POLITECNICO
MILANO 1863

Set A Game

DOCUMENTATION ON A WEB APPLICATION

FOR THE AMATEUR WORLD OF TENNIS

Software Engineering (for Automation)

Bonacina Emanuele

Ceruti Federico

Spagnuolo Walter

Summary

INTRODUCTION **2**
FEASIBILITY STUDY **3**
Problem Description 3
Social Campaign 4
Benefits 5
REQUIREMENTS ANALYSIS AND SPECIFICATIONS **6**
Stakeholder Identification 7
Software Specifications 7
UML requirements diagram 9
Sequence Diagram 10
DESIGN AND DEVELOPMENTS **12**
The Architecture 13
Database 15
Submodules 16
Off-Log Interface 16
Login Interface 20
Finite State Machine 25
General Finite State Machine 25
Challenge Instructions modelled with Finite State Machine 26
Challenge logical management modelled with Finite State Machine 27
TESTING **28**
Introduction 28
Modules 29
CONCLUSIONS **36**

INTRODUCTION

"**Set-A-Game**" is a web-based software, regarding the amateur world of tennis, thought and tested by a group of students from Politecnico di Milano. We started from the feasibility study, and then we went through the specifications, the development and through the testing and validation of the software created. The requirements were implemented in a modular way with the aim of creating a flexible system that can be further expanded. At the end of the to the implementation discussion, we supply a testing part in which the features implemented are validated, stressing every exception that may arise in a daily use of the application.

We divided the entire work in the following chapters:

- FEASIBILITY STUDY
- REQUIREMENTS ANALYSIS AND SPECIFICATIONS
- DESIGN AND DEVELOPMENT
- TESTING AND VALIDATION

FEASIBILITY STUDY

Problem Description

The world of the sport deals a huge part in everyone's life. Mostly in the last years, the world of the health, the welfare and the sport activities is more and more followed, also at an amateur level. Simultaneously there was a big development in the digital world, and we can say that everything is almost connected. In fact, are available and are more and more downloaded and followed applications or pages related to the world of the sport.
A lot of Apps exist in which everyone can read results or rankings of a certain competition, deepen some news, or App in which are published live results of a contest or in which is described the live progress of a match or of a race. All these stuffs are development not only in the world of the professional sport but also at amateur level.

In particular, our interest covers the world on the amateur tennis, that is a hobby for us, but done with a certain diligence, seriousness and organization. For this reason we though to a solution that might be a support for amateur players in the organization of a match, a support to read results of other matches, to know the statistics of the opponents and their position in the chart ,with the final aim to make them more interested and involved in their sport activities. Due to this reason, the project is entirely based on the creation of a web application for the amateur world of tennis in which every player can look at the clubs nearby, can find the field on which play the match, can book the field (if available),can select some services as the rent of rackets, tennis balls, a referee, can see the statistics of the other players, can insert the points and the final result of the matches and have a confirmation from the opponent.
This can bring benefits not only for the amateur players, but also for the managers of the different clubs that in this way might plan and handle in a better way the different requests from the clients.

Social Campaign

The feasibility of the project was studied also through a social campaign, with the aim of catching opinions about the different services we want to offer with our application. Indeed, we shared our idea with most of our friends that are involved in the amateur tennis world to get to know what they think about it. We then created a survey and the questions were:

1. Do you think might be useful an app that allow you to organize and set tennis matches?

2. How would you rate the possibility of sending a challenge request to one of the other players and managing the organization of the match through an app? (where and when it will be played)

3. How would you rate the possibility of knowing in advance the number of available clubs and fields nearby you?

4. What do you think about the idea of inserting the result of the matches and modifying automatically your chart position and your opponents' chart position?

5. How would you rate the possibility of being able to know the statistics of your opponents and their chart position?

We got a positive feedback from the 92% of our friends.
Moreover, the most interesting point, for the players, is the possibility to send challenges request and insert results of the matches and the fact that there is the modification of the chart position in an automatic way.

Benefits

To sum up the overall idea, here are listed the main benefits for both players and tennis clubs.

For the players:

- The possibility of sending a challenge request, booking a field and organize a match, requiring in advance, if necessary, some services.
- The possibility of inserting the points and the final results of the matches (and wait the confirmation by the opponent).
- The automatic modification of the chart position depending on the result of the disputed match.

While for the clubs' side:

- Reduction of management costs due to a platform able to organize the matches in the available fields, able to manage the points and the results of the different matches and able to verify if the players are registered with a certain type of card (accepted or not by the club itself).

REQUIREMENTS ANALYSIS AND SPECIFICATIONS

This part of the report gives a description of the features involved in the predevelopment stage of our Web Application project. Therefore, it will emphasize the specifications and how they have been integrated inside the app's flows, with the use of UML diagrams.
The requirements of the application can be summarized in this way:

- Amateur tennis players can send a challenge request to one of the other users, selecting him from the player list. The opponent can refuse or accept the challenge request. Once the challenge is accepted, one can retire (and wait again the confirm from the opponent).

- The players can insert the points and the results of the matches in the app, or check the results inserted by their opponents.

The application is a resource not only for the players but also for the clubs, that might manage in a better way their activities.

Stakeholder Identification

Stakeholder	Number	Role in the project	Subgroup
Amateur tennis players (members of a federation)	5k	Active users of the application	Client
External guests	300	Active users of the application	Client
Tennis clubs	1k	Management of the matches and of the tennis court	Providers
Developers	3	Owner of the idea and responsible for the design	Majority stakeholder

Software Specifications

All the specifications are discussed in detail in the following section:

- When one user registers himself, he must insert name, surname, a valid e-mail and a password.
- In the application, each user to login must insert the e-mail and the password.
- The application will collect the credentials of each user in a database not to allow the duplication of e-mails.
- The application will display the name and a Logout option.
- In the section, “my profile”, one can modify his personal data.
- Each registered user will be characterized by its number of card, its type of card, with the release and expiration date of the card itself.

