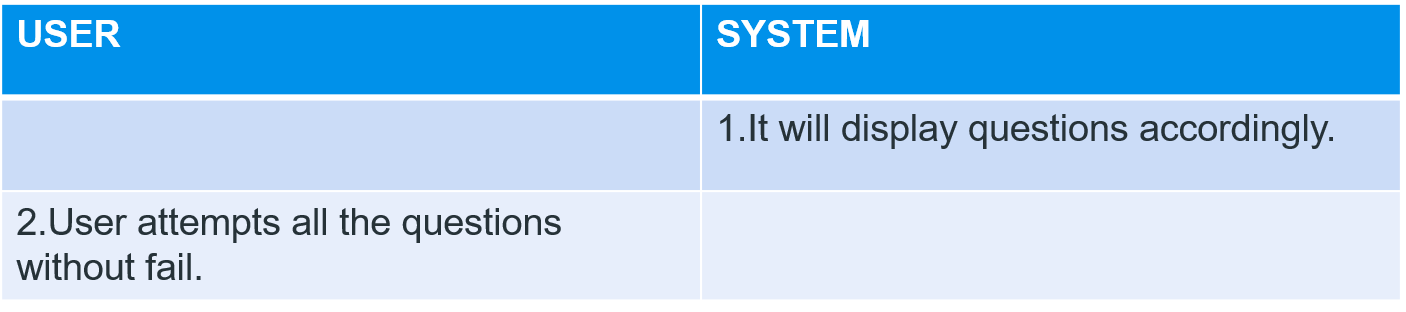
Name:Test.

Actors:User.

Description:Starts taking test.

Pre-conditions:User should login successfully.

Post-conditions:User attempts the test.



Use case id:UC04

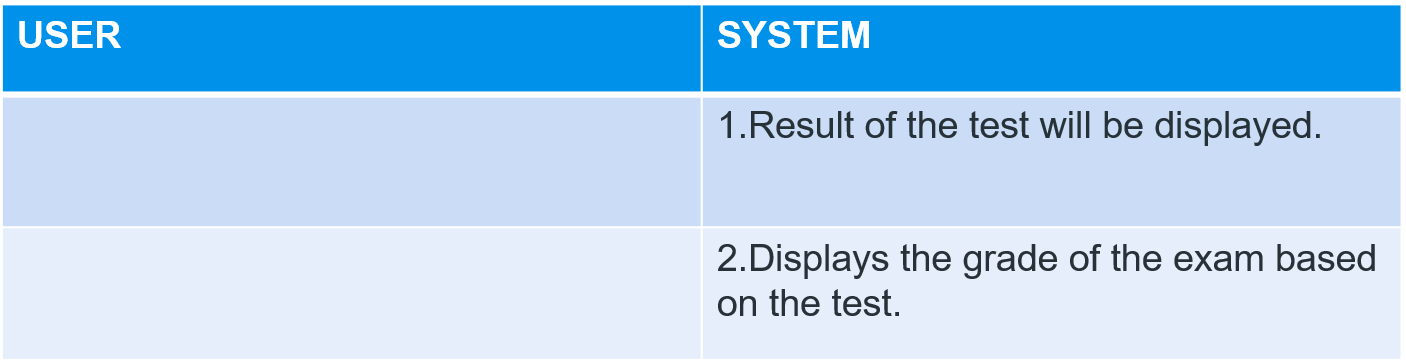
Name:Result.

Actors:System.

Description:Result will be displayed.

Pre-conditions:User should attempt their test.

Post-conditions:Result will be dsiplayed.



Use case id:UC05

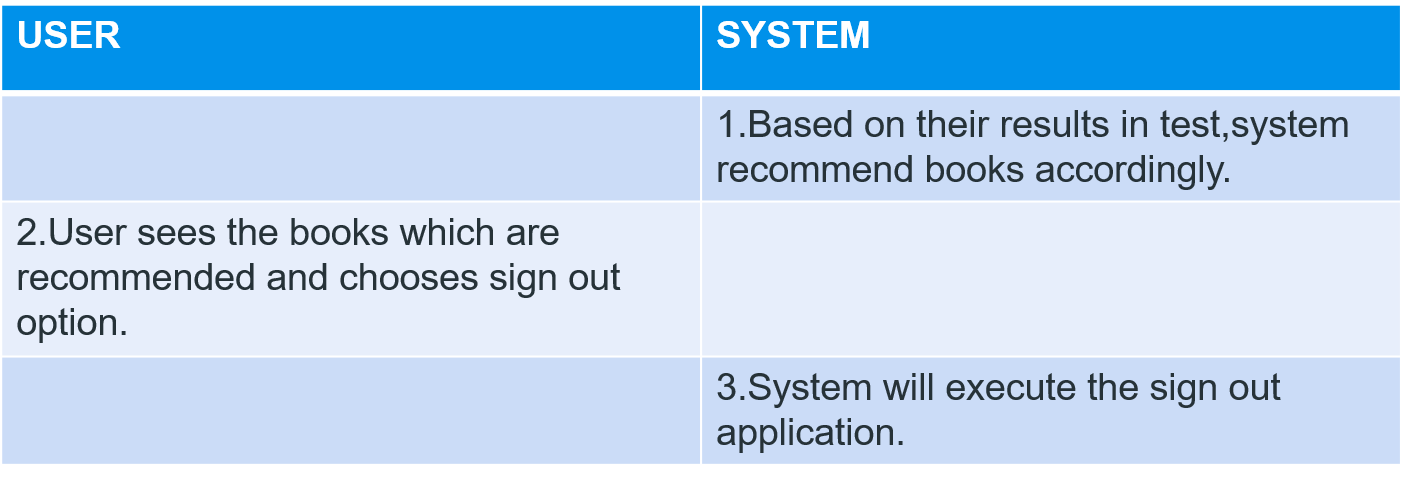
Name:Books Recommendation.

