

Si consideri lo studio di un sistema software client-server in grado di gestire un circolo velico con le seguenti caratteristiche:

- ogni socio è memorizzato tramite il nome, cognome, indirizzo e codice fiscale;
- ogni socio può possedere una o più imbarcazioni, può aggiungerne e rimuoverle. Ogni barca ha un nome, ma, visto che imbarcazioni di diversa proprietà possono avere lo stesso nome, hanno anche un identificativo univoco assegnato dal circolo;
- ogni socio deve pagare una quota di associazione che ha una durata annuale;
- ogni socio deve pagare una quota per il rimessaggio per ogni sua imbarcazione. Anche questa quota ha una durata annuale ed è calcolata in base alla lunghezza delle imbarcazioni;
- il circolo organizza delle gare tra le imbarcazioni dei suoi soci e ogni socio può iscrivere un'imbarcazione pagando una quota di iscrizione.
- Soci e personale del circolo possono collegarsi al server, tramite nome utente e password, in concorrenza attraverso i client.

In particolare, il sistema software verrà utilizzato dal personale del circolo per:

- Gestire le informazioni sui soci, sulle imbarcazioni e sulle gare;
- Inviare le notifiche che informano i soci che devono rinnovare la quota di associazione e/o le quote di rimessaggio;
- Mantenere la traccia dei pagamenti di associazione, rimessaggio e iscrizione alle gare.

Tutti i pagamenti devono essere effettuati attraverso il servizio di pagamento offerto dal sistema di gestione del circolo che accetta carte di credito e ricevute di bonifico.

Partendo dalla precedente descrizione è necessario:

1. definire i requisiti funzionali e il diagramma UML dei casi di uso;
2. definire il diagramma UML delle classi evidenziando gli attributi delle classi, le associazioni e la loro molteplicità. In particolare, il diagramma deve presentare una vista ad alto livello del sistema (e.g., non deve includere dettagli implementativi come le classi grafiche che definiranno la GUI).
3. la definizione della descrizione testuale/tabellare e dei corrispondenti diagrammi UML di sequenza dei seguenti casi di uso: il rinnovo della quota di associazione, l'aggiunta di una nuova imbarcazione e l'iscrizione ad una gara;

Notare che:

- la documentazione per i punti 1-3 deve essere rappresentata da un documento PDF;
- il codice java deve essere suddiviso in opportuni package e commentato con javadoc;
- Il client deve fornire un'interfaccia grafica sviluppata con Javafx.
- I dati devono essere mantenuti tramite un database SQL;
- Le principali classi devono essere testate con JUnit e se necessario con Mockito;
- il codice del sistema deve essere accompagnato, oltre che dal Javadoc, da un documento PDF (manuale) che descrive come eseguire le applicazioni sviluppate (i.e., quali classi hanno il compito di eseguire un'applicazione (i.e., hanno un main) e quali argomenti devono eventualmente essere passati per permettere una esecuzione corretta, e quali azioni deve fare l'utente per "guidare" la loro esecuzione.

**il progetto va consegnato entro tre mesi dall'assegnamento.**