

---

---

# Realizzazione di un'applicazione web per la ricerca e la visualizzazione di eventi

WatchDogs • 10/05/2024

---

# Distribuzione del Team

Cuciniello Alex : ANALISTA

Del Fatto Andrea : QA

Morciano Gabriele : ARCHITETTO

---

---

---

# Evidenze delle cerimonie

Nel nostro team, abbiamo adottato la cerimonia stand-up, contraddistinta da brevità e concentrazione, garantendo così la sua massima efficienza. Durante questi incontri, ogni membro ha avuto l'opportunità di esprimersi liberamente senza essere interrotto, creando un clima di rispetto reciproco e assicurando che le voci di tutti fossero ascoltate.

Durante la cerimonia, ci siamo concentrati sui nostri obiettivi a breve termine e sulle attività fondamentali da completare entro la fine della settimana. Questo approccio ha mantenuto il team focalizzato sugli obiettivi comuni e ha facilitato l'individuazione tempestiva di eventuali ostacoli o sfide da affrontare.

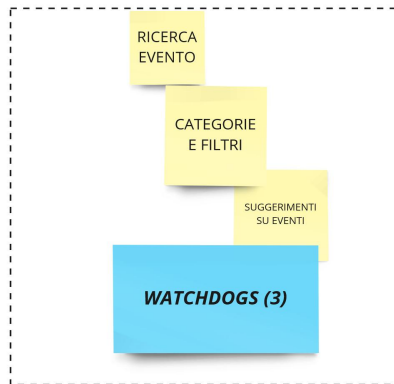
Inoltre, abbiamo promosso la trasparenza e la collaborazione fornendo aggiornamenti chiari e trasparenti sullo stato del lavoro. Questa pratica ha favorito un ambiente di lavoro aperto e cooperativo, consentendo al team di affrontare le sfide in modo collaborativo e di trovare soluzioni efficaci.

