



FINAL PROJECT

This project consists of the development of a computer system to register physical activity of users of a gym. The system will have three roles: client, monitor, and administrator.

The system has to meet the following requirements:

- ☐ **Authentication:** The program will ask the user to write his/her role, choosing among: client, monitor, and administrator. After selecting the role, (s)he has to specify the username and password to check if (s)he is already in the system.
- ☐ **Data of users:** each user of the system will have the following data:
 - ☐ **Data common to all roles:**
 - ☐ **Role:** user, monitor, administrator.
 - ☐ **Given name and surname.**
 - ☐ **username.**
 - ☐ **password.**
 - ☐ **Client's data:** this information is specific to the role client
 - ☐ **Registered activity:** the user can enroll in **1 activity**.
 - ☐ **Weekly average time spent on the gym:** how many hours the client usually spends on the gym every day.
 - ☐ **votes:** numerical score assigned to the client (**between 1 -10 points**).
 - ☐ **Level:** level of experience of the client depending on their score (**beginner, advance, or expert**).
 - ☐ **Monitors data:**
 - ☐ **Activities:** each monitor can lead **1 activity**.
 - ☐ **votes:** votes received by clients.
 - ☐ **Level:** level of experience of the monitor depending on their score (**beginner, advance, or expert**).
 - ☐ **Administrator data:** this role only have the data common to all roles.
- ☐ **Client's options:**
 - ☐ **Activity registration:** the client can enroll to an activity, when the user enter to this option, (s)he will see the list of activities and the monitor

of the activity. Then the user will type the name of the activity to enroll to the activity.

- ☐ **Search monitors:** if the client enter to this option, (s)he can see the list of monitors arranged alphabetically or arranged by the number of votes. For each monitor the system will show the name, the activity that (s)he is monitoring, and the number of votes.
- ☐ **Vote a monitor:** the client can vote for one monitor of the gym specifying the number of votes (i.e. between 1 and 12) and the monitor's name. After voting, the system will update the number of **received votes** of the monitor.

☐ **Monitor's options:**

- ☐ **Consult registered clients:** the monitor can see the list of registered clients arranged alphabetically or by its score.
- ☐ **Vote a client:** the monitor can vote one client of the gym specifying the number of votes (i.e. between 1 and 12) and the client's name. After voting, the system will update the number of **received votes** of the client.

☐ **Administrator's options:**

- ☐ **Add a client:** add one client to the client's list.
- ☐ **Add a monitor:** add a monitor to the monitor's list.
- ☐ **Add activities:** add an activity to the activity's list.
- ☐ **Enroll a monitor to an activity:** each monitor can have only one activity.
- ☐ **Update client's level:** when this option is chosen, the system will update the clients level depending on the clients votes (0-20 votes beginner, 21-50 votes advanced, 51 or more expert).
- ☐ **Update monitor's level:** when this option is chosen, the system will update the monitors level depending on the monitors votes (0-20 votes beginner, 21-50 votes advanced, 51 or more expert).



Part One - Lab 6 (Due date March 16th 23:55)

During this session, the group will create an Eclipse project and define the methods needed for representing all the system's requirements. It will not be necessary to implement them. The group has to specify the return value, the name, the input parameters, and a comment with a short description. At the end of the class, the group will submit the project created during the laboratory session through AulaGlobal.



Part two - Lab 7 and Lab 8 (Due date March 31th 9:00)

During the subsequent two sessions, the group will work on the Eclipse project created during Part One, implementing the methods. The methods that display the arranged lists must use each of the algorithms shown during the theory sessions (Bubble Sort, Selection Sort, Insertion Sort, Heap Sort, Merge Sort, Quick Sort). At the end of the second session, each group will submit the current version of Eclipse project through AulaGlobal.



Part Three - Final Exercise (Due date April 7th 23:55)

During the last two sessions, the methods defined during the first two parts will be reviewed, taking into account the new concepts introduced in theory sessions. The group will also initialize the necessary data to have users (clients, monitors, and administrators) and activities with monitors and clients already subscribed to the system. Finally, each group will submit the final version of the project through AulaGlobal.