**PROGETTO ISPW**

**ANNO 2021/2022**

**Italian Ballerz**

1. **Introduction** 
   1. Aim of the document
      1. Overview of the defined system
      2. Operational settings
      3. Related systems (at least 2), Pros and Cons.
   2. User Stories (3 per member)
   3. Functional Requirements (3 per member)
   4. Use Cases: Overview Diagram (1)
2. **Storyboards** 
   1. 2 screens per member, covering all the functionalities described in SRS, developed using Draw.io or similar
3. **Design** 
   1. Class diagram:
      1. 1 VOPC per member.(analysis)
      2. 1 design-level diagram per member (e.g. that includes patterns, or specific solutions that improve the engineering level of the system)
   2. Design patterns: 1 different pattern per member. Possibly try to apply the pattern within the context of the project.
   3. Activity diagram: 1 per member.
   4. Sequence diagram: 1 per member.
   5. State diagram: 1 per member.
4. **Testing** 
   1. Develop at least 3 test cases per person. In each test (class) file, please report (via Java comments) the name of the person in charge.
5. **Code** 
   1. Similar functionality implemented as Desktop (JavaFX) + DB or File System.
   2. Exceptions: at least 2 per member (do not just catch and back-propagate the exceptions, but properly handle them. Possibly define your own error logic by means of exceptions)
   3. Be able to show that Svn(or Git) + SonarCloud is correctly installed in one of your computer and it is able to analyze your project for rule violations. No rule must be violated.
6. **Analytics**

6.1 Provide a Proces Control Chart as explained in the slides.

1. **Video**

7.1 A 1 to 2 minutes recorded video of the developed system performing the expected functionalities.

1. **Introduction**

1.1 Aim of the document

Lo scopo di questo documento è quello di esplicitare i vari passi della progettazione nella realizzazione del software Italian Ballerz, come progetto per l’esame di Ingegneria del software.

* + 1. Overview of the defined system

Il software realizzato si occupa della gestione delle prenotazioni per campi da basket. Inoltre, si vuole far avere un account all’utente in grado di tener traccia delle proprie attività, come ad esempio permettere all’utente di salvare le proprie statistiche.

* + 1. Operational settings

Per lo sviluppo del software ho usato:

-Figma.com per ottenere il codice HTML delle storyboard.

-Star UML per la realizzazione dei diagrammi relativi alla progettazione del sistema

-Scene Builder per la creazione di file FXML per realizzare l’interfaccia grafica (e quindi le view del MVC)

-JavaFX per gestire e ampliare i file FXML (controller grafico del MVC)

-Intellij e in particolare Java come IDE e ambiente di programmazione. Per utilizzare il software si deve estrarre la cartella ItalianBallerz dal file zip e aprirla come progetto in Intellij. Nella cartella è già presente il file .idea per configurare come fare il run dell’applicazione. Se questo non dovesse essere possibile la classe con la responsabilità di avviare l’applicazione è MainInterface.java .

* + 1. Related systems

App correlate CourtFinder e In a pickle

* 1. User stories
* **As a user, I want to search for basketball court next to me, so that I can play with other people. (Implementato)**
* **As a registered user, I want to track my basketball statistics, so that I can see if I’m improving or not. (Implementato)**
* **As a registered user, I want to log in, so that I can enter my profile. (Implementato)**
* As a user, I want to inform when I play in a specific court, so that everyone could know if the court is not empty and join me. (Non implementato)
* As a user, I want to review a court, so that everyone knows which court is better. (Non implementato)
* As a not so skilled basketball player user, I want to find other players at my same level, so that I will have fun while playing. (Non implementato)
  1. Functional requirements

- **The system shall provide a list and a map on which are located the basketball courts in a city of choice. (Implementato tranne mappa)**

- **The system shall provide registration using an email, a password and a username. (Implementato)**

