

Progetto di Programmazione WEB & MOBILE

Prenotazione di un servizio Beauty & Fashion

Su un portale Web .

Eduardo Romero



INDICE

1. Introduzione	3
2. Descrizione generale	3

2.1. Descrizione delle tecnologie utilizzate.....	3
4. Descrizione dello sviluppo del progetto.....	4

1. Introduzione

Negli ultimi anni si è assistito ad uno sviluppo delle ICT (Information and Communication Technology che sono l'insieme dei metodi e delle tecniche utilizzate nella trasmissione, ricezione ed elaborazione di dati e informazioni) e delle relative applicazioni come fulcro funzionale allo sviluppo; nonostante ciò gran parte dei consumatori e dei gestori di attività di erogatori di servizi, (centri estetici, alberghi, spa ecc.) considerano l'impiego della tecnologia ed internet una componente alternativa ed essenziale.

A partire dalle prime innovazioni tecnologiche si denota come l'impatto di internet abbia suscitato notevoli migliorie sia nel campo della gestione/progettazione che nel campo economico.

Il principale obiettivo di questo progetto è la realizzazione di un "sistema applicativo" web, che permetta la prenotazione di una prestazione di un servizio erogato dal centro-estetico "fittizio", che permetta la registrazione e all'accesso all'area personale. Fondamentalmente ho suddiviso in due parti: una parte di Front-End, e l'altra di Back-End. Volendo con l'eventuale implementazione di un Framework che consenta di sfruttare al meglio delle tecnologie web si può sviluppare un applicazioni mobile.

2. Descrizione generale del progetto

Il progetto nasce principalmente con l'idea di sviluppare un applicazione Web per un ipotetico centro estetico.

Ho diviso il progetto in due parti :

- La prima rappresenta fondamentalmente l'intera applicazione web
- La seconda invece rappresenta un applicativo di servizio REST che interlocherà con il DB

La prima parte, quindi, si identifica il lato client, e per lo sviluppo dell' applicativo ho deciso di utilizzare un Angular.

Angular è un framework che si basa sul linguaggio di programmazione TypeScript(superset di JavaScript) per lo sviluppo di applicazioni Web client side. Esso ha riscosso un notevole successo dovuto all'approccio di sviluppo proposto e all'infrastruttura fornita che incoraggia l'organizzazione del codice e la separazione dei compiti nei vari componenti. E' stato sviluppato dalla collaborazione di Microsoft e Google due giganti dell'informatica moderna.

Le applicazioni sviluppate in Angular sono chiamate SPA , cioè una Single-Page-Application. In informatica con Single-Page-Application si intende un'applicazione web o un sito web che può essere usato o consultato su una singola pagina web con l'obiettivo di fornire una esperienza utente più fluida e simile alle applicazioni desktop dei sistemi operativi tradizionali.

Esse vengono eseguite interamente dal Browser dopo essere state scaricate dal web-server(elaborazione quindi nel lato client).

Questo comporta il risparmio di dover spedire indietro la pagina web al web-server ogni volta che c'è una richiesta di azione da parte dell'utente. Il codice generato da Angular gira su tutti i principali web browser moderni quali ad esempio Chrome,Microsoft,Firefox, Opera, Safari ecc....

Angular è stato progettato per fornire uno strumento facile e veloce per sviluppare applicazioni che girano su qualunque piattaforma inclusi smartphone e tablet. Infatti le applicazioni web in Angular in combinazione con il toolkit open source Bootstrap diventano responsive, ossia il design del sito web si adatta in funzione alle dimensioni del dispositivo utilizzato¹.

Altro toolkit che facilita la progettazione in Angular è Angular-Material, una serie di componenti che permette di creare una pagina web molto velocemente: con l'utilizzo combinato di Flex-Layout ed Angular Material si possono creare siti e applicazioni web responsive molto avanzate basate su Angular.

Per la seconda parte ho deciso di utilizzare Node.js, ed Express.js(un framework per applicazioni web per Node.js).

Esso è un runtime system open source multipiattaforma orientato agli eventi per l'esecuzione di codice JavaScript, costruita sul motore JavaScript V8 di Google Chrome.. Questo consente agli sviluppatori di realizzare web application con JavaScript non più solo lato client, ma anche sfruttandolo come linguaggio di programmazione lato server.

Node.js è Molti dei suoi moduli base sono scritti in JavaScript, e gli sviluppatori possono scrivere nuovi moduli nello stesso linguaggio.

La caratteristica principale di Node.js risiede nella possibilità che offre di accedere alle risorse del sistema operativo in modalità event-driven(progr. ad eventi) e non sfruttando il classico modello basato su processi o thread concorrenti, utilizzato dai classici web server. Il modello event-driven, o “programmazione ad eventi”, si basa su un concetto piuttosto semplice: si lancia una azione quando accade qualcosa. Ogni azione quindi risulta asincrona a differenza dei pattern di programmazione più comune in cui una azione succede ad un'altra solo dopo che essa è stata completata.

Ciò dovrebbe garantire una certa efficienza grazie ad un sistema di callback gestito a basso livello dal runtime.

Grazie al comportamento asincrono, durante le attese di una certa azione il runtime può gestire qualcos'altro che ha a che fare con la logica applicativa.

Per la parte del DB ho deciso di utilizzare MongoDB.

Esso è un DBMS non relazionale, orientato ai documenti. Classificato come un database di tipo NoSQL, MongoDB si allontana dalla struttura tradizionale basata su tabelle dei DBSql

in favore di documenti in stile JSON con schema dinamico (MongoDB chiama il formato BSON, Binary-JSON), rendendo l'integrazione di dati di alcuni tipi di applicazioni più facile e veloce.

I documenti BSON (oggetti) consistono in liste ordinate di elementi. Ciascun elemento consiste in un nome di campo, un tipo e un valore. I nomi di campo sono stringhe. I tipi includono:

- stringhe
- interi (32 o 64 bit)
- double (numeri a virgola mobile a 64 bit, standard IEEE754)
- date (numeri interi in millisecondi dall'*epoca Unix* come riferimento, 1° gennaio 1970)
- byte array (dati binari)
- boolean (true e false)
- NULL
- oggetto BSON
- array BSON
- espressioni regolari

- codice Javascript

I tipi BSON sono, nominalmente, un superset dei tipi JSON (JSON non include tipi data o byte array, per citarne alcuni), con la differenza notevole rappresentata dall'assenza di un tipo "numerico" universale, incluso in JSON.

4. Descrizione dello sviluppo del progetto

Il mio applicativo deve essere lanciato su due differenti righe di comando,
con il comando:

-“npm run start” .

L'applicazione web “Centro-estetico” una volta lanciata si può raggiungere al seguente indirizzo direttamente dal Browser “http://localhost:4200” . Esso è raggiungibile grazie ad un server built-in di Angular, che si appoggia nel indirizzo IP locale: 127.0.0.1 (l'equivalente di localhost ed identifica il nostro calcolatore), ed è in ascolto sulla porta 4200(default).

L'applicativo del servizio REST , è in ascolto nella porta:3000, ed espone le seguenti API:

Method-POST :http://localhost:3000/api/utente/login

Method-POST :http://localhost:3000/api/utente/signup

Method-GET :http://localhost:3000/api/booking

Il termine REST rappresenta un sistema di trasmissione di dati su HTTP senza ulteriori livelli. I sistemi REST non prevedono il concetto di sessione, ovvero sono stateless.

L'architettura REST si basa su HTTP. Il funzionamento prevede una struttura degli URL ben definita che identifica univocamente una risorsa o un insieme di risorse e l'utilizzo dei metodi HTTP specifici per il recupero di informazioni (GET), per la modifica (POST, PUT, PATCH, DELETE) e per altri scopi (OPTIONS, ecc.). Questo particolare aspetto è approfondito nella sezione "Relazione tra gli URL e i metodi HTTP".

Grazie a queste API esposte, posso gestire la Registrazione-Signup da lato client,

l' Accesso all'area personale, Login, e richiedere della disponibilità di Un orario libero fittizio.

Purtroppo è da completare il calendario, e l'accesso AUTH con i servizi di google, facebook e twitter.