

JAYAPRIYA VIDYALAYA
SENIOR SECONDARY SCHOOL
[Affiliated to CBSE, New Delhi ,Affiliation Code-1930685]

ALL INDIA SENIOR SCHOOL CERTIFICATE EXAMINATION [AISSCE] – 2023-24



COMPUTER SCIENCE (083) – PROJECT REPORT ON

Time Caretaker

Submitted By

TYPE OF THE PROJECT : INDIVIDUAL			
SL.NO	ROLL NO.	NAME OF THE STUDENT	CLASS &SECTION
1		Chiranjeev.S	XII-CS-MATH

Under the Guidance of

Mr.S.BASKARAN M.Sc., B.Ed., PGDCA

Department of Computer Science

Teacher In-Charge

Principal

DECLARATION

I declare that the work presented in this project titled **”Time Caretaker “**, submitted to **Mr.S.Baskaran, PGT-Computer Science , Jayapriya Vidyalaya Sr. Sec. School , Virudhachalam** for the award of the **All India Senior School Certificate Examination [AISSCE] – 2023-24.**

I have not plagiarized or submitted the same work for the award of any other examination.

January, 2024

**Place : Jayapriya Vidyalaya Sr.Sec.School,
Virudhachalam**

**Chiranjeev S
XII-CS-MATH**

CERTIFICATE

This is to certify that S.Chiranjeev_of Class XII CS-MATH has prepared the record on the Project entitled

" Time Caretaker "

The report is the result of **his** efforts & endeavours.

The report is found worthy of acceptance as final project report for the subject **Computer Science (083)** of *Class XII – CS-MATH*.

he has prepared the report under my guidance.

Mr.S.BASKARAN M.Sc.,B.Ed.,PGDCA
Department of Computer Science
Jayapriya Vidyalaya Sr. Sec. School
Virudhachalam

**JAYAPRIYA VIDYALAYA
SENIOR SECONDARY SCHOOL**

[Affiliated to CBSE, New Delhi ,Affiliation Code-1930685]

ALL INDIA SENIOR SCHOOL CERTIFICATE EXAMINATION [AISSCE] – 2022-23



CERTIFICATE

The project report entitled

" Time Caretaker ",

Submitted by

Chiranjeev.S of Class XII CS-MATH for the

All India Senior School Certificate Examination [AISSCE] –2023-24 of Computer Science at Jayapriya Vidyalaya Senior Secondary School, Virudhachalam has been Examined.

Signature of Internal Examiner

Signature of External Examiner

ACKNOWLEDGEMENT

I would like to register my sense of gratitude to our **Principal Mrs.Sri Bala G Raj** for his immense encouragement that made this project successful.

I would like to express a deep sense of thanks to my my project guide **Mr.S.Baskaran PGT in Computer Science** for guiding me immensely through the course of the project. He always evinced keen interest in my work.

His constructive advice & constant motivation have been responsible for the successful completion of this project.

I wish to thank **my parents** for their motivation & support. and my special thank to my classmates for their timely help & support for the completion of this project.

We owe our sincere gratitude towards **Jayapriya Vidyalaya Senior Secondary School, Virudhachalam.**

Our heartfelt thanks to **CBSE** for given this opportunity to do this project.

Last but not the least, I would like to thank all those who had helped directly or indirectly towards the completion of this project.

S.Chiranjeev
XII-CS-MATH

CONTENT

SL.NO.	PARTICULARS	PAGE NO.
1.	Introduction about the project	
2.	System, Software and Hardware requirements	
3.	Introduction about Python	
4.	Introduction about MySQL	
5.	System Development Life Cycle (SDLC)	
6.	Phases of system development life cycle	
7.	Project Design	
8.	Python code	
9.	Database screenshots	
10.	Output screenshots	
11.	Advantages of project	
12.	Further development areas	
13.	Bibliography	

Introduction about the project : Time Caretaker

The project Time Caretaker is a boon for every individual who are suffering to manage their time , who are forgetting their schedule ,who wants to listen to jokes and who wants to take lecture notes. This project can majorly do 5 tasks. They are as follows

1)Set Alarm -The alarm will get the required details and set the alarm.

2)Set Reminder -The reminder will get the re required details and set the reminder.

3)Crack a Joke - The jokes will be randomly selected from the database and displayed.

4)record lectures – Students can record their lectures and as well as they can store it as text notes.

5)set schedule – People can set the schedule in database and check it from the database just by entering the date in the correct format .

System Requirements of the Project

Recommended System Requirements

Processors: Intel® Core™ i3 processor 4300M at 2.60 GHz.

Disk space: 2 to 4 GB.

Operating systems: Windows® 10, / MAC OS / UBUNTU.

Python Versions: 3.X.X or Higher.

MySQL- 8.0.X

Minimum System Requirements

Processors: Intel Atom® processor or Intel® Core™ i3 processor.

Disk space: 1 GB.

