

Previsioni Meteo

Matteo Celardo 984457

PWM - secondo semestre A.A. 2021/2022

Indice

1	Introduzione	2
1.1	Analisi dei requisiti	2
1.1.1	Destinatari	2
1.1.2	Modello di valore	2
1.1.3	flusso di dati	3
1.1.4	Aspetti tecnologici	3

Capitolo 1

Introduzione

1.1 Analisi dei requisiti

1.1.1 Destinatari

L'applicazione è progettata per andare incontro a qualunque tipo di utente, dal più al meno esperto, grazie ad un'interfaccia semplice ed intuitiva, la quale permette di trovare facilmente le informazioni relative al meteo in tempo reale della città a cui si è interessati tramite la barra di ricerca o tramite le città impostate tra i propri preferiti, previa registrazione e introduzione delle stesse nell'apposito spazio all'interno dell'area personale.

Nella versione attuale, non sono posti particolari vincoli di banda grazie alla sola presenza degli elementi utili alla navigazione, riducendo così la latenza il più possibile. Inoltre, le immagini relative alle condizioni meteorologiche delle città visualizzate saranno caricate in modalità asincrona, riducendo al minimo il periodo in cui la pagina non risponde agli input dell'utente.

Attualmente, si consiglia la navigazione via PC per una maggiore semplicità di lettura, ma è comunque possibile utilizzare un telefono grazie alla responsività degli elementi presenti nelle pagine.

Gli utenti che si recano sulla piattaforma web saranno principalmente persone che hanno la necessità di sapere le condizioni meteo di una città di loro interesse, motivo per cui i contenuti sono organizzati in modo tale da fornire informazioni su richiesta esplicita dell'utente tramite una barra di ricerca oppure tramite gli elementi salvati nell'area personali una volta effettuato il login

1.1.2 Modello di valore

L'applicazione si contraddistingue per essere intuitiva e veloce da usare, elementi sempre graditi durante l'esperienza utente.

Grazie alla disponibilità in tempo reale di informazioni relative al tempo, quest'applicazione potrebbe essere integrata con un'API in grado di fornire informazioni a sistemi automatici in grado usati per svolgere operazioni in base al tempo atmosferico, risparmiando così ingenti somme di denaro in sensori e cablaggi.

Benché attualmente non presenti, sarebbe facile inserire banner pubblicitari o sponsor di sorta grazie alla struttura modulare del progetto.

Entrambi questi elementi accrescerebbero notevolmente il valore economico dell'applicazione in dipendenza, rispettivamente, al numero di sistemi automatici collegabili per utente o al numero di banner inseriti, motivo per cui risulta difficile formulare una stima esatta di valore economico.

1.1.3 flusso di dati

Il flusso dei dati all'interno dell'applicazione è unicamente in formato JSON: lo scambio di dati tra client e server o tra client e API di siti esterni è infatti interamente gestito attraverso l'inoltro di oggetti e stringhe JSON.

Grazie ad una struttura REST, le comunicazioni saranno unicamente aperte dal client nel momento in cui necessiterà di una risorsa (come il meteo relativo ad una città oppure un'altra pagina del sito), il quale richiederà ciò di cui necessita tramite oggetti JSON inviati al server, ricevendone altri in risposta che permetteranno di aggiornare un frammento della pagina o di effettuare il reindirizzamento.

I contenuti salvati sono interamente archiviati lato server tramite l'uso di un database MongoDB ad eccezione della preferenza espressa dall'utente per avere la pagina in dark mode oppure in light mode, i quali sono salvati localmente attraverso localStorage.

Allo stato attuale, il progetto prevede solo costi per la manutenzione in up del server, senza che siano necessari particolari interventi di manutenzione periodici.

Il progetto utilizza unicamente librerie e API reperibili gratuitamente, ma sarebbe perfettamente possibile modificarle ed adottarne di colsed source a pagamento con pochi e semplici aggiustamenti grazie ad una gestione modulare del codice.

1.1.4 Aspetti tecnologici

La trasmissione di dati può essere effettuata in chiaro ad eccezione della password usata per accedere alla propria area personale, la quale viene trasmessa sotto forma di SHA256 per evitare che sia leggibile.

Il database prevede la creazione di una singola collezione contenente:

- *_id*: id univoco assegnato dal database in automatico
- *email*: email dell'utente
- *pw*: password dell'utente sotto forma di digest SHA256
- *pref*: array contenente le città messe tra i preferiti dall'utente. può essere vuoto

Tecnologie utilizzate:

- *HTML 5 e bootstrap*: realizzazione della struttura delle pagine e gestione degli stili con bootstrap;
- *JavaScript*: realizzazione delle richieste al server e alle API, nonché del toggle della darkmode e l'aggiornamento di frammenti di pagina con dati ricevuti dal server;
- *Node JS*: implementazione del server con tutte le sue funzionalità (caricamento di tutte le pagine, invio di oggetti JSON per aggiornare frammenti di pagina, interrogazione dell'API per ottenere il nome dei comuni e delle regioni italiani);
- *Express*: framework utilizzato per semplificare il deploy del server;
- *JSON*: formato usato per la trasmissione dei dati;
- *MongoDB*: database utilizzato per salvare le informazioni relative agli utenti registrati sull'app.