Conception Phase – Habit App

Habit tracker application is a tool for tracking habits and by tracking develop good habits and eliminate bad ones.

User can create and delete habits, track, modify, deselect from tracking and check off habits. As well user can load list of all habits, list of currently tracked habits, list of habits with same periodicity. User can obtain a longest streak from all the habits and selected habits.

User has to periodically check off habits. There 2 periods: daily habits and weekly habits. User can check off habit within the whole day or 7 days. For daily habits the cut off time is 24:00 hours on the next day when habit tracking was started. For weekly habits the cut off time is 24:00 hours on the 7th day when habit tracking was started. If user didn’t check off habit within required period, the habit’s streak will be reset to 0 (broken).

An application will start on execution of menu.py file.

On an application startup user will get on the screen the list of all currently tracked habits and below the list of habits list of menu options (See menu diagram).

Interaction with user is done via Command Line Interface. User select a desired option in the application menu by typing a number of option in CLI. Then, a required command will be performed and current menu level options will be reprinted. On lower levels of menu will be option “Back” which will allow to return to the upper level of menu / main menu.

There are 6 modules in Habit Tracker App, they are listed below:

* habit.py – module contains class Habit, which is a subclass of a Base class. Module imports class Base from module base and attributes Column, Integer, UnicodeText, Boolean, Date, PickleType from module sqlalchemy.
* habits.py – module contains class Habits. Module imports class Habit from module Habits.
* menu.py – module contains command line interface and functions which interact with habits.py module. Imports sys, datetime, sqlalchemy, imports habit class from Habit module, import habits class from HabitsList, imports predefined\_habits from predefined habits, import all the functions from analytics module, imports session from base.
* analytics.py – module contains functions to analyze habits.
* base.py – module contains database setup and class Base imported from sqlalchemy. Module imports create\_engine from sqlalchemy, declarative\_base from sqlalchemy.ext.declarative, sessionmaker from sqlalchemy.orm.
* predefined\_habits.py – module contains a list with predefined habits stored as dictionaries.

In application is planned 3 classes (Base, Habit and HabitsList).

Data in the project will be stored in Sqlite3 database in combination with Object Relation Mapper SQLAlchemy. Sqlite3 database db.db file will be initialized on first execution of menu.py file and then habits from module predefined\_habits.py will be recorded in database. In order to prevent overwriting of predefined habits on each startup menu module will check if database file exist. If database file exists, then menu module will not write in database habits form predefined\_habits module.