

Documentazione per il progetto didattico intracorso 2020/2021

Studenti del gruppo 1 (grp_01) partecipanti:

Baio Antonio (S1087266: contributo: 25 %)

Galeazzi Margherita (S1087714: contributo: 25 %)

Caporusso Chiara Amalia (S1087171: contributo: 25 %)

Giannattasio Kevin (S1086831: contributo: 25 %)

Descrizione del sito:

Il nostro progetto prevede lo sviluppo di un app web per l'acquisto di biglietti per concerti.

Il sito è strutturato su quattro livelli di utenti: cliente non registrato, cliente registrato, organizzatore eventi e admin. A seconda del livello di utente, si hanno diverse funzioni disponibili.

Livello pubblico:

Il livello pubblico del sito è accessibile senza la necessità di autenticarsi (o eventualmente registrarsi). Dalla homepage è possibile navigare, attraverso la navbar, in diverse sezioni:

- Sezione '**Catalogo**': lista di tutti gli eventi disponibili. Tramite una pagina specifica, è possibile avere varie informazioni e dettagli su ogni evento. Gli eventi sono memorizzati nel database, collegato al sito, che va a popolare il catalogo. Sempre nella sezione 'catalogo' è possibile effettuare una ricerca degli eventi basata sulla descrizione dell'evento di interesse, sulla sua data (mese ed anno), il luogo di svolgimento (regione) e la società organizzatrice, specificando uno o più parametri (descrizione, luogo, data).
- Sezione '**Info**': è possibile informarsi sui servizi offerti dalla nostra organizzazione e sui vari contatti.
- Sezione '**FAQ**': è possibile consultare le domande comuni su acquisto, metodo di pagamento ecc. (realizzato tramite jQuery-JavaScript)
- Sezione '**Accedi**': si viene rimandati alla pagina di autenticazione (o eventualmente registrazione), dove una volta autenticati, si ha l'accesso all'area personale (e si ottiene anche la possibilità di acquistare i biglietti, non disponibile per utenti non registrati). L'autenticazione "lato server" è costruita mediante il package laravel/ui, mentre "lato client" tramite form del package laravel/Collective.

Livello utente:

L'utente che ha effettuato l'accesso verrà reindirizzato verso la sua area privata.

La prima funzionalità disponibile di questo livello di utenza è quella dell'acquisto dei biglietti: tramite il catalogo, si può scegliere un evento tra quelli disponibili e acquistare uno o più biglietti relativi a quest'ultimo.

Inoltre, sempre nella propria sezione privata, è possibile:

- modificare i propri **dati di registrazione**;
- consultare lo **storico acquisti**: si verrà reindirizzati in una pagina dove sono presenti i biglietti acquistati e le informazioni ad esso associati, con un codice QR, univoco che se scansionato, identifica il biglietto dell'utente con alcune informazioni sull'evento;
- consultare la sezione "**parteciperò**", dove si ha un elenco di tutti gli eventi a cui si intende partecipare (senza aver ancora acquistato il biglietto). È possibile indicare la propria intenzione a partecipare ad un evento mediante click sul bottone parteciperò nella pagina contenente i dettagli dell'evento. I parteciperò sono gestiti nel database in maniera tale che un utente possa esprimere la sua intenzione di partecipare una sola volta per evento.

Viene implementata ovviamente anche la funzionalità di logout, per terminare la propria sessione.

Livello organizzatore:

Un utente del livello organizzatore ha come prima funzionalità disponibile quella di gestione dei propri eventi. Più nello specifico può (con conseguente aggiornamento del database):

- **inserire** un nuovo evento sul catalogo del sito (utilizzo di jQuery per la validazione della form e per semplificare l'inserimento della mappa che indica il luogo in cui si svolge l'evento);
- **visualizzare i dettagli** di un evento specifico;
- **visualizzare le statistiche** riguardanti un evento specifico, quali: la quantità di biglietti venduti per un evento specificato; la percentuale di biglietti venduti per un evento specificato in rapporto a quelli disponibili; l'incasso di ogni evento;
- **modificare** i dati di un evento specifico (tramite una form dove inserire i nuovi dati);
- **eliminare** un evento specifico.

Viene implementata anche qui la funzionalità di logout, per terminare la propria sessione.

Livello admin:

L'utente di livello amministratore ha una propria area personale, dove tramite navigazione può raggiungere due aree specifiche:

- **gestione FAQs**: è possibile inserire, eliminare o modificare tramite apposite form le FAQs da presentare poi nella relativa sezione, con conseguente aggiornamento del database;
- **gestione utenti**: è possibile eliminare, tramite un'apposita tabella, gli utenti registrati al sito. Inoltre, è possibile eliminare, aggiungere e modificare gli organizzatori di eventi, con la visualizzazione dei dati di vendita di ciascuna organizzazione, previa selezione di quella d'interesse (intesi come il numero cumulativo di biglietti venduti per tutti gli eventi organizzati e l'incasso totale).

Viene implementata anche qui la funzionalità di logout, per terminare la propria sessione.

Sperimentazione in autonomia (pacchetti aggiuntivi):

Abbiamo inoltre deciso di aggiungere alcuni pacchetti opzionali:

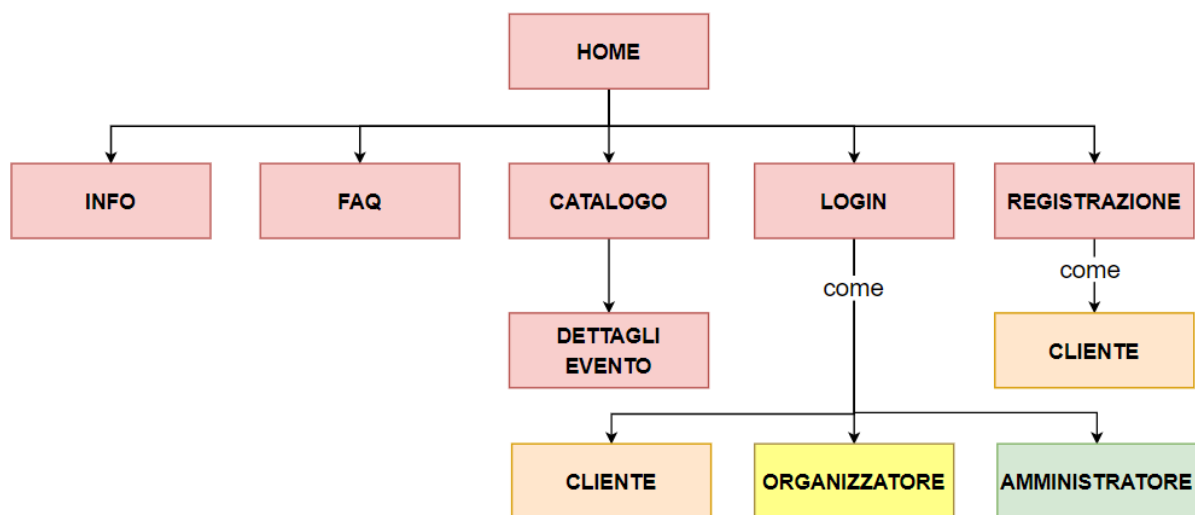
- package Carbon utilizzato per gestire la data e l'ora in PHP in modo molto più semplice e semantico. Per gestire il formato di data e ora nel database, abbiamo utilizzato il formato timestamp.
- un middleware "PreventBackHistory" : ci permette di risolvere il problema legato al fatto che successivamente al Logout, cliccando sul bottone che permette di tornare indietro del browser, le pagine riservate dei vari utenti rimanevano visibili, contrariamente a quanto doveva accadere secondo gli altri middleware da noi definiti. Questo perché quando l'utente clicca il back button e non è loggato, il browser visualizza la pagina salvata nella cache. L'utente non potrà navigare né interagire con nulla, ma anche per evitare che possa semplicemente vedere l'ultima pagina dell'utente che ha appena effettuato il Logout, questo middleware "ripulisce" la cache a seguito del Logout.
- creazione delle pagine da visualizzare in caso di errore 403 (l'accesso alla risorsa richiesta è vietato per il richiedente) e 404 (il server non ha trovato ciò che è stato richiesto).
- Per la generazione "istantanea" dei codici QR associati ai biglietti, abbiamo installato e successivamente configurato un package aggiuntivo per Laravel (tramite Composer), chiamato "simple-qr-code".

