**Relazione tecnica Web Application CarSEA (BlaBlaCar)**

**Studente:** Gherardi Gianmattia  
**Matricola:** STA07049/L8  
**Progetto:** BlaBlaCar  
**Repository:** https://github.com/ggherardi/CarSEA

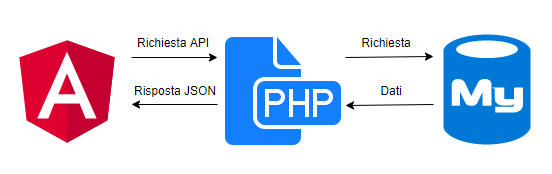
1. **Tecnologie utilizzate**

Per la realizzazione del progetto sono state utilizzate le seguenti tecnologie:

* Angular 5.x
* PHP 7.0
* DB MySQL
* CSS3

Essendo realizzato in Angular, il progetto avrà una struttura che segue il paradigma MVC (Model, View, Controller) e il codice sorgente sarà scritto in HMTL e CSS3 per quanto riguarda le View, e in TypeScript per quanto concerne invece Controller e Model. Per propria natura, il codice TypeScript viene compilato tramite un transpiler in JavaScript (compatibile ECMAScript 5), e dunque l’applicazione sarà eseguibile su qualunque piattaforma che supporti lo standard ECMAScript 5.   
Le chiamate REST si interfacceranno con dei servizi scritti in PHP i quali, a loro volta, comunicheranno con un database MySQL.

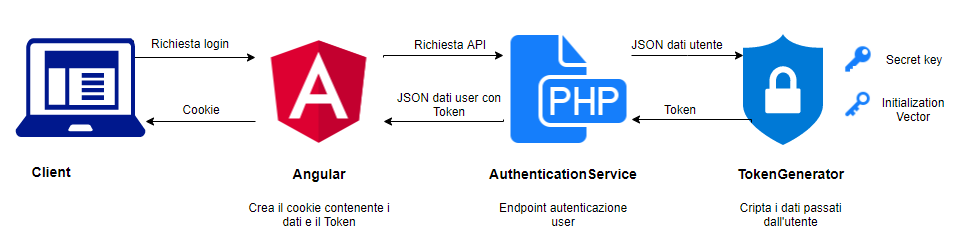
1. **Comunicazione con il database**

Alcune operazioni svolte dagli utenti implicheranno la persistenza di dati sul database MySQL. Questi dati verranno comunicati tramite delle chiamate REST a dei servizi PHP, che a loro volta andranno ad interrogare il database per restituire infine al client una response in formato JSON. 

Da notare che non tutte le richieste API potranno essere effettuate se l’utente non ha effettuato l’accesso al sito: alcuni endpoint necessitano, infatti, dei Token di sicurezza staccati durante l’autenticazione.

1. **Autenticazione e sicurezza**

Quando l’utente crea una nuova utenza, viene generata una tupla nel DB contenente i dati forniti durante la registrazione; le password vengono criptate in modo da non essere lasciate in chiaro e per renderle più sicure.  
Ogniqualvolta l’utente immetterà le sue credenziali di accesso, se l’autenticazione andrà a buon fine verrà creato un cookie per mantenere la sessione di quell’utente attiva: questo cookie ha durata di 5 ore e conterrà Username, ID, Nome e un Token criptato. La creazione del Token avviene utilizzando due diverse chiavi di cifratura e la stringa JSON contenente i dati sopracitati dell’utente. In questa maniera viene generato un Token che sarà indissolubilmente legato a queste informazioni, così da impedire tentativi di manomissione dei cookie e quindi delle chiamate ai servizi: infatti, alcuni endpoint (in particolare quelli che andranno a scrivere sul DB) necessitano della validazione del Token che verrà passato durante la chiamata; se la decriptazione non andasse a buon fine, la chiamata al servizio termina e viene restituito al client un messaggio di errore di autenticazione. In tal modo si può impedire che una persona esterna possa causare danni ad altri utenti.



1. **Sistema di logging**

E’ stato creato un sistema di logging che permette di tracciare le operazioni eseguite dai servizi PHP. Verrà creato ogni giorno (eventualmente da suddividere ulteriormente) un file di log che scriverà operazioni eseguite ed eventuali eccezioni. Ogni singola chiamata al servizio verrà dotata di un ID di correlazione (CorrelationID) che permetterà di seguire tutte le operazioni di quella richiesta sul log, fino al termine della stessa.

1. **Analisi codice sorgente e script PHP**

Di seguito si troverà la descrizione del funzionamento dei componenti Angular e degli script PHP:

*Componenti Angular*