

Tasker : the to-do list

GROUP 7:

TEAM MEMBERS:

- 1.Sohan Kale (lead)
- 2.Anivedh Auradkar
- 3.Aditya Bakare
- 4.Purva Gaikwad

1.Introduction :

The document lays out a plan regarding the development of the 'Tasker' application.

A full fledged layout of the various aspects involved in the development of the app, the different functional requirements, specifications, varied constraints and how those constraints can be mitigated, from the "team's" perspective.

2. Overview:

"Getting bombarded with a load of tasks" , is what most people cannot handle. The basic underlying problem is prioritization of the tasks that have to be done. Once that factor is sorted out, things get a bit easier to handle.

Hence we aim to help better prioritization of daily tasks through our application.

3. Customers:

Everyone. Anyone can use this application ranging from a child to an old-age person.

4.Requirements Analysis:

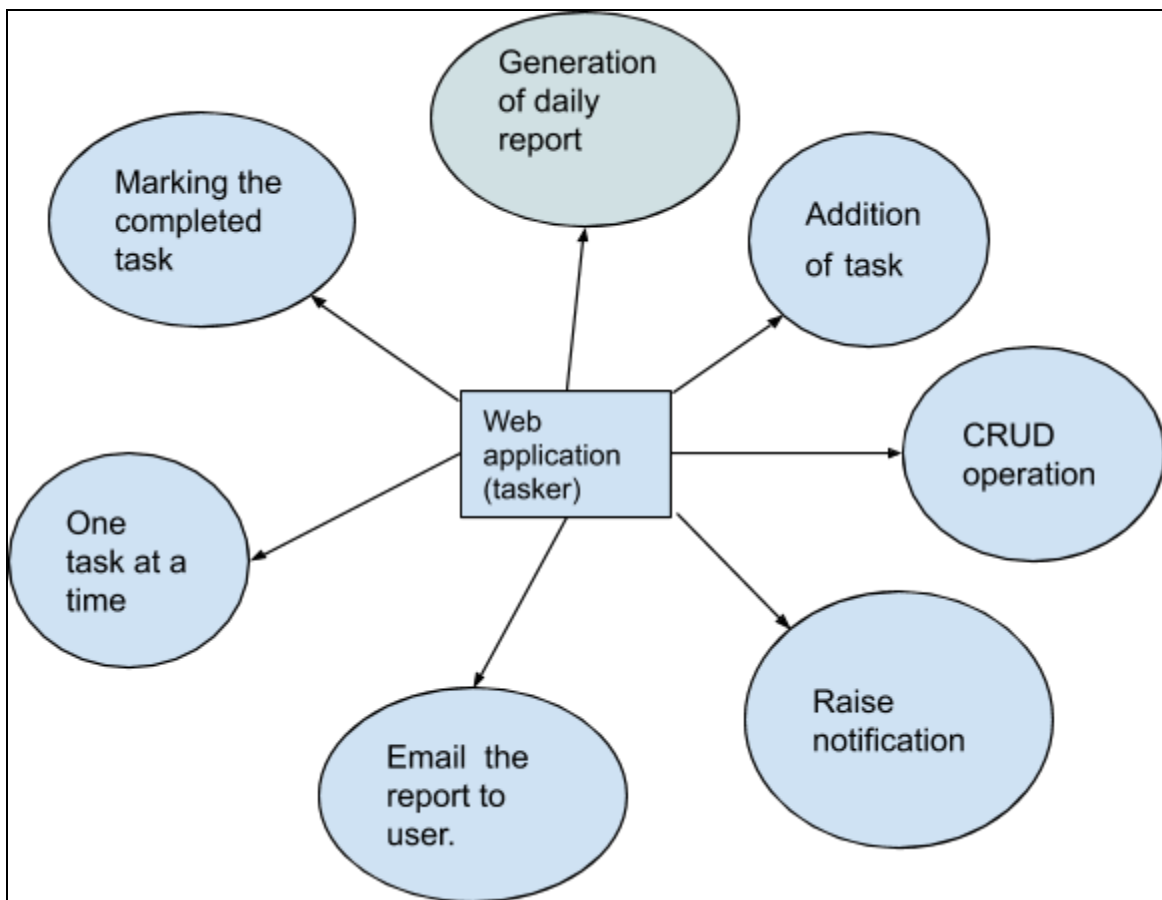
What problem are we trying to solve?

The difficulties in managing various tasks and the confusion created in managing tasks and their respective priorities.

● Functional Specifications:

- Creation,reading,updation,deletion of task.
- Raising alert for specific task at the given time.
- Handling conflicts in tasks.
- Mark completed tasks.
- Creation of a daily report(completed and uncompleted tasks).
- Simple but interactive UI for the task manager application
- *Addition of a text to speech feature to the tasker*
- *Addition of a pop and countdown timer to the notification part.*

P.S: Points mentioned in italics are features that can be inculcated later.