---

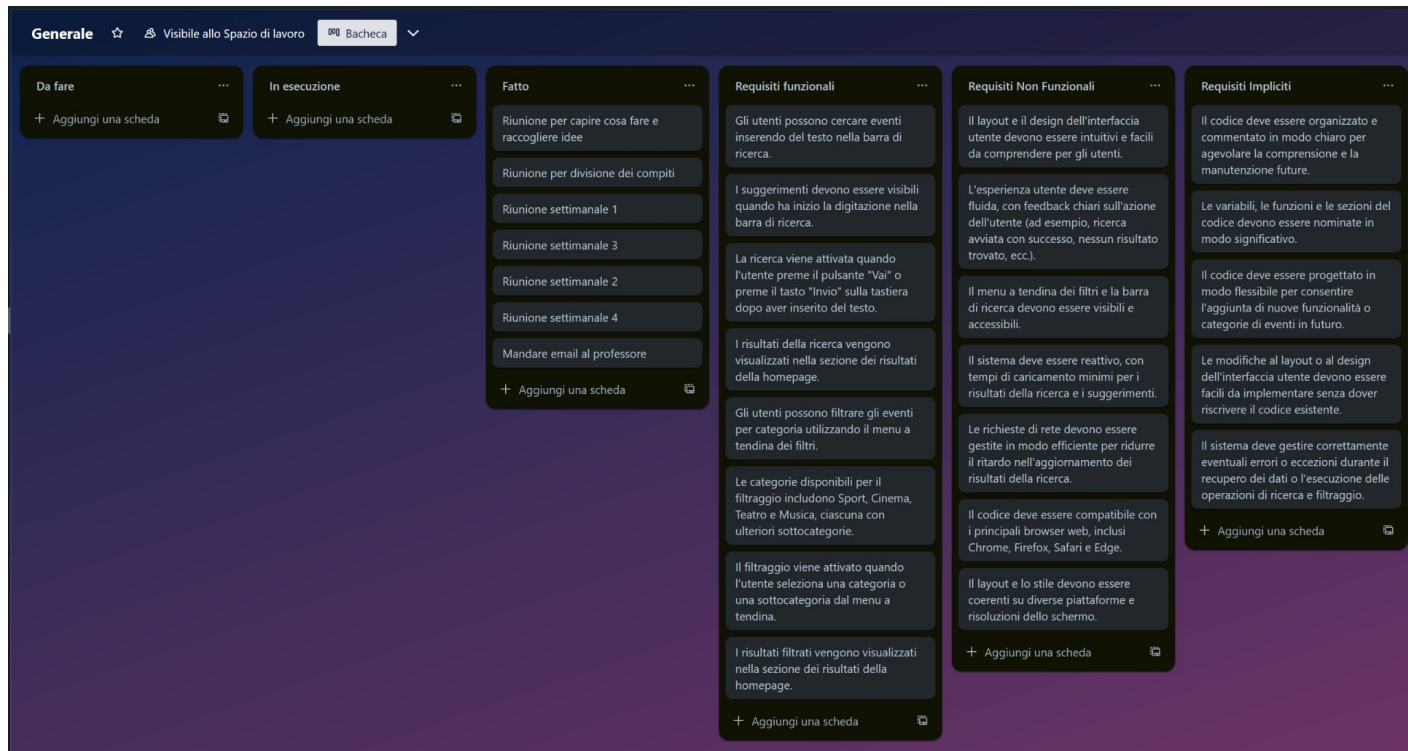
---

# Estrazione del disegno funzionale

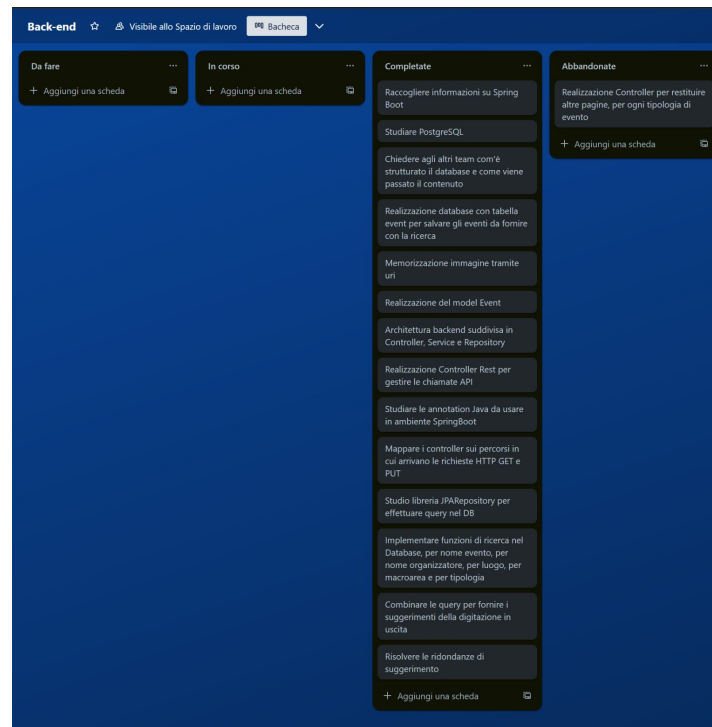
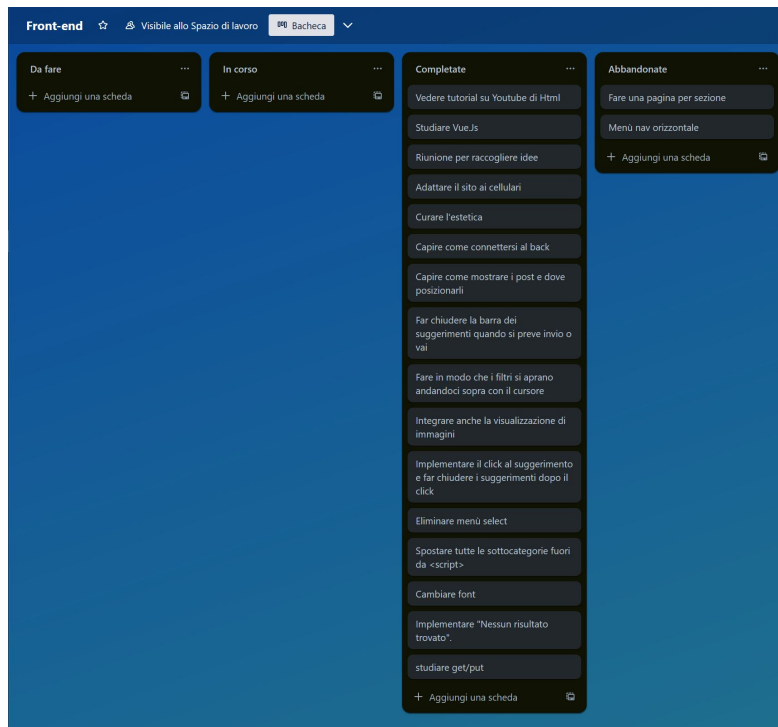
PORTALE  
PUBBLICO



# Screenshot di Trello



# Screenshot di Trello



---

# Test cases

## Test di navigazione dei filtri:

Descrizione: questo test verifica che i filtri del menu siano navigabili e che la selezione delle opzioni di filtro produca i risultati attesi.