Actors:System.

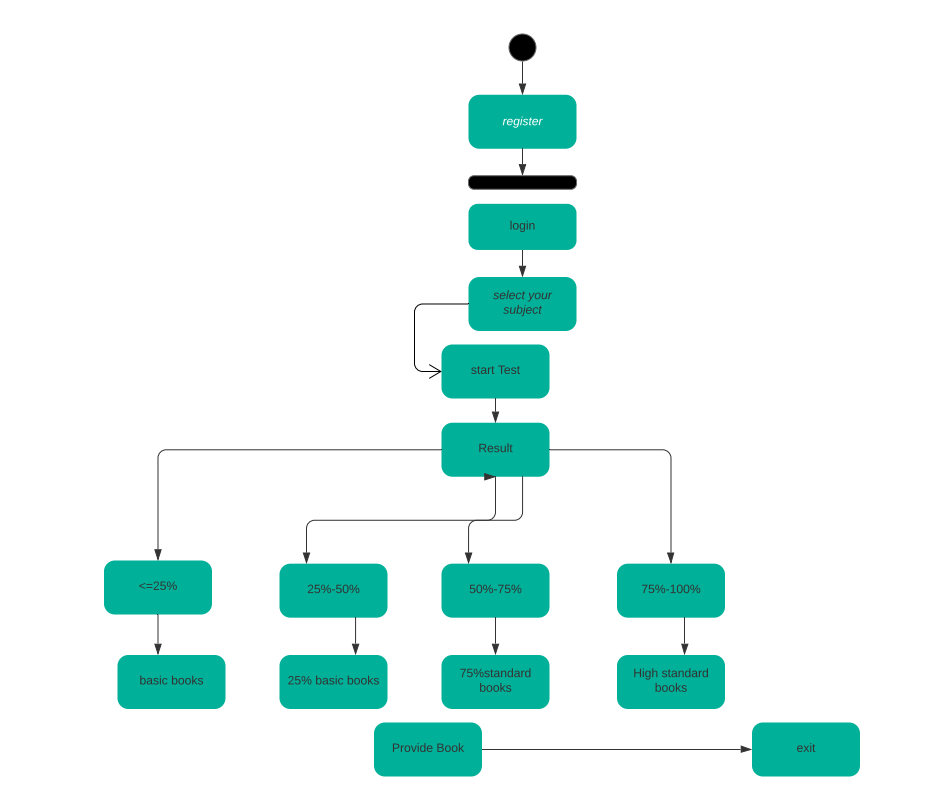
Description:Based on their result,books will be recommended.

Pre-conditions:User should get their result of exam.

Post-conditions:Exit from app.



**Activity diagram**

****

**Implementation**

**Source code-**

import tkinter as tk

from tkinter import \*

import random

import sqlite3

import time

import sys

def loginPage(logdata):

sup.destroy()

global login

login = Tk()

user\_name = StringVar()

password = StringVar()

login\_canvas = Canvas(login,width=720,height=440,bg="purple")

login\_canvas.pack()

login\_frame = Frame(login\_canvas,bg="white")

login\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

heading = Label(login\_frame,text="Goalgetters App Login",fg="black",bg="white")

heading.config(font=('calibri 40'))

heading.place(relx=0.1,rely=0.1)

#USER NAME

ulabel = Label(login\_frame,text="Username",fg='black',bg='white')

ulabel.place(relx=0.21,rely=0.4)

uname = Entry(login\_frame,bg='#d3d3d3',fg='black',textvariable = user\_name)

uname.config(width=42)

uname.place(relx=0.31,rely=0.4)

#PASSWORD

plabel = Label(login\_frame,text="Password",fg='black',bg='white')

plabel.place(relx=0.215,rely=0.5)

pas = Entry(login\_frame,bg='#d3d3d3',fg='black',show="\*",textvariable = password)

pas.config(width=42)

pas.place(relx=0.31,rely=0.5)

def check():

for a,b,c,d in logdata:

if b == uname.get() and c == pas.get():

menu()

break

else:

error = Label(login\_frame,text="Wrong Username or Password!",fg='black',bg='white')

error.place(relx=0.37,rely=0.7)

#LOGIN BUTTON

log = Button(login\_frame,text='Login',padx=5,pady=5,width=5,command=check)

log.configure(width = 15,height=1, activebackground = "#33B5E5", relief = FLAT)

log.place(relx=0.4,rely=0.6)

login.mainloop()

def signUpPage():

root.destroy()

global sup

sup = Tk()

fname = StringVar()

uname = StringVar()

passW = StringVar()

email = StringVar()

sup\_canvas = Canvas(sup,width=720,height=440,bg="purple")

sup\_canvas.pack()

sup\_frame = Frame(sup\_canvas,bg="white")

sup\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

heading = Label(sup\_frame,text="Goalgetters App SignUp",fg="black",bg="white")

heading.config(font=('calibri 40'))

heading.place(relx=0.1,rely=0.1)