- **The system shall provide to insert the statistics of a registered user:**

* 1. **Points**
  2. **Rebounds**
  3. **Assists**
  4. **Minutes**

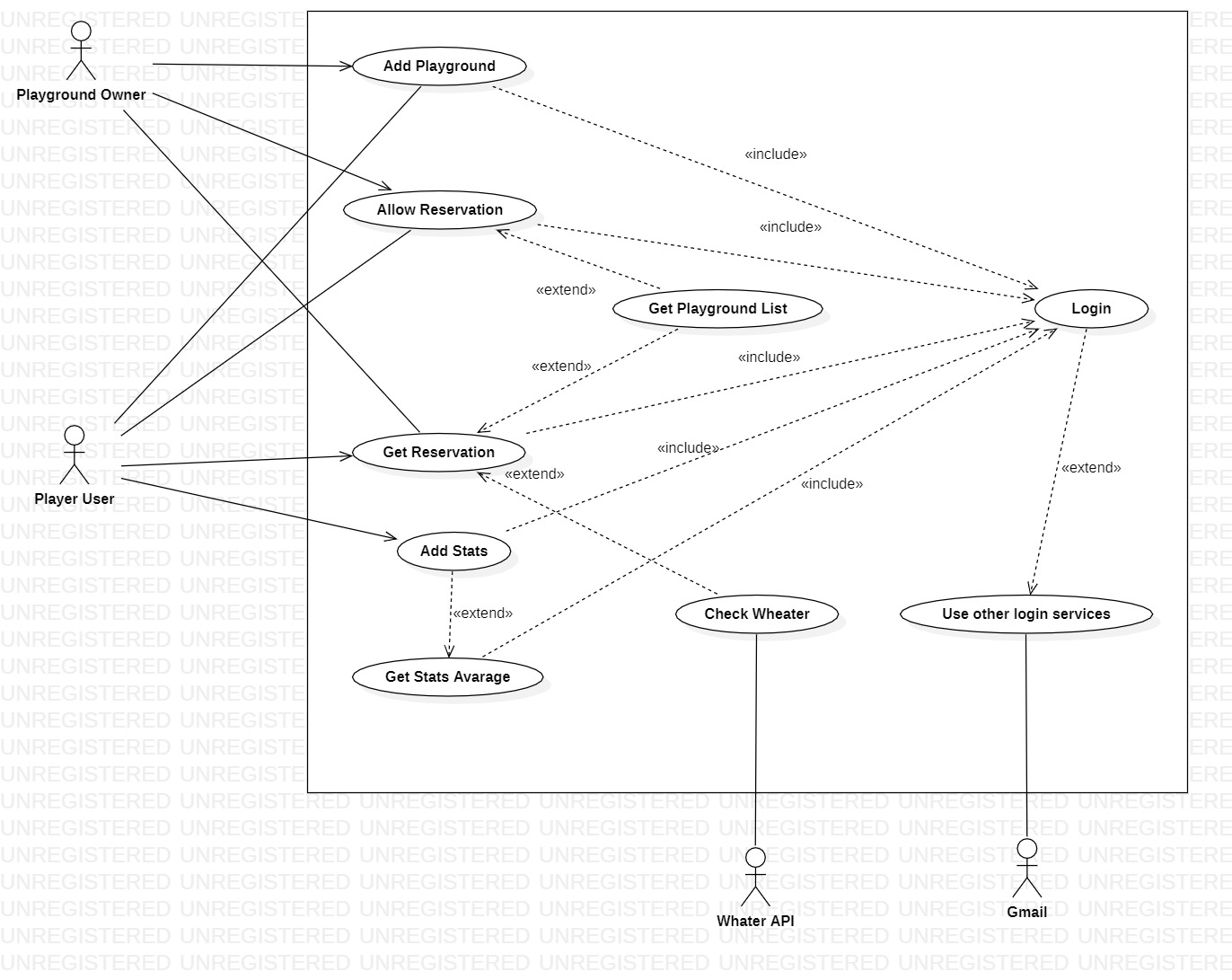
**(Implementato)**

- The system shall provide log in using a password and a username. (Implementato)