Operating systems: Windows 7 or later / MAC OS / UBUNTU.

Python Versions: 3.8.X.

MySQL- 8.0.X

Pre requisites before installing MySQL Connector Python

You need root or administrator privileges to perform the installation process.

Python must be installed on your machine.

Note: – MySQL Connector Python requires python to be in the system's PATH.

Installation fails if it doesn't find Python.

On Windows, If Python doesn't exist in the system's PATH, please manually add the directory containing python.exe yourself.

Introduction about Python

Python is a high level, interpreted and general purpose dynamic programming language that focuses on code readability. It has fewer steps when compared to Java and C. It was founded in 1991 by developer Guido Van Rossum. It is used in many organizations as it supports multiple programming paradigms. It also performs automatic memory management.

Advantages :

- 1) Presence of third-party modules
- 2) Extensive support libraries(NumPy for numerical calculations, Pandas for data analytics etc.)
- 3) Open source and community development
- 4) Easy to learn
- 5) User-friendly data structures
- 6) High-level language
- 7) Dynamically typed language(No need to mention data type based on value assigned, it takes data type)
- 8) Object-oriented language
- 9) Portable and Interactive
- 10) Portable across Operating systems

Applications :

- 1) GUI based desktop applications(Games, Scientific Applications)
- 2) Web frameworks and applications
- 3) Enterprise and Business applications
- 4) Operating Systems
- 5) Language Development
- 6) Prototyping

Introduction about MySQL

MySQL is a relational database management system (RDBMS) based on the SQL (Structured Query Language) queries. It is one of the most popular languages for accessing and managing the records in the table. MySQL is open-source and free software under the GNU license. Oracle Company supports it.

The following are the most important features of MySQL:

Relational Database Management System (RDBMS)

MySQL is a relational database management system. This database language is based on the SQL queries to access and manage the records of the table.

Easy to use

MySQL is easy to use. We have to get only the basic knowledge of SQL. We can build and interact with MySQL by using only a few simple SQL statements.

It is secure

MySQL consists of a solid data security layer that protects sensitive data from intruders. Also, passwords are encrypted in MySQL.

Client/ Server Architecture

MySQL follows the working of a client/server architecture. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server; that is, they can query data, save changes, etc.

Free to download

MySQL is free to use so that we can download it from MySQL official website without any cost.

It is scalable

MySQL supports multi-threading that makes it easily scalable. It can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4 GB. However, we can increase this number to a theoretical limit of 8 TB of data.

Speed

MySQL is considered one of the very fast database languages, backed by a large number of the benchmark test.

High Flexibility

MySQL supports a large number of embedded applications, which makes MySQL very flexible.

Compatible on many operating systems

MySQL is compatible to run on many operating systems, like Novell NetWare, Windows* Linux*, many varieties of UNIX* (such as Sun* Solaris*, AIX, and DEC* UNIX), OS/2, FreeBSD*, and others. MySQL also provides a facility that the clients can run on the same computer as the server or on another computer (communication via a local network or the Internet).

Allows roll-back

MySQL allows transactions to be rolled back, commit, and crash recovery.

Memory efficiency

Its efficiency is high because it has a very low memory leakage problem.

High Performance

MySQL is faster, more reliable, and cheaper because of its unique storage engine architecture. It provides very high-performance results in comparison to other databases without losing an essential functionality of the software. It has fast loading utilities because of the different cache memory.

High Productivity

MySQL uses Triggers, Stored procedures, and views that allow the developer to give higher productivity.

Platform Independent

It can download, install, and execute on most of the available operating systems.

Partitioning

This feature improves the performance and provides fast management of the large database.