#full name

flabel = Label(sup\_frame,text="Full Name",fg='black',bg='white')

flabel.place(relx=0.21,rely=0.4)

fname = Entry(sup\_frame,bg='#d3d3d3',fg='black',textvariable = fname)

fname.config(width=42)

fname.place(relx=0.31,rely=0.4)

#username

ulabel = Label(sup\_frame,text="Username",fg='black',bg='white')

ulabel.place(relx=0.21,rely=0.5)

user = Entry(sup\_frame,bg='#d3d3d3',fg='black',textvariable = uname)

user.config(width=42)

user.place(relx=0.31,rely=0.5)

#password

plabel = Label(sup\_frame,text="Password",fg='black',bg='white')

plabel.place(relx=0.215,rely=0.6)

pas = Entry(sup\_frame,bg='#d3d3d3',fg='black',show="\*",textvariable = passW)

pas.config(width=42)

pas.place(relx=0.31,rely=0.6)

#email

elabel = Label(sup\_frame,text="email",fg='black',bg='white')

elabel.place(relx=0.215,rely=0.7)

e = Entry(sup\_frame,bg='#d3d3d3',fg='black',textvariable = email)

e.config(width=42)

e.place(relx=0.31,rely=0.7)

def addUserToDataBase():

fullname = fname.get()

username = user.get()

password = pas.get()

email = e.get()

conn = sqlite3.connect('quiz.db')

create = conn.cursor()

create.execute('CREATE TABLE IF NOT EXISTS userSignUp(FULLNAME text, USERNAME text,PASSWORD text,COUNTRY text)')

create.execute("INSERT INTO userSignUp VALUES (?,?,?,?)",(fullname,username,password,email))

conn.commit()

create.execute('SELECT \* FROM userSignUp')

z=create.fetchall()

print(z)

# L2.config(text="Username is "+z[0][0]+"\nPassword is "+z[-1][1])

conn.close()

loginPage(z)

def gotoLogin():

conn = sqlite3.connect('quiz.db')

create = conn.cursor()

conn.commit()

create.execute('SELECT \* FROM userSignUp')

z=create.fetchall()

loginPage(z)

#signup BUTTON

sp = Button(sup\_frame,text='SignUp',padx=5,pady=5,width=5,command = addUserToDataBase,bg='green')

sp.configure(width = 15,height=1, activebackground = "#33B5E5", relief = FLAT)

sp.place(relx=0.4,rely=0.8)

log = Button(sup\_frame,text='Already have a Account?',padx=5,pady=5,width=5,command = gotoLogin,bg="white",fg='purple')

log.configure(width = 16,height=1, activebackground = "#33B5E5", relief = FLAT)

log.place(relx=0.4,rely=0.9)

sup.mainloop()

def menu():

login.destroy()

global menu

menu = Tk()

menu\_canvas = Canvas(menu,width=720,height=440,bg="light cyan")

menu\_canvas.pack()

menu\_frame = Frame(menu\_canvas,bg="white")

menu\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

wel = Label(menu\_canvas,text=' W E L C O M E T O G O A L G E T T E R S ',fg="white",bg="#101357")

wel.config(font=('Broadway 22'))

wel.place(relx=0.1,rely=0.02)

level = Label(menu\_frame,text='Select your Subject !!',bg="white",font="calibri 18")

level.place(relx=0.25,rely=0.3)

var = IntVar()

mathsR = Radiobutton(menu\_frame,text='maths',bg="white",font="calibri 16",value=1,variable = var)

mathsR.place(relx=0.25,rely=0.4)

physicsR = Radiobutton(menu\_frame,text='physics',bg="white",font="calibri 16",value=2,variable = var)

physicsR.place(relx=0.25,rely=0.5)

chemistryR = Radiobutton(menu\_frame,text='Chemistry',bg="white",font="calibri 16",value=3,variable = var)

chemistryR.place(relx=0.25,rely=0.6)

def navigate():

x = var.get()

print(x)

if x == 1:

menu.destroy()

maths()

elif x == 2:

menu.destroy()

physics()

elif x == 3:

menu.destroy()

chemistry()

else:

pass

letsgo = Button(menu\_frame,text="Let's Go",bg="white",font="calibri 12",command=navigate)

letsgo.place(relx=0.25,rely=0.8)

menu.mainloop()

def maths():

global e

e = Tk()

maths\_canvas = Canvas(e,width=720,height=440,bg="#101357")

maths\_canvas.pack()

maths\_frame = Frame(maths\_canvas,bg="white")

maths\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

def countDown():

check = 0

for k in range(20, 0, -1):

if k == 1:

check=-1

timer.configure(text=k)

maths\_frame.update()

time.sleep(1)

timer.configure(text="Times up!")

if check==-1:

return (-1)

else:

return 0

global score

score = 0

mathsQ = [

[

"7+2x=15 find x",

"4",

"3",

"2",

"1"

],

[

"Find the number of triangles in an octagon",

"326",

"120",

"56",

"Cannot be determined"

] ,

[

"An event in probability that will never happen is called as",

"Unsure event",

"Sure event",

"Possible event",

"Impossible event"

],

[

"The total number of integers between 200 and 400,each of\n "

"which either begins with3 or ends with 3 or both,is",

"10",

"100",

"110",

"120"

