Contract CO2: signUp

Operation: SignUp()
Cross Reference: Sequence Diagram: Client Sign Up
Preconditions: None
Postconditions:

- A client entry is registered to the client database
- Redirect user to the menu

Contract CO2: approveClient

Operation: approveClient(email:string)
Cross Reference: Sequence Diagram: Administrator approve client
Preconditions:

- User logged in as administrator
- There are unapproved clients in the database

Postconditions:

- A client entry is updated in the database
- Redirect user to the menu

Contract CO2: Login

Operation: authenticateUser(email: string, password: string
Cross Reference: Sequence Diagram: Client Login , Sequence Diagram: Expert Login
Preconditions:

- User must be in the appropriate database
- Credentials must match those in the database
- If user is Client
    - Administrator has approved the client user

Postconditions:

- An appropriate User instance (client, expert or administrator) is created
- Attributes email and password are set
- Calls the method showMenu()

Contract CO2: viewObjects

Operation: viewObjects()
Cross Reference: Sequence Diagram: Client view objects
Preconditions:

- User must be logged in
- There are objects in the database

Postconditions:

- List of available objects in the institution (or auction) is shown

Contract CO2: AddObject

Operation: registerObject(name: string, description: string, owned: boolean)
Cross Reference: Sequence Diagram: Administrator adds object
Preconditions:

- Administrator must be logged in
- Object is not already in the database

Postconditions:

- Object is added to the database
- Success message is shown to the administrator

Contract CO2: AddAuction

Operation: insertAuction(name: string,auctionHouse: string, location: string, date: string, startTime: string, endTime: string, specialty: string)
Cross Reference: None
Preconditions:

- Administrator must be logged in
- There isn’t another auction during the time of the auction being added

Postconditions:

- Auction is successfully added to the database
- Success message is shown to the administrator

Contract CO2: BookExpert

Operation: bookExpert(email: string)
Cross Reference: Sequence Diagram: Client booking expert
Preconditions:

- Client must be logged in
- Expert has not already been booked for that time by another client

Postconditions:

- Addition of booking is added to database
- Client is notified of success of booking

Contract CO2: addAvailability

Operation: addAvailability(date: string, time: string)
Cross Reference: Sequence Diagram: Expert adding availability
Preconditions:

- Expert must be logged in
- Availability is not already in the system

Postconditions:

- Addition of availability is added to database
- Expert is notified of success of the addition