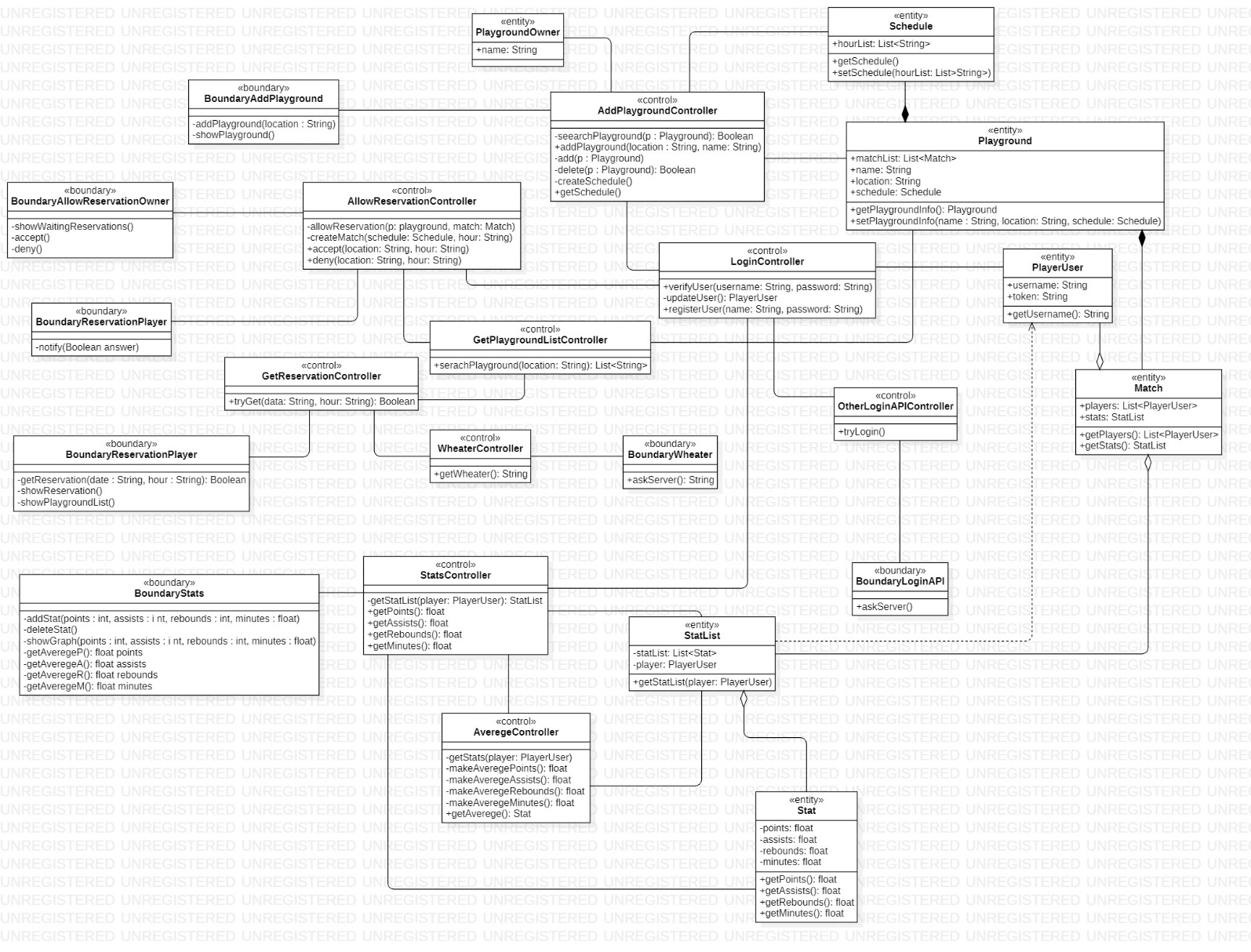
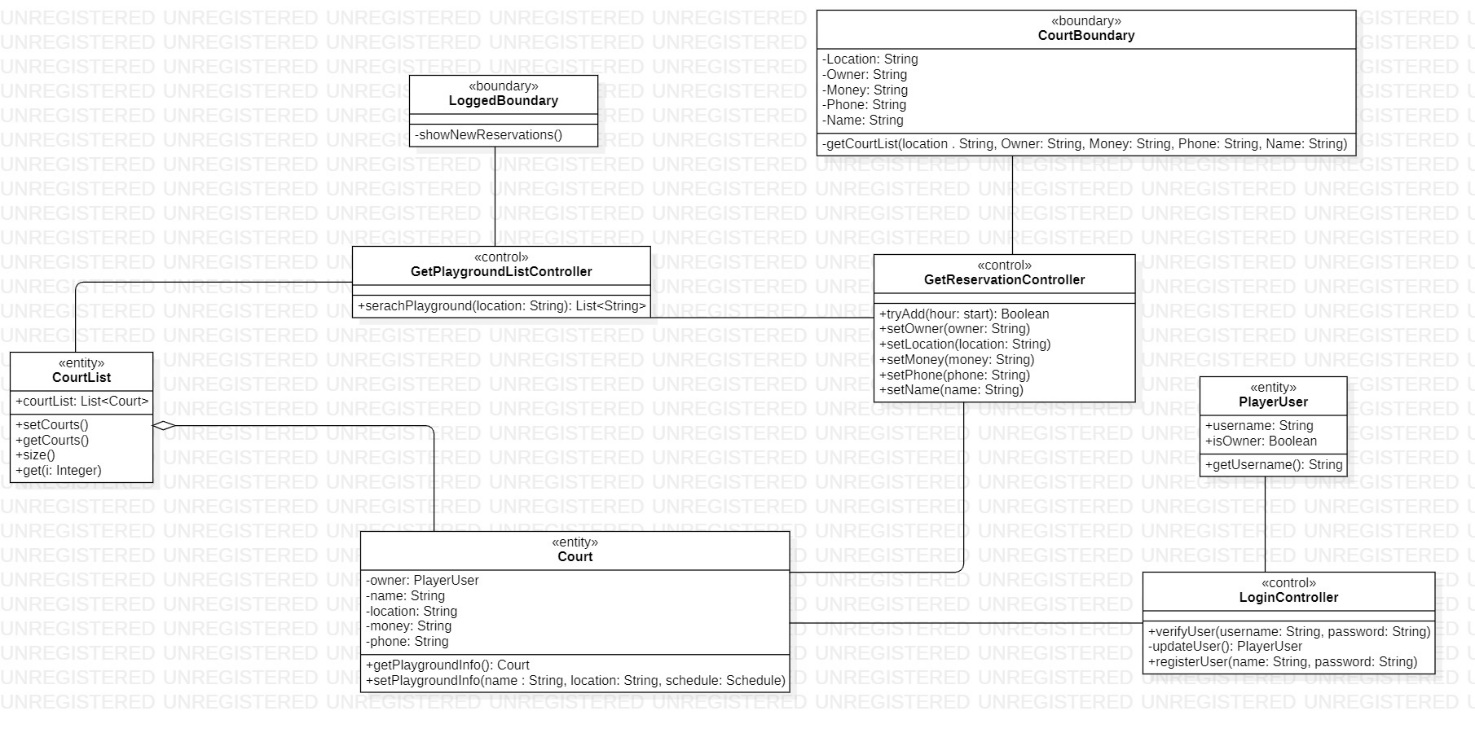
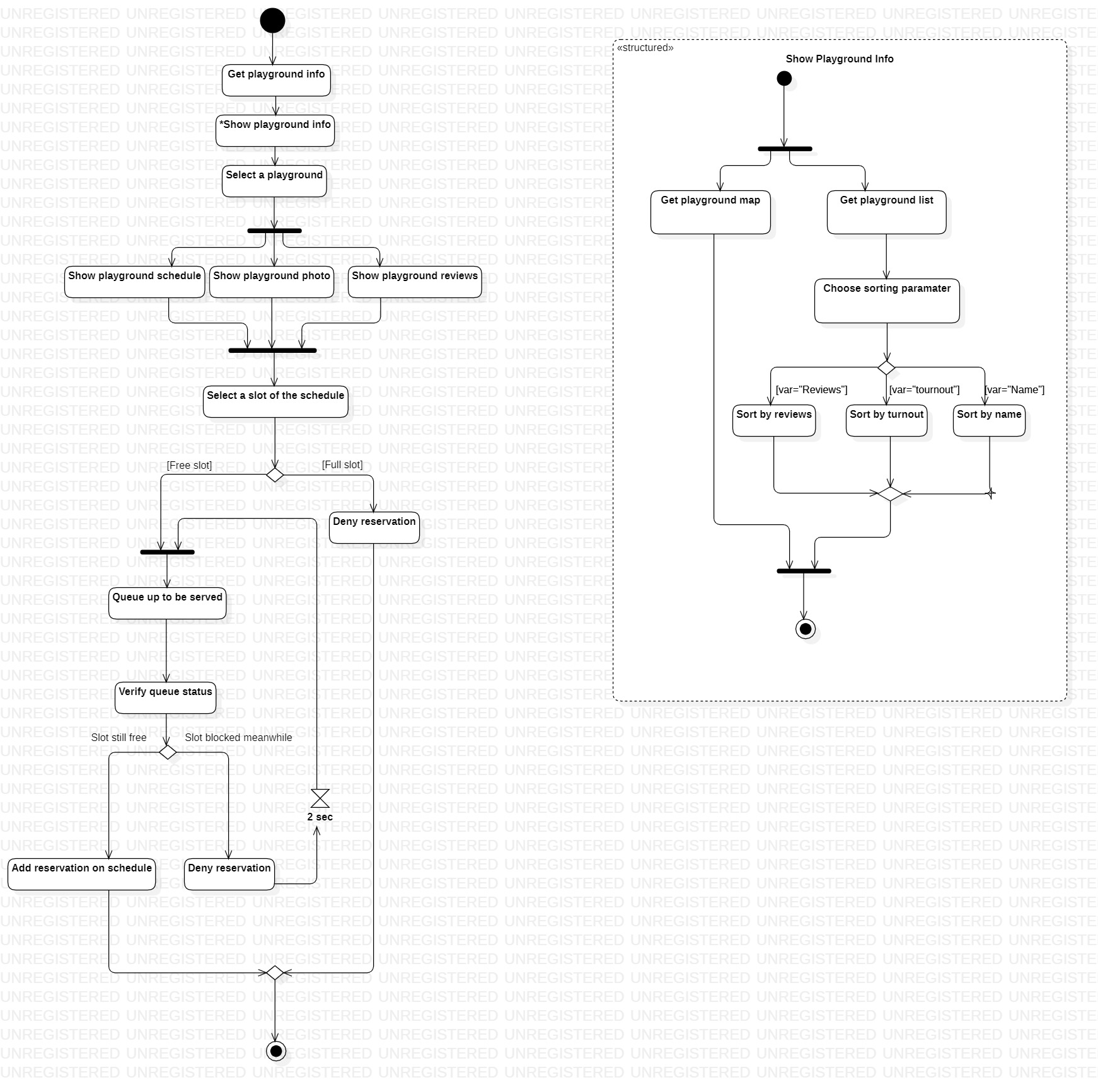
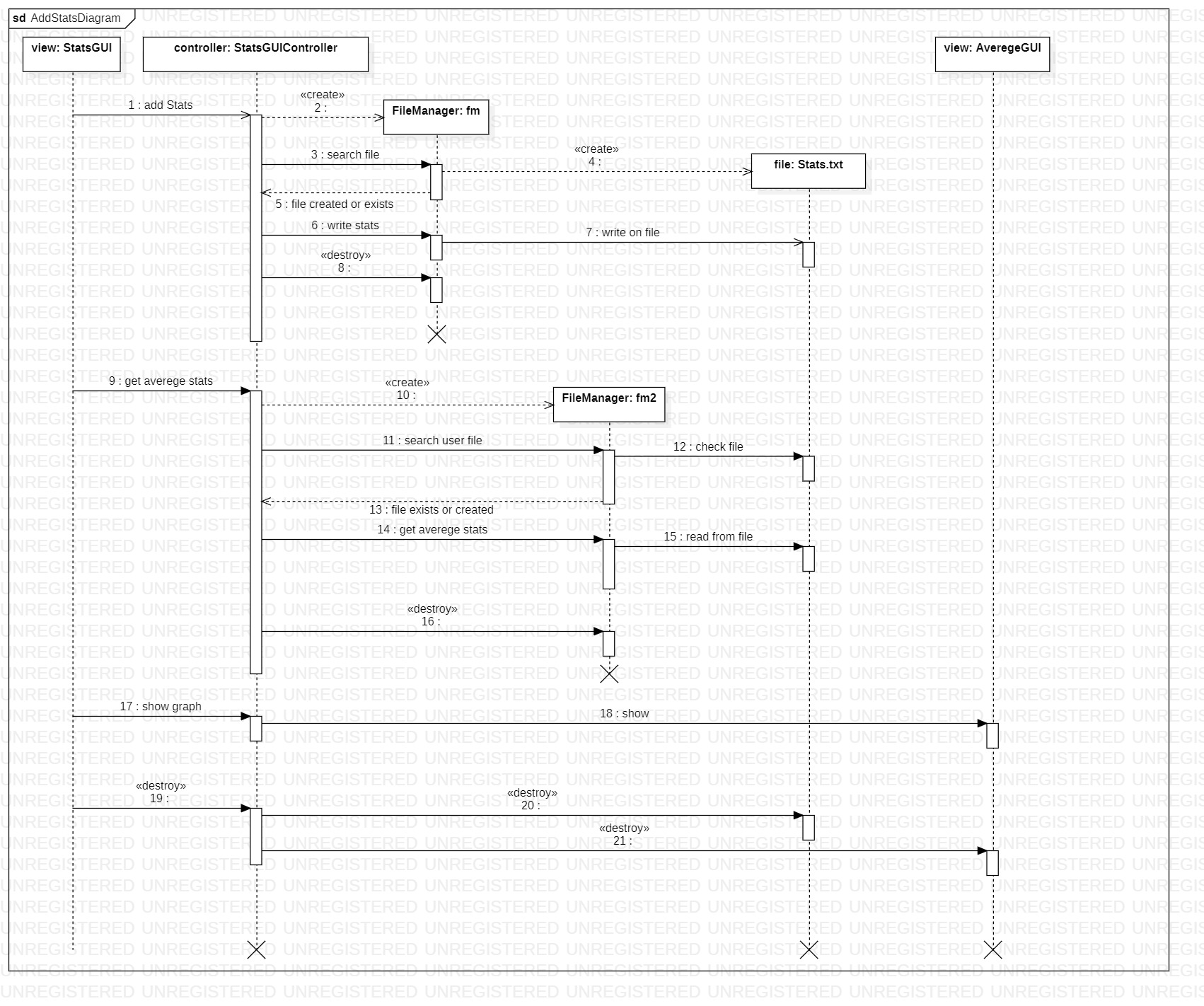
- The system shall provide rated reviews, using a rating scale based stars, from 1 star up to 5. (Non implementato)

- The system shall provide a schedule for each registered court to be shown to both registered and guest users. (Implementato)

1.4 Use case overview diagram



1. Immagine che contiene testo

   Descrizione generata automaticamente**Storyboard** (Il file HTML si trova nella cartella StoryBoard)
2. **Design**
   1. Class diagram
      1. VOPC (BCE class diagram)
      2. Design-level diagram (MVC diagram)
   2. Design pattern
   3. Activity diagram
   4. Sequence diagram
   5. State diagram