],

[

"Find the area of the sector covered by the hour hand after\n"

"it has moved through 3 hours if the length of the hour hand is 7cm",

"77 sq.cm",

"38.5 sq.cm",

"35 sq.cm",

"70 sq.cm"

],

[

"What is the probability of getting 1 and 5 if a dice is thrown once",

"1/6",

"1/3",

"2/3",

"8/9"

],

[

"if x=y+z then x^3-y^3-z^3 is",

"0",

"3xyz",

"-3xyz",

"1"

],

[

"What is the ratio measures of in-radius,circum radius and\n"

"one of the ex-radius of an equilateral traingle?",

"1:2:5",

"1:3:5",

"1:2:3",

"1:1.14142:2"

],

[

"What is the probability of getting sum as 3 if a dice is thrown?",

"2/18",

"1/18",

"1/24",

"1/36"

]

]

answer = [

"4",

"56",

"Impossible event",

"110",

"38.5 sq.cm",

"1/3",

"3xyz",

"1:2:3",

"1/18"

]

li = ['',0,1,2,3,4,5,6,7,8]

x = random.choice(li[1:])

ques = Label(maths\_frame,text =mathsQ[x][0],font="calibri 12",bg="white")

ques.place(relx=0.5,rely=0.2,anchor=CENTER)

var = StringVar()

a = Radiobutton(maths\_frame,text=mathsQ[x][1],font="calibri 10",value=mathsQ[x][1],variable = var,bg="white")

a.place(relx=0.5,rely=0.42,anchor=CENTER)

b = Radiobutton(maths\_frame,text=mathsQ[x][2],font="calibri 10",value=mathsQ[x][2],variable = var,bg="white")

b.place(relx=0.5,rely=0.52,anchor=CENTER)

c = Radiobutton(maths\_frame,text=mathsQ[x][3],font="calibri 10",value=mathsQ[x][3],variable = var,bg="white")

c.place(relx=0.5,rely=0.62,anchor=CENTER)

d = Radiobutton(maths\_frame,text=mathsQ[x][4],font="calibri 10",value=mathsQ[x][4],variable = var,bg="white")

d.place(relx=0.5,rely=0.72,anchor=CENTER)

li.remove(x)

timer = Label(e)

timer.place(relx=0.8,rely=0.82,anchor=CENTER)

def display():

def showMark(mark):

sh = Tk()

show\_canvas = Canvas(sh,width=720,height=440,bg="#101357")

show\_canvas.pack()

show\_frame = Frame(show\_canvas,bg="white")

show\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

st = "Your score is "+str(mark)

mlabel = Label(show\_canvas,text=st,fg="black")

mlabel.place(relx=0.5,rely=0.2,anchor=CENTER)

def exit():

sh.destroy()

if(mark==0 or mark==1 or mark==2 or mark==3):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,font="arial 10",fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="Fundamentals of algebra by career Launcher"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="School geometry by H.S Hall"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="An introduction to probability by william feller"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

elif(mark==4 or mark==5 or mark==6):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="Algebra and Trignometry by Robert F.Blitzer"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="The plane geometry by Sl loney"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="Probability and Statistics by Morris H.DeGroot"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

elif(mark==7 or mark==8 or mark==9):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="higher algebra by hall and knight"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="Advanced problems in geometry by Vikas Gupta"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="Challenging problems in probabilty by federick"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

exit = Button(show\_frame,command=exit,text="exit")

exit.place(relx=0.5,rely=0.9,anchor=CENTER)

if len(li) == 1:

e.destroy()

showMark(score)

if len(li) == 2:

nextQuestion.configure(text='End',command=calc)

if li:

x = random.choice(li[1:])

ques.configure(text =mathsQ[x][0])

a.configure(text=mathsQ[x][1],value=mathsQ[x][1])

b.configure(text=mathsQ[x][2],value=mathsQ[x][2])

c.configure(text=mathsQ[x][3],value=mathsQ[x][3])

d.configure(text=mathsQ[x][4],value=mathsQ[x][4])

li.remove(x)

print(li)

y = countDown()

if y == -1:

display()

def calc():

global score

if (var.get() in answer):

score+=1

display()

submit = Button(maths\_frame,command=calc,text="Submit")

submit.place(relx=0.5,rely=0.82,anchor=CENTER)

nextQuestion = Button(maths\_frame,command=display,text="Next")

nextQuestion.place(relx=0.87,rely=0.82,anchor=CENTER)

y = countDown()

if y == -1:

display()

e.mainloop()

def physics():

global m

m = Tk()

physics\_canvas = Canvas(m,width=720,height=440,bg="#101357")

physics\_canvas.pack()

physics\_frame = Frame(physics\_canvas,bg="white")

physics\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

def countDown():

check = 0

for k in range(20, 0, -1):

if k == 1:

check=-1

timer.configure(text=k)

physics\_frame.update()

time.sleep(1)

timer.configure(text="Times up!")

if check==-1:

return (-1)

else:

return 0

global score

score = 0

physicsQ = [

[

"The working principle of washing machine is ",

"reverse osmosis",

"diffusion",

"centrifugation",

"dialysis"

],

[

"What is the ratio of normal stress to the volumetric strain \nwithin the elastic limits is called as?",

"Bulk modulus",

"Young modulus",

"Modulus of Rigidity",

"None of the above"

],

[

"Among various electrical safety devices, one based on the\n heating effect of electric current is called a:",