Procedure:

Fare clic sul pulsante "Filtri" per aprire il menu dei filtri.

Navigare attraverso le opzioni di filtro per le varie categorie (Sport, Cinema, Teatro, Musica).

Selezionare una categoria e verificare che vengano visualizzate le sottocategorie corrispondenti.

Selezionare una sottocategoria e verificare che vengano visualizzati i risultati relativi alla categoria e sottocategoria selezionate.

## Test di ricerca degli eventi:

Descrizione: questo test verifica che la funzionalità di ricerca restituisca i risultati attesi in base all'input dell'utente.

Procedure:

Inserire un termine di ricerca nella barra di ricerca.

Premere il pulsante di ricerca o premere "Invio" sulla tastiera.

Verificare che vengano visualizzati i risultati correlati al termine di ricerca inserito.

Verificare che i risultati contengano informazioni accurate sugli eventi, inclusi il nome dell'evento, l'organizzatore, il luogo, la data e la descrizione.

---

---

# Test cases

## **Test di gestione degli errori:**

Descrizione: questo test verifica che l'applicazione gestisca correttamente gli errori e fornisca feedback utente chiaro in caso di problemi.

Procedure:

Provare a cercare un termine che non produce risultati.

Verificare che venga visualizzato un messaggio appropriato indicando che non sono stati trovati risultati.

Provare a cercare mentre la connessione di rete è disconnessa.

Verificare che venga visualizzato un messaggio di errore chiaro indicando il problema di connessione.

## **Test di compatibilità del browser:**

Descrizione: questo test verifica che l'applicazione sia compatibile con una varietà di browser web comuni.

Procedure:

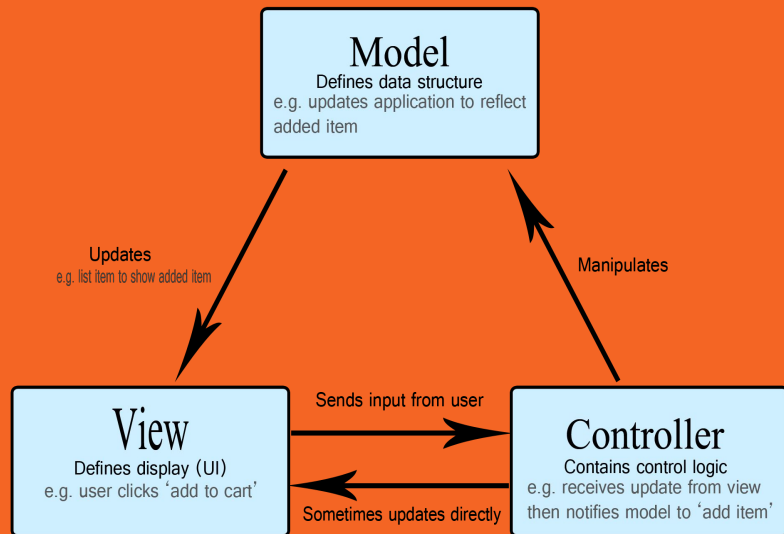
Aprire l'applicazione utilizzando browser web diversi come Chrome, Firefox, Safari e Edge.

Verificare che l'applicazione funzioni correttamente su ciascun browser, inclusa la corretta visualizzazione dell'interfaccia utente e il funzionamento delle funzionalità interattive.

---



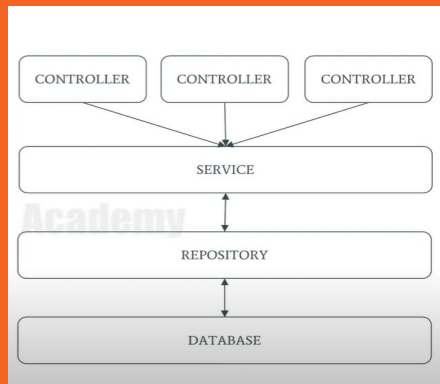
# Separazione frontend-backend (Vuejs + Springboot)



## Architettura MVC

L'architettura MVC è un pattern architetturale molto usato nel web, per separare la logica della presentazione (view, parte grafica), dalla logica di business (model, cuore dell'app che contiene dati e metodi per operare).

