

# Fencing Club Management System (Desktop Application)

DOCUMENTATION



## CONTENT

<u>GOAL</u>	<u>3</u>
<u>SCOPE</u>	<u>3</u>
<u>Possible Features</u>	<u>3</u>
<u>End Users</u>	<u>5</u>
<u>Integration of the End users with the project (user stories)</u>	<u>5</u>
<u>Areas covered with this project</u>	<u>6</u>
<u>PROJECT USERS, ACTORS, VENDORS, ACTUATORS</u>	<u>7</u>
<u>PROJECT PROPERTIES</u>	<u>7</u>
<u>Project Functionality</u>	<u>8</u>
<u>1. Login to the system</u>	<u>9</u>
<u>2. Menu for navigation.</u>	<u>9</u>
<u>3. Manage Members. Add New Member</u>	<u>11</u>
<u>4. Manage Members. Select by ID</u>	<u>12</u>
<u>5. Manage Members. Update Member by ID</u>	<u>12</u>
<u>6. Manage Members. Delete Member by ID</u>	<u>13</u>
<u>7. Manage Members. Filter</u>	<u>13</u>
<u>8. Manage Coaches</u>	<u>14</u>
<u>9. List of Members</u>	<u>14</u>
<u>10. Member Information</u>	<u>15</u>
<u>11. Camp registration</u>	<u>16</u>
<u>12. Inventory</u>	<u>16</u>
<u>14. Attendance</u>	<u>17</u>
<u>System Messages</u>	<u>18</u>
<u>REST API implemented</u>	<u>19</u>
<u>Test cases implemented using NUnit</u>	<u>21</u>
<u>Project Setup</u>	<u>22</u>
<u>Work Progress in Future</u>	<u>22</u>

# GOAL

The goal of the **Fencing Club Management System** Software is for use by club owners and his employees to organize all processes working and linked in one system. The goal is to simplify all processes and manage it by software, so each user could have more time to grow in his performance of doing his tasks. Things that now are kept in mind or have tracking on paper or some of them even have no tracking, will be submitted in one system. Also this software will give a report system to analyze situations and make decisions for improvement.

# SCOPE

## Possible Features

### **1. Club Member registration**

Club members will be able to register in the system. He will create own login and password. Also he will have access to modify/update his information later. User has to update his license in his profile.

### **2. Login encrypted system**

Each user will have access to his profile. Access is secured with an encrypted password.

### **3. Attendance management**

Club members will confirm his attendance after entering the system the same day when they have training.

### **4. Schedule management**

There will be a possibility to choose the exact group to see its schedule.

### **5. Individual appointment management**

Coaches will see their calendar and will be able to add individual training with club members.

## **6. Individual performance tracking**

A coach will have a possibility to evaluate his fencer regularly and to give notes with recommendations according to its training plan. All records will be kept for future progress records.

## **7. Training exercises planning**

There will be a possibility to add a training plan information for a year, month and a week

## **8. Calendar of activities (competition)**

There will be a Competition calendar available for each member and coaches. Information about each competition, as well as results (after completing it) will be displayed there.

## **9. E-mail *communication* management**

For a regular email newsletter it will be possible to choose a group of recipients. Also some letter template will be available to choose if you need to send it as a reminder (to pay, to update license, etc)

## **10. Camp registration**

Information about planned Camps will be presented according to its time schedule.

## **11. Inventory order**

There will be an order form, where each Club Member can choose the equipment for him, choose size and submit this order under his account.

## **12. Reports**

Some reports will be available for the owner (general information about club members quantity, quantity of visited days, average performance of fencers), reports for coaches (monthly performance and attendance ) , reports for accountant department that they could use this numbers in their accounting system (number of individual trainings hours for each coach, each member, camp participants, total equipment order), reports for members, to see in their profile cumulative information.

## End Users

There is a list of end-users and those who will receive notification through this management system software owner, coaches, club members, parents of club members, accountant department.

- **Owner** will use it for reporting, analyzing, planing, equipment pre order, work time control, competition calendar
- **Coaches** will use it for scheduling, attendance, keep tracking of performance of each club member, annual and weekly planning program for classes, to plan and analyze performance results.
- **Club members** will register their attendance, place orders for equipment, register for camps and confirm participation in competition, e-mail reminders and notifications about schedule changes.
- **Parents of young club members** and club members will receive regular email info letters, emails about attendance of their children, news about upcoming competitions and camps.
- Accountants/administrators will have access to reports about attendance, hours worked by coaches, and individual classes. All that for use in the contability.

## Integration of the End users with the project (user stories)

- As an owner, I need to see a whole picture of what is going with my business. Reports should be on a general level, and then go to details. Based on that I want to do my calculations and to plan my business growth.
- As a coach, I want to keep track of the performance of my fencers, their attendance and to keep comments and advice for them in future.
- As an accountant I want to have access to reports that show me numbers: worked hours, inventory orders, individual classes to use in my accountant system.
- As an administrator, I want to see if any member misses something (license, payment) to send them a reminder by email. Alo a want to see club members grouped by different parameters to send them different email templates.

### **Areas covered with this project**

1. **Customer management.** This system allows a club members database management , with their detailed information, attached licenses, email reminders, information and notifications.
2. **Staff management.** System will keep the actual schedule and keep records about coach work hours and availability.
3. **Appointment scheduling.** Coaches can show personalized calendars for coaches with their individual classes.
4. **Inventory order.** System will provide a list of possible inventory, the order form and will summarize that information for the accounting department.
5. **Reporting.** This system will generate different reports for different users to provide summary information that can be used for analysis and planning. For example, individual performance.

## **PROJECT USERS, ACTORS, VENDORS, ACTUATORS**

1. Project users. The main users are the club owner and coaches. The owner will see the whole picture, summary of business and will see the progress to make, analyze and plan. Coaches will use software for individual class agenda, for training plan submitting and for club members information tracking and performance records.
2. Actors. Software companies can provide support for this product as well as development of additional features.
3. Vendors. Hardware and Software vendors can provide equipment and software that can be used and integrated with this software.
4. Actuators. Servers, Databases, APIs.

## **PROJECT PROPERTIES**

- Functionality. Software systems should function correctly and perform described functions for which they are designed.
- Usability (User-friendly). Easy to use, reach what you need in less clique. Understandable even without manuals.
- Security. System has to be secure and protective for user information.
- Flexibility. Should be able to easily adapt to future changes and growth.
- Reliability. Should operate without failure.
- Maintainability. Should be easily maintainable with user backups.
- Integrity. Should have the possibility to be integrated in future with other applications such as finance management or marketing.