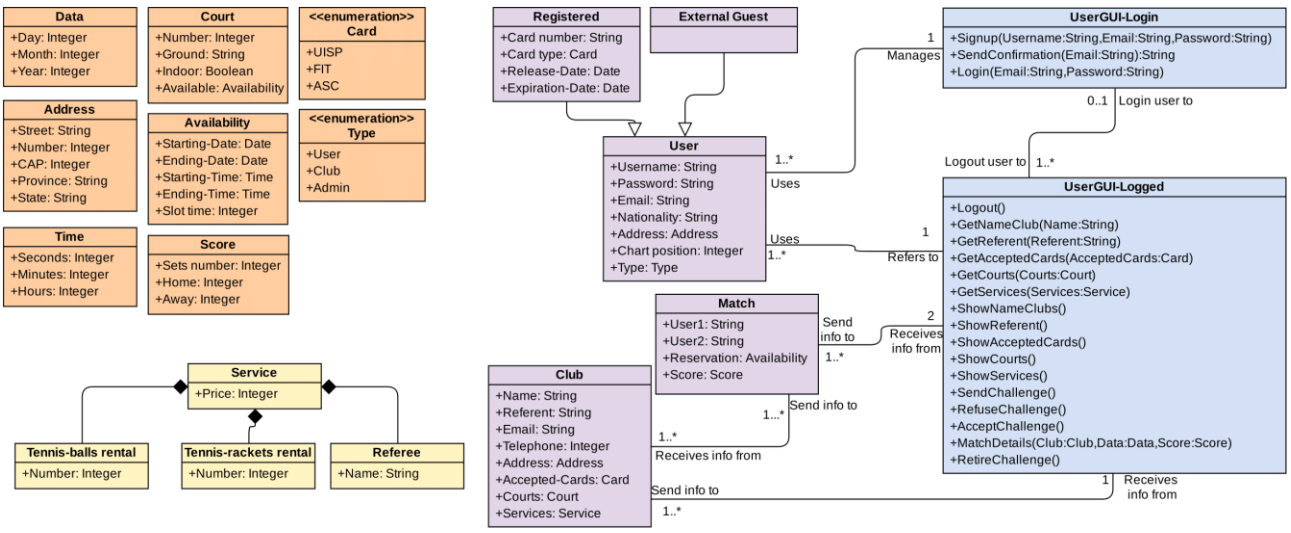
- Each club will be characterized by the name of its referring, its telephone number, the type of cards that are accepted in that club, the number and the type of the fields present, the services available.
- The user will access to a section to select one player to send him a challenge request.
- The opponent will be able to accept or refuse the challenge request.
- Once the challenge request is accepted, both the users can access to a details challenge section in which will be possible to select the club in which the match will be played, the data and in which will be possible to insert the result.
- Each user can play a maximum of 4 matches in a month and challenge the same player a maximum of 6 times in a year.
- Once the challenge request will be accepted, one of the two users can send a request of retirement, and this request must be accepted by the opponent.
- The app will present to the user the different fields present in the selected club.
- The players will be able to read some additional services (rental of tennis rackets, tennis balls, the presence of a referee).
- One of the players will be able to insert the final result of the match, the opponent will be able to confirm and check that result, before that it is registered by the system.
- For each match win 3 points are awarded, 2 points for each match drawn, and finally 1 point for each match lost.
- The application provides an automatic calculation of the new chart position of the players involved in that match.
- The not-registered users will be able to read results of past matches, read the statistics of the players registered and the features of the tennis clubs registered (valid also for the registered users).
- The application will send the complete match information to the clubs' database and to the players' database.

UML requirements diagram

Once the specifications are listed, let's have a look at the system in terms of classes of the objects. A class is a template used in object-oriented programming to create objects, which are instances of that class. This is the preliminary stage for the design of the software implementation, with which we created a scheme that tries to recreate the relationships between objects.

We used composition relationship to describe the subcomponents of the SERVICE class, whose existence is mandatory for its children's existence. USER, on the other hand, is inherited respectively from Registered/External Guest. It's important to underline that MATCH is at the same time sent to the USERS and to the CLUB.

Finally, we want to remark that the application, for our purposes is limited at the interactions with the user, not caring about its management nature.



Sequence Diagram

To determine the flow of the application, it's possible to describe the interaction among objects through sequence diagram. Doing that, the behaviour of the system is well defined over time and the app's cycle is properly shown. The sequence diagram below is divided into four parts:

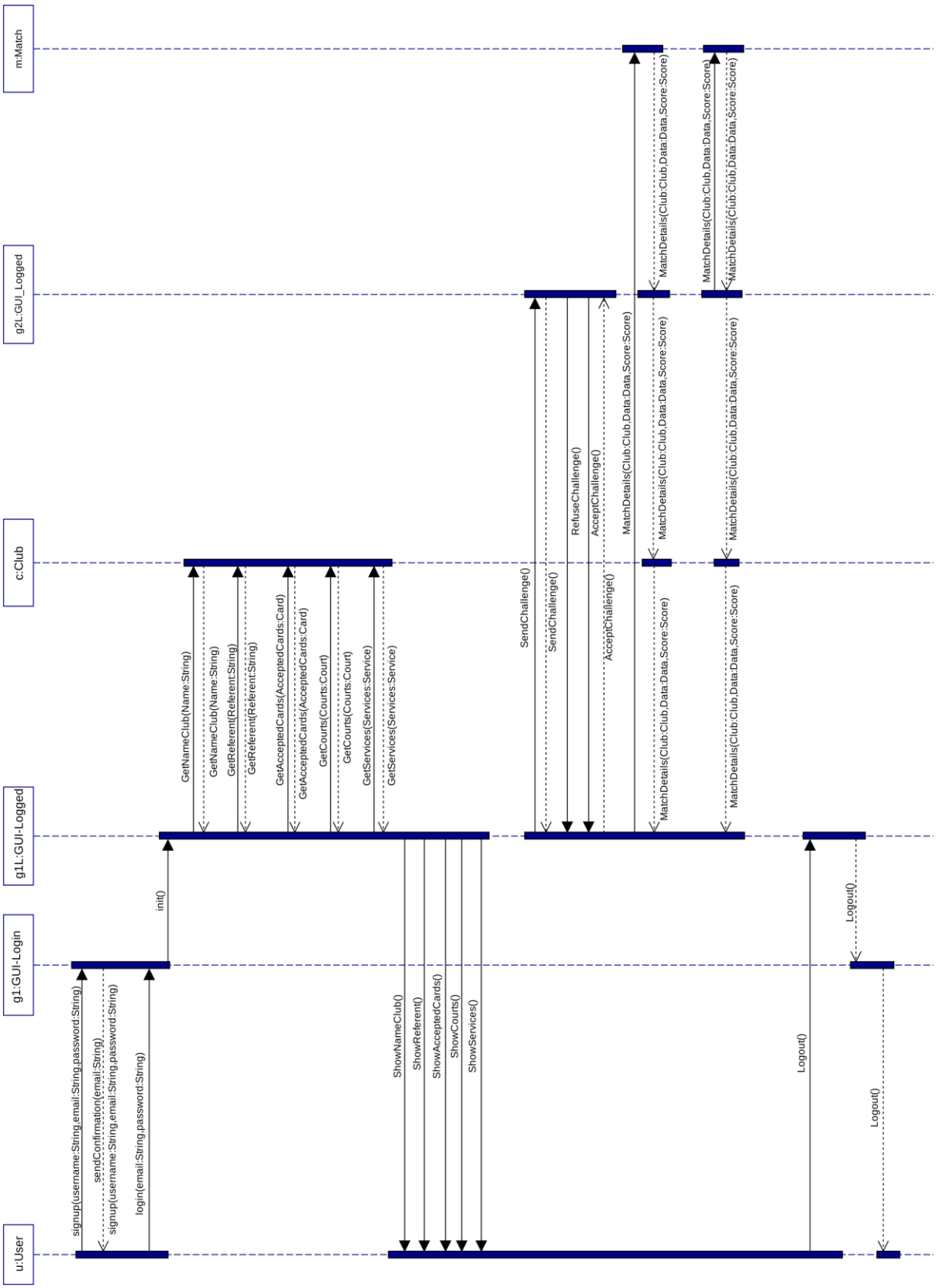
- **Signup or login**: where the user creates is credential for login or if already got it, enters in the application.

- **Getting-info**: the logged user interface interacts with users and clubs asking for information needed to display name of the clubs, referents, type of courts, services, and cards accepted.

- **Challenges**: the interaction between two users and the club. A user can: send a challenge request to another user, which can accept it or refuse it. Once the request is accepted, it is possible to select the club in which the match will be performed or it is possible to send a request of retirement from the match to the opponent.

- **Logout** quitting the session.