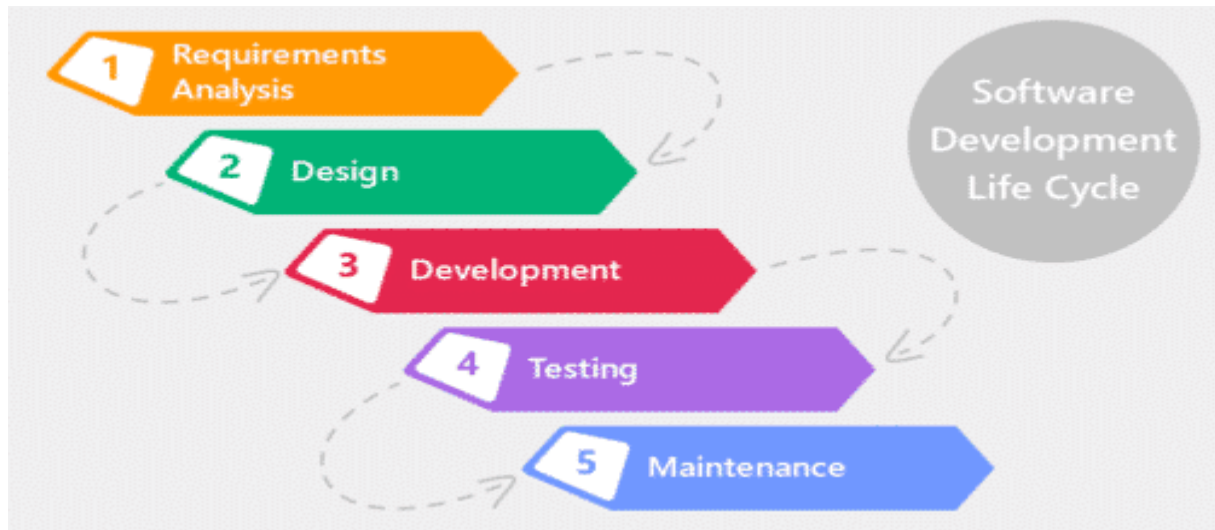
GUI Support

MySQL provides a unified visual database graphical user interface tool named "**MySQL Workbench**" to work with database architects, developers, and Database Administrators. [MySQL Workbench](#) provides SQL development, data modeling, data migration, and comprehensive administration tools for server configuration, user administration, backup, and many more. MySQL has a fully GUI supports from MySQL Server version 5.6 and higher.

Dual Password Support

MySQL version 8.0 provides support for dual passwords: one is the current password, and another is a secondary password, which allows us to transition to the new password.

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.

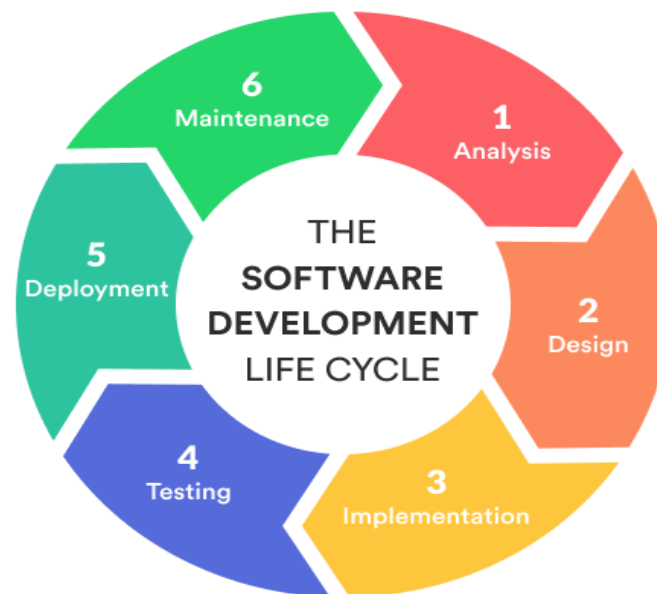
SYSTEM CONCEPT DEVELOPMENT PHASE

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.

PICTORIAL REPRESENTATION OF SDLC:



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.

