***1]Requirement analysis:-***

1.What are the problems we are trying to solve?

For the given problem statement of creating a "To Do List"application,we are trying to

make our daily schedule which includes an alarm notifying that particular task which

should happen by that individual at that specified time .Also if on a given day a task

has more priority than the other one which earlier had more priority than it,then the

alarm should first rise for that task by shifting of that task in earlier time slot.

We have to activate a new and modified "To Do List" daywise.Our application should be

able to do all these tasks efficiently.We need to use a system timer to manage alarm

mechanism for each task in the"To Do List".

**2]FUNCTIONAL SPECIFICATION:-**

FUNCTION 1 : **ADD ACTIVITY** contains **four subfunction**

1.1 **CREATE**  - This function will help user to create new activity like

Waking up,taking medicines,attending meeting etc for any upcoming or

Present day and for any time**.It will provide the user with the option to set the priority**

**PRIORITY** : User can set the priority for any activity .

1.2 **READ** - This function will help user to read any activity created by him of any day.

1.3  **UPDATE** - This function will help user to update any activity created by him of any day.he can make changes to any activity eg can change time of event to occur,can change the complete event.**It will provide the user with the option to CHANGE the priority**

**PRIORITY :**User can change the priority of any activity or can delete the priority for any activity.

1.4 **DELETE** - This function will help user to delete any activity created by him of any day.user will be provided option to delete the entire activity of day or any particular time.

FUNCTION 2 : **RAISE ALERT**

This function will give the user the feature to set reminder for any activity of any day at any given time by user.

FUNCTION 3 : **REPORT GENERATION**

This function will help the user to get the detailed report of all the activity in the date range given by the user.

3]External interface specification:-

**\*USER INTERFACE :-**

* Front-end software:Python
* Back-end software:SQLite

When the user submits the data ,it will be stored into a database using SQLite.Then the program will be running in the background until it is time for a remainder to pop-up.When this occurs,a pop-up will appear on the screen informing the user that it is time for them to do whatever they had planned. GUI will be created using tkinter framework.

Requirement 1:-Use SQLite.

Requirement 2:-Have GUI pop-ups for scheduling tasks

|  |  |
| --- | --- |
| **Time** | **Task** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Fig.-Data stored in above format.

**Time**  **Task**

Fig.-Pop menu will appear at scheduled time.

**3]HARDWARE SPECIFICATION**

**Ubuntu 18.04 LTS SPECIFICATION:**

2 GHz dual core processor or better

2 GB system memory

25 GB of free hard drive space

Either a DVD drive or a USB port for the installer media

Internet access is helpful

**PyCharm Windows SPECIFICATIONS**:

Microsoft Windows 10/8/7/Vista/2003/XP (incl.64-bit)

1 GB RAM minimum

2 GB RAM recommended

1024x768 minimum screen resolution

Python 2.4 or higher, Jython, PyPy or IronPython

**Database Backend:-**

SQL server database backend is required.

CRUD operation can be done on data stored in backend

* C-Create data in to-do list
* R-Read the complete to-do list
* U-Update the data with status as “not started”,”in progress”,or “complete”
* D-Delete the data from the list.

Users’ data should be protected. Data security should be maintained.

4.Performance constraint

What is the speed , availability,response time of the software is necessary to be known.

Speed of an application depends on time and space complexity required by the code of

that program.For the "To Do List" application,if priority of tasks gets easily managed

by the software the time complexity would be comparatively less.Performance of this

application can also be improved if working of alarm mechanism is properly handled.

Maintenace of code of this application also contributes to its good performance.

5..Memory

Memory allocation and management for this "To Do List" application involves mostly

allocating memory in form of priority queue by considering specific time slot for

that respective task taking place at particular point of time on that day.Chances

of memory getting wasted unnecesarily are less as mostly tasks in the daily schedule

and their time slots which consume memory are getting memory allocated dynamically.The

alarm notifying the execution of a particular task consumes negligible amount of memory.

If memory leakage takes place the OS may crash.So memory management will be monitored

at every instant and unnecessary memory will be released using memory allocation operators

so that another user can utilise that space whenever required.