Reservation system project plan Y2 2022

Vladimir Kurazhev

Student number: 587345

Automation and Information technology

Since 2016

25.02.2022

1 General description and difficulty level

From project description:

Create a reservation system for an imaginary company. The system contains information on scheduling, customers, and the company's resources.

The company has a limited amount of resources. These resources can be, for example, maintenance areas and mechanicians in a car repair shop, chairs in a barbershop, or hotel rooms. The information on the resources has been added into the program.

The program saves information about reservations into a calendar and marks the chosen resources as reserved for that time. When making a reservation causes a conflict (the resource has already been reserved), the reservation doesn't go through. The program adds relevant information on the customer to each reservation.

The program can be used to check the reservations on a given point of time and to print out information on reservations within a time interval determined by the user.

Difficulty levels:

Easy

- A text-based user interface which can be used to check if a date/time is reserved and to make a reservation for a chosen time.
- The program prevents concurrent reservations and, if needed, informs the user about the resource already being reserved at the given point of time.
- The program marks the chosen resource as reserved for the given points of time.
- The program can be used to save information on customers (e.g. name, address, phone number and e-mail) and to connect it to a reservation.
- The reservations don't disappear when the program is closed, meaning they can be saved into and loaded from separate files.
- In the easy version all reservations can be of equal length.

Medium

- Requirements of the easy version.
- A graphical user interface which includes a calendar view that shows reservation times of the resources and can be used to make new reservations.

- The program saves history data on customers/services provided (e.g. basic information and maintenance history of a car), which can be searched and viewed.
- The reservations can be of different lengths.
- Unittests for at least part of the program

Hard

- Requirements of the medium version.
- Make the user interface as usable, clear and informative as possible.
- Additional services can be attached to reservations (e.g. do a routine inspection for a car brought in for a tyre change, have a bottle of champagne in the hotel room waiting for the customer).
- The program uses the length of the time slot and the reserved services to calculate a price, and prints it out.
- Add other useful features, for example statistics or automatic reminders.

First of all will be completed all easy and medium difficulty level's requirements. If there is enough time left all the hard level requirements will be done as well.

2 Use case description and draft of the user interface

The program communicate with the user with a graphical user interface which includes a calendar view that shows reservation times of the resources and can be used to make new reservations. Will be implemented using click-buttons to reserve time. Ideally bypassing the need to enter any text other than the required personal data.

