

# Presentazione del progetto

Osichess-Team 2

Marco Coppola 0001020433

Valerio Pio De Nicola 0001043353

Chiara Tosadori 0001043695

Bogdan Chirila 0001028516

Ahmed Niouer 0001027288

<https://osichess.murkrowdev.org>



# Ruoli

**Chiara Tosadori:**  
Sviluppatrice, Frontend,  
Scrum Master

**Marco Coppola:** Sviluppatore,  
Backend, Amministratore di  
sistema

**Valerio Pio De Nicola:** Product  
Owner, Frontend, Backend

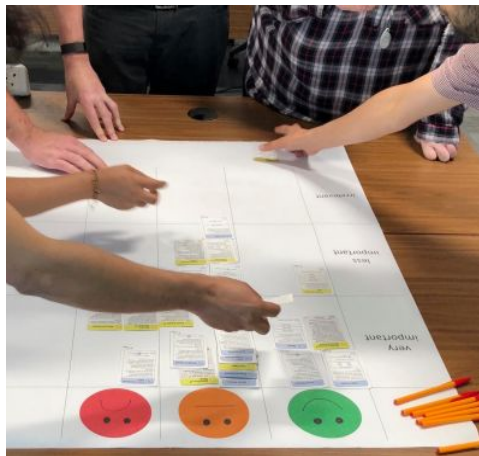
**Bogdan Chirila:** Analista delle  
prestazioni

**Ahmed Niouer:** Designer,  
UI/UX



# Tecniche per le retrospettive

## Carte essence



## La stella marina



## Tabella di autovalutazione

Cards	Valerio Pio De Nicola PO	Marco Coppo la DEV	Chiara Tosado ri SM	Bogdan Chirila DEV	Ahmed Nouei DEV	motivazioni
scrum master						
Product Owner						
Scrum Team						
Develop ers						
Product backlog						
Sprint Planning						
Daily scrum						
Sprint Goal						
Self manage ment						

# Mezzi di comunicazione

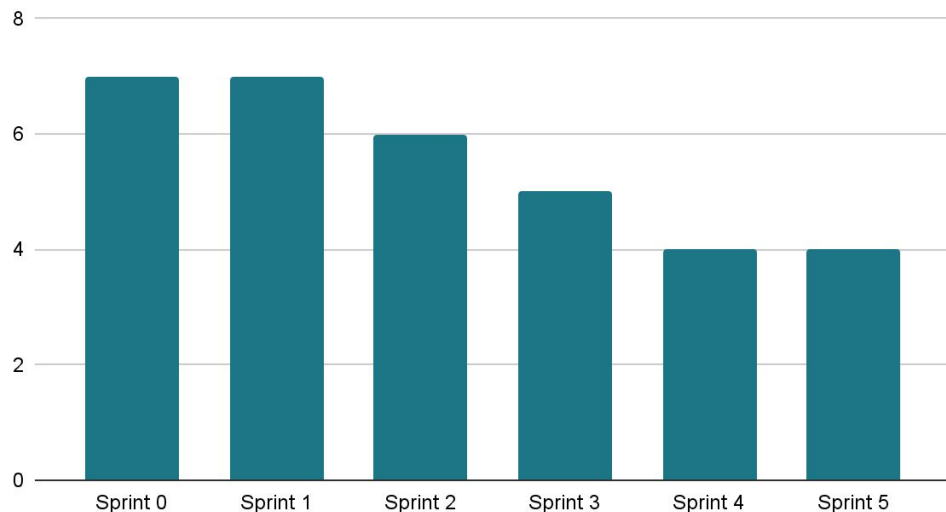
Il principale mezzo di comunicazione testuale è stato il gruppo Telegram, su cui appunto venivano organizzate le riunioni e avvenivano varie comunicazioni. Ci eravamo preposti di utilizzare Mattermost, ma abbiamo preferito continuare su Telegram per la sua immediatezza e soprattutto per le notifiche.

Invece per gli scrum è stato utilizzato Microsoft Teams, che ci ha anche aiutato a tener traccia delle durate delle varie riunioni.