"Capacitor",

"Fuse",

"Protective relay",

"Surge protector"

],

[

"What is the term used to describe the angular distance\n of a place north or south of Earth's equator?",

"Pole",

"Longitude",

"Latitude",

"Hemisphere"

],

[

"What will happen to a person's Weight when he is in a moving elevator?",

"Increase",

"Decrease",

"Weight will not change",

"May increase or decrease"

],

[

"Which of the following objects would possess potential energy?",

"Raised hammer",

"Blowing wind",

"A rolling stone",

"Moving bullet"

],

[

"The term \_\_ is prefixed to scientific terms to describe something that is constant.",

"Photo",

"Iso",

"Mega",

"Quasi"

],

[

"\_ expansion makes the Eiffel Tower taller during summers.",

"Thermal",

"Chemical",

"Gravitational",

"Gradient"

],

[

"A nautical mile is equal to ",

"2000 meters",

"1852 meters",

"1672 meters",

"2450 meters"

]

]

answer = [

"centrifugation",

"Bulk modulus",

"Fuse",

"Latitude",

"Weight will not change",

"Raised hammer",

"Iso",

"Thermal",

"1852 meters"

]

li = ['',0,1,2,3,4,5,6,7,8]

x = random.choice(li[1:])

ques = Label(physics\_frame,text =physicsQ[x][0],font="calibri 12",bg="white")

ques.place(relx=0.5,rely=0.2,anchor=CENTER)

var = StringVar()

a = Radiobutton(physics\_frame,text=physicsQ[x][1],font="calibri 10",value=physicsQ[x][1],variable = var,bg="white")

a.place(relx=0.5,rely=0.42,anchor=CENTER)

b = Radiobutton(physics\_frame,text=physicsQ[x][2],font="calibri 10",value=physicsQ[x][2],variable = var,bg="white")

b.place(relx=0.5,rely=0.52,anchor=CENTER)

c = Radiobutton(physics\_frame,text=physicsQ[x][3],font="calibri 10",value=physicsQ[x][3],variable = var,bg="white")

c.place(relx=0.5,rely=0.62,anchor=CENTER)

d = Radiobutton(physics\_frame,text=physicsQ[x][4],font="calibri 10",value=physicsQ[x][4],variable = var,bg="white")

d.place(relx=0.5,rely=0.72,anchor=CENTER)

li.remove(x)

timer = Label(m)

timer.place(relx=0.8,rely=0.82,anchor=CENTER)

def display():

def showMark(mark):

sh = Tk()

show\_canvas = Canvas(sh,width=720,height=440,bg="#101357")

show\_canvas.pack()

show\_frame = Frame(show\_canvas,bg="white")

show\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

st = "Your score is "+str(mark)

mlabel = Label(show\_canvas,text=st,fg="black")

mlabel.place(relx=0.5,rely=0.2,anchor=CENTER)

def exit():

sh.destroy()

if(mark==0 or mark==1 or mark==2 or mark==3):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,font="arial 10",fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="Fundamentals of Physics by R.Shankar"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="College Physics by Randall Knight"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="From Classical to Quantum fields"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

elif(mark==4 or mark==5 or mark==6):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="Tensors by Namrata dewan Soni"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="Concepts of physics by Shivami"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="H.C Verma"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

elif(mark==7 or mark==8 or mark==9):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="Quantum mechanics concepts and applications "

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="Applied physics by Manasi karkare"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="Physics for IIT by Ashwani kumar Sharma"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

exit = Button(show\_frame,command=exit,text="exit")

exit.place(relx=0.5,rely=0.9,anchor=CENTER)

if len(li) == 1:

m.destroy()

showMark(score)

if len(li) == 2:

nextQuestion.configure(text='End',command=calc)

if li:

x = random.choice(li[1:])

ques.configure(text =physicsQ[x][0])

a.configure(text=physicsQ[x][1],value=physicsQ[x][1])

b.configure(text=physicsQ[x][2],value=physicsQ[x][2])

c.configure(text=physicsQ[x][3],value=physicsQ[x][3])

d.configure(text=physicsQ[x][4],value=physicsQ[x][4])

li.remove(x)

print(li)

y = countDown()

if y == -1:

display()

def calc():

global score

if (var.get() in answer):

score+=1

display()

submit = Button(physics\_frame,command=calc,text="Submit")

submit.place(relx=0.5,rely=0.82,anchor=CENTER)

nextQuestion = Button(physics\_frame,command=display,text="Next")

nextQuestion.place(relx=0.87,rely=0.82,anchor=CENTER)

y = countDown()

if y == -1:

display()

m.mainloop()

def chemistry():

global h

h = Tk()

chemistry\_canvas = Canvas(h,width=720,height=440,bg="#101357")

chemistry\_canvas.pack()

chemistry\_frame = Frame(chemistry\_canvas,bg="white")

chemistry\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

def countDown():

check = 0

for k in range(20, 0, -1):

if k == 1:

check=-1

timer.configure(text=k)

chemistry\_frame.update()

time.sleep(1)

timer.configure(text="Times up!")

if check==-1:

return (-1)

else:

return 0

global score

score = 0

chemistryQ = [

[

"COD is a measure of",

"Organic substances in water",

"Oxides of S,P and N in air",

"Inorganic pollutants in water",

"Salinity of water"

],

[

"Percent of water precent in oceans",

"2.04",

"6.2",

"94.8",

"97.3"