Sequence Diagram



DESIGN AND DEVELOPMENTS

In Design and Development chapter are shown all the technical details of the "Set a Game" project. It starts with explaining our principal design goals until the description of the general architecture (high level) and its components.
It's not in the author's intention to go deeply through the code implementation since the quite big amount of code wrote could be less interesting with respect to the overall designed framework.

Before going in the building details of the project, the fourth principal property in order to define the approach of the app's implementation are explained:

- Scalability: Our App must be scalable in terms of being able to support multiple players and clubs as well as support many potential Club with their own properties.

- Modularity: it allows adding or extending multiple features both from server and from client side in order to be open to its futures developers.

- Usability: it is not necessary to have a user manual because the user's interface is very simple and intuitive.

- Cross-Device Compatibility: The users should have access to our App wherever and whenever needed.

The Architecture

The system owns two user interfaces as mentioned in specification chapter in term of logged or not logged user. These interfaces own multiple sub-modules, which can be seen as a standalone interface, but for simplicity, they referred to a single user interface. In the following table these sub-modules are defined, comparing the first and second GUI in terms of functionalities the user could have access.

Off-Logged User interface	Logged User Interface
• Open a form popup where the player enters his information for signing-up • Open a form popup where the club enters his information for signing-up • Read past matches • Read Statistic of the registered players • Read features of the registered tennis club. • Control already existent users	• Logout • Send a challenge request to another user • Accept or refuse the challenge request • Select the club in which the match will be played • Select the data • Once the challenge request is accepted, one of the two user can send a retirement request • Insert final result of the match • Accept points if you are an opponent player • Show the winner • Modify user's data

The distinction between these two interfaces is been done in order to have a powerful tool also for the not logged users because in these latter years sport activities are really followed from all the users.

In this case, our application is designed to not only show the final result of the match but the not logged player can see also the history of all matches and the statistic of the players, in this way who love this sport can follow it totally. This means that the "Set a Game" is directly designed on the user's needs regardless the fact that the user is logged or not, usual or occasional to the application.

For both users our service can be considered a useful tool to have more knowledge on the various matches and on the statistics of the other players.

The development of this app can be faced up from multiple point of view as mentioned in the previous chapters. The problem we are aimed to solve with our application refers to players (clients), but clearly the same problem can be approached and solved from the clubs side (providers).

Database

We organized the database with different tables, to which the different modules of our application can repeatedly access.

- Table USER contains name, surname, birthday, e-mail, password, address, telephone number, type of card of each registered user.
- Table CLUB contains information about the different clubs registered(name of the referent, telephone number, address, type of tennis courts, cards accepted).
- The table MATCHES stores all the data related to disputed matches (players, score, results)
- The table CHARTS contains the list of the chart points.
- The table NATIONALITY and PROVINCES collects respectively the nationality and the provinces of each user.

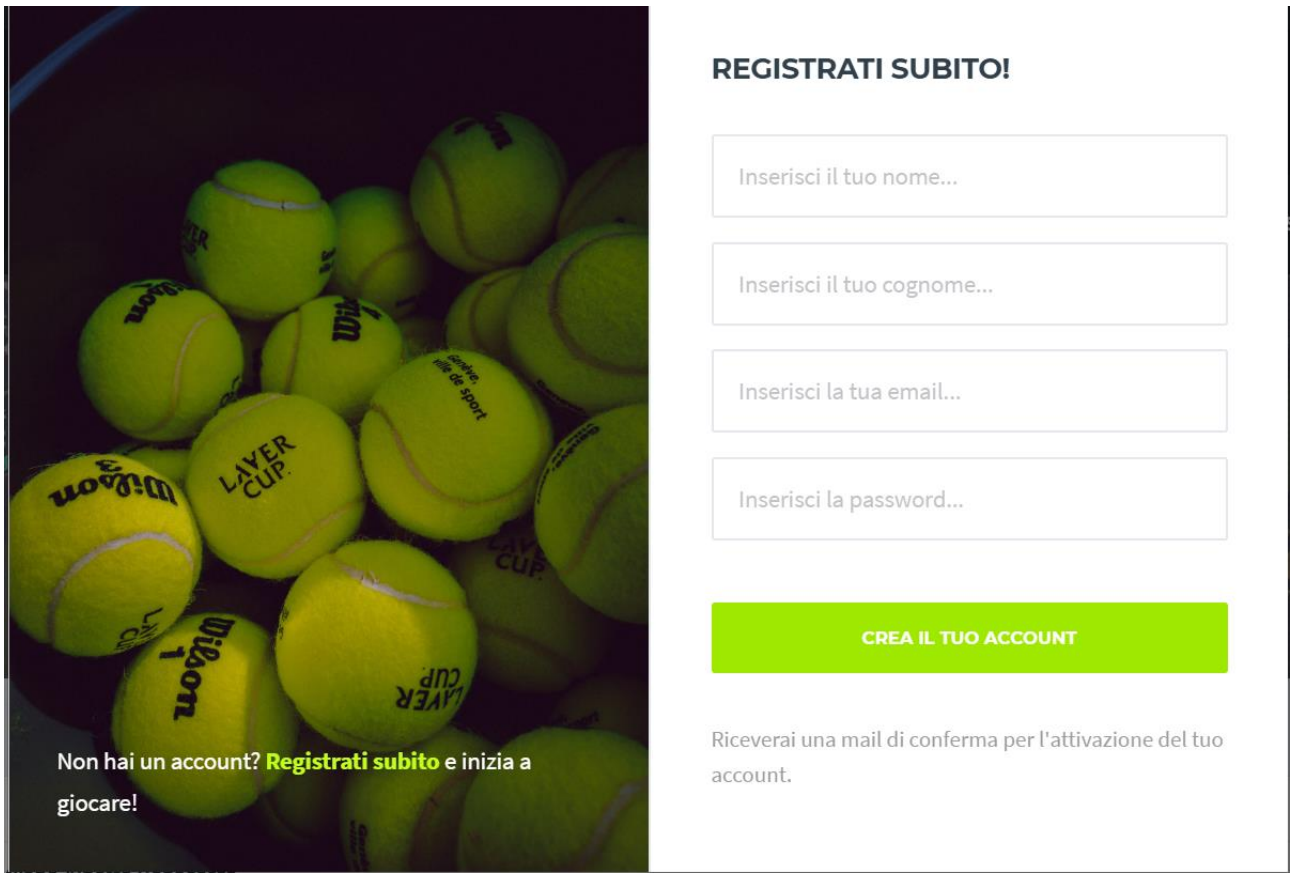
Submodules

In this section is provided a description of each submodules collected in the two user interfaces (OFF-LOG and LOGIN).

Off-Log Interface

The off-logged interface allows to view the external core of the web app. This interface offers various submodules.