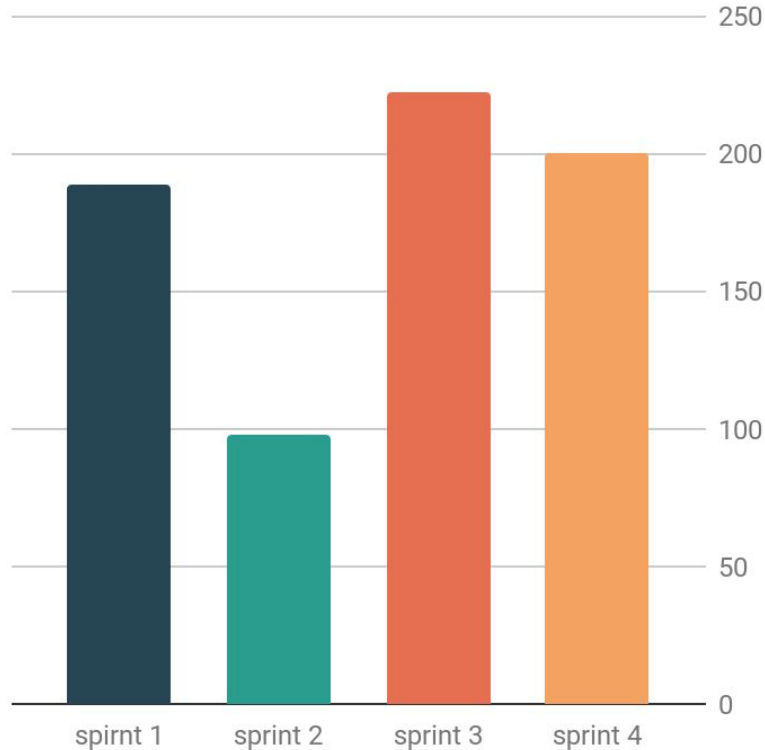
# Statistiche Scrum

In totale abbiamo accumulato circa **80 ore** di lavoro, di cui **30 ore** sono state impiegate per gli scrum, in media la durata delle riunioni era di **1 ora**

Scrums per sprint



# Velocity



Velocity sprint 1: 189

Velocity sprint 2: 98

Velocity sprint 3: 222

Velocity sprint 4: 200

Velocity media: 177.25

# Roadmap Infographics

- Configurazione ambiente di lavoro
- Team building
- Prime US



- Partita contro Bot
- DB strutturato
- Autenticazione
- Frontend essenziale



- Partite PVP
- Timer di gioco
- Statistiche
- Modifica profilo



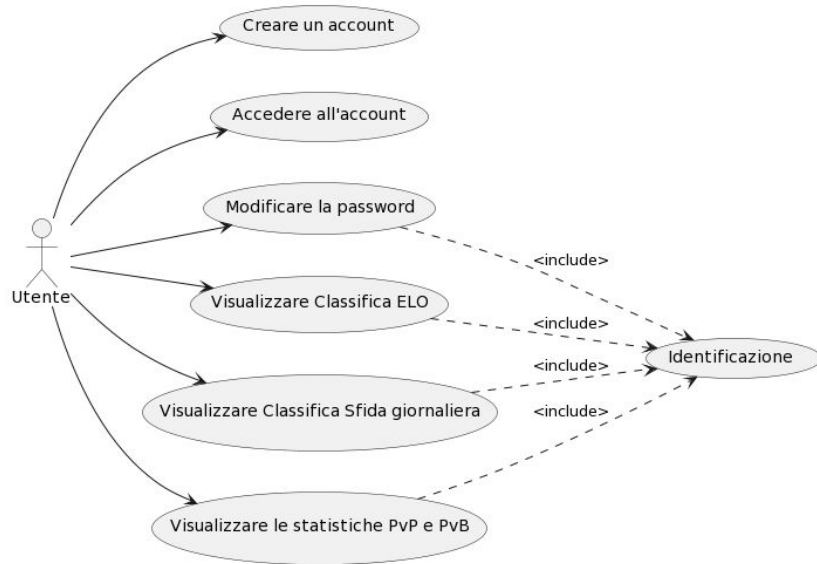
- Rating ELO
- Classifica
- Bot di Telegram



