**Personal manager**

**Description**

This project is aimed to practice different data structures and algorithms and to improve GUI design skills. The program should serve as a personal manager allowing to create appointments, store them, display them and manipulate them. It should have functioning and simple GUI.

The appointment fields should include: date, time, place and short description of the appointment.

Also, from within the program you should be able to save, load, edit, add and remove appointments.

**Functional requirements:**

* Program must display user the list of his appointments
* Program should allow user to filter the displayed appointments
* Program must allow user to find an appointment by id
* Program must allow user add appointments
* Program must allow user to remove appointments
* Program must allow user to edit the existing appointments
* Program must allow user to save or load appointment files

**Non-functional requirements:**

* Program must display the list of appointments as a table.
* Program must display the appointment description when the user double-clicks the table entry.
* Program must ask user if he wants to save the changes when he is exiting the program
* Program must save the appointments to a file once every minute while the program is running
* Program must show clear and readable error messages to the user in a form of dialog windows.

**Personal manager v2.0**

**Description**

Manager should store all of the appointments in the database and instead of loading them from file, load them from the database. Will have multiple users and users will be presented with the program login screen. The program will check the user database and try to match login and password to the ones in database and on successful attempt it will load them. User will be able to create an account on the program website. User will be able to download the program from the website. And user will be able to see his 5 nearest appointments from the website. Program also should save data to the file every 20 seconds automatically and then upload the file on the database upon exiting. (Should have ‘load from file’ button).

Step 1

Implement the login system without the database. Login data will be stored in a file as “Login – Password\nLogin2 - Password2…”. And user appointments would be stored in the file “login.list” (a separate file for every user e.g ”alex.list” and “john.list”).

Create login window.

Make password check.

Make it load appropriate user file.

Make the program save current list to a file every 20 seconds.

Step 2

Create a database table containing sample the users and sample tables containing sample user appointments.

Create a webpage that will allow:

* to create new user accounts
* to change username/password for the current user
* to see the 5 nearest appointments for the current user
* to download the program

Step 3

Connect the program to the database and make it send and receive data to and from the server instead of loading them from the file.

* Make the program receive data from the database pack into the AppointmentList and display it
* Make the program send data to the database. Process the current AppointmentList and save it to the database in an appropriate table
* Make the login system work with the database instead of the file.