Here is some software that could be used for the development of this software:

.Net Framework: 4.5.2 / WPF / MySQL: 8.0.32

# Project Functionality

## Project Functionality

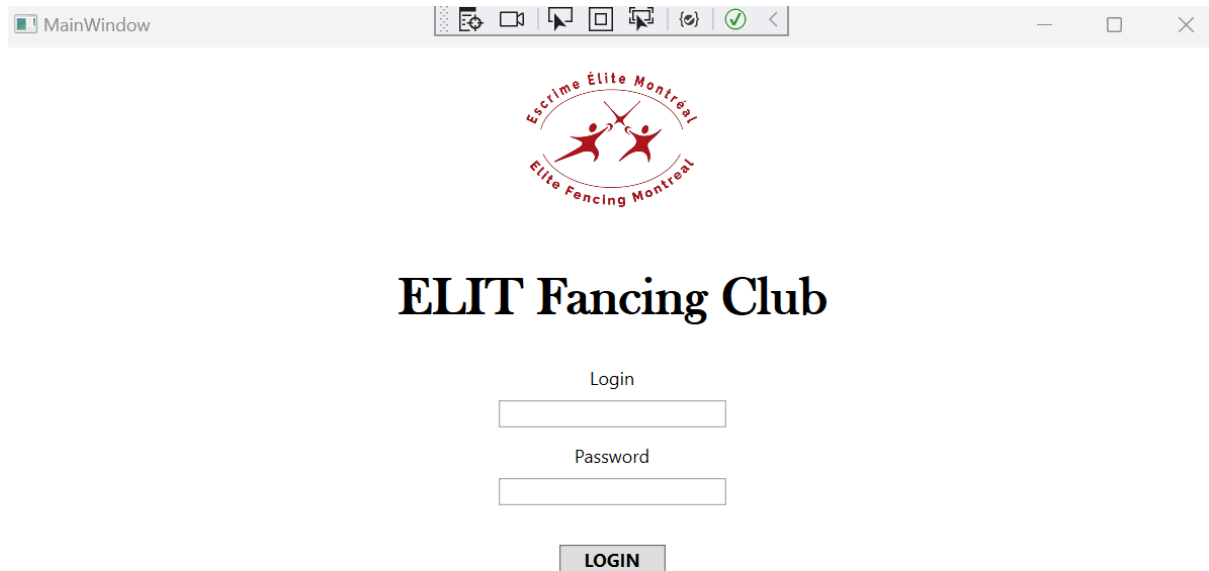
- 1. Login to the system*
- 2. Menu for navigation.*
- 3. Manage Members. Add New Member*
- 4. Manage Members. Select by ID*
- 5. Manage Members. Update Member by ID*
- 6. Manage Members. Delete Member by ID*
- 7. Manage Members. Filter*
- 8. Manage Coaches*
- 9. List of Members*
- 10. Member Information*
- 11. Camp registration*
- 12. Inventory*
- 14. Attendance*
- 15 System Messages*



## 1. Login to the system

The first window that appears after running the application is **Login window**. User has to enter his **Login** and **Password** to enter the system and then to press **LOGIN button**.

Note. In this application there is only an Admin role (it can be extended to Coach and Member roles in the future).



MainWindow

Escrime Elite Montreal  
Elite Fencing Montreal

# ELIT Fencing Club

Login

Password

LOGIN

## 2. Menu for navigation.

If the Login and Password were correct, then the Menu Window will appear. User can choose one of the proposed Menu:

**Manage Users**

**Manage Members**

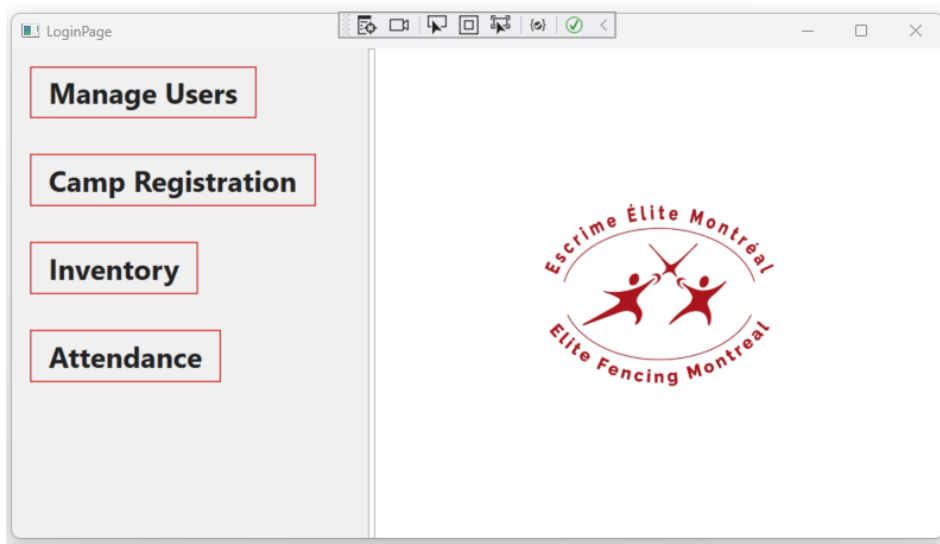
**Manage Coaches**

**Member List**

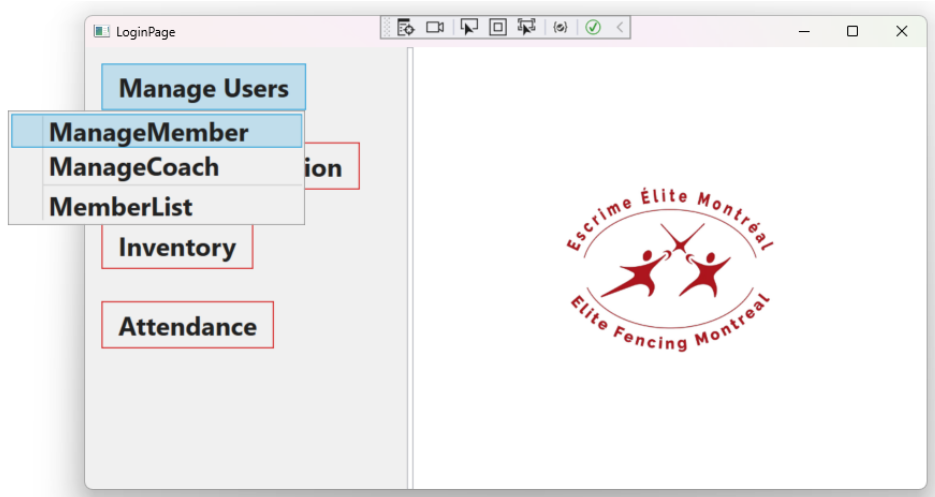
**Camp Registrations**

**Inventory**

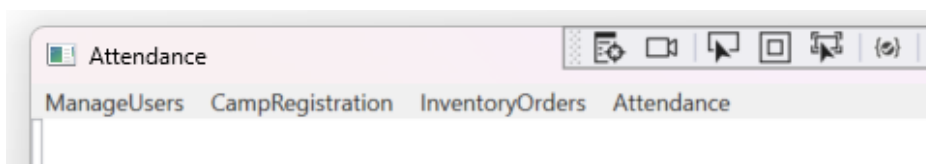
**Attendance**



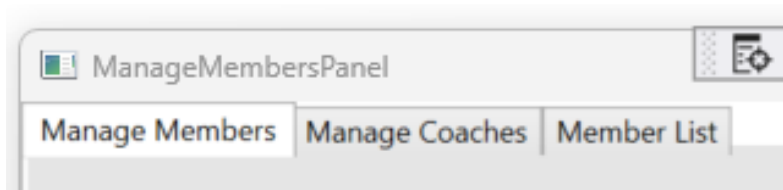
There is a submenu as well



Also **Top Menu** is available from the top of each window

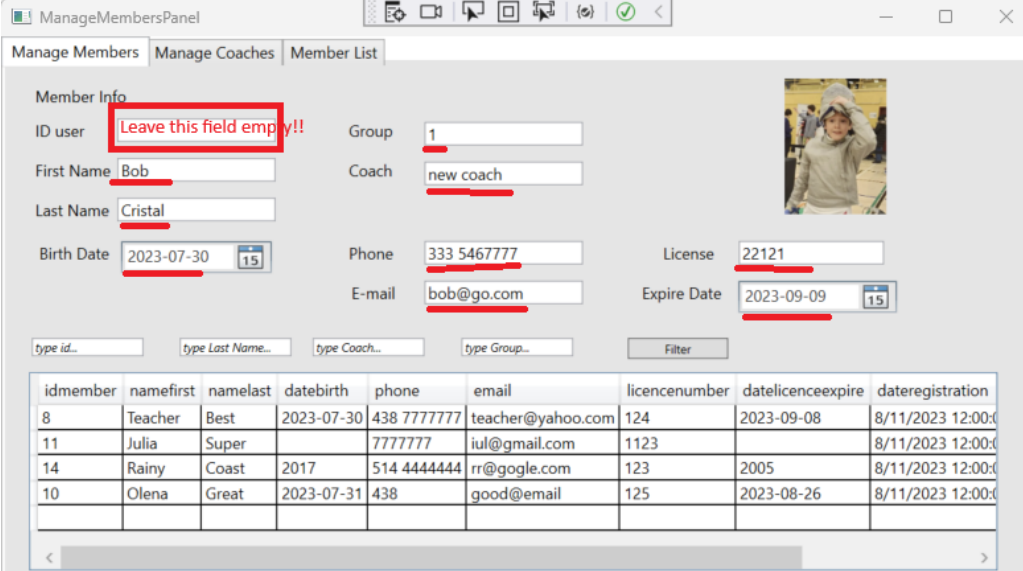


**Tab Menu** to **Manage Users** window it is implemented with another navigation, using Tabs. They are **Manage Members**, **Manage Coaches**, **Member List**



### 3. Manage Members. Add New Member

To **Add New Member**, the user has to fill in text boxes (sauf ID, that is autogenerated) and date pickers with information about the new member. **Add photo** functionality can be finished in the future. New Members will appear on the table below.



ManageMembersPanel

Manage Members | Manage Coaches | Member List

Member Info

ID user **Leave this field empty!!** Group 1

First Name Bob Coach new coach

Last Name Cristal

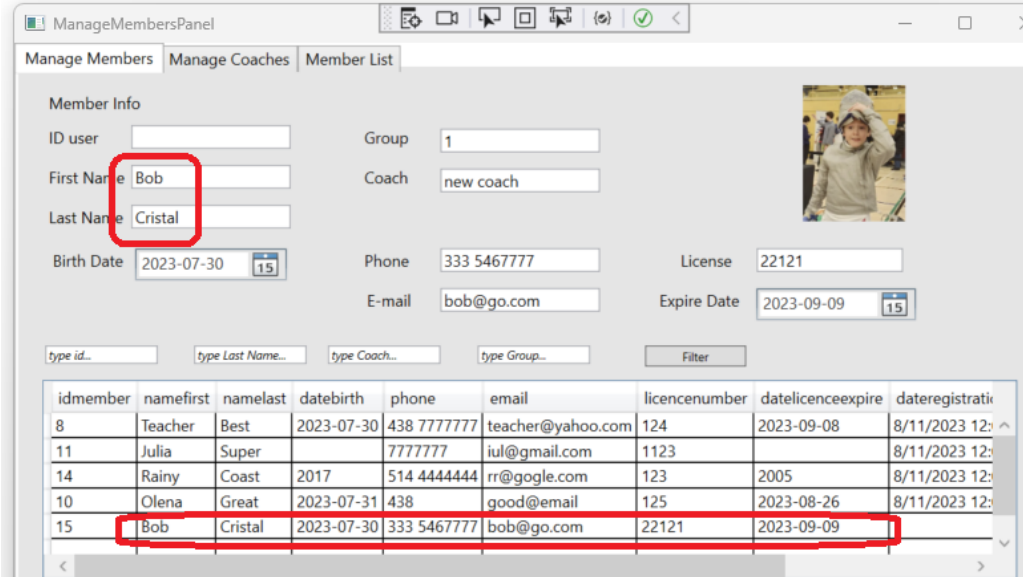
Birth Date 2023-07-30 15 Phone 333 5467777 License 22121

