

Indirizzo del sito: tecweb2021:8080/cmichele/
Gli account utente usano una e-mail per effettuare l'accesso. Fanno
eccezione le seguenti coppie:
user / user admin / admin

Autori

christian.micheletti@studenti.unipd.it
jacopo.fichera@studenti.unipd.it

Indice

| | | |
|----------|---|----------|
| 1 | Analisi | 2 |
| 1.1 | Utenza | 2 |
| 1.2 | Funzionalità pubbliche | 2 |
| 1.3 | Funzionalità per utenti | 2 |
| 1.4 | Funzionalità di amministrazione | 3 |
| 2 | Navigazione | 3 |
| 3 | Sviluppo | 3 |
| 3.1 | Database | 3 |
| 3.2 | Backend | 3 |
| 3.2.1 | Il framework Component | 3 |
| 3.2.2 | Test | 4 |
| 3.3 | Interfaccia | 4 |
| 3.3.1 | Test | 4 |
| 3.3.2 | Registrazione | 4 |

1 Analisi

1.1 Utenza

Sono previsti tre tipi di utenza, in base al livello di autorità: **utente non autenticato**, **utente autenticato** e **amministratore**. Le funzionalità sono associate alla classe di utenza a cui sono rivolte, e un livello di autorità ha accesso anche alle funzionalità dei livelli minori. In altre parole, le funzionalità pubbliche sono disponibili per ogni classe di utenza, le funzionalità dell'utente standard sono disponibili sia per questi ultimi sia per gli amministratori e le funzionalità amministrative sono esclusive degli amministratori.

1.2 Funzionalità pubbliche

Accesso pubblico. Una sezione dell'applicativo e alcune funzionalità sono di accesso pubblico. L'accesso pubblico è attivo per default, quando ci si collega alla pagina `/home.php`. La pagina principale è composta da una sezione di ricerca e navigazione, una fascia dove si riportano le informazioni sulla propria posizione e il **feed**. Il feed è composto da tre modalità di ricerca: tra i più recenti, i più popolari e i più dibattuti.

Registrazione di un nuovo account. Si effettua alla pagina `/register.php`. Occorre inserire il proprio nome utente, l'indirizzo e-mail e una password.

Ricerca post. La funzionalità si utilizza tramite l'input di ricerca in alto (desktop) o nel menù laterale (mobile), scrivendo il testo da cercare, per poi premere "invio" oppure cliccare il tasto "ricerca".

Visualizzazione di un post. Che si sia arrivati dal feed o dai risultati della ricerca, ogni visitatore può visualizzare le informazioni del post: il titolo, il nome dell'utente che l'ha postato, i commenti e le immagini dell'avvistamento. Un visitatore non autenticato da qui può anche visitare il profilo degli utenti che hanno commentato o dell'utente che ha postato.

Catalogo. Il catalogo permette di cercare delle informazioni sugli uccelli, suddivisi per ordine, famiglia e classe.

1.3 Funzionalità per utenti

Accesso all'area riservata. Si effettua dalla pagina `/login.php`. Occorre inserire l'e-mail e la password scelte al momento della creazione dell'account.

Funzionalità di interazione con il post. Un utente autenticato può lasciare un commento, lasciare un like o un dislike sul post che sta leggendo.

Modifica del profilo personale. TODO

Aggiungere degli amici. TODO

Creazione di un Post. TODO

1.4 Funzionalità di amministrazione

Gestione raw delle tabelle. TODO

2 Navigazione

3 Sviluppo

3.1 Database

3.2 Backend

Il backend è stato sviluppato interamente in php 7.0.2. Non sono stati usati framework esterni, ma è stato implementato un framework liberamente ispirato alle tecnologie più recenti.

3.2.1 Il framework Component

Il framework è organizzato secondo il paradigma ad oggetti. La classe principale è la classe astratta `Component`. Ogni componente eredita le funzionalità di questa classe. La public interface di ogni component si può riassumere come segue:

- un costruttore che prende una stringa che corrisponde al contenuto da mostrare, con dei placeholder per i dati da sostituire;
- un metodo `build` che sostituisce i placeholder, costruendo la pagina e restituendola;
- un metodo `returnComponent` che costruisce la pagina se non era ancora stata costruita;

- un metodo di conversione automatica in stringa, che corrisponde ad una chiamata `returnComponent()`;
- un metodo `notBuilt` che indica di ricostruire la pagina alla prossima `returnComponent()`;
- un metodo `resolveData` che restituisce un array associativo contenente i dati da sostituire nei placeholder. Deve essere sovrascritto nelle implementazioni delle classi figlie.

3.2.2 Test

3.3 Interfaccia

3.3.1 Test

3.3.2 Registrazione

| Nome utente | Password | e-mail | Risultato |
|----------------|----------|---------------|-----------------|
| Alfonso Cuaron | alcua | alfie@cuar.it | Successo |
| John Wick | wick | john@wick.gun | Successo |