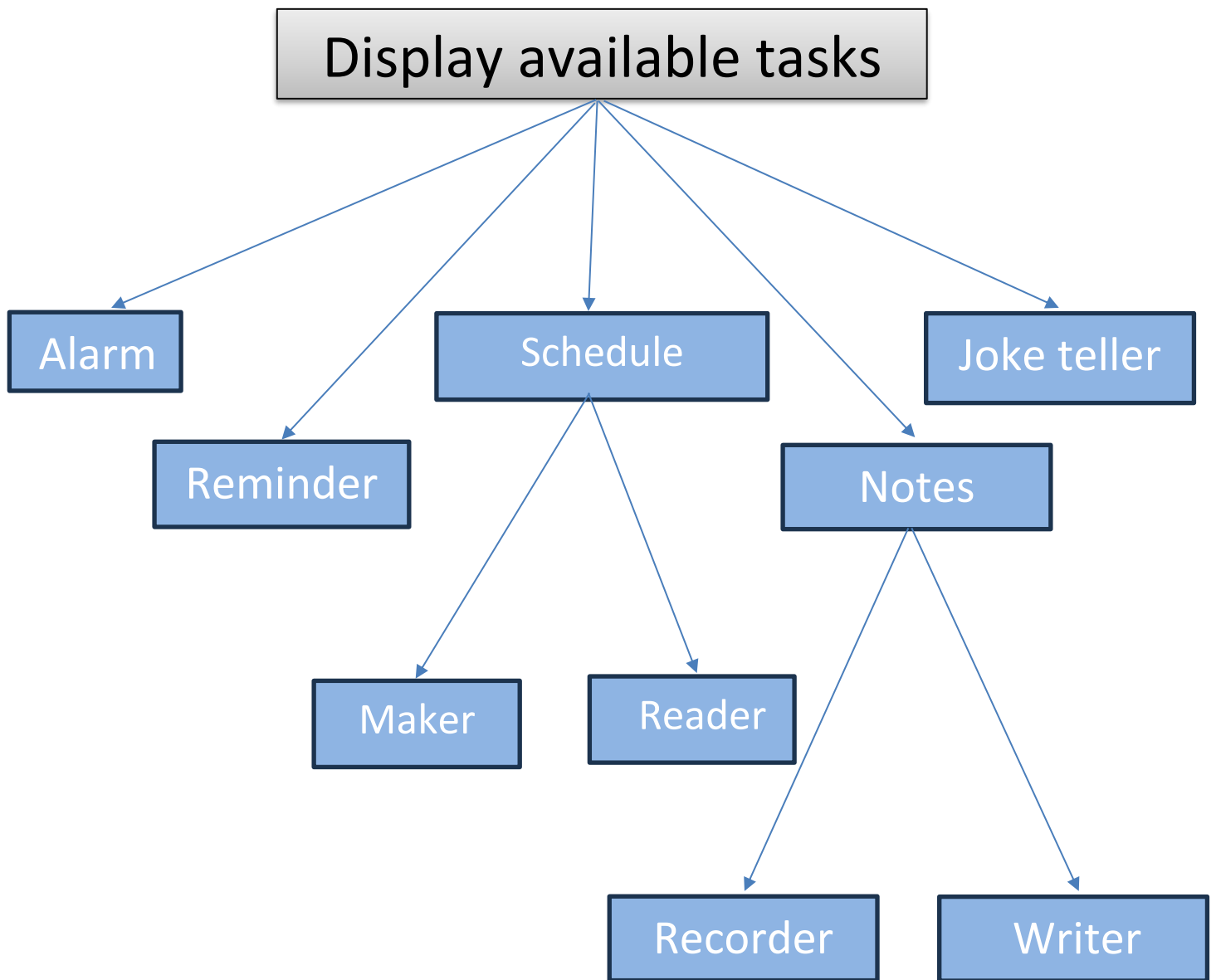
OPERATIONS AND MAINTENANCE PHASE

The system operation is ongoing. 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.

PROJECT DESIGN



PYTHON CODE

```
import speech_recognition as sr
```

```
import pyttsx3 as s
```

```
import playsound
```

```
import datetime as dt
```

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

```
import random as r
```

**#THE FOLLOWING ARE THE USER
DEFINED FUNCTIONS**

```
import pralarm_f as z1
```

```
import prreminder_f as z2
```

```
import prdt_f as z3
```

```
import prjoke_f as z4
```

```
import prnotestaker as z5
```

```
import prschedulerreminder_f as z6
```

```
import Tprdt_f as tz3
```

```
ma = s.init()
```

```
mC = ma.getProperty('rate')
```

```
ma.setProperty('rate', 120)
```

```
mvoices = ma.getProperty('voices')
```

```
ma.setProperty('voices', mvoices[1].id)
```

```
try:
```

```
    with sr.Microphone() as source:
```

```
        rx = sr.Recognizer()
```

```
        print('you can give the input to this  
project by voice as well as text .which way  
do you prefer to select the task:')
```

```
        ma.say('you can give the input to this  
project by voice as well as text .')
```

```
        ma.say('which way do you prefer to  
select the task')
```



```
ma.runAndWait()
```

```
inp= rx.listen(source)
```

```
if 'voice' in rx.recognize_google(inp):
```

```
    print('you can do the following  
tasks with this program just by saying the  
name of that task')
```

```
    ma.say("you can do the following  
tasks with this program just by saying the  
name of that task")
```

```
    print('THEY ARE')
```

```
    ma.say('they are')
```

```
    print('1) alarm')
```

```
    ma.say( '1) alarm')
```

```
    print('2) reminder')
```

```
    ma.say('2) reminder')
```

```
    print('3) joke')
```

```
ma.say('3) joke')
print('4) date and time')
ma.say('4) date and time')
print('5) record lectures')
ma.say('5) record lectures')
print('6) set schedule in preyer')
ma.say('6) set schedule in preyer')
print('now you can say the name of
the task:')

ma.say('now you can say the name
of the task:')

ma.runAndWait()

audio = rx.recognize_google(audio,
key=my_google_speech_recognition_api_
key, language='en-US', show_all=True)

try:
```

```
        if 'alarm' in
rx.recognize_google(audio):
            z1.alarm()

        elif 'reminder' in
rx.recognize_google(audio):
            z2.reminder()

        elif 'time' in
rx.recognize_google(audio) or 'date' in
rx.recognize_google(audio):
            z3.dt()

        elif 'joke' in
rx.recognize_google(audio):
            z4.joke()

        elif 'lecture' in
rx.recognize_google(audio) or 'record' in
rx.recognize_google(audio) :
            z5.notes_taker()
```

```
elif 'schedule' in  
rx.recognize_google(audio):  
    z6.prschedulere minder_f()
```

```
except sr.UnknownValueError:  
    print("Could not understand  
audio")  
except sr.RequestError as e:  
    print("Could not request results;  
{0}".format(e))
```

```
elif 'text' in rx.recognize_google(inp):  
    try:  
        with sr.Microphone() as source:  
            rx = sr.Recognizer()
```