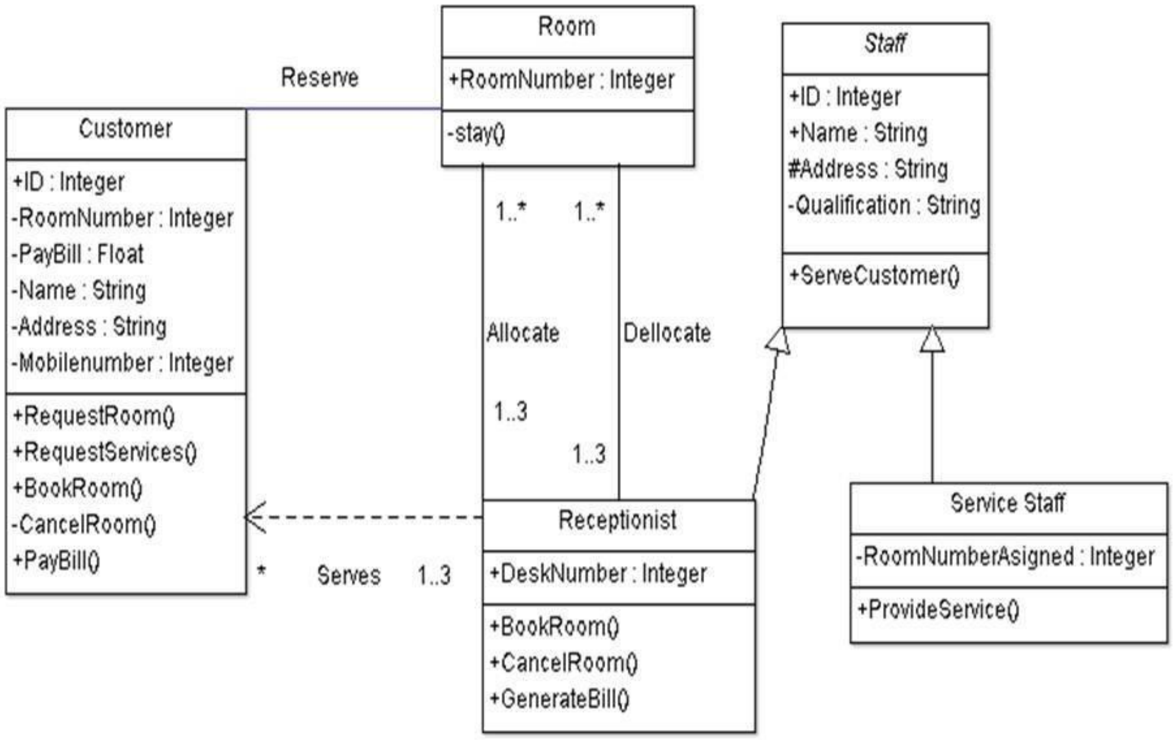
Normal situation

Enter the program -> checking for free(suit) schedule window -> select one -> type personal or required data -> the time reserved

3 Program's structure plan

The structure is divided into two main parts "Main menu" for introducing main functions as reserve, schedule/calendar(maybe some reservation history), delete reservation, other functions depending on hard level phase and "Buttons" which will includes buttons class for creating clickable blocks in menu. Buttons for example: "Select_Time_Button" will be getting data from selected window from calendar like (reserved or not, time and date, selected function), "Reserve_Button" to make reservation and make it reserved in calendar. All the functional buttons will be subclasses of the main class button.

Here is an example of hotel reservation:



Class Diaram for Hotel Management System

4 Data structures

Most of the data will be string, bool and date formats, and will be recorded in dictionaries and lists, since the data will only change its content.

5 Files and file formats

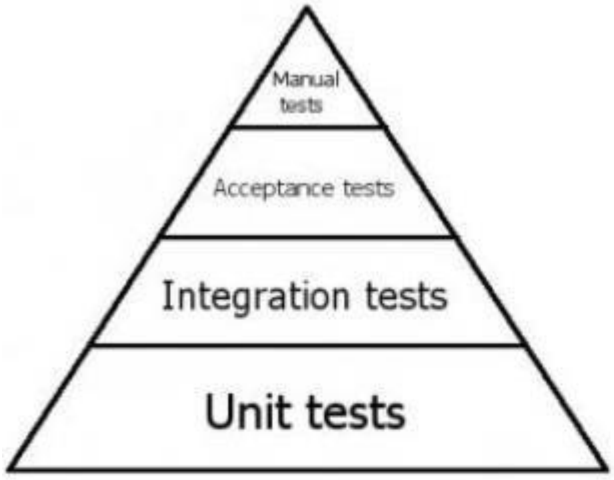
Txt format files will be used for the first time, next might be used some different if it will be more suit for the project. The data will be presented as a full schedule as a calendar with dates and reserved time or not by bool also name of client which reserved it.

6 Algorithms

For creating menu, buttons and calendar some constants will be used, boards first schedule will be made for one week and then multiplied into months. Bool data will be most important because it will pass the value reserved(F) or free(T) for all reservations. Mostly it will be checking for reservation and reserving new ones using buttons for it.

7 Testing plan

Testing will take place by using test methods for errors, testing individual modules of the application source code(unit tests) and manual tests.



8 Libraries and other tools

PYQT 5 , maybe some libraries for calendar and for graphical interfaces. QT Designer isprohibited in project instructions.

9 Schedule

Most of the work must be done before the end of the next month. everything will be planned out by weeks, in the first week it will go to the menu and creating a raw version of the main functions. Then next week finishing some interface and maybe rest of time will go for finishing all requirements, testing blocks and etc.

10 Literature references and links

Course materials and all included links, https://www.python.org, https://tutsplus.com/,https://www.sololearn.com/,https://www.techbeamers.com/,https://hackr.io/tutorials/learn-python, https://realpython.com/, Head First Python, 2nd Edition(book).