],

[

"The most abundant hydrocarbon polluatnt",

"Ethane",

"Methane",

"Propane",

"Butane"

],

[

"The volume of two moles of oxygen at STP",

"22.4L",

"11.2L",

"40L",

"44.8L"

],

[

"PH of an aqueous solution of NH4Cl is",

"7",

">7",

"<7",

"1"

],

[

"When a solid melts,enthalpy",

"Increase",

"Decrease",

"No change",

"None"

],

[

"Structure of -BH2 group is",

"Linear",

"Planar",

"Tetrahedral",

"Octahedral"

],

[

"Among the following,BP is maximum for",

"Isobutane",

"n-butane",

"Propane",

"Ethane"

],

[

"The bonds not present in diborane are",

"B-H",

"B-H-B",

"B-B",

"H-B-H"

]

]

answer = [

"Organic substances in water",

"97.3",

"Methane",

"44.8L",

">7",

"Increase",

"Planar",

"n-butane",

"B-B"

]

li = ['',0,1,2,3,4,5,6,7,8]

x = random.choice(li[1:])

ques = Label(chemistry\_frame,text =chemistryQ[x][0],font="calibri 12",bg="white")

ques.place(relx=0.5,rely=0.2,anchor=CENTER)

var = StringVar()

a = Radiobutton(chemistry\_frame,text=chemistryQ[x][1],font="calibri 10",value=chemistryQ[x][1],variable = var,bg="white")

a.place(relx=0.5,rely=0.42,anchor=CENTER)

b = Radiobutton(chemistry\_frame,text=chemistryQ[x][2],font="calibri 10",value=chemistryQ[x][2],variable = var,bg="white")

b.place(relx=0.5,rely=0.52,anchor=CENTER)

c = Radiobutton(chemistry\_frame,text=chemistryQ[x][3],font="calibri 10",value=chemistryQ[x][3],variable = var,bg="white")

c.place(relx=0.5,rely=0.62,anchor=CENTER)

d = Radiobutton(chemistry\_frame,text=chemistryQ[x][4],font="calibri 10",value=chemistryQ[x][4],variable = var,bg="white")

d.place(relx=0.5,rely=0.72,anchor=CENTER)

li.remove(x)

timer = Label(h)

timer.place(relx=0.8,rely=0.82,anchor=CENTER)

def display():

def showMark(mark):

sh = Tk()

show\_canvas = Canvas(sh,width=720,height=440,bg="#101357")

show\_canvas.pack()

show\_frame = Frame(show\_canvas,bg="white")

show\_frame.place(relwidth=0.8,relheight=0.8,relx=0.1,rely=0.1)

st = "Your score is "+str(mark)

mlabel = Label(show\_canvas,text=st,fg="black")

mlabel.place(relx=0.5,rely=0.2,anchor=CENTER)

def exit():

sh.destroy()

if(mark==0 or mark==1 or mark==2 or mark==3):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,font="arial 10",fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="Basic concepts of chemistry by Pegasus"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="General Chemistry:Principles and modern applications"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="O.P Tandon"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

elif(mark==4 or mark==5 or mark==6):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="Handbook of chemistry by arihant experts"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="Atkins physical chemistry by peter atkins"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="Inorganic chemistry by Cooper H.langfor"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

elif(mark==7 or mark==8 or mark==9):

st1="Based on your score,the books recommended are:"

klabel=Label(show\_canvas,text=st1,fg="black")

klabel.place(relx=0.5,rely=0.3,anchor=CENTER)

st2="The elements by Nick Man"

klabel=Label(show\_canvas,text=st2,fg="black")

klabel.place(relx=0.5,rely=0.4,anchor=CENTER)

st3="Stuff matters by Mark miodowink"

klabel=Label(show\_canvas,text=st3,fg="black")

klabel.place(relx=0.5,rely=0.5,anchor=CENTER)

st4="Advanced organic chemistry by Richard J.Sunberg"

klabel=Label(show\_canvas,text=st4,fg="black")

klabel.place(relx=0.5,rely=0.6,anchor=CENTER)

st5="Thankyou for taking the Goal Getters test!!"

klabel=Label(show\_canvas,text=st5,fg="black")

klabel.place(relx=0.5,rely=0.7,anchor=CENTER)

exit = Button(show\_frame,command=exit,text="exit")

exit.place(relx=0.5,rely=0.9,anchor=CENTER)

if len(li) == 1:

h.destroy()

showMark(score)

if len(li) == 2:

nextQuestion.configure(text='End',command=calc)

if li:

x = random.choice(li[1:])

ques.configure(text =chemistryQ[x][0])

a.configure(text=chemistryQ[x][1],value=chemistryQ[x][1])

b.configure(text=chemistryQ[x][2],value=chemistryQ[x][2])

c.configure(text=chemistryQ[x][3],value=chemistryQ[x][3])

d.configure(text=chemistryQ[x][4],value=chemistryQ[x][4])

li.remove(x)

print(li)

y = countDown()

if y == -1:

display()

def calc():

global score

if (var.get() in answer):

score+=1

display()

submit = Button(chemistry\_frame,command=calc,text="Submit")

submit.place(relx=0.5,rely=0.82,anchor=CENTER)

nextQuestion = Button(chemistry\_frame,command=display,text="Next")

nextQuestion.place(relx=0.87,rely=0.82,anchor=CENTER)

y = countDown()

if y == -1:

display()

h.mainloop()

def start():

global root

root = Tk()

canvas = Canvas(root,width = 720,height = 440)

canvas.grid(column = 0 , row = 1)

img = PhotoImage(file="back.png")

canvas.create\_image(50,10,image=img,anchor=NW)

button = Button(root, text='Start',command = signUpPage)

button.configure(width = 102,height=2, activebackground = "#33B5E5", bg ='green', relief = RAISED)

button.grid(column = 0 , row = 2)

root.mainloop()

if \_name=='main\_':

start()