- Daily Challenge
- Resign Partite
- Classifica daily challenge



# Use case





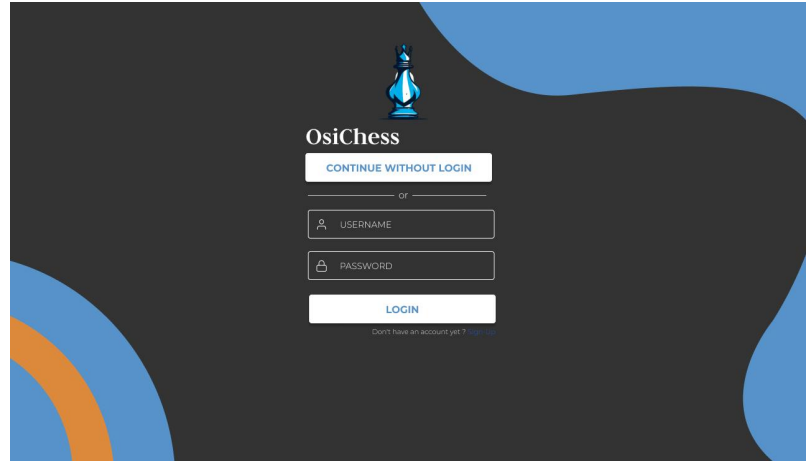
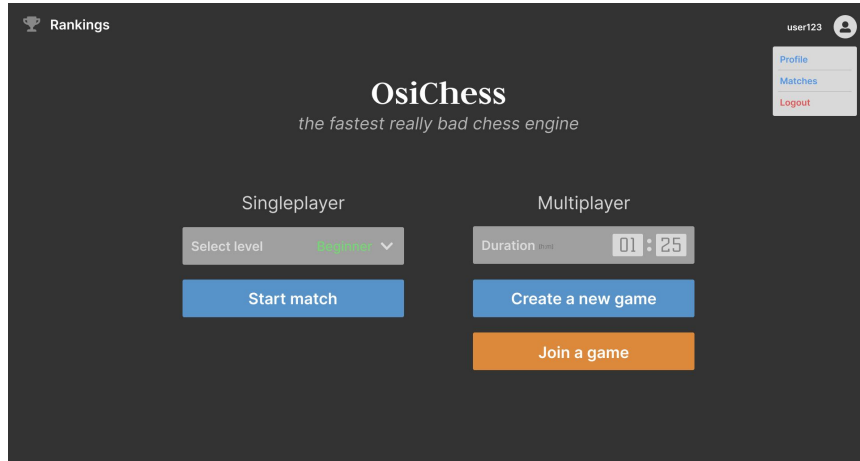
# Backlog del prodotto

- Come giocatore vorrei poter giocare una partita in PvB. **36 punti**
- Come utente non registrato vorrei poter accedere all'app con funzionalità limitate. **14 punti**
- Come giocatore vorrei poter avviare una partita in PvB. **28 punti**
- Come utente vorrei poter creare un account. **48 punti**
- Come utente vorrei poter accedere al mio account. **38 punti**
- Come giocatore, voglio poter visualizzare le mosse legali per il pezzo selezionato durante la partita. **25 punti**
- Come utente vorrei poter accedere al mio profilo e modificare la password **14 punti**
- Come giocatore vorrei poter giocare una partita in PvP . **30 punti**
- Come giocatore, voglio poter leggere il tempo a disposizione residuo durante la partita .**28 punti**
- Come giocatore, voglio poter selezionare la modalità di gioco (PvB o PvP) durante la creazione di una partita. **11 punti**
- Come giocatore vorrei potermi unire ad una partita inserendo il suo codice **29 punti**
- Come giocatore voglio poter annullare la mia mossa . **16 punti**
- Come giocatore, voglio poter selezionare la difficoltà della partita con uno slider che rappresenta la differenza di potenza delle configurazioni .**11 punti**
- Come giocatore vorrei poter impostare la durata della partita prima di crearla. **21 punti**
- Come utente vorrei poter vedere la classifica dei migliori utenti basandosi sul rating ELO. **35 punti**
- Come giocatore vorrei essere identificato anche con un ranking ELO che guadagno o perdo giocando le partite. **20 punti**
- Come giocatore, vorrei poter leggere il mio nome utente e quello del mio avversario durante la partita . **25 punti**
- Come utente vorrei poter vedere il numero di mosse che sono state effettuate in una partita contro il bot. **18 punti**
- Come utente vorrei poter abbandonare una partita in corso premendo il tasto resign. **34 punti**
- Come utente vorrei poter visualizzare una classifica dei giocatori che hanno completato in meno mosse la sfