**Assumptions**

1. Since I was not instructed about how a user is recognized by the system, I just implemented a simple "login" to get the user's username and email. Of course, this is not a proper login, it is missing a password and it is in no way secure.
2. The identification of each user is based on a randomized id saved in his local storage. The direct assumption from this method is that only one user can register from each browser. This method was used to give the user a way to enter later and to see which classes he is registered to or on their waiting list.
3. The implementation is for one week only and the assumption is that the classes remain the same each week. Therefore, the relevant rows of a class should be deleted from the registered and waiting tables in the database after the class occurred. This is so people can register and enter the waiting list for the following week. Of course, because of that, users can register and enter the waiting list only a week before the class itself.
4. When a user enters a waiting list, he is given a timestamp and according to this timestamp, the first user that joined the list will enter the class in case of cancellation. Since the cancel function is a bonus that I didn't have time to implement, the announcement doesn't occur. For future implementation, the users' email was saved.

**Please note:** Over the course of the week, I tried to get an answer about what I need to do with the fourth point (notifying the customers on the waiting list should a vacancy open), since I didn't do the bonus of allowing cancellation. Sadly, I couldn't get an answer so according to logic, this part was not implemented (because it can't happen if no one cancels his booking).