**RELAZIONE APP EVENTI**

**Gruppo**

* Giacomo Matteucci (matricola 109371)
* Leonardo Pittori (matricola 110680)

**Descrizione generale**

Per l’esame di Progettazione Applicazioni Web e Mobili (PAWM) abbiamo pensato di realizzare una semplice applicazione che consente di potersi prenotare ad alcuni eventi. Per poterla utilizzare è previsto che l’utente debba possedere un account, con il quale potersi loggare ed effettuare le operazioni previste.

Nel caso in cui l’utente non possieda un account, l’applicazione consente a quest’ultimo di **registrarsi** facendo inserire:

* nome
* cognome
* data di nascita
* email, usata per *identificare* l’utente. Nel caso in cui l’email inserita sia già stata usata per un altro account, la registrazione non viene effettuata
* password, la quale deve avere una lunghezza minima di 6 caratteri

Una volta creato un account ed effettuato il **login** (attraverso email e password), l’applicazione mostrerà all’utente la propria pagina personale nella quale si potranno visualizzare, oltre al proprio nome e cognome, le prenotazioni effettuate per eventi che si devono ancora svolgere: le stesse possono anche essere cancellate.

Dalla sua pagina personale l’utente può scegliere di:

* effettuare il **logout**
* **visualizzare informazioni** relative ai **creatori** **dell’applicazione**
* **modificare** i propri **dati personali**, cioè email e password
* **eliminare** il proprio **account**
* **visualizzare** le **prenotazioni effettuate** **per** **eventi** che si sono **svolti**
* **effettuare** una **nuova prenotazione**

Quando l’utente sceglie di effettuare una prenotazione, l’applicazione mostra tutti gli eventi disponibili (se non ne è presente nessuno da poter prenotare, l’applicazione avvisa l’utente di ciò e lo riporta alla sua pagina personale in modo automatico): l’utente digita il numero di partecipanti per l’evento al quale desidera prenotarsi e conferma attraverso l’apposito bottone (nel caso in cui l’utente abbia già una prenotazione per quell’evento il numero di partecipanti di quest’ultima verrà incrementato).

**Tecnologie utilizzate**

Nella realizzazione di quest’applicazione si possono distinguere 2 parti fondamentali:

* front-end
* back-end

Tecnologie utilizzate (**back-end**)

Per la “costruzione” del back-end si è scelto di utilizzare **Spring Boot**. Quest’ultimo è un framework utilizzato per facilitare la costruzione/scrittura di applicazioni basate su Java

Inoltre, poiché viene utilizzato un database per poter memorizzare tutti i dati riguardanti utenti, eventi e prenotazioni per questi ultimi, sono state adottate altre 2 tecnologie:

* **JPA**, una libreria ORM (Object Relational Mapping) che consente di interrogare e rappresentare le tabelle di un database come se fossero oggetti. Permette, cioè, di effettuare l’accesso, gestire e rendere persistenti dati fra database relazionali e oggetti (classi)
* **HIBERNATE**, una piattaforma che si occupa di fornire un servizio ORM (in questo caso JPA)

Per la gestione degli account è stato utilizzato **Firebase** (sviluppato da Google): attraverso questa tecnologia è possibile delegare la gestione degli account, e quindi delle password (con tutte le problematiche correlate), ad un sistema di autenticazione esterno.

Tecnologie utilizzate (**front-end**)

Per la “costruzione” del front-end si è scelto **Ionic**.

Ionic è un framework utilizzato per sviluppare, sfruttando tecnologie web come HTML, Javascript e CSS, app mobile ibride: permette, cioè, di scrivere app che funzionano su piattaforme diverse con la stessa base di codice.

Per quanto riguarda il framework Javascript con il quale è stata sviluppata la parte front-end è stato scelto **Angular**, utilizzato per sviluppare di Single Page Application (SPA). Questo tipo di applicazioni prevede che:

* alla prima richiesta di una pagina web (cioè quando l’utente vuole visitarla per la prima volta) il browser fa una richiesta al back-end, il quale risponde restituendo la pagina html e una serie di file javascript (scritti con framework come Angular)
* alle successive interazioni dell’utente il browser effettua una chiamata AJAX (tecnologia che, tramite una chiamata asincrona nel codice sorgente javascript, permette di fare chiamate http ad un back-end per ottenere un’informazione) al back-end, che non restituisce una pagina HTML ma un’informazione (nella quasi totalità dei casi file json, ma anche xml). Mediante i file javascript scaricati durante la prima richiesta, e analizzando le informazioni nei file json, il front-end modificherà il DOM

Ionic fa uso di **Capacitor**, un progetto open source che permette di eseguire web app (cioè applicazioni basate su tecnologie web) su piattaforme diverse, fornendo un’interfaccia che permette di accedere agli SDK e API nativi di una piattaforma: ciò significa che permette di accedere alle funzionalità native di ogni piattaforma con la stessa base di codice.