**Git hub link:**

[SaduBhavana/miniproject (github.com)](https://github.com/SaduBhavana/miniproject)

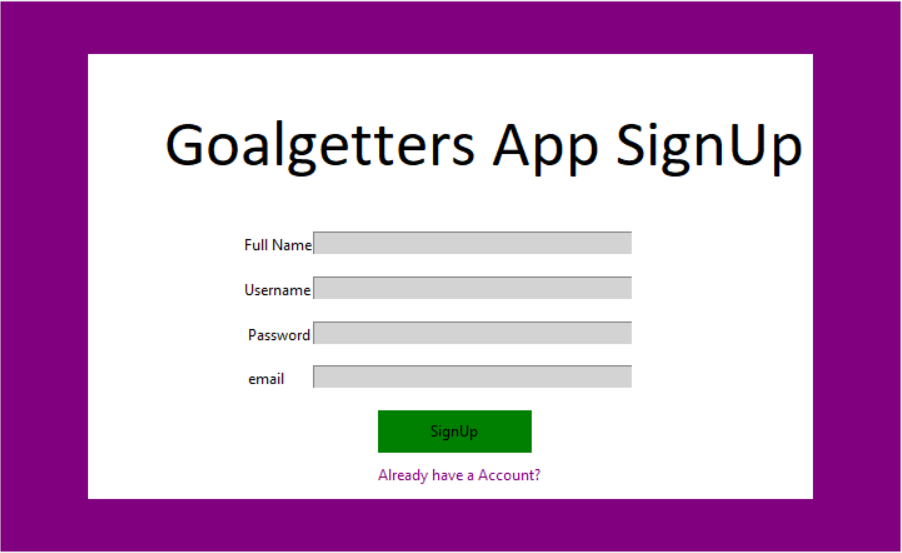
https://github.com/nithya2704/goalgetters.git

[Files · main · IT B Team18 / Goal Getters · GitLab (vce.ac.in)](https://gitlab.vce.ac.in/team18-b-web/goal-getters/-/tree/main)

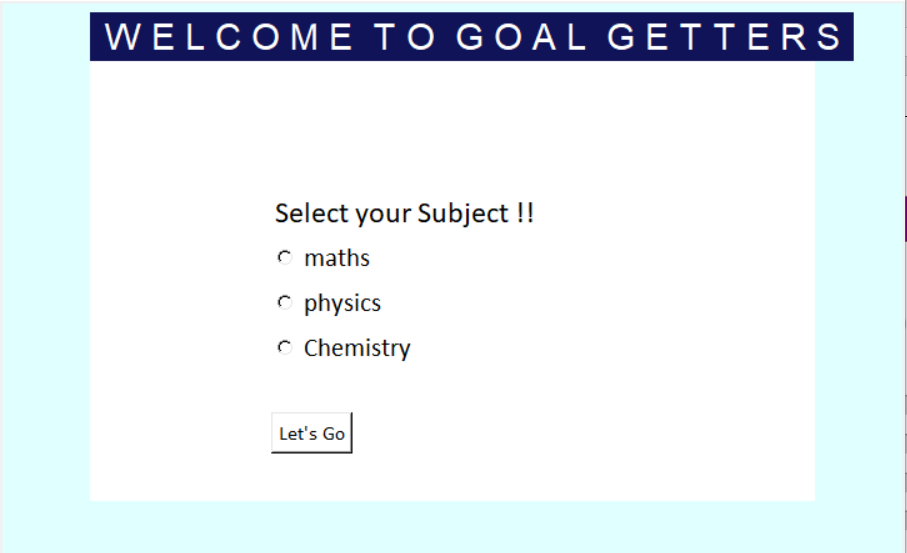
**Screen shots of the application testcases:**