● External Interface Specification:

- User Interface:

Software Requirements:

1. Operating System : Microsoft Windows 10
2. Browser : Google Chrome(v70.0 or later)
Microsoft Edge
3. Python : v3.3 or above
4. Web Server : Flask

Hardware Requirements:

1. Processor : Qualcomm Snapdragon 410
and above, MediaTek MT6738 or
Newer, Intel 2nd gen core i3
or newer

- Communication Interfaces: Flask v1.1
- Hardware Interfaces: A pc, laptop with windows 10 (UI)
- Database Backend: SQLAlchemy database

5. Technical Specification: Following would be the languages to develop web application within the stipulated time period:

Front-end development: HTML, CSS, jinja2.

Back-end development: Python, SQLAlchemy

Framework: FLASK

6. Deliverables: The following things will be delivered during the course of development:

- Feature specification
- Product design
- Source code

7. Scheduling and Estimates:

Milestone	Description	Release Date	Release iteration
M1	Application View and Design (Front End development)	Nov 20 2019	R1
M2	Database for the application (Back End)	Nov 20 2019	R2
M3	Notification Addition	Nov 21 2019	R3
M4	Integrating front-end and back end	Nov 21 2019	R4
M5	Testing for initial release	Nov 21 2019	R5
M6	Issue tracker.	Nov 22 2019	R6
M7	Final Release	Nov 22 2019	R7

8. Risk Identification:

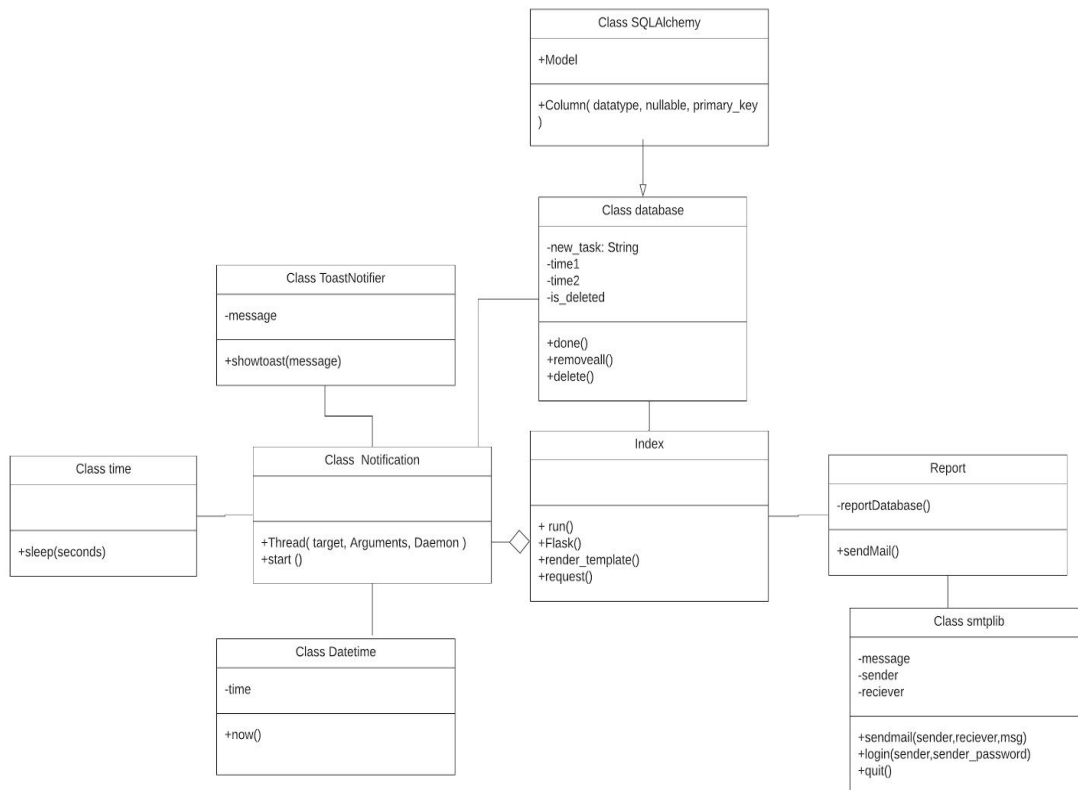
Many to-do lists are available with more features so less crowd will be attracted towards this idea initially.

9. Risk Mitigation:

Its a web based application not consuming anyspace on your device and no need of downloading an app for the same. Use it whenever you need.

Design

Class Diagram:



SEQUENCE DIAGRAM