print('you can do the
following tasks with this program just by
entering the name of that task')

ma.say("you can do the
following tasks with this program just by
entering the name of that task")

print('THEY ARE')

ma.say('they are')

print('1) alarm')

ma.say('1) alarm')

print('2) reminder')

ma.say('2) reminder')

print('3) joke')

ma.say('3) joke')

print('4) date and time')

ma.say('4) date and time')

```
print('5) record lectures')
ma.say('5) record lectures')
print('6) set schedule in
preyer')

ma.say('6) set schedule in
preyer')

ma.say('now you can enter the
name of the task:')

ma.runAndWait()

x1=input('enter the name of
the task:')

if 'alarm'==X1:
    z1.alarm()

elif 'reminder'==X1:
    z2.reminder()

elif 'time'==X1 or 'date'==x1:
```

```

        tz3.dt()

        elif 'joke' in
rx.recognize_google(audio):
            z4.joke()

        elif 'lecture' in
rx.recognize_google(audio) or 'record' in
rx.recognize_google(audio) :
            z5.notes_taker()

        elif 'schedule' in
rx.recognize_google(audio):
            z6.prschedulereReminder_f()

except sr.UnknownValueError:
    print("Could not understand
audio")

except sr.RequestError as e:

```

```
        print("Could not request results;  
{0}".format(e))
```

```
except sr.UnknownValueError:
```

```
    print("Could not understand audio")
```

```
except sr.RequestError as e:
```

```
    print("Could not request results;  
{0}".format(e))
```

“***THANK YOU***”

USER DEFINED FUNCTIONS

ALARM

```
import playsound
```

```
import datetime as dt
```

```
import pyttsx3 as s
```


#perfectly works

na = s.init()

nC = na.getProperty('rate')

na.setProperty('rate', 120)

nvoices = na.getProperty('voices')

na.setProperty('voices', nvoices[1].id)

def alarm():

 print('enter the following requirments to
set the alarm')

 na.say('enter the following requirments
to set the alarm')

 na.runAndWait()

 na.say('enter the hour:')

```
na.runAndWait()

c1 = int(input('enter the hour:'))

na.say('enter the  minute:')

na.runAndWait()

c2 = int(input('enter the  minute:'))

na.say('enter AM/PM:')

na.runAndWait()

c3 = input('enter AM/PM:')

if c3.lower() == 'pm':

    c1 = c1 + 12

while True:

    if c1 == dt.datetime.now().hour and

c2 == dt.datetime.now().minute:

    print('alarm playing...')

    playsound.playsound('alarmsong.mp3')

    break
```

REMINDER

```
import pytsx3 as s
```

```
import datetime as dt
```

```
def reminder():
```

```
    ma = s.init()
```

```
    mC = ma.getProperty('rate')
```

```
    ma.setProperty('rate', 120)
```

```
    mvoices = ma.getProperty('voices')
```

```
    ma.setProperty('voices', mvoices[1].id)
```

```
print('enter the following requirments to  
set the reminder')
```

```
ma.say('enter the following requirments  
to set the reminder')
```

```
ma.runAndWait()
```

```
ma.say('enter the hour:')
```

```
ma.runAndWait()
```

```
c1 = int(input('enter the hour:'))
```

```
ma.say('enter the minute:')
```

```
ma.runAndWait()
```

```
c2 = int(input('enter the minute:'))
```

```
ma.say('enter AM/PM:')
```

```
ma.runAndWait()
```

```
c3 = input('enter AM/PM:')  
ma.say('enter the statement to remind  
you:')  
ma.runAndWait()  
  
c4 = input('enter the statement to remind  
you')  
ma.say('enter how many times the  
statement should repeat:')  
ma.runAndWait()  
  
c5 = int(input('enter how many times the  
statement should repeat'))  
if c3 == 'pm':  
    c1=c1+12
```

while True :

 if c1 == dt.datetime.now().hour and
 c2 == dt.datetime.now().minute:

 a = s.init()

 C = a.getProperty('rate')

 a.setProperty('rate', 120)

 voices = a.getProperty('voices')

 a.setProperty('voices', voices[1].id)

while c5>0:

 c5=c5-1

 print(c4)

 b = c4

 a.say(b)

 a.runAndWait()

break

NOTES RECORDER & NOTES TAKER

```
import pytsx3 as s
```

```
import pyaudio as py
```

```
import wave
```

```
import speech_recognition as sr
```

```
ra = sr.Recognizer()
```

```
ma = s.init()
```

```
mC = ma.getProperty('rate')
```

```
ma.setProperty('rate', 120)
```

```
mvoices = ma.getProperty('voices')
```

```
ma.setProperty('voices', mvoices[1].id)
```

```
# working perfectly
```

```
def notes_taker():
```

```
    fpb=3200
```

```
    f=py.paInt16
```

```
    c=1
```

```
    r=16000
```

```
    p=py.PyAudio()
```

```
    st=p.open(format=f,channels=c,rate=r,input  
t=True,frames_per_buffer=fpb)
```

```
        ma.say('enter the time duration of the  
lecture in seconds')
```

```
        ma.runAndWait()
```



```
x=int(input('enter the time duration of  
the lecture in sec'))
```

```
ma.say('enter the audio file name')
```

```
ma.runAndWait()
```

```
y=input('enter the audio file name')
```

```
ma.say('start the lecture')
```

```
ma.runAndWait()
```

```
print('start the lecture')
```

```
sec=x
```

```
fr=[ ]
```

```
for i in range(0,int(r/fpb*sec)):
```

```
    da=st.read(fpb)
```

```
    fr.append(da)
```

```
st.stop_stream()
```

```
st.close()
```

```
p.terminate()
```

```
o=wave.open(y+'.wav','wb')
o.setnchannels(c)
o.setsampwidth(p.get_sample_size(f))
o.setframerate(r)
o.writeframes(b"".join(fr))
o.close()
print('the lecture is stored successfully')
ma.say('the lecture is stored
successfully')
ma.runAndWait()
ma.say('enter the text file name:')
ma.runAndWait()
f=input('enter the text file name:')
tp=sr.AudioFile(y+'.wav')
with tp as source:
```

```
    audio = ra.record(source)

try:

    s = ra.recognize_google(audio)

    print("Text: "+s)


    with open(f+'.txt','w') as tf:

        tf.write(s)


    print('the notes is stored successfully
')


except Exception as e:

    print("Exception: "+str(e))
```

SCHEDULE REMINDER

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

```
import pytsx3 as s
```

```
ma = s.init()
```

```
mC = ma.getProperty('rate')
```

```
ma.setProperty('rate', 120)
```

```
mvoices = ma.getProperty('voices')
```

```
ma.setProperty('voices', mvoices[1].id)
```

```
def swrite():
```

```
a=my.connect(host='localhost',user='root',  
password='Sch00l@',database='chiranjeev  
pro')
```

```
c=a.cursor()
ma.say('enter the schedule:')
ma.runAndWait()
s=input('enter the schedule:')
ma.say('enter the date :')
ma.runAndWait()
d=input('enter the date :')
q=" insert into sreminder
values('{ }','{ }')".format(d,s)
c.execute(q)
a.commit()
a.close()

def sread():

a=my.connect(host='localhost',user='root',
```

```
password='Sch00l@',database='chiranjeev  
pro')
```

```
c=a.cursor()
```

```
ma.say('enter the date at which you want  
to know the schedule in yyyy-mm-dd  
format')
```

```
ma.runAndWait()
```

```
x=input('enter the date at which you  
want to know the schedule in yyyy-mm-dd  
format:')
```

```
q="select schedule from sreminder  
where rdate='{ }'".format(x)
```

```
c.execute(q)
```

```
k=c.fetchone()
```

```
print(k[0])
```

```
a.close()
```

```
def ax():  
    x=input('do you want to set the schedule  
/ see the schedule')  
    if 'see' in x:  
        sread()  
    elif 'set' in x:  
        swrite()
```

DATE AND TIME WITH AUDIO AS INPUT

```
import datetime as dt  
import pytsx3 as s  
import speech_recognition as sr  
import pyaudio  
def dtf():  
    try:
```

with sr.Microphone() as source:

```
d = dt.datetime.now()
```

```
v = str(d)
```

```
a = s.init()
```

```
C = a.getProperty('rate')
```

```
a.setProperty('rate', 120)
```

```
voices = a.getProperty('voices')
```

```
a.setProperty('voices', voices[1].id)
```

```
rx = sr.Recognizer()
```

```
g = 'enter what you  
want(date/time/date and time)'
```

```
print(g)
```

```
a.say(g)
```



```
a.runAndWait()
```

```
ip=rx.listen(source)
```

```
if rx.recognize_google(ip, language  
='en-IN', show_all=True)== 'time':
```

```
    a1 = v[11:19:1]
```

```
    a_ = v[11:13]
```

```
    a2 = int(a_)
```

```
    if a2 > 12:
```

```
        a3 = a2 % 12
```

```
        a4 = v[14:16]
```

```
        a5 = v[18:20]
```

```
    else:
```

```
        a3 = a2
```

```
        a4 = v[14:16]
```

```
        a5 = v[18:20]
    print(a3, 'hours', a4, 'minuts', a5,
'seconds')

    b = a3, 'hours', a4, 'minuts', a5,
'seconds'

    a.say(b)

    a.runAndWait()
```

```
        elif 'date' in
rx.recognize_google(ip):

            b_ = dt.date.today()

            b1 = v[0:4]

            b2 = v[5:7]

            b3 = v[8:10]
```

```
print('today the date is :', b_)  
b =( 'today the date is :', b1, b2,  
b3)
```

```
a.say(b)
```

```
a.runAndWait()
```

```
elif 'date and time' in  
rx.recognize_google(ip) or 'both' in  
rx.recognize_google(ip):
```

```
b_ = dt.date.today()
```

```
b1 = v[0:4]
```

```
b2 = v[5:7]
```

```
b3 = v[8:10]
```

```
a1 = v[11:19:1]
```

```
a_ = v[11:13]
```

```
a2 = int(a_)
```

```
if a2 > 12:
```

```
    a3 = a2 % 12
```

```
    a4 = v[14:16]
```

```
    a5 = v[18:20]
```

```
else:
```

```
    a3 = a2
```

```
    a4 = v[14:16]
```

```
    a5 = v[18:20]
```

```
    print('today the date is :', b_, 'and  
now the time is:', a3, 'hours', a4, 'minuts',  
a5, 'seconds')
```

```
    b = 'today the date is :', b1, b2,  
b3, 'and now the time is:', a3, 'hours', a4,  
'minuts', a5, 'seconds'
```

```
        a.say(b)

        a.runAndWait()

except sr.UnknownValueError:

    print("Could not understand audio")

except sr.RequestError as e:

    print("Could not request results;
{0}".format(e))
```

