

## Compito di Introduzione alla Programmazione Web – 15 luglio 2021

Una WebApp implementa un sistema di chat tra utenti. Si sviluppi il codice usando il pattern MVC.

Per accedere alle pagine del sito gli utenti devono autenticarsi. Accessi di utenti non autenticati devono essere ridiretti alla pagina di login.

### Autenticazione

1) L'autenticazione avviene chiedendo all'utente username e password, preinizializzate con le seguenti coppie username – password:

- admin – admin
- user1 – pw1
- user2 – pw2

*Esempio di pagina di login:*

### Inserisci le tue credenziali per accedere

2) la verifica delle credenziali avviene tramite accesso a un DB (Derby) che deve:

- chiamarsi MyDerbyDB,
- essere memorizzato all'interno del progetto Netbeans,
- essere creato con user WEBENGINE e pw WEBENGINE
- avere una tabella USERS con campi username e password.
- Essere preinizializzato con i tre utenti menzionati

**NOTA: in prima battuta, l'autenticazione può essere fatta solo in memoria senza usare un DB. Si consiglia di lasciare l'utilizzazione del DB a fine sviluppo, se avanza tempo.**

3) Tutte le pagine, eccetto quella di autenticazione, hanno in alto un banner con il nome dell'utente e un bottone "Log out". Questo bottone fa sì che il collegamento dell'utente scada, e che l'utente stesso sia riportato alla pagina di autenticazione.

### La pagina utente

4) La pagina utente mostra una lista di stanze (inizialmente vuota) ed un bottone che permette di aggiungere una stanza, dandole un titolo che riflette l'argomento a cui la pagina è dedicata.

*Esempio di pagina utente inizialmente senza stanze:*

user3

---

**Nessuna stanza disponibile al momento**

---

### Aggiungi una nuova stanza

Esempio di pagina utente con stanze esistenti:

user3

---

## Entra in una delle stanze disponibili:

- [Campionati Europei](#)
  - [Tour de France](#)
- 

## Aggiungi una nuova stanza

Se vi sono stanze disponibili, cliccando sul nome di una di esse si passa alla relativa pagina.

Quando si aggiunge un nuova stanza, appena creata si entra subito nella sua pagina

### Le pagine delle stanze

- 5) Una stanza raccoglie messaggi su un tema, e la sua pagina li mostra in ordine cronologico.
- 6) Ciascun messaggio è composto di tre parti: nome dell'utente che ha scritto il messaggio, il testo del messaggio e il timestamp di creazione del messaggio.
- 7) In fondo alla pagina della stanza un campo di testo permette di aggiungere un nuovo messaggio alla stanza.
- 8) La pagina di una stanza si ricarica automaticamente ogni 10 sec., e/o manualmente con il tasto "refresh page"

user3

---

## Sei nella stanza "Campionati Europei"

1. user1 : Che finale! (Mon Jul 12 13:10:56 CEST 2021)
  2. user3 : Sì, grande squadra! (Mon Jul 12 13:11:20 CEST 2021)
- 

## Inserisci un nuovo messaggio

---

La pagina si ricarica automaticamente ogni 10 sec, ma se vuoi puoi ricaricarla manualmente:

---

[Esci dalla stanza](#)

- 9) Il ricaricamento della pagina(automatico e manuale) è inibito se l'utente sta scrivendo (ovvero se il campo "Inserisci nuovo messaggio" non è vuoto).
- 10) Un link "esci dalla stanza" permette di tornare alla pagina utente.
- 11) Non è prevista persistenza per stanze e messaggi, e quindi le une e gli altri scompaiono al riavvio della webApp.

12) Se vi avanza tempo, formattate il sito tramite css o bootstrap.

13) In una pagina README.html scrivete eventuali comunicazione o commenti per il docente.

## Suggerimenti

Si consiglia di iniziare lo sviluppo definendo gli oggetti di business (Stanza, Messaggio) e i relativi insiemi.

Attenzione: spazi nel nome della stanza potrebbero darvi problemi di gestione se usate il nome nelle url.

Potete in prima battuta evitare di usare spazi (e segnalarlo nel README.html), ma se avete tempo potete raffinare usare l'encoding, ad esempio con il seguente codice:

```
String getEncodedTitle(String title) {
    String encodedTitle=null;
    try {
        encodedTitle=URLEncoder.encode(title, "UTF-8");
    } catch (UnsupportedEncodingException ex) {
        ex.printStackTrace();
    }
    System.out.println(title+"<->" + encodedTitle);
    return encodedTitle;
}
```