



UNIVERSITÀ POLITECNICA DELLE MARCHE

CORSO DI TECNOLOGIE WEB

Sito web per la gestione di prestazioni
ambulatoriali specialistiche

Health Facility

Gianluca Bonifazi

1082209

Professore
Alessandro CUCCHIARELLI

Indice

1	Riferimenti	2
2	Descrizione del sito	2
3	Mappa del sito	3
4	Soluzioni adottate	4
4.1	Responsive Design	4
4.2	Chat	4
4.3	Prenotazioni	5
4.4	Grafici	5
4.5	Real-Time Search	5
5	Informazioni Utili	6

1 Riferimenti

Il presente progetto è stato sviluppato dal **Gruppo 52**, composto esclusivamente da **Gianluca Bonifazi** (matricola 1082209), durante il corso di laurea magistrale in *Ingegneria Informatica e dell'Automazione* nell'anno accademico 2017-2018 per l'esame di *Tecnologie Web*.

2 Descrizione del sito

Il progetto aveva come obbiettivo la realizzazione di un sito web per la gestione delle prestazioni ambulatoriali specialistiche di una struttura sanitaria che è stata idealmente chiamata **Health Facility**.

Questa struttura è divisa in dipartimenti (es. *Cardiologia, Neurologia, Radiologia, ...*), i quali a loro volta, mettono a disposizione varie prestazioni specialistiche (es. *Elettrocardiogramma, Ecografia, Mappatura dei nei, ...*). Queste prestazioni, anche chiamate *servizi*, sono prenotabili direttamente online, previa registrazione e autenticazione mediante le pagine predisposte. Una volta che l'utente ha compiuto queste operazioni può, tramite la pagina di dettaglio di ogni servizio (o prestazione), scegliere data e ora e premere sul pulsante che verifica l'eventuale disponibilità e, se l'orario risulta essere valido, prenota la visita, inserendo nel database una riga che fa riferimento all'utente, al servizio e alla data e all'ora scelta.

Inoltre è anche disponibile una pagina di ricerca, la quale permette di reperire in modo veloce le informazioni riguardanti il servizio che l'utente desidera prenotare. Sono disponibili due modi differenti di ricerca: tramite selezione del dipartimento e successivamente dei servizi ad esso collegati o tramite inserimento di caratteri.

Ogni utente registrato possiede un'area riservata nella quale sono elencati tutti gli appuntamenti: sia futuri che passati. Vi è inoltre la possibilità di modificare email e password associati all'account ed è presente anche una chat utile per richiedere assistenza direttamente allo staff della struttura o per ricevere comunicazioni riguardanti lo spostamento o la cancellazione di una visita.

Anche il personale della struttura, dopo essersi autenticato, viene reindirizzato ad una propria area riservata composta da un'agenda, contenente gli appuntamenti da svolgere, ed una chat speculare a quella presente nella zona riservata all'utente.

Infine, al livello più alto della policy di autenticazione, troviamo l'utente amministratore. Anch'esso, dopo aver effettuato il login tramite la stessa pagina degli altri due utenti (user e staff) viene reindirizzato in un'area riservata dove gli vengono messe a disposizione le funzionalità per gestire in autonomia l'intero sito web. Infatti può:

- Cancellare o modificare gli utenti registrati al sito (user)
- Cancellare, modificare ed inserire personale della struttura (staff)
- Cancellare, modificare ed inserire i dipartimenti della struttura
- Cancellare, modificare ed inserire i servizi di ogni dipartimento
- Cancellare, modificare ed inserire prenotazioni ai vari servizi
- Cancellare, modificare ed inserire le FAQ

Inoltre, nella dashboard dell'amministratore sono presenti alcuni grafici che servono ad effettuare analisi statistiche sulle prenotazioni alla struttura. Infatti, inserendo un intervallo di tempo verranno visualizzati: il numero di appuntamenti totali suddivisi per tale intervallo, il numero totale di appuntamenti suddivisi per tipo di prestazione ed il numero totale di appuntamenti suddivisi per dipartimento.

Un altro tipo di analisi che può effettuare l'amministratore è quella relativa ai grafici di ogni utente. Basterà infatti cliccare sul pulsante presente nella lista degli utenti registrati per poterli visualizzare.

3 Mappa del sito

- Homepage - /index
- Dipartimenti - /department
 - Dettaglio dipartimento - /department/details/id/:department_id
- Dettaglio servizio - /service/index/id/:service_id
- Cerca prestazione - /search
- Contatti - /contact
- FAQ - /faq
- Registrati - register
- Login - /login
- Area Riservata Utente - /user
 - Dashboard - /user
 - Modifica Profilo - /user/profile
 - Chat Assistenza - /user/chat
- Area Riservata Staff - /staff
 - Dashboard - /staff
 - Agenda - /staff/agenda
 - Lista Chat Utenti - /staff/userlist
 - * Chat utente - /staff/chat/id/:user_id
- Area Riservata Amministratore - /admin
 - Dashboard - /admin
 - Gestione Utenti - /admin/user
 - * Statistiche Utente - /admin/userstats/id/:user_id
 - Gestione Staff - /admin/staff
 - * Inserisci nuovo staff - /admin/insertstaff
 - Gestione Dipartimenti - /admin/department
 - * Inserisci nuovo dipartimento - /admin/insertdepartment
 - Gestione Servizi - /admin/service
 - * Inserisci nuovo servizio - /admin/insertservice
 - Gestione Prenotazioni - /admin/booking
 - * Inserisci nuova prenotazione - /admin/insertbooking
 - Gestione FAQ - /admin/faq
 - * Inserisci nuova FAQ - /admin/insertfaq

4 Soluzioni adottate

Come da specifiche progettuali il back-end del sito web è stato scritto utilizzando Zend Framework 1. Lato front-end invece è stato utilizzato il framework Bootstrap nella sua versione 4.1.3 insieme a diverse librerie Javascript, tra cui: jQuery, Moment, DatetimePicker e ChartJS. Di seguito sono elencate le soluzioni tecnologicamente più interessanti adottate in questo progetto.

4.1 Responsive Design

L'intera applicazione web è stata realizzata utilizzando un *Design Responsivo*, infatti il sito è in grado di adattarsi graficamente in modo automatico al dispositivo col quale viene visualizzato riducendo al minimo la necessità dell'utente di ridimensionare e scorrere i contenuti. Questo è stato possibile grazie all'ampio utilizzo di Bootstrap, all'utilizzo delle CSS media query e ad un attento studio preliminare sul posizionamento degli elementi nella pagina.

4.2 Chat

La chat, presente nell'area riservata dell'utente e in quella dello staff, è stata implementata utilizzando la tecnica AJAX. Il funzionamento è il seguente:

1. L'utente apre la pagina della chat.
2. Il controller istanzia una variabile, la quale conterrà i messaggi scritti fino all'istante in cui la query al database è stata eseguita.
3. La chat viene popolata utilizzando gli eventuali messaggi presenti in questa variabile. Se la variabile è vuota, e quindi non sono stati ancora mandati messaggi, la chat rimarrà vuota.
4. Durante il caricamento della pagina viene anche istanziata una variabile chiamata `dateNow` con la data, l'ora, i minuti ed i secondi in cui sono stati reperiti gli ultimi messaggi della chat. Questa variabile infatti servirà per verificare quando è stato eseguito l'ultimo download dei messaggi.
5. Tutto ciò che è stato descritto fino ad ora è opera del back-end, cioè del PHP. Da adesso in poi, a prendere il controllo della situazione è il lato front-end con Javascript che, non appena il documento è stato correttamente caricato, avvia una funzione `setInterval`. Questa funzione, eseguita per l'appunto ad ogni intervallo di `n` millisecondi, emette una chiamata AJAX al server richiedendo tutti i messaggi inseriti nel database con data successiva al valore presente all'interno della variabile `dateNow`, che verrà poi aggiornata all'istante in cui il server risponde.
6. Se ci sono nuovi messaggi allora vengono aggiunti alla chat, altrimenti non succede nulla.

La funzione `setInterval`, descritta nei punti 5 e 6, viene eseguita continuamente finché il browser rimane aperto sulla pagina mantenendo la chat sempre aggiornata con gli ultimi messaggi. Anche l'invio di nuovi messaggi, è sempre gestito da Javascript. Infatti una funzione AJAX viene lanciata nel momento in cui si preme il pulsante d'invio e inserisce il nuovo messaggio nel database. Questi nuovi messaggi non sono istantaneamente inseriti nella chat perchè verranno poi scaricati al prossimo "giro" della funzione `setInterval`.

PS: Per completezza è giusto evidenziare che la chat potrebbe presentare alcuni problemi nel momento in cui l'utente non abbia una buona connessione o nell'eventualità che il server sia sovraccaricato e che quindi non abbia un'elevata velocità di risposta. In effetti la tecnica AJAX non è la più indicata per la creazione di una chat real-time, bensì sarebbe stato più opportuno l'utilizzo delle Web-Socket che però non sono argomento del corso e che quindi potrebbero essere implementate come sviluppo futuro del progetto.

4.3 Prenotazioni

Le prenotazioni all'interno del sito web sono gestite interamente con richieste AJAX al server. Inoltre, come funzionalità opzionale, è stata implementata la possibilità di far scegliere all'utente una data dal calendario evitando così di dover prenotare il primo "slot" libero nell'agenda. Questa libertà di scelta ha avuto implicazioni sul metodo di controllo della prenotazione. Infatti, prima di confermare una prenotazione o una eventuale modifica, viene controllato se, sia l'utente che il servizio, non hanno altre prenotazioni che si andrebbero a sovrapporre e se gli orari di apertura e di chiusura della struttura sono inerenti alla data e all'ora scelta.

4.4 Grafici

Nell'area riservata all'amministratore vengono utilizzati dei grafici per visualizzare le statistiche sulle prenotazioni effettuate nella struttura sanitaria. Questi grafici sono disegnati grazie ad una famosa libreria open source scritta in Javascript chiamata ChartJS (<https://www.chartjs.org/>). Per rappresentare graficamente i dati sono stati utilizzati dei grafici a linea ma, con pochi piccoli accorgimenti, si possono trasformare in grafici a torta o a barre.

4.5 Real-Time Search

La ricerca, per nome, delle prestazioni avviene in tempo reale. Infatti i dati vengono estratti direttamente mentre l'utente scrive e non dopo aver effettuato l'invio al server. Questo è possibile grazie all'uso di Javascript.

I passaggi sono i seguenti: durante il caricamento della pagina, il controller richiede al model i dati che a sua volta risponde incapsulando il tutto all'interno di un JSON. Quest'ultimo durante la digitazione dell'utente, viene "sfogliato" verificando che i caratteri digitati siano presenti all'interno del nome identificativo del servizio. Quelli che danno risultato positivo sono stampati a video, gli altri vengono ignorati.

5 Informazioni Utili

L'intero progetto è stato sviluppato utilizzando Git come software per il controllo della versione ed è caricato in una repository **Github** pubblica, raggiungibile al seguente link:

- <https://github.com/Biuni/HealthFacility-TW>
-

Di seguito la lista di librerie e framework utilizzati con i link relativi alle loro documentazioni.

Javascript:

- jQuery - <https://jquery.com>
- MomentJS - <https://momentjs.com>
- DataTable - <https://datatables.net/>
- DateTime Picker - <https://eonasdan.github.io/bootstrap-datetimepicker/>

CSS:

- Bootstrap - <https://getbootstrap.com/>
- Fontawesome - <https://fontawesome.com/>

PHP:

- Zend Framework - <https://framework.zend.com/>
-

I dati di autenticazione per gli utenti di test sono:

Account amministratore:

- *Username*: admin
- *Password*: IJnT

Account staff:

- *Username*: staff
- *Password*: IJnT

Account utente:

- *Username*: user
 - *Password*: IJnT
-

Le immagini sono state prese da **Flaticon** (<https://www.flaticon.com/>)