DATE AND TIME WITH TEXT AS INPUT

```
import datetime as dt

import pyttsx3 as s

def dt():

    ip = input('enter what you
want(date/time/date and time)')
```

```
d = dt.datetime.now()
```

```
v = str(d)
```

```
a = s.init()
```

```
C = a.getProperty('rate')
```

```
a.setProperty('rate', 120)
```

```
voices = a.getProperty('voices')
```

```
a.setProperty('voices', voices[1].id)
```

```
if ip == 'time':
```

```
    a1 = v[11:19:1]
```

```
    a_ = v[11:13]
```

```
    a2 = int(a_)
```

```
    if a2 > 12:
```

```
        a3 = a2 % 12
```

```
        a4 = v[14:16]
```

```
a5 = v[18:20]
```

```
else:
```

```
    a3 = a2
```

```
    a4 = v[14:16]
```

```
    a5 = v[18:20]
```

```
    print(a3, 'hours', a4, 'minuts', a5,  
'seconds')
```

```
    b = a3, 'hours', a4, 'minuts', a5,  
'seconds'
```

```
    a.say(b)
```

```
    a.runAndWait()
```

```
elif ip == 'date':
```

```
    b_ = dt.date.today()
```

```
b1 = v[0:4]
```

```
b2 = v[5:7]
```

```
b3 = v[8:10]
```

```
print('today the date is :', b_)
```

```
b = 'today the date is :', b1, b2, b3
```

```
a.say(b)
```

```
a.runAndWait()
```

```
elif ip == 'date and time':
```

```
    b_ = dt.date.today()
```

```
    b1 = v[0:4]
```

```
    b2 = v[5:7]
```

```
    b3 = v[8:10]
```

```
    a1 = v[11:19:1]
```



```
a_ = v[11:13]
```

```
a2 = int(a_)
```

```
if a2 > 12:
```

```
    a3 = a2 % 12
```

```
    a4 = v[14:16]
```

```
    a5 = v[18:20]
```

```
else:
```

```
    a3 = a2
```

```
    a4 = v[14:16]
```

```
    a5 = v[18:20]
```

```
    print('today the date is :', b_, 'and now  
the time is:', a3, 'hours', a4, 'minuts', a5,  
'seconds')
```

```
    b = 'today the date is :', b1, b2, b3,  
'and now the time is:', a3, 'hours', a4,  
'minuts', a5, 'seconds'
```

a.say(b)

a.runAndWait()

DATABASE SCREENSHOTS

slno	jokes
1	What do kids play when their mom is using the phone? Bored games
2	What do you call an ant who fights crime? A vigilANTE!
3	Why are snails slow? Because they're carrying a house on their back.
4	What's the smartest insect? A spelling bee!
5	What does a storm cloud wear under his raincoat? Thunderwear.
6	What is fast, loud and crunchy? A rocket chip
7	How does the ocean say hi? It waves!
8	What do you call a couple of chimpanzees sharing an Amazon account? PRIME-mates.
9	Why did the teddy bear say no to dessert? Because she was stuffed.
10	Why did the soccer player take so long to eat dinner? Because he thought he couldn't use his hands.
11	Name the kind of tree you can hold in your hand? A palm tree!
12	What do birds give out on Halloween? Tweets.
13	What has ears but cannot hear? A cornfield.
14	What's a cat's favorite dessert? A bowl full of mice-cream.
15	Where did the music teacher leave her keys? In the piano!
16	What did the left eye say to the right eye? Between us, something smells!
17	What do you call a guy who's really loud? Mike
18	Why do birds fly south in the winter? It's faster than walking!
19	What did the lava say to his girlfriend? ?I lava you!?
20	Why did the student eat his homework? Because the teacher told him it was a piece of cake.