SIGN-UP MODULES

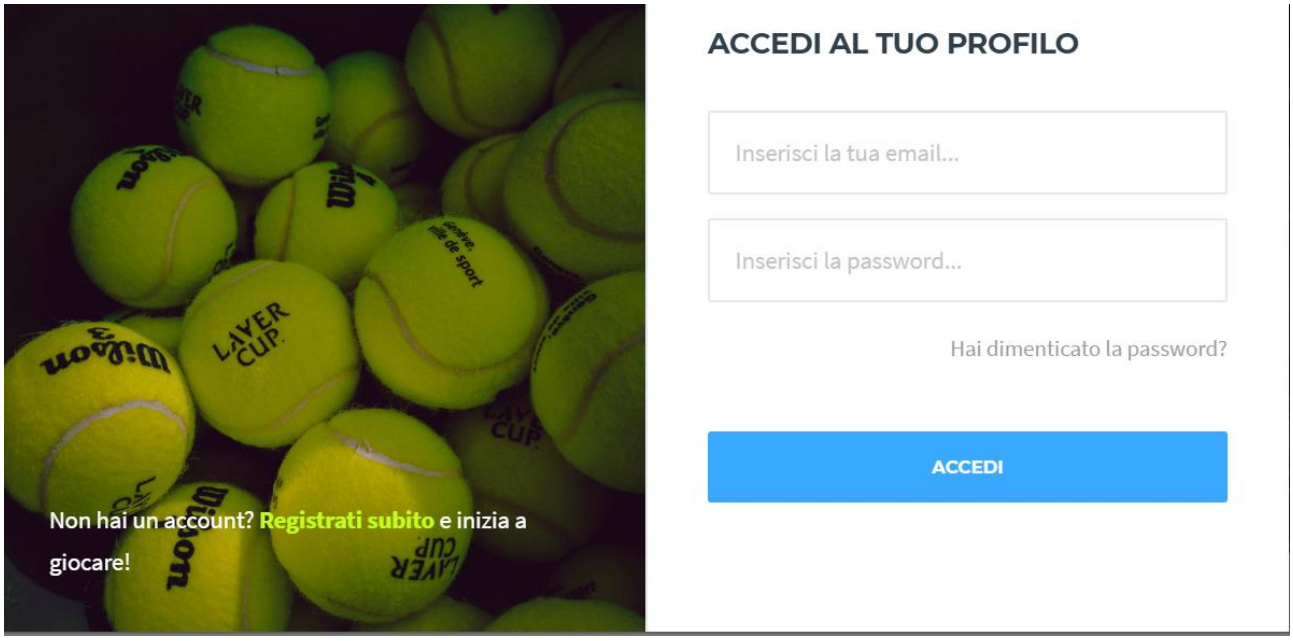


Through this module new users are able to register themselves by inserting:

- Name
- Surname

- E-mail address (it must not already exist)
- Password

LOGIN MODULE



With this module a registered user can access to the main functionalities offered by the app and can switch to the other GUI interface. To do it, the user has to insert:

- E-mail address
- password

HOME MODULE

This is the first module that appears, in our Web app. Through this is possible to read the last disputed matches and the chart.

CLUBS MODULE

This module allows to a logged-off user to see the list of the clubs registered to the site with their main characteristics, such as the referent, the address, the types of courts, the cards accepted, the offered services.

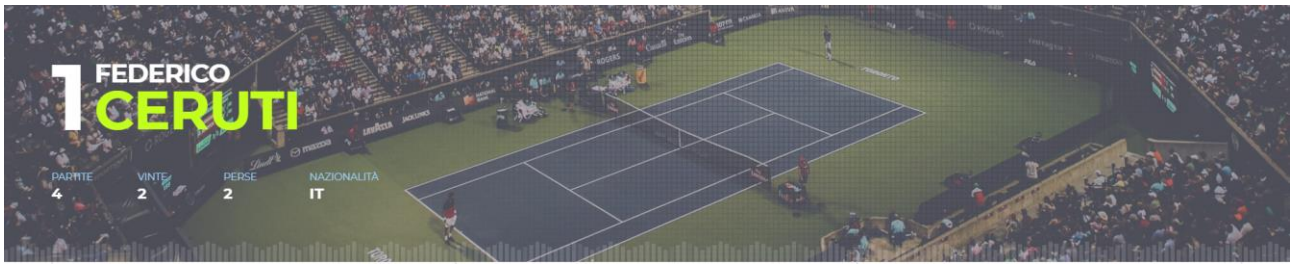
CIRCOLO	REFERENTE	INDIRIZZO	CARTE	CAMPI	SERVIZI
Tennis Club Milano Alberto Bonacossa	Mario Rossi segreteria@tcmbonaccosa.it - 02330012249	Via G. Arimondi, 15 - 20155, Milano (MI)	FIT, UISP	16 campi da tennis, 12 terra rossa, 1 cemento e 2 in mateco	Noleggio racchette, noleggio palline, Arbitro
Tennis Club Ambrosiano	Guido Verdi info@tcambrosiano.com - 0226414392	Via Feltre, 33 - 20155, Milano (MI)	UISP	16 campi da tennis, 13 in terra rossa e 3 in greenset	Noleggio racchette

PLAYERS MODULE

Through this module is possible to read the list of the players registered in alphabetical order by name, with how many matches one player has won, lost or drawn and his position in the chart.

If one specific player is selected, the last disputed matches by him are listed and is possible to read his statistics and his nationality.

GIOCATORE	PARTITE	VINTE	PERSE	PAREGGIATE	FINALI	TITOLI	CLASSIFICA
Emanuele Bonacina	3	1	2	0	SOON	SOON	2
Federico Ceruti	4	2	2	0	SOON	SOON	1
Walter Spagnuolo	1	1	0	0	SOON	SOON	3



27-11-2019	**Tennis Club Ambrosiano** **Referente**: Guido Verdi **Contatti**: info@tcambrosiano.com - 0226414392 **Indirizzo**: Via Feltre, 33 - 20155, Milano (MI) **Carte accettate**: UISP **Campi**: 16 campi da tennis, 13 in terra rossa e 3 in greenset **Servizi**: Noleggio racchette	Federico Ceruti	3 - 6	Emanuele Bonacina
25-11-2019	**Tennis Club Milano Alberto Bonacossa** **Referente**: Mario Rossi **Contatti**: segreteria@tcmbonaccosa.it - 02330012249 **Indirizzo**: Via G. Arimondi, 15 - 20155, Milano (MI) **Carte accettate**: FIT, UISP **Campi**: 16 campi da tennis, 12 terra rossa, 1 cemento e 2 in mateco **Servizi**: Noleggio racchette, noleggio palline, Arbitro	Emanuele Bonacina	2 - 6	Federico Ceruti
24-11-2019	**Tennis Club Ambrosiano** **Referente**: Guido Verdi **Contatti**: info@tcambrosiano.com - 0226414392 **Indirizzo**: Via Feltre, 33 - 20155, Milano (MI) **Carte accettate**: UISP **Campi**: 16 campi da tennis, 13 in terra rossa e 3 in greenset **Servizi**: Noleggio racchette	Federico Ceruti	0 - 3	Walter Spagnuolo
22-11-2019	**Tennis Club Milano Alberto Bonacossa** **Referente**: Mario Rossi **Contatti**: segreteria@tcmbonaccosa.it - 02330012249 **Indirizzo**: Via G. Arimondi, 15 - 20155, Milano (MI) **Carte accettate**: FIT, UISP **Campi**: 16 campi da tennis, 12 terra rossa, 1 cemento e 2 in mateco **Servizi**: Noleggio racchette, noleggio palline, Arbitro	Federico Ceruti	4 - 2	Emanuele Bonacina

CHART MODULE

In this module is possible to read the chart of the users. The list is the same of that explained in the previous module, with the difference that is ordered in a different way.

CLASSIFICA	GIOCATORE	PUNTI	PARTITE	VINTE	PERSE	PAREGGIATE	FINALI	TITOLI
1	**Federico** Ceruti	8	4	2	2	0	0	0
2	**Emanuele** Bonacina	5	3	1	2	0	0	0
3	**Walter** Spagnuolo	3	1	1	0	0	0	0

MATCHES MODULE

This module allows to read the list of the last matches played and confirmed on the web app (as in the player's matches modules).