Il controller è il tramite tra view e model.



Architettura interna che prevede più livelli di astrazione per favorire sviluppi futuri, riuso del codice e riduzione delle ridondanze

---

# Funzionalità implementate

## **Barra di ricerca:**

- Consente all'utente di inserire un testo di ricerca.
- Filtra gli elementi in base al testo inserito dall'utente.
- Mostra suggerimenti di ricerca mentre l'utente digita.

## **Menu dei filtri:**

- Fornisce categorie di filtro per gli eventi, come Sport, Cinema, Teatro e Musica.
- Permette di selezionare una categoria principale.
- Mostra sottocategorie quando si passa sopra a una categoria principale.
- Invia richieste al server per ottenere eventi filtrati in base alla categoria selezionata.

## **Visualizzazione degli eventi:**

- Mostra i risultati della ricerca o del filtro degli eventi.
- Visualizza informazioni dettagliate sugli eventi, come nome, organizzatore, luogo, data e descrizione.

## **Pulsante di toggle dei filtri:**

- Permette di aprire e chiudere il menu dei filtri.
-

---

# Funzionalità implementate

## **Stile e layout:**

Utilizza stili CSS per la formattazione del layout e l'aspetto grafico dell'applicazione.

Utilizza un font personalizzato fornito da Google Fonts per l'aspetto estetico.

Utilizza JS in sinergia con HTML e CSS per animazioni dinamiche ed effetti in tempo reale.

## **Integrazione con Vue.js e Axios:**

Utilizza Vue.js per gestire la logica dell'interfaccia utente e la comunicazione con il back-end.

Utilizza Axios per effettuare richieste HTTP al server.

## **Recupero informazioni dal database:**

Back-end organizzato in Controller, Service e Repository per l'accesso al database, recupero e invio delle informazioni al front-end.

## **Risorse esterne:**

Importa librerie esterne come Vue.js e Axios per facilitare lo sviluppo e l'interazione con il back-end.

## **Compatibilità cross-browser:**

Utilizza un DOCTYPE HTML valido e definisce il charset e il viewport per garantire la compatibilità cross-browser e la corretta visualizzazione su dispositivi diversi.

---

# Funzionalità principale

La principale funzionalità del codice consiste nell'offrire agli utenti la possibilità di cercare eventi utilizzando una barra di ricerca e di filtrare gli eventi in base a categorie predefinite tramite un menu a discesa.

Attraverso l'utilizzo di Vue.js e Axios, il testo inserito viene trasmesso al backend per ricevere suggerimenti di ricerca in tempo reale.

Le comunicazioni tra Vue.js, Axios e i Controller REST del back-end sono la base del funzionamento della barra di ricerca: nell'applicazione Vue.js vengono definite funzioni grazie alle quali il componente Axios invia delle richieste HTTP ai path sui quali sono mappati i Controller. Questi ultimi, ottenuta la richiesta, interrogano il database tramite le funzionalità messe a disposizione dal framework Spring Data JPA, per fornire poi i risultati al front-end. Grazie a Vue.js la risposta viene gestita per fornire interattività e i risultati richiesti all'utente.

I suggerimenti vengono mostrati sotto la barra di ricerca mentre l'utente digita, consentendo un'esperienza interattiva e immediata.

I risultati della ricerca o del filtraggio degli eventi sono presentati in una sezione dedicata situata sotto la barra di ricerca e il menu dei filtri, fornendo agli utenti un'organizzazione chiara e intuitiva delle informazioni visualizzate.

---

---

---

# Il nostro feedback

- Il progetto ha fornito una preziosa opportunità di apprendimento pratico, collaborazione interpersonale e sviluppo professionale nel campo dello sviluppo web.
  - Ha richiesto una predisposizione all'apprendimento autonomo e al problem solving.
  - La partecipazione all'interno di un team ha favorito un ambiente collaborativo e stimolante, che ha contribuito alla realizzazione efficace del progetto.
  - L'interazione con persone appartenenti ad altri team ha ampliato le prospettive e favorito lo scambio di idee e conoscenze.
  - La gestione efficiente delle risorse, incluse il tempo e le competenze disponibili, è stata essenziale per il successo del progetto. La capacità di organizzazione e pianificazione è stata pertanto fondamentale.
-