+-----+-----+	
rdate	schedule
+-----+-----+	
2023-11-01	1)unit test -1(cs) ,2)revision Amines
2023-12-02	1)dance class , 2)swimming class
2023-11-03	1)maths exam revision, 2)physics revision
+-----+-----+	

OUTPUT SCREENSHOTS

you can give the input to this project by voice as well as text .which way do you prefer to select the task:

you can do the following tasks with this program just by entering the name of that task

THEY ARE

- 1) alarm
- 2) reminder
- 3) joke
- 4) date and time
- 5) record lectures
- 6) set schedule in preyer

enter the name of the task:alarm

enter the following requirments to set the alarm

enter the hour:4

enter the minute:8

enter AM/PM:pm

alarm music is playing

you can do the following tasks with this program just by entering the name of that task

THEY ARE

- 1) alarm
- 2) reminder
- 3) joke
- 4) date and time
- 5) record lectures
- 6) set schedule in prayer

enter the name of the task:reminder

enter the following requirments to set the reminder

enter the hour:4

enter the minute:15

enter AM/PM:pm

enter the statement to remind youwake up

enter how many times the statement shoud repeat8

wake up

wake up

wake up

wake up

wake up

wake up

wake up

wake up

>>>|

you can give the input to this project by voice as well as text .which way do you prefer to select the task:

you can do the following tasks with this program just by entering the name of that task

THEY ARE

1) alarm

2) reminder

3) joke

4) date and time

5) record lectures

6) set schedule in prayer

enter the name of the task:joke

Name the kind of tree you can hold in your hand? A palm tree!

>>>

you can give the input to this project by voice as well as text .which way do you prefer to select the task:

you can do the following tasks with this program just by entering the name of that task

THEY ARE

1) alarm

2) reminder

3) joke

4) date and time

5) record lectures

6) set schedule in preyer

enter the name of the task:date

enter what you want(date/time/date and time)date

today the date is : 2023-11-13

>>> |

you can give the input to this project by voice as well as text .which way do you prefer to select the task:

you can do the following tasks with this program just by entering the name of that task

THEY ARE

- 1) alarm
- 2) reminder
- 3) joke
- 4) date and time
- 5) record lectures
- 6) set schedule in preyer

enter the name of the task:lecture

enter the time duration of the lecture in sec15

enter the audio file name:audiolecture_1

start the lecture

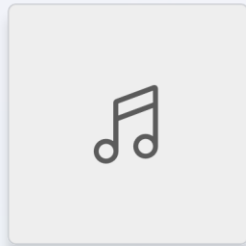
the lecture is stored successfully

enter the text file name:lecture_1notes

Text: today we are going to see about Python Python is a programming language with which we can do logical operations

the notes is stored successfully

>>>



00:00:00  00:00:15

audiolecture_1



|today we are going to see about Python Python is a programming language with which we can do logical operations

you can give the input to this project by voice as well as text .which way do you prefer to select the task:

you can do the following tasks with this program just by entering the name of that task

THEY ARE

1) alarm

2) reminder

3) joke

4) date and time

5) record lectures

6) set schedule in preyer

enter the name of the task:schedule

do you want to set the schedule / see the schedule:yes

enter the schedule:1)english exam,2)CS revision

enter the date :2023-11-6

.>>>

rdate	schedule
2023-11-01	1)unit test -1(cs) ,2)revision Amine
2023-12-02	1)dance class , 2)swimming class
2023-11-03	1)maths exam revision, 2)physics revision
2023-11-06	1)english exam,2)CS revision

you can give the input to this project by voice as well as text .which way do you prefer to select the task:

you can do the following tasks with this program just by entering the name of that task

THEY ARE

1) alarm

2) reminder

3) joke

4) date and time

5) record lectures

6) set schedule in preyer

enter the name of the task:schedule

do you want to set the schedule / see the schedule:see

enter the date at which you want to know the schedule in yyyy-mm-dd formate:2023-11-06

1)english exam,2)CS revision

>>>|

ADVANTAGES OF PROJECT

>>> The main moto of this project is to make people productive.

>>>Students can just listen the class
this project will take the notes

>>>People can see their schedule on
the respective dates .

>>>The reminder will remind the task
at the specified time so people will not
forget to do the task.

>>>The alarm will be useful for the
people to take a short nap

FURTHER DEVELOPMENT AREAS

>>>>The alarm in this could have access to the voice input.

>>>>The number of music choices could have been more in alarm.

>>>>GUI can be given to this project.

>>>>The alarm and reminder can be made to run in background.

BIBLIOGRAPHY

1) www.geeksforgeeks.org

2) www.python.org

3) www.freecodecamp.org

4) www.simplilearn.com

5) www.parade.com

6) Computer science with python-
Sumita Arora