Login Interface

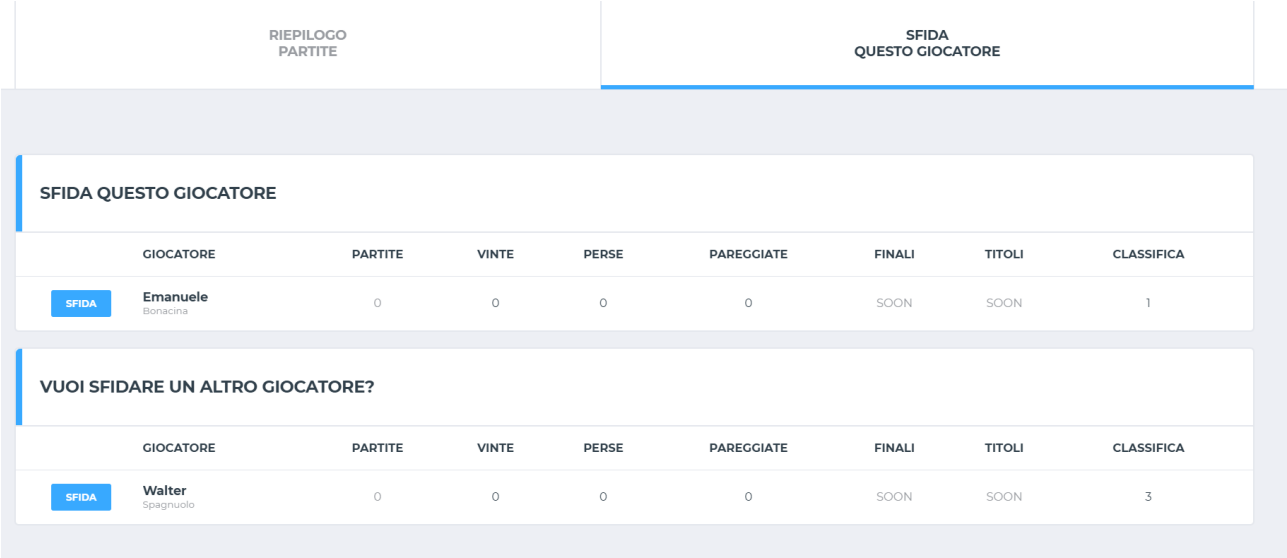
With the login operation, the user can go into the core of the application and exploit all its functionalities.
Once the login has been made, one can:

- select a player to challenge in a specific club in a specific court or receive a challenge from another user;
- select if accept or deny the request of challenge and once the challenge is accepted the user can select the details about it (like the club in which the challenge will be played and the day) or can send a request of retirement to the opponent (he can in turn accept the retirement or refuse it);
- insert the final result, and confirm the score;
- modify his personal data.

In this section are described the different modules from which the logged interface is composed by remembering that shares all them of the off-log GUI at which others are added.

CHALLENGE A PLAYER MODULE

Through this module is possible to send a challenge to one player. From the list of all the player is possible to select a specific player. The main difference with respect to the off-logged interface is that over the last matches, of that specific player, the option “challenge this player” is available. Therefore, is visible again the list of all other players and the user can select to challenge each of them.



MY MATCHES MODULE

In this module are shown all the matches that are outstanding, both sent and received and pending confirmation. If the challenge request is sent, the user can retire it first that the opponent accepts it.

If the request is sent and the opponent has not yet responded, with the option “retire the challenge” the challenge request is deleted automatically.

If the request is received, with the option “refuse the challenge”, it is deleted automatically

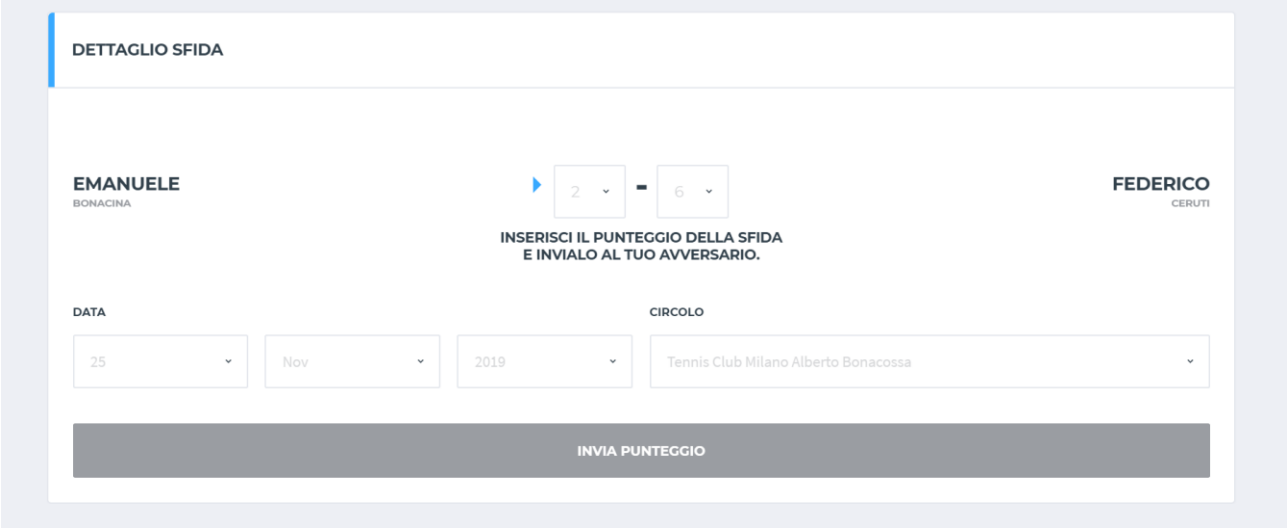
Once the challenge request is accepted by the opponent, the options "details" and "retire the challenge" appear.

RETIRE THE CHALLENGE MODULE

Once "retire the challenge" is selected, a request of retirement is sent to the opponent, which has to accept or to refuse the request.

CHALLENGE'S DETAIL MODULE

Through this module, is possible to insert the result of the match, the day and the club in which it was played, and then sent the information to your opponent.



