

Specifiche del progetto

Il progetto finale richiede lo sviluppo di una applicazione web per la creazione e gestione di blog personali. Le informazioni su questi blog sono memorizzate in un database collegato al servizio web. La web application permette all'utente di registrarsi e di loggarsi per poter usufruire dei servizi interni.

FASE DI REGISTRAZIONE:

L'utente si deve registrare specificando almeno nome utente, email e password. Nel caso in cui la registrazione vada a buon fine, l'utente potrà entrare nel servizio tramite un'apposita schermata di login.

LOGIN E UTILIZZO DELLA PIATTAFORMA:

Un utente loggato può accedere ai propri dati personali, ovvero quelli inseriti in fase di registrazione. E' prevista la possibilità di passare ad un servizio **premium**, il quale genera una classe di utenti che hanno delle potenzialità in più rispetto ai normali utenti. Questi utenti *premium* hanno accesso a tutte le funzionalità degli utenti regolari, più alcune funzionalità dedicate solo a loro. Le funzionalità dedicate *sono a scelta dello studente* e devono essere significative e possono ad esempio riguardare il numero di blog che si possono creare o i formati grafici a disposizione o l'ordine in cui i blog appaiono nella ricerca.

La funzionalità centrale dell'applicazione è quella di creare e gestire blog, quindi dopo la login l'utente deve poter creare uno (o più) blog personali i quali devono avere almeno un titolo, una descrizione e opzionalmente un'immagine. Al blog è associata una **categoria**, ovvero quale tema tratta il blog (esempio: cucina, sport). Ogni categoria può avere una o più **sottocategorie** (esempio: la categoria "cani" e la categoria "gatti" possono essere sottocategorie della categoria "animali").

Una volta creato un blog, deve essere prevista la funzionalità di pubblicazione di contenuti (es. post) sul blog stesso. Ogni contenuto è composto da titolo, data e ora, testo, e, se l'utente lo richiede, almeno un'immagine. È possibile prevedere più immagini. Tutti gli utenti registrati al servizio possono inoltre interagire con i contenuti creati, mediante dei commenti e tramite feedback (esempio: "mi piace" di Facebook, oppure up-/down-vote di Reddit).

Per ogni blog, il creatore può specificare anche uno o più utenti che faranno da co-autori, ovvero utenti che sono abilitati a creare post per un blog nonostante non ne siano i creatori. Tali utenti possono soltanto creare contenuti ed eliminare quelli da loro creati, ma non possono cancellare i contenuti prodotti dal creatore e non possono cancellare il blog.

Deve essere anche implementata una funzionalità di ricerca per blog e contenuto tramite parole che compaiono nel titolo (blog e contenuto), categoria, nome (co-)autore (blog) o testo. Funzionalità aggiuntive rispetto a quelle riportate nelle specifiche sono importanti e vanno ad impattare positivamente sulla valutazione finale.

Ulteriori vincoli:

- Su tutti i dati inseriti dagli utenti deve essere possibile eseguire le operazioni [CRUD](#): Create, Read, Update, Delete;
- Devono essere implementati controlli sugli input degli utenti, ad esempio: la stringa passata come email è formattata come una email;
- Il progetto deve essere sviluppato utilizzando i linguaggi visti a lezione (PHP, HTML, JS, CSS). NO FRAMEWORK PHP;
- Deve essere usato AJAX.

Se il progetto viene svolto in coppia: il carico di lavoro deve essere adeguato ad un lavoro in coppia. Le funzionalità descritte sopra sono quelle base per un lavoro di una singola persona. Di conseguenza, un lavoro in coppia comporta l'aggiunta di almeno una funzionalità aggiuntiva, una grafica più curata e un sistema di ricerca più sofisticato.

La relazione

Il progetto deve essere accompagnato da una relazione scritta che illustri il progetto. Nella relazione è utile spiegare quali sono state le scelte fatte e la motivazione che ha portato a farle.

La relazione dovrà contenere il diagramma E-R, lo schema logico, ed una discussione riguardo eventuali ridondanze presenti nel database. La relazione non può contenere screenshot dell'applicazione, nè screenshot del codice.

Modalità di consegna

La consegna deve consistere di 3 file:

1. un folder contenente tutti i file da copiare nella directory httpdocs;
2. L'esportazione del database tramite phpmyadmin. Il nome del database deve essere del tipo NUMERODIMATRICA o COGNOME_NOME;
3. La relazione.

Questi 3 file devono essere messi in un archivio (zip, 7z) ed inviati via mail ai docenti Barbara Guidi barbara.guidi@unipi.it e Andrea Michienzi andrea.michienzi@unipi.it entro la data di scadenza per la consegna. La data di scadenza delle consegne è fissata ad almeno 10 giorni prima dell'appello. Tutti gli aggiornamenti e/o modifiche ai progetti consegnati e le consegne ricevute dopo la scadenza non verranno considerate.

Non verranno considerati i progetti in cui non sono presenti i tre files richiesti, o che non presentano un nome del database come quello indicato.

Prova finale

La prova finale consiste nella discussione del progetto, durante la quale verranno poste domande individuali per accertarci che ogni membro del gruppo conosca il codice consegnato. Le domande possono riguardare tutti gli argomenti del corso (sia BdD che LPW).

Criteri di valutazione

La corretta esecuzione del codice non è sufficiente per il superamento dell'esame. Il giudizio finale dipende dalla qualità del sistema prodotto che dipende da caratteristiche quali:

- Capacità di gestire input errati o maliziosi da parte degli utenti senza errori o crash
- Utilizzo di tutte le tecnologie richieste durante il corso
- Tempo di risposta alle richieste di un utente
- Numero di blog e utenti che si è in grado di gestire senza sperimentare rallentamenti
- Quantità di memoria utilizzata
- Personalizzazioni offerte ad un blog
- Funzionalità aggiuntive non presenti nelle specifiche