Schema dei link:

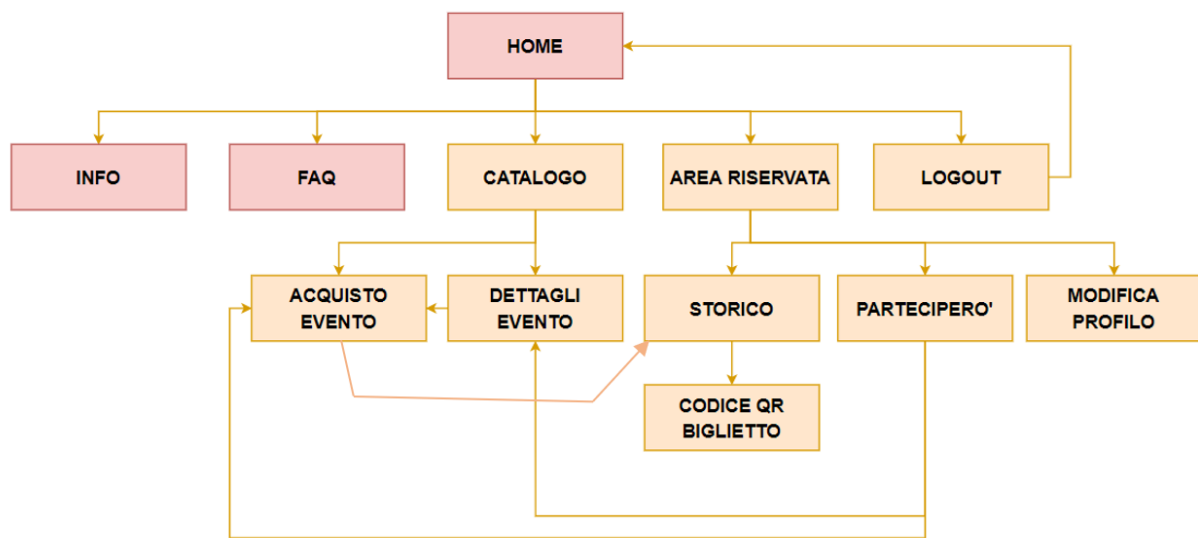
LEGENDA:

OSPITE	=	Zone visibile a tutti gli utenti
CLIENTE	=	Zone visibile ai clienti
ORGANIZZATORE	=	Zone visibile agli organizzatori
AMMINISTRATORE	=	Zone visibile all'amministratore

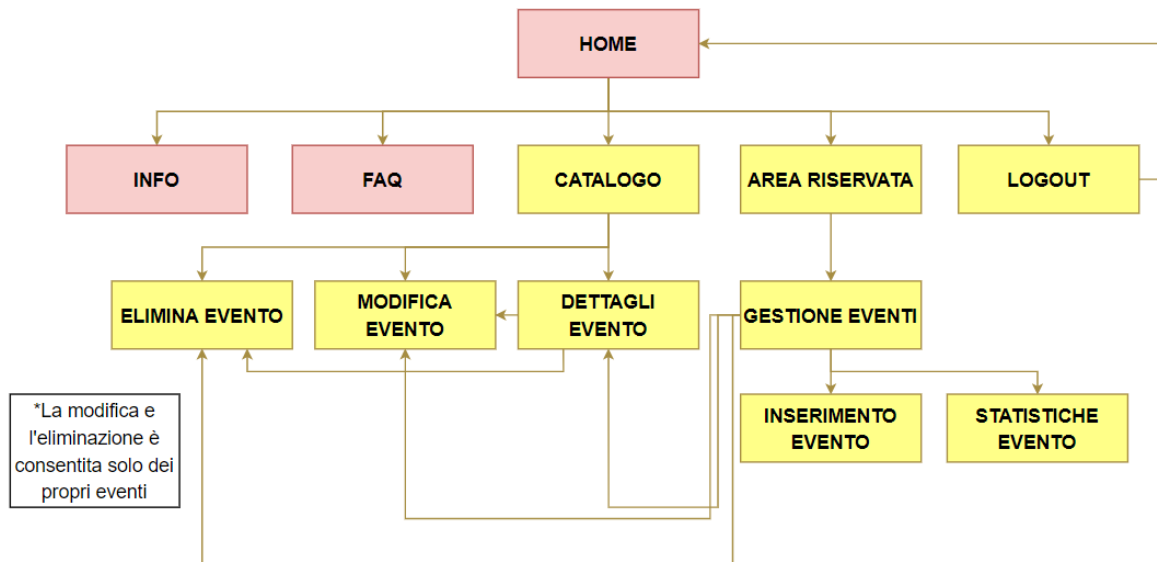
Livello "OSPITE":



Livello "CLIENTE":



Livello "ORGANIZZATORE":



Livello "AMMINISTRATORE":