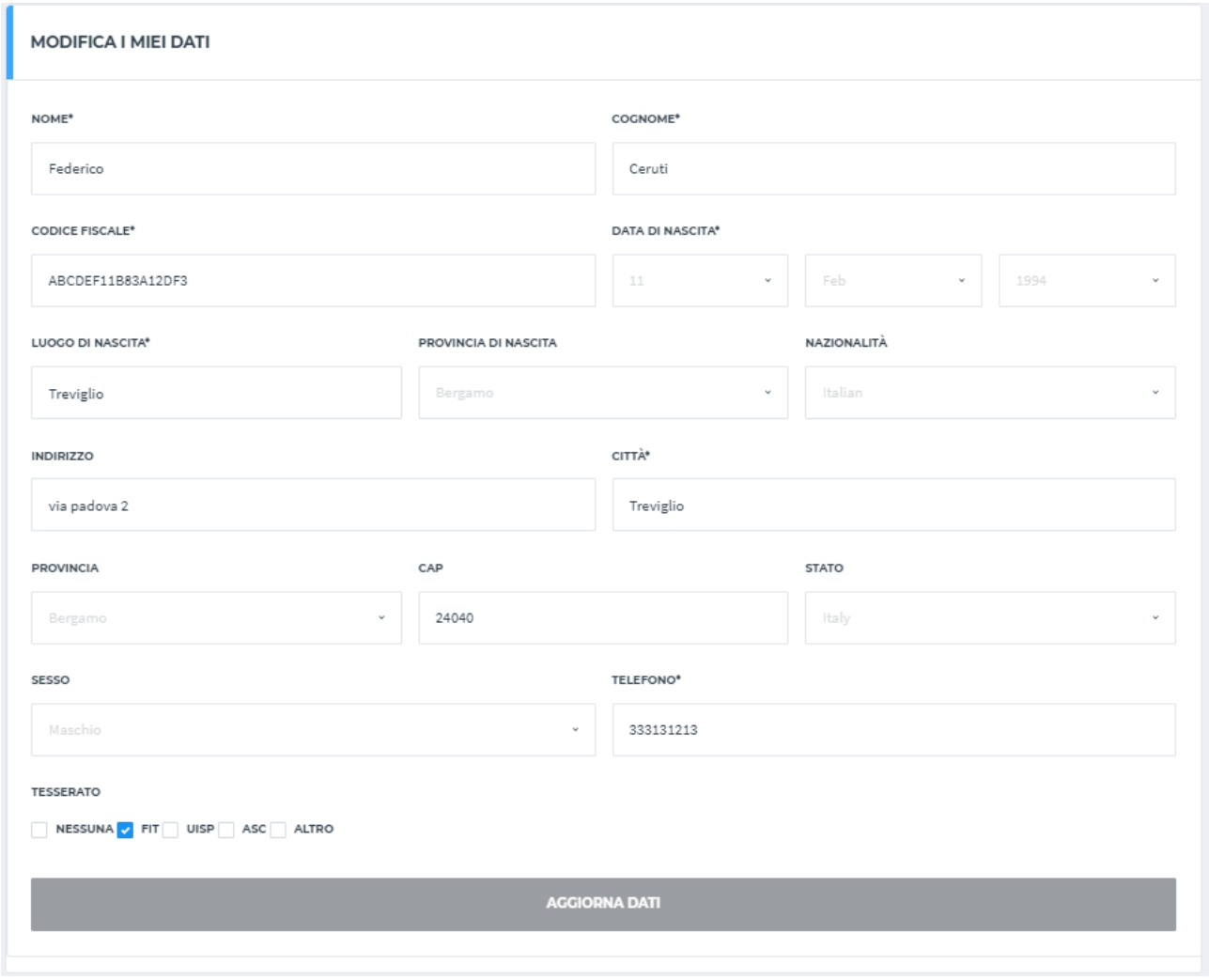
Once the result is sent, the challenge appears in the state of "waiting confirmation".

CONFIRM RESULT MODULE

In case the result is inserted by the opponent, I have to confirm or to contest the result. If the score is confirmed, it is registered in the database, otherwise, if it is contested, the question is managed by the club. *(not implemented)*

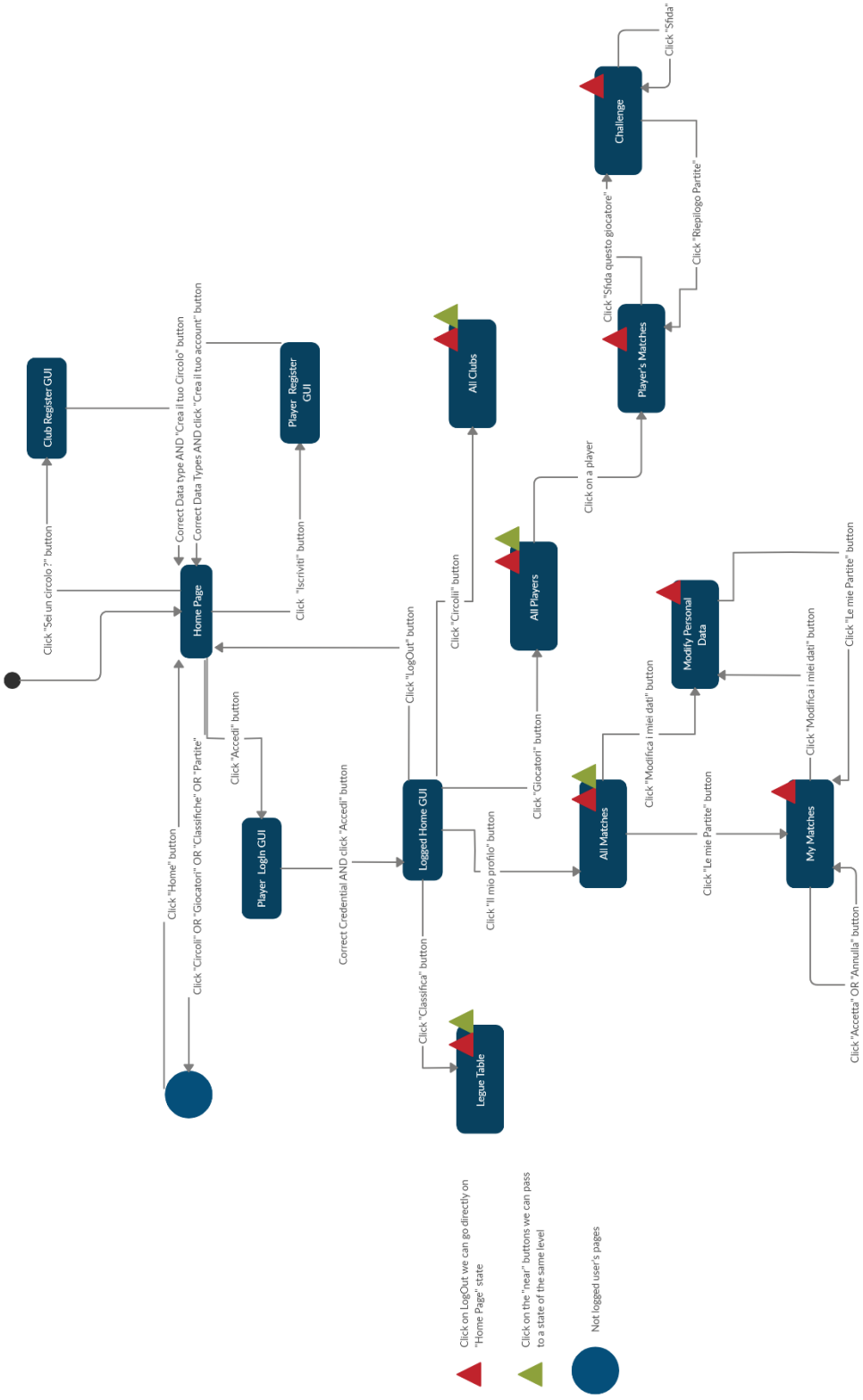
MODIFY PERSONAL DATA

This module is in the "my profile" section and allows to modify and update the personal data.



