

T3 Chess Cake

Presentazione del lavoro svolto

Petru Marcel Marincas - PO



23/11/2023

Sprint 2 goals

I nostri obbiettivi per questo sprint erano i seguenti:

- Terminare le User Stories lasciate incomplete dello Sprint 1
 - Questo quindi voleva dire avere scacchi senza bug, avere una generazione casuale migliore, utente contro utente locale, player contro computer, salvare le partite e registrarsi ed accedere usando un DB.
- Impostare l'ambiente di produzione
- Creare il timer, in particolare per giocatore contro giocatore (locale)
- Creare l'interfaccia delle leaderboard e vedere i dati utili.

Sprint 2 Review

Per la review della prima sprint è stata fatta come video e caricata YouTube (come video non in elenco). Si trova a [questo link](#).

Per queste slide, in quanto servono alla presentazione, nella prossima diapositiva si ha il video caricato per essere presentato.



Chess Cake

Username



Password



[Registrati](#)

Login →

[Continua come anonimo](#)

Stato attuale dello sprint

- Abbiamo recuperato il debito accumulato dallo scorso sprint.
- Abbiamo un gioco funzionante e una gestione delle utenze.
- Gestiamo correttamente il tempo come richiesto per questo sprint.
- Giocare contro il computer è competitivo e interessante.
- Stiamo raffinando il nostro algoritmo di generazione della scacchiera in base al rank: dobbiamo trovare un equilibrio tra partite più interessanti e partite più ragionevoli, e gestendone la casualità.
- La modalità utente contro utente locale è quella più vicina ad una versione definitiva presente nel prodotto finale.

Aggiornamenti su traguardi tecnici

- Siamo riusciti a metterlo in produzione (a [questo link](#))
- Stiamo utilizzando MongoDB per i dati
 - Per ora stiamo usando MongoDB Atlas, ovvero un servizio gratuito che offre basse prestazioni per un database di MongoDB. (più informazioni [qui](#))
 - In future passeremo ad usare quello offerto dai server del dipartimento.
- Ci stiamo appoggiando a Stockfish
 - L'implementazione del computer come avversario di scacchi usa internamente Stockfish, o meglio una sua libreria scaricabile per JavaScript.

Testing

Abbiamo iniziato con il testing di diverse componenti. Per ora abbiamo una testing suite complete per:

- La generazione della board e scacchi
- Le API di login, registrazione e le route protette

```
PASS ./boardFunctions.test.js
PASS ./userAPI.test.js (5.351 s)
```

```
Test Suites: 2 passed, 2 total
Tests:      29 passed, 29 total
Snapshots:  0 total
Time:       6.537 s
Ran all test suites.
```

Problemi riscontrati e attuali

- Durante il primo weekend dello sprint ci sono stati dei problemi con il server DISI che ci hanno impedito di lavorare.
 - Consideriamo che questo abbia impattato negativamente le prestazioni durante questo sprint, in quanto abbiamo potuto sfruttare meno del nostro tempo libero per sviluppare.
- Dobbiamo trasferire la gestione della partita di scacchi dal frontend al backend, per gestire la sicurezza da attacchi del gioco, in modo da non poterne simulare il comportamento in modo errato.
- Non stiamo usando interamente tutto self-hostando, in quanto vi è il placeholder attuale di MongoDB Atlas

Piani per il prossimo sprint

Per il prossimo sprint abbiamo i seguenti obbiettivi:

- Finire la gestione delle leaderboard e il salvataggio delle partite
- Implementare l'area utente, dove poter vedere le partite salvate
- Integrare delle funzionalità con i social network (condividere partite)
- Sviluppare la modalità Player vs Mob

Da un punto di vista tecnico, inoltre:

- Portare scacchi dal frontend al backend
- Usare MongoDB offerto dal dipartimento piuttosto che Atlas