E-mail bob@go.com Expire Date 2023-09-09 15

type id... type Last Name... type Coach... type Group... Filter

idmember	namefirst	namelast	datebirth	phone	email	licencenumber	datelicenseexpire	dateregistration
8	Teacher	Best	2023-07-30	438 7777777	teacher@yahoo.com	124	2023-09-08	8/11/2023 12:00:00
11	Julia	Super		7777777	iul@gmail.com	1123		8/11/2023 12:00:00
14	Rainy	Coast	2017	514 4444444	rr@gogle.com	123	2005	8/11/2023 12:00:00
10	Olena	Great	2023-07-31	438	good@email	125	2023-08-26	8/11/2023 12:00:00

Add new Member Select by ID Update Member Delete Member



ManageMembersPanel

Manage Members | Manage Coaches | Member List

Member Info

ID user Group 1

First Name **Bob** Coach new coach

Last Name Cristal

Birth Date 2023-07-30 15 Phone 333 5467777 License 22121

E-mail bob@go.com Expire Date 2023-09-09 15

type id... type Last Name... type Coach... type Group... Filter

idmember	namefirst	namelast	datebirth	phone	email	licencenumber	datelicenseexpire	dateregistrati
8	Teacher	Best	2023-07-30	438 7777777	teacher@yahoo.com	124	2023-09-08	8/11/2023 12:00:00
11	Julia	Super		7777777	iul@gmail.com	1123		8/11/2023 12:00:00
14	Rainy	Coast	2017	514 4444444	rr@gogle.com	123	2005	8/11/2023 12:00:00
10	Olena	Great	2023-07-31	438	good@email	125	2023-08-26	8/11/2023 12:00:00
15	Bob	Cristal	2023-07-30	333 5467777	bob@go.com	22121	2023-09-09	

Add new Member Select by ID Update Member Delete Member

#### 4. Manage Members. *Select by ID*

Use ID user text box to enter Member ID and then the **SELECT by ID button** to click. Member information will appear in the correspondent fields.

The screenshot shows the 'ManageMembersPanel' application with the 'Member Info' tab selected. The 'ID user' field is highlighted with a red box. Below the form is a table of members. The 'Select by ID' button is highlighted with a red arrow.

idmember	namefirst	namelast	datebirth	phone	email	licencenumber	datelicenseexpire	dateregistrati
8	Teacher	Best	2023-07-30	438 7777777	teacher@yahoo.com	124	2023-09-08	8/11/2023 12:00 AM
11	Julia	Super		7777777	iul@gmail.com	1123		8/11/2023 12:00 AM
14	Rainy	Coast	2017	514 4444444	rr@google.com	123	2005	8/11/2023 12:00 AM

#### 5. Manage Members. *Update Member by ID*

Users can change any field and then click **UPDATE Member button**. Updated information will appear in the table below.

The screenshot shows the 'ManageMembersPanel' application with the 'Member Info' tab selected. The 'Update Member' button is highlighted with a red arrow. A success message dialog is displayed: 'Member info was updated successfully' with an 'OK' button.

st	datebirth	phone	email	licencenumber	datelicenseexpire	dateregistration	image	groupe	coach
	2023-07-30	438 7777777	teacher@yahoo.com	124	2023-09-08	8/11/2023 12:00:00 AM		1	Best
		7777777	iul@gmail.com	1123		8/11/2023 12:00:00 AM		2	New
	2017	514 4444444	rr@google.com	123	2005	8/11/2023 12:00:00 AM		2	my
	2023-07-31	438	good@email	125	2023-08-26				

## 6. Manage Members. **Delete Member by ID**

Use ID user text box to enter Member ID and then the **DELETE Member button**

The screenshot shows the 'ManageMembersPanel' window with the 'Member Info' tab selected. The 'ID user' text box is highlighted with a red rectangle and contains the value '10'. Below the form is a table with member data. At the bottom, the 'Delete Member' button is highlighted with a red arrow.

idmember	namefirst	namelast	datebirth	phone	email	licencenumber	datelicenseexpire	dateregistrati
8	Teacher	Best	2023-07-30	438 7777777	teacher@yahoo.com	124	2023-09-08	8/11/2023 12:
11	Julia	Super		7777777	iul@gmail.com	1123		8/11/2023 12:
14	Rainy	Coast	2017	514 4444444	rr@gogle.com	123	2005	8/11/2023 12:

## 7. Manage Members. **Filter**

Filter By Member ID . Use the shown *type id...* text box to enter Member ID and then click **FILTER button**. Selected Members will be in the table below.

In the future it can be implemented to filter by last name, coach, group.

The screenshot shows the 'ManageMembersPanel' window with the 'Member Info' tab selected. The 'type id...' text box is highlighted with a red rectangle and contains the value '10'. The 'Filter' button is highlighted with a red arrow. Below the form is a table with member data.

idmember	namefirst	namelast	datebirth	phone	email	licencenumber	datelicenseexpire	dateregistration	image
10	Olena	Great	2023-07-31	438	good@email	125	2023-08-26	8/11/2023 12:00:00 AM	

## 8. Manage Coaches

To Add Coaches with individual login and password. To be implemented in the future.

ManageMembersPanel


Manage Members | **Manage Coaches** | Member List

Coach Card

ID user  Login

First Name  Password

Last Name



## 9. List of Members

In this tab all members are displayed in the table

ManageMembersPanel

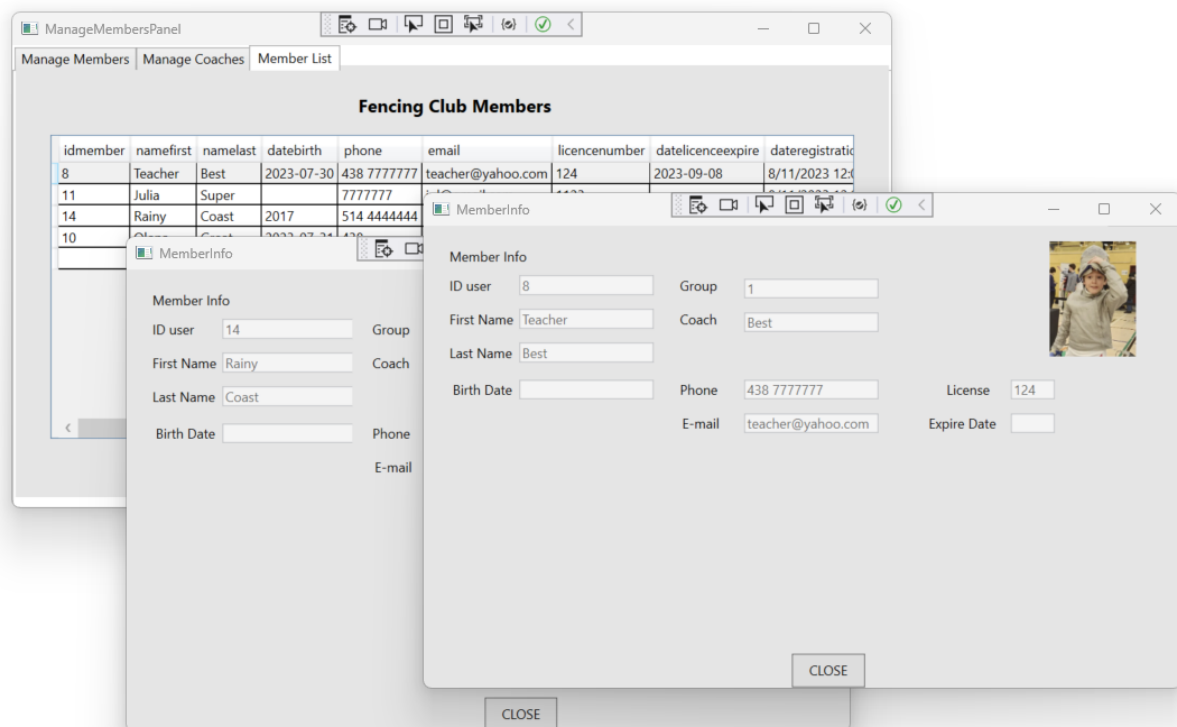
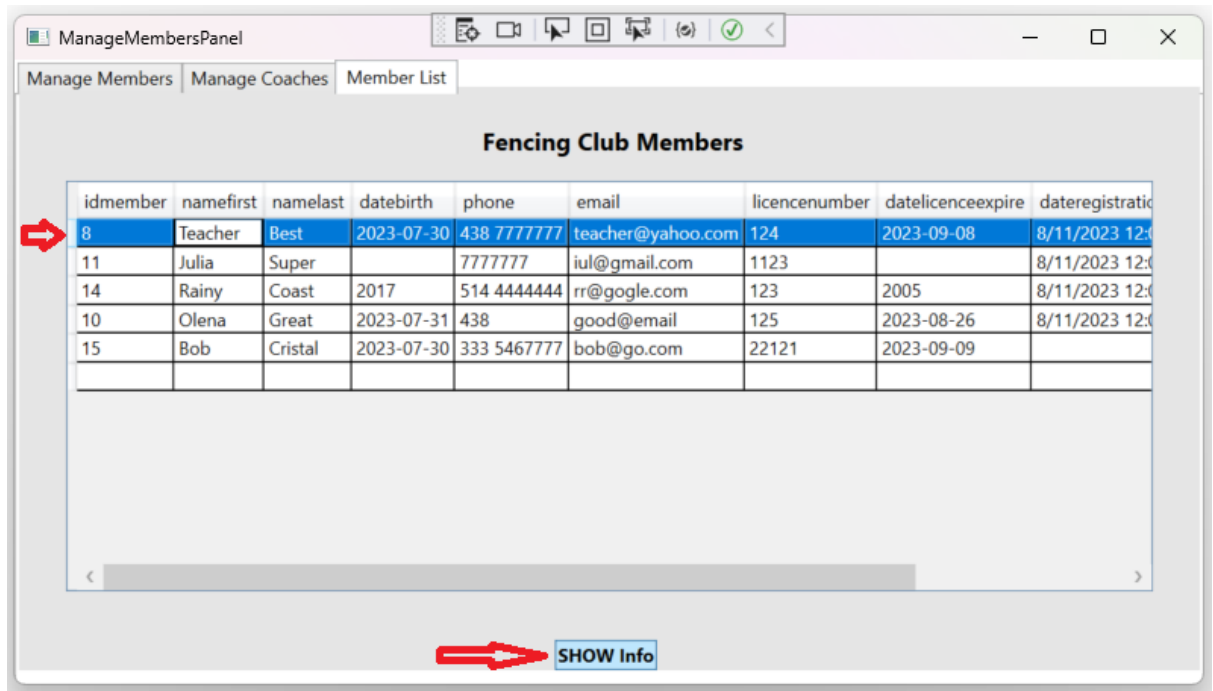
Manage Members | Manage Coaches | **Member List**

**Fencing Club Members**

idmember	namefirst	namelast	datebirth	phone	email	licencenumber	datelicenseexpire	dateregistratio
8	Teacher	Best	2023-07-30	438 7777777	teacher@yahoo.com	124	2023-09-08	8/11/2023 12:0
11	Julia	Super		7777777	iul@gmail.com	1123		8/11/2023 12:0
14	Rainy	Coast	2017	514 4444444	rr@gogle.com	123	2005	8/11/2023 12:0
10	Olena	Great	2023-07-31	438	good@email	125	2023-08-26	8/11/2023 12:0

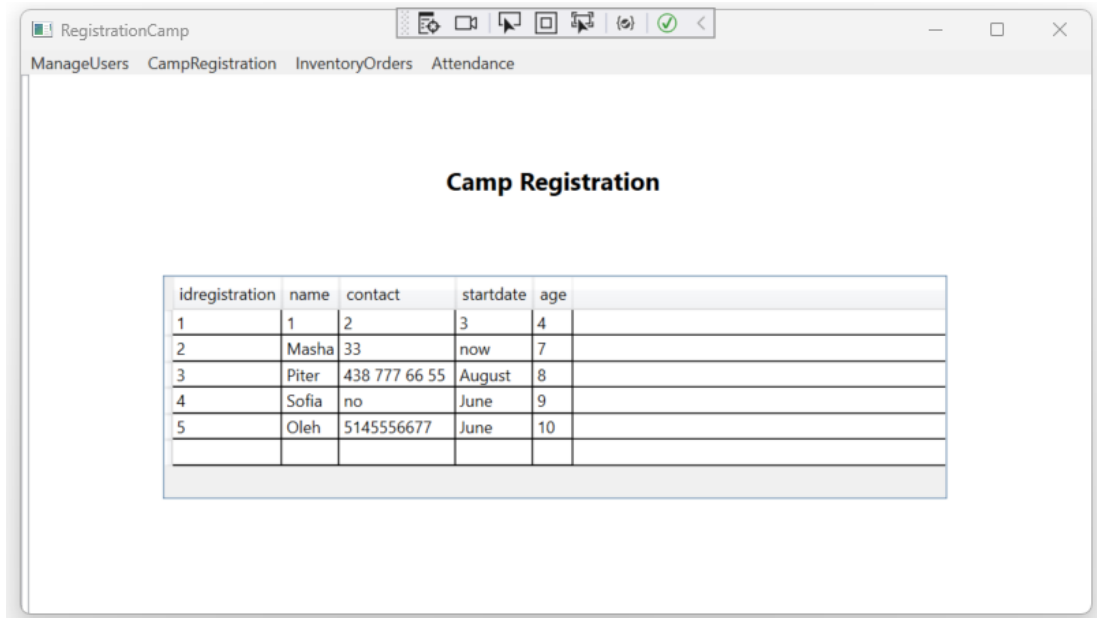
## 10. Member Information

To see Member information, click on any line in the Member table, then click on the SHOW Info button. The information about the selected member will appear in a new window. User can open many windows.



## 11. Camp registration

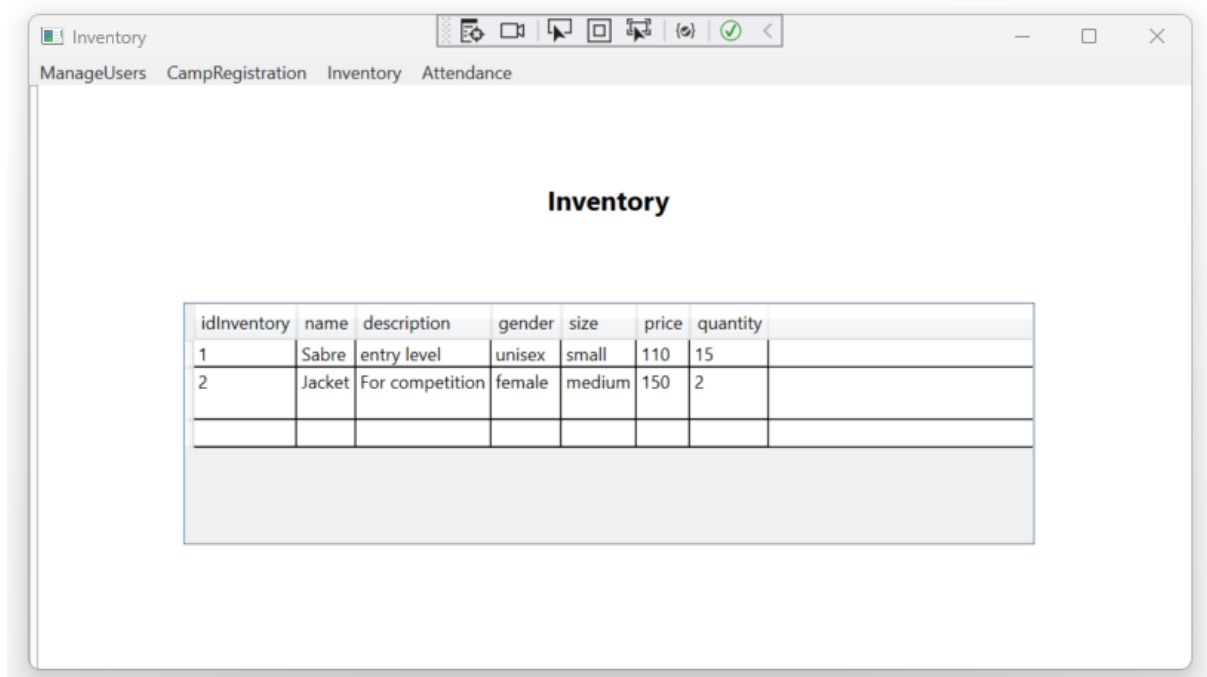
Using REST API, a person can fill in a Camp registration Form and submit it. Information will be recorded into the database. It will be updated in the table at the **Camp Registration window**.



idregistration	name	contact	startdate	age	
1	1	2	3	4	
2	Masha	33	now	7	
3	Piter	438 777 66 55	August	8	
4	Sofia	no	June	9	
5	Oleh	5145556677	June	10	

## 12. Inventory

The list of available Inventory is shown in the table of Inventory window. Can be extended with additional features in the future.



idInventory	name	description	gender	size	price	quantity	
1	Sabre	entry level	unisex	small	110	15	
2	Jacket	For competition	female	medium	150	2	



#### 14. Attendance

To keep records of attendance, every class, a coach has to create a new Attendance list. For that need to choose Date, Group, Coach and Members and click the **Mark as PRESENT on this class button**. Implementation can be modified in the future.

**Attendance**

Choose information for todays class:

2023-08-16

Select group:

Group #1

Group #2

Select coach:

Coach

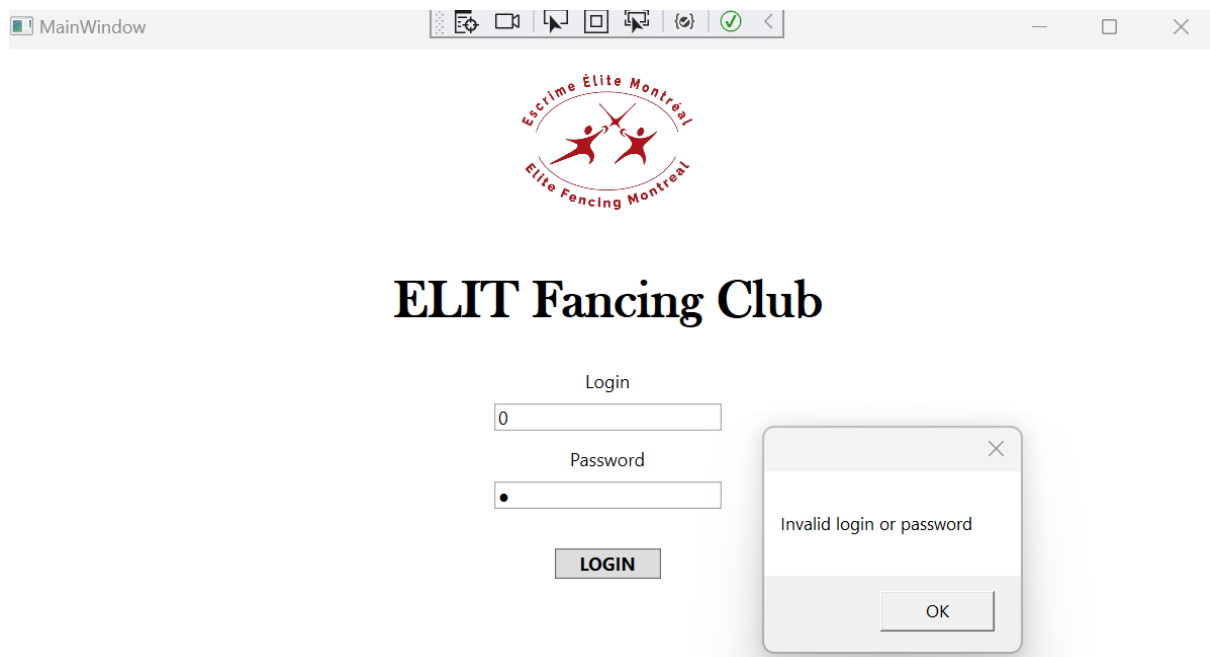
Kid's coach

Mark as present at this class

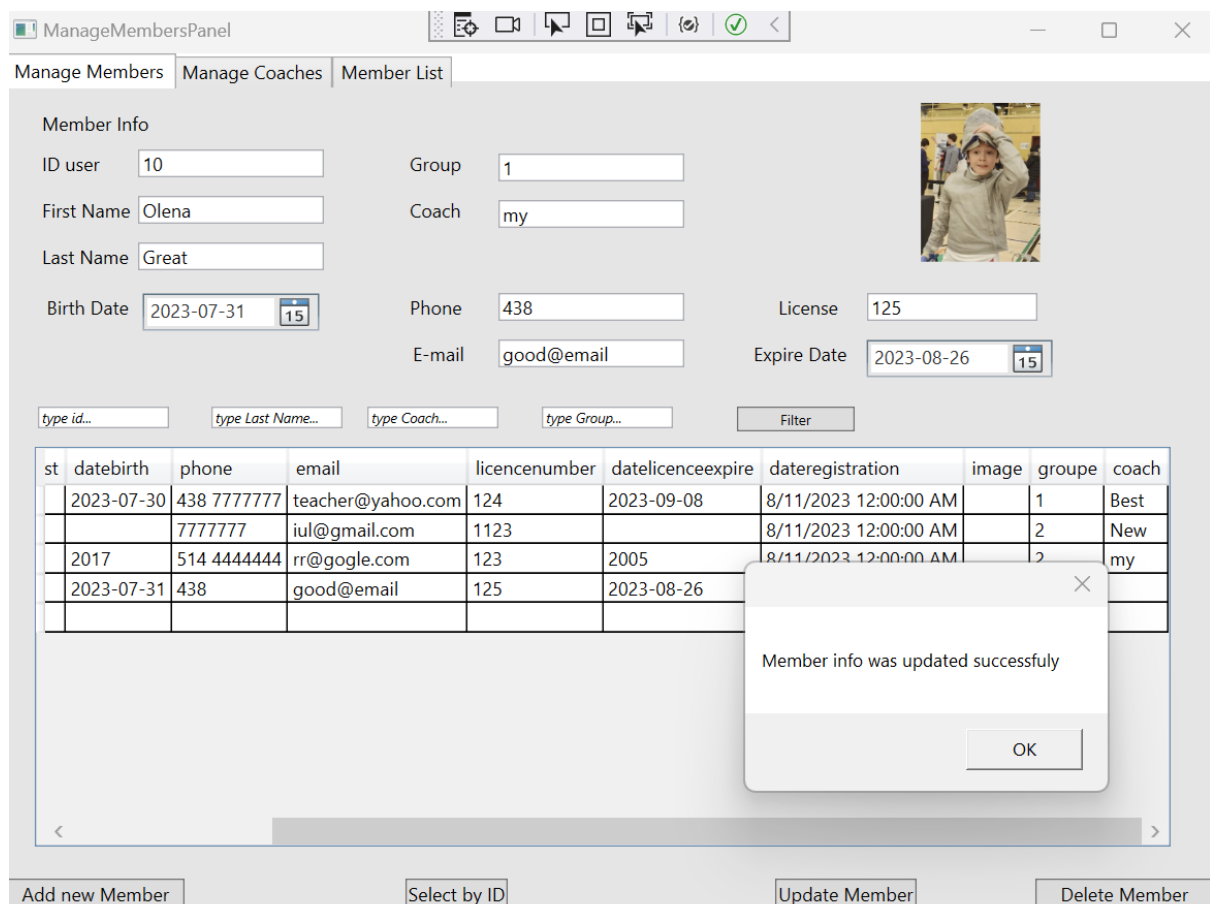
idmember	namefirst	namelast	datebirth	phone	en
8	Teacher	Best	2023-07-30	438 7777777	tea
11	Julia	Super		7777777	iul
14	Rainy	Coast	2017	514 4444444	rr@
10	Olena	Great	2023-07-31	438	go

## System Messages

ERROR. Incorrect password



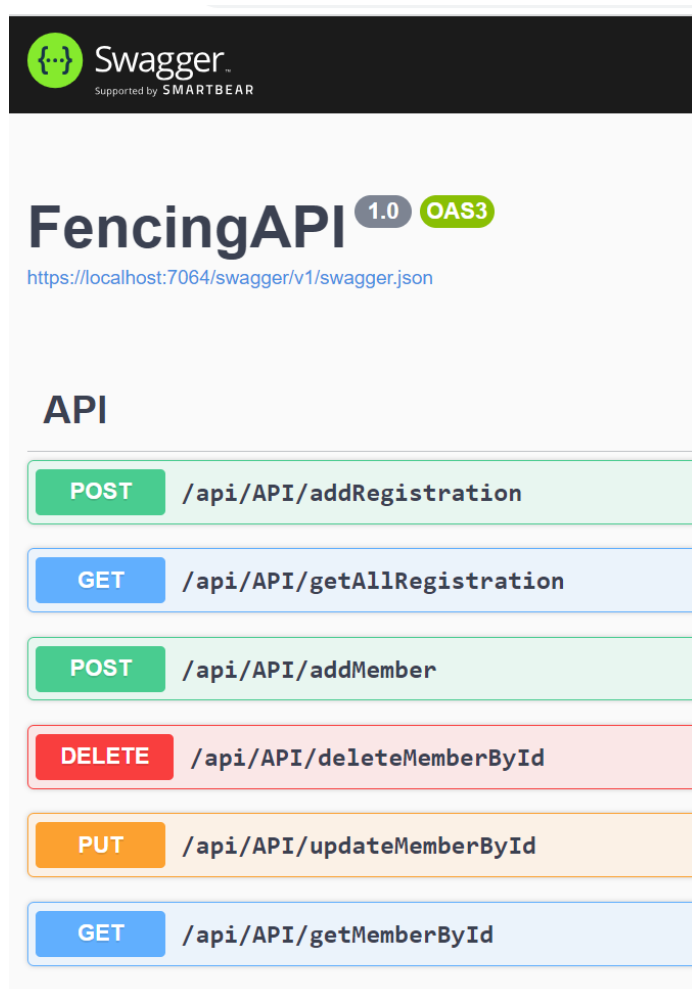
Confirmation message



## REST API implemented

REST API implemented in this project:

1. Add Camp Registration . For members to register for camps. Can be used by external web or mobile or other applications.
2. Get All Registration. All Camp registrations are visible in this application.
3. Add Member. All CRUD operations can be done by administrator, for example using mobile application.
4. Add Delete
5. Add Update
6. Get Member



Example of Piter's registration for a camp. Then we see it in the database.

POST

/api/API

Parameters

No parameters

Request body

```
{
  "idRegistration": 0,
  "name": "Piter",
  "contact": "438 777 66 55 ",
  "startDate": "August",
  "age": "8"
}
```

Execute

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7064/api/API' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "idRegistration": 0,
    "name": "Piter",
    "contact": "438 777 66 55 ",
    "startDate": "August",
    "age": "8"
  }'
```

	idregistration [PK] integer	name character varying (100)	contact character varying (100)	startdate character varying (100)	age character varying (100)
1	1	1	2	3	4
2	2	Masha	33	now	7
3	3	Piter	438 777 66 55	August	8

## Test cases implemented using NUnit

In this project some tests are implemented:

### 1. Test1\_getPasswordByLogin

This test gets the password from the database that corresponds to the login input;

2. **Test2\_isValidLogin** - This test checks if the password input corresponds to the password in the database for input login.

Given: Login and password input is correct.

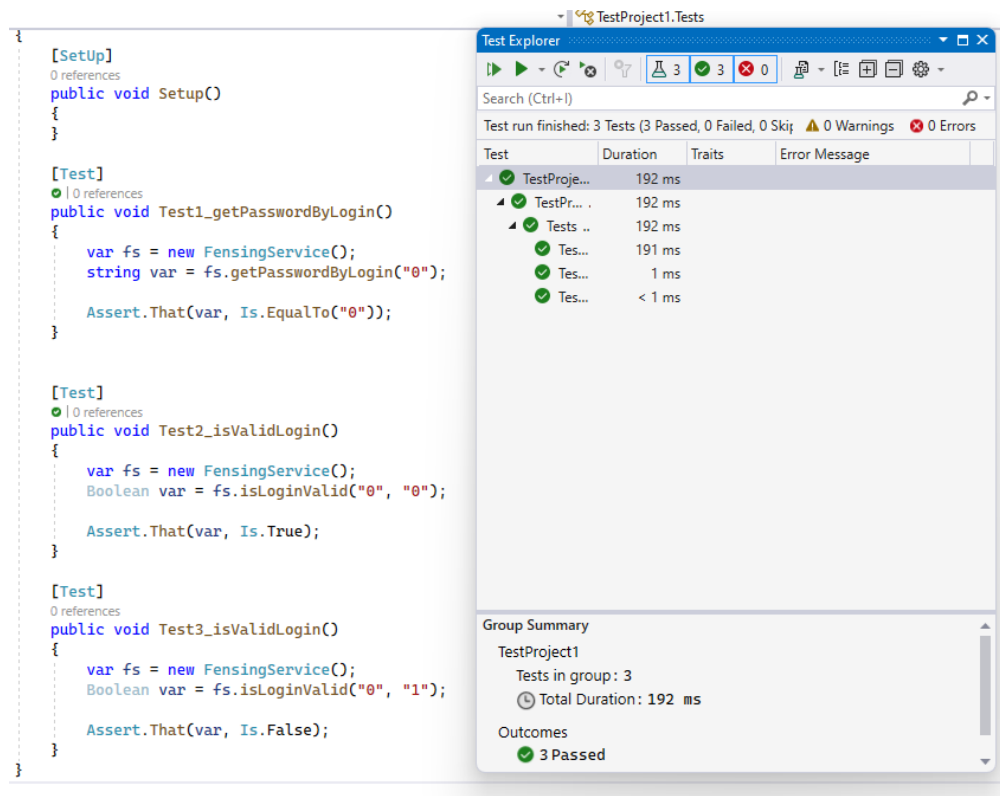
Then the test is True.

3. **Test3\_isValidLogin** - This test checks if the password input corresponds to the password in the database for input login.

Given: Correct Login but wrong Password input.

Then the test result is False.

As most implemented methods are calling the database, we have to use mock, so that database could stay without changes. That kind of testing can be implemented in the future.



## **Project Setup**

With the idea to provide to a client a whole service, we propose a service of the first set up (including Database, Visual Studio, etc) as well as a small introduction course.

## **Work Progress in Future**

- **Add photo** functionality can be finished in the future.
- As most implemented methods are calling the database, we have to use mock, so that database could stay without changes. That kind of **additional Testing** can be implemented in the future.
- In the future it can be implemented to **Filter by last name, coach, group**.
- To **Add Coaches** with individual login and password. To be implemented in the future.
- The list of available **Inventory** is shown in the table of Inventory window. Can be extended with additional features in the future.
- **Attendance**. Implementation can be modified in the future.
- **Different Reports** can be implemented in the future.
- All not implemented features from the first plan of this project.

This application was developed for Fencing Club but it can be well adapted for any sport or creativity club that has members, attendance, events, inventory, etc.