Finite State Machine

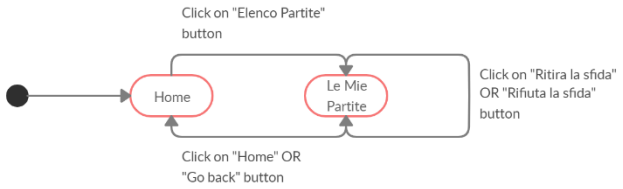
General Finite State Machine



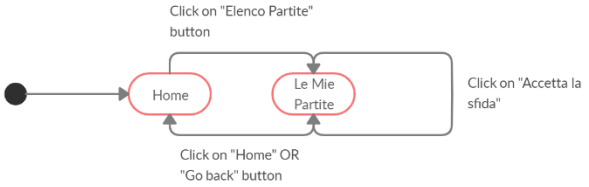
Challenge Instructions modelled with Finite State Machine

LAUNCH A CHALLENGE

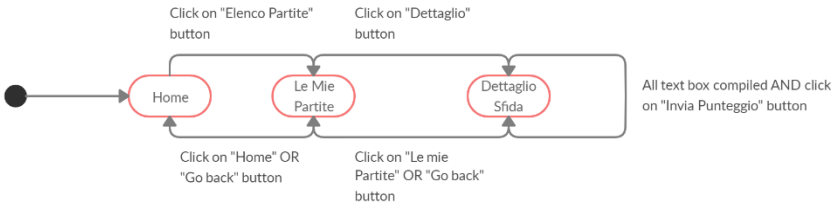
DELETE A CHALLENGE



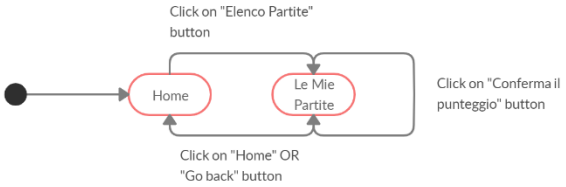
ACCEPT A CHALLENGE



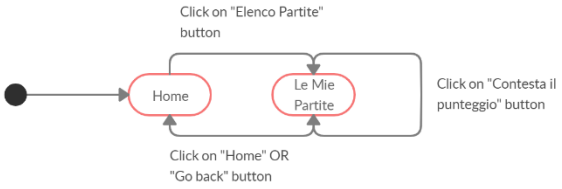
INSERT SCORE



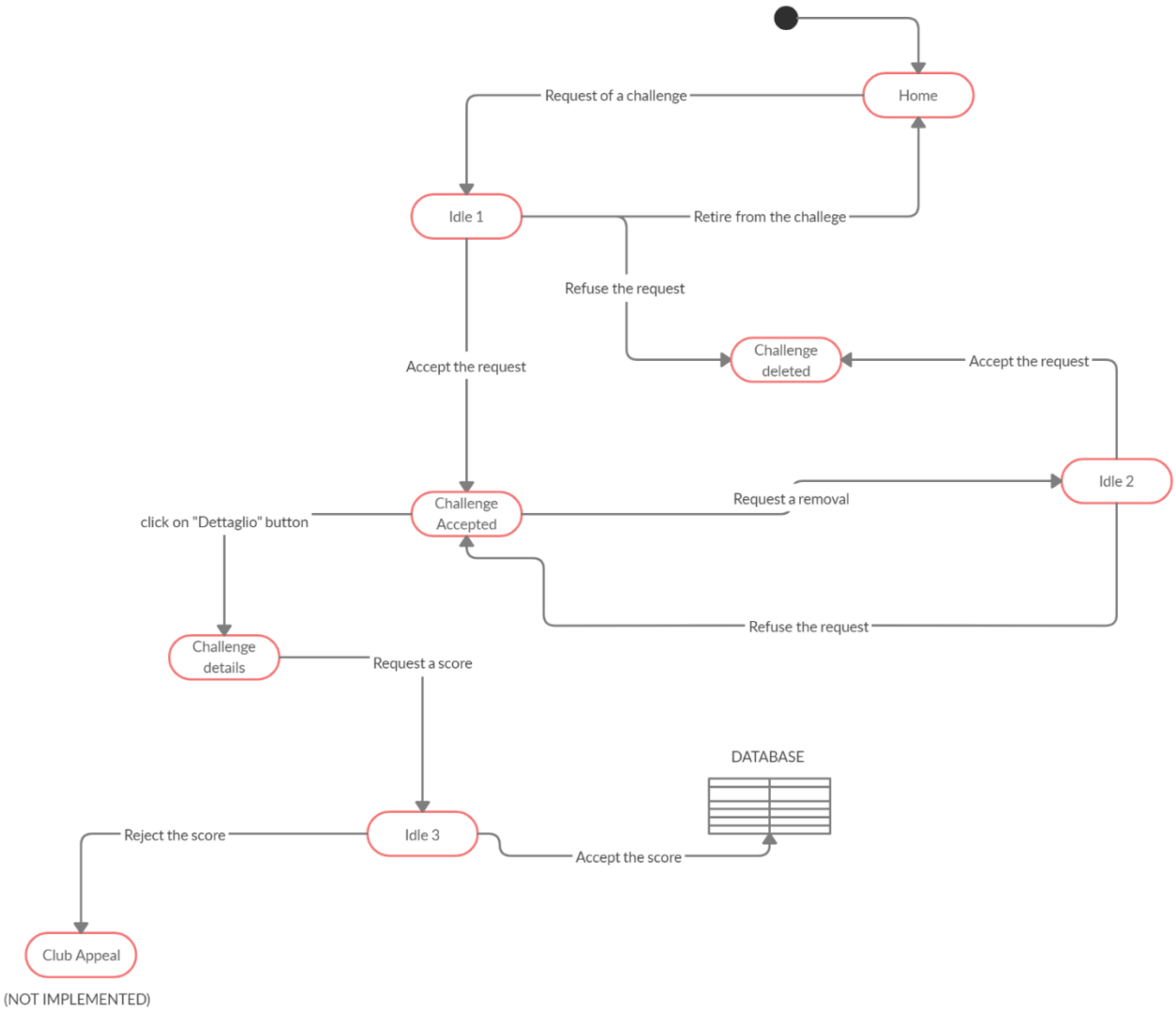
ACCEPT A SCORE



REJECT A SCORE



Challenge logical management modelled with Finite State Machine



TESTING

Introduction

In this chapter, the phase of testing is reported to evaluate the functionality of our web software application, with the intent to find whether it met the specified requirements or not and to identify if there is any defect. The test cases are organized per module, and more specifically in:

- Signing up module
- Logging in module
- Challenges module
- Chart module
- My Profile module

The testing phase is made on the final application, where all the modules were integrated into the system ready for the final user.

Modules

SIGNING UP MODULE

Requirements: The user inserts his name, his surname, an e-mail (not already existing in the database, duplicates are not allowed) and a password.

Test Case 1

<name ='Federico', surname='Ceruti', email='federico@sag.it', password='test'>

First a check is performed to avoid email duplicate, then, in the table USER in the database, a new row is created.

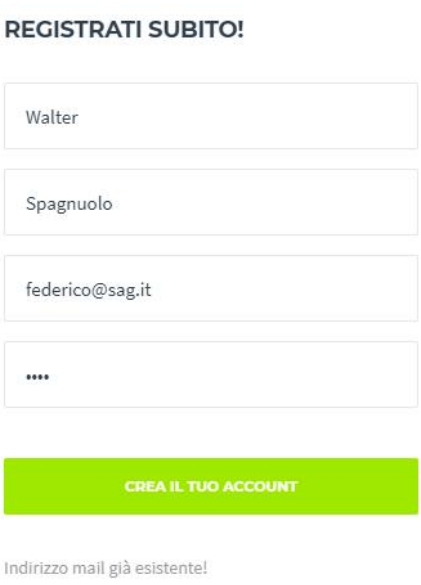
SIGN UP: SUCCEEDS

Test Case 2

<name= 'Walter', surname='Spagnuolo', email='federico@sag.it', password='test'>

The module show that the email is already existing, so it is not possible to proceed with the sign up.