****

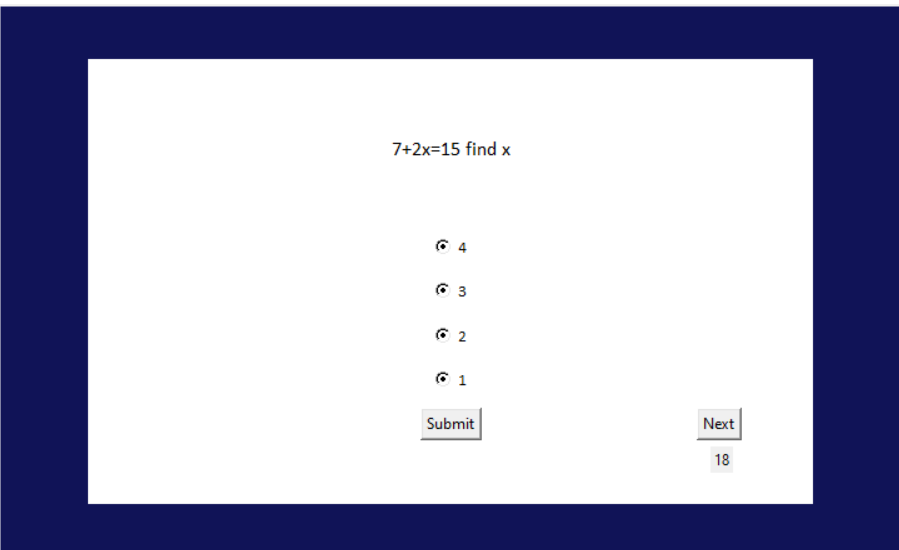
**Click on start to login**

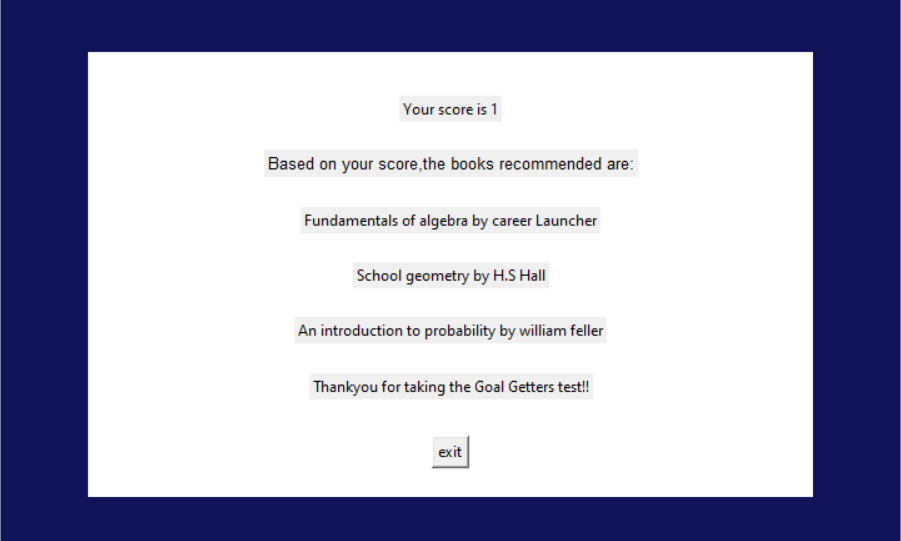
****

**If your new user register yourself or else click on already have a account**

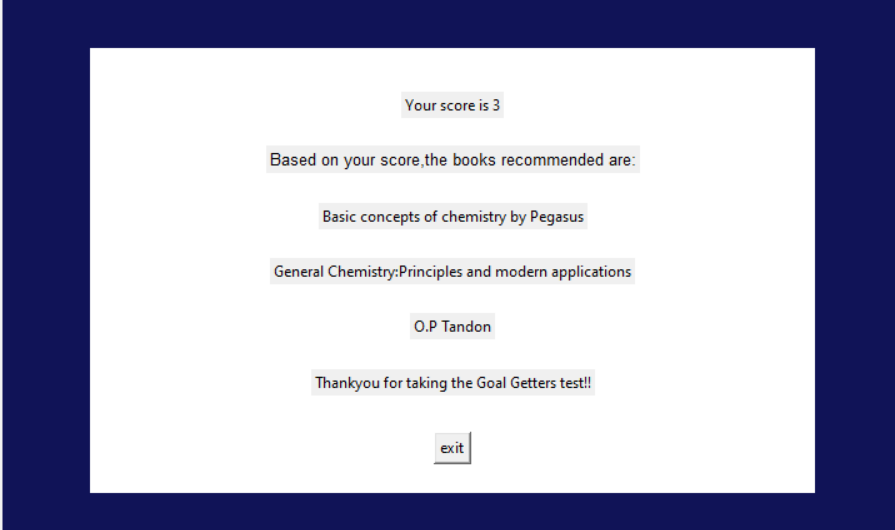
****

**Select your subject and click on let’s go**

****

****

**For chemistry**

****

**Click on exit to come out the page**

**ADDITIONAL KNOWLEDGE ACQUIRED**

* Python GUI programming using the module tkinter.
* Using the excel sheet for storing the data using the module tkinter.
* Use case diagrams.
* Activity diagram.

**CONCLUSION AND FUTURE WORK –**

We know that most of the members are confused to choose what book they want In a library so here we will provide books based on their iq

In future we can this particular by mentioning each and every topic

. by this may know his/her iq levels

**REFERENCES –**

Basic Python

1. Course covered during 2nd semester by Dr. Ramesh Vassapanavara Sir.
2. Ppts and handouts provided by the sir.
3. Python Programming - Using Problem Solving Approach First Edition by Reema Thareja.

Tkinter tutorials

1.<https://www.youtube.com/watch?v=YXPyB4XeYLA>

2. <https://www.youtube.com/watch?v=VMP1oQOxfM0&t=271s>

3. <https://www.youtube.com/watch?v=NkAwxoQkdOA>

Openpyxl tutorials

1. <https://www.youtube.com/watch?v=nsKNPHJ9iPc&t=1940s>
2. <https://www.youtube.com/watch?v=7YS6YDQKFh0&t=4s>

Use cases and activity diagram

1. Hand-outs provided by DRL PRASANNA ma’am.
2. <https://www.youtube.com/watch?v=zid-MVo7M-E>
3. <https://www.youtube.com/watch?v=knM8BGY9yVI&t=161s>