

Corso di laurea in Informatica - Dipartimento di Informatica Università di Pisa

Progetto – appello 3 2021

IS: Consegna entro il 12 luglio 2021 ore 23:59

BD: Consegna entro il 17 luglio 2021 ore 23:59

Introduzione comune per i corsi di IS e BD

EasyRegatta

Si vuole realizzare EasyRegatta, Sistema nel seguito, un sistema software di supporto agli organizzatori e ai partecipanti alle regate veliche, in particolare di regate di flotta in cui barche di dimensioni diverse gareggiano insieme, usando poi un meccanismo di compenso, basato sui ratings delle barche, per redigere la classifica.

L'Autorità Organizzatrice deve pubblicare un bando di regata che indica dove e quando l'evento si svolgerà, quali classi sono ammesse e le condizioni per partecipare. Il Sistema memorizza il bando che diventa accessibile a tutti consultando un catalogo di bandi. Il bando di regata può essere modificato, in questo caso il Sistema aggiorna il catalogo e invia una notifica a tutte le barche che si erano iscritte prima della modifica.

L'Autorità Organizzatrice deve inoltre nominare un Comitato di Regata. Il Comitato di Regata deve pubblicare le istruzioni di regata, che comprendono la definizione del sistema di compenso, e poi deve condurre la gara. Infine, il Comitato di Regata deve calcolare la classifica della regata e ha la responsabilità di esaminare e decidere le *proteste* (mancate precedenza, boe toccate, motori accesi...). Le decisioni comprendono “protesta invalida” oppure penalità in termini di posizione in classifica o la squalifica.

Il Sistema deve permettere l'iscrizione a barche ed equipaggio, memorizzando per ogni barca: nome, rating e numeri velici, per ogni membro dell'equipaggio: nome, numero tessera FIV, certificato medico.

Il Sistema deve gestire le comunicazioni pre-partenza in modo che le comunicazioni arrivino agli equipaggi (senza dover più usare le bandiere e i segnali acustici della Barca Comitato). Verranno mandati i seguenti segnali: *Avviso*, 5 minuti prima della partenza, *Ultimo minuto*: 1 minuto prima della partenza, *Partenza*.

Il Sistema deve permettere anche le comunicazioni dalle imbarcazioni verso il Comitato di Regata e verso le altre imbarcazioni per segnalare le proteste.

Il Sistema deve memorizzare i tempi di passaggio alle boe (al solo fine informativo per valutare le prestazioni delle barche) e all'arrivo (per stilare la classifica finale) e supportare il Comitato di Regata per calcolare la classifica tenendo conto dei rating delle imbarcazioni.

I passaggi in boa vengono comunicati dagli equipaggi, assieme ai numeri velici della barca che precede e della barca che segue, ove possibile. Il tempo all'arrivo è preso invece dall'equipaggio della Barca Comitato.

Ogni regata ha un tempo limite deciso dal Comitato di Regata. Le imbarcazioni che non arriveranno entro il tempo limite saranno considerate “non arrivate”. E’ possibile (decisione del Comitato di Regata) tenere conto del tempo di passaggio a una data boa per la classifica delle barche che arrivano oltre tempo limite. Le imbarcazioni che sono passate dalla boa indicata e che non hanno tagliato la linea di arrivo entro il tempo limite, saranno comunque poste in classifica in coda all’ultimo arrivato, in base all’ordine (compensato) di passaggio alla boa. In tal caso ci sarà una Barca Comitato anche in pressi della boa per prendere i tempi.

Le proteste dovranno essere inserite entro 60 minuti dall’arrivo dell’imbarcazione protestante.

Una barca che si ritira dovrà darne comunicazione al più presto, tranne che in una situazione di emergenza.

Una barca che non parta entro 30 minuti dal proprio segnale di partenza sarà classificata “Non Partita” senza possibilità di protesta.

Progetto di Ingegneria del Software

1. Descrivere con un diagramma UML tutti i casi d’uso del Sistema. Per uno di essi dare la narrativa.
2. Dare un diagramma di sequenza che descriva un caso d’uso non banale.
3. Dare un diagramma delle classi che modelli il dominio del Sistema.
4. Dare un diagramma di macchina a stati che modelli gli stati che può avere un’imbarcazione relativamente a una regata.
5. Dare un diagramma di attività che modelli il processo di indizione e conduzione di una regata.
6. Definire una possibile architettura (dare sia la vista C&C che di dislocazione) per realizzare il Sistema.
7. Si assuma di usare il metodo *Time on Distance* (spiegato di seguito) per calcolare il tempo compensato e quindi definire il vincitore. Il rating indica la prestazione media di una imbarcazione per differenti intensità di vento e angoli, cioè la velocità media espressa in secondi necessari a percorrere un miglio. Questo valore, moltiplicato per il numero di miglia percorse, dà il tempo che va sottratto dal tempo reale per avere il tempo compensato (su cui si basa la classifica).

$$\text{Tempo Compensato} = \text{Tempo reale} - (\text{Rating} * \text{Distanza})$$

Progettare una batteria di test e un driver di test per testare il metodo `calcolaVincitore` fornito di seguito. Specificare quali criteri sono stati usati.

```
static String calcolaVincitore(Report[] rep, int distanza){
    float min = 10000;
    String vincitore = null;
    float f;
    for (int i=0; i<rep.length; i++){
        f = rep[i].tempoReale - rep[i].rating*distanza;
        if (f < min) {
            min=f;
            vincitore = rep[i].nome;
        }
    }
    return vincitore;
}
```

Dove la classe Report è definita come segue:

```
class Report{
    String nome;
    float rating;
    int tempoReale;
    Report(String s, float r, int t){
        nome = s;
        rating = r;
        tempoReale = t;
    }
}
```

Regole di consegna del progetto di Ingegneria del Software

Oltre alle informazioni pubblicate su didawiki (e replicate sul sito valutami):

1. Deve essere consegnato un archivio contenente:
 - a. il file .vpp del progetto (un file unico e non tanti file spezzettati)
 - b. un file pdf che possiamo stampare per correggere più velocemente i progetti.
Deve contenere: nome cognome e matricola dei membri del gruppo; le parti testuali del progetto; i diagrammi prodotti, rispettando l'ordine degli esercizi dati.
2. Sia i file che l'archivio devono chiamarsi col cognome del referente del gruppo (chi fa la sottomissione del progetto).
3. Il progetto deve essere inviato per email alla propria docente entro la data fissata: roberta.gori@unipi.it laura.semmini@unipi.it
4. Per correttezza nei confronti dei compagni di gruppo, chi sottomette mette in CC tutti i membri del gruppo.
5. La mail deve avere subject: Progetto IS_P3_easyregatta
6. Tutte le mail (mittente e destinatari) devono essere istituzionali (@unipi.it / @studenti.unipi.it).