SIGN UP: FAILS



LOGGING IN MODULE

Requirements: the user inserts his email and his password, registered during the signing up module, and passes from the logged out interface to the logged in interface.

Test Case 1

<email = 'federico@sag.it', password='test'>

LOG IN: SUCCEDS

Test Case 2

<email= 'federico', password= 'abcd'>

If the email or the password or both of them, do not correspond to those inserted during the signing up process, so they do not exist in the database, the user remains in the logged out interface.

LOG IN: FAILS

CHALLENGES MODULE

A logged user wants to send a challenge request

Requirements: the logged in user, from the list of the registered players can select one of them and, through the button "challenge this player", send him or to another player the challenge request through the button "send".

SEND CHALLENGE REUQEST: SUCCEEDS

Retire the challenge request

Requirements: the user that sends the challenge request is allowed to retire it before that the opponent accepts or refuse it. Through the button "Retire the challenge", the request is deleted automatically for both the users.

RETIRE CHALLENGE: SUCCEEDS

Accept or refuse the challenge request

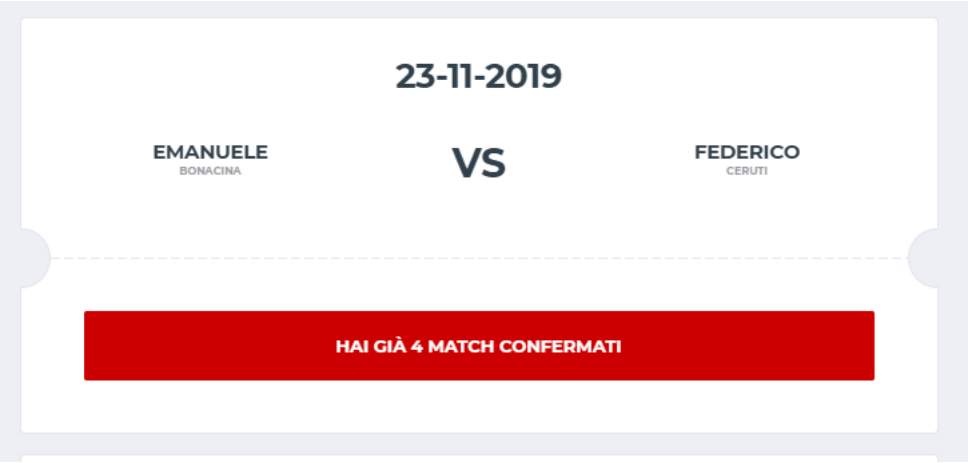
Requirements: the user that receives the challenge request can choose if accept or refuse the request. Through the button "accept" the challenge is created, through the button "refuse" the request is deleted automatically for both the users.

ACCEPT OR REFUSE CHALLENGE: SUCCEEDS

The user reaches 4 matches in a month

Requirements: if the user that sends the challenge request or the user that receives it has reached the 4 matches in that month, when the button "accept" is pushed, the message "you already have 4 confirmed matches"

ACCEPT CHALLENGE: FAILS



Challenge detail

REQUIREMENTS: Once the challenge is accepted, both of the users can enter in the "challenge detail" section, through the specific button, in which is possible to insert the day, the club and the result of the match.

CHALLENGE DETAIL: SUCCEEDS

Retire challenge

REQUIREMENTS: the main difference between the option "retire request" is that, here, the "challenge request" was already accepted, and the "removal request" can be sent by both the users. Through the button "retire challenge" the message "removal request sent" appears.

RETIRE CHALLENGE: SUCCEEDS

The removal request is sent to the opponent which has to accept or refuse it. If the request is accepted, the challenge is deleted

REMOVAL REQUEST ACCEPTED: SUCCEEDS

If the button "cancelling" is pushed, the request is refused, and the challenge remains in the section "outstanding challenges".

REMOVAL REQUEST REFUSED: SUCCEEDS

Select a club for the match

Requirements: through the button "select a club", one of the two users involved in the match can select the club in which the challenge will be played. To the user that selects the club, appears the list of the club available and that accepted the type of card that the 2 users have registered, and through this list he can choose the club that prefers.

Test Case 1

Both the 2 users have a type of card accepted by at least one of the available clubs.

SELECT CLUB: SUCCEEDS

Test Case 2

If only one of the 2 users, does not have a type of card accepted by the available clubs, no club appears in the selection list.

SELECT CLUB: FAILS

Insert result

Requirements: one of the 2 users involved in the match can insert the match result and send it to the opponent. Once the result is sent, the message "pending confirmation" appears, until the opponents do not confirm it.

INSERT RESULT AND SEND IT TO YOUR OPPONENT: SUCCEEDS

Confirm the result

Once the result is inserted by one of the 2 players, the opponent has to select if to confirm it or if to contest it. Through the button "confirm the result", the match is considered closed, is registered in the database and is added to the list of "past matches" of the two players. The button "contest the result" is not implemented, because should be managed by the club side.

CONFIRM THE RESULT: SUCCEEDS

A not logged user want to send a request challenge

Requirements: If a not logged user, try to select a player from the "players list" to send him a challenge request, the button "challenge this player" does not appear , but he can only read the statistics and the past matches of that specific player.

SEND CHALLENGE REQUEST: FAILS

CHART MODULE

Modify the chart

Requirements: when the result of a match is inserted, the winner player gains 3 points, the loser one gains 1 point, while, in case of drawn, each player gains 2 points. Consequently, the chart changes and the also the statistics of the players involved in the match.

CHART MODIFICATION: SUCCEEDS

MY PROFILE MODULE

Match summary

Requirements: the logged user, clicking the button "my profile" can access to his account's page. In this page I can read the summary of his matches, with their details like the club in which they were played, the results and the opponents.

MATCH SUMMARY: SUCCEEDS

Modify my data

Requirements: through the button "modify my data" the user can insert, modify and update his personal data. Once the modification is made, it is applicable by pushing the button "update my data". When the updating happens the message "update successful" is shown.

TEST CASE 1: all the fields identified with the star (*) are filled, the update happens with success.

MODIFY DATA: SUCCEEDS

TEST CASE 2: one of the fields identified with the star (*) is left empty, the message "complete all the fields" appears.

MODIFY DATA: FAILS

CONCLUSIONS

Since the software has been developed in a modular way, it should be extended in the future by adding some functionalities not implemented in our project. The idea is to insert the possibility of creating tournaments, and of subscribing to them. Another addition regards the introduction of the ranking of each player. The software will be developed also from the clubs' side that should be able to manage the matches, introducing the availability of the courts in a certain slot time, the booking of courts and services, and the handling of the dispute of the scores.

During the development of our web-App, we realized the importance of the role covering by the requirements and specifications since only having clear the objectives of the application it was possible to implement it completely.

Big relevance was given also to the testing part, exploited to verify the correct functioning and behaviour of the developed software.

The creation of "Set a Game" has been useful to apply, in a practical way, some theoretical knowledges learned during the course, and to understand and to face up some problems that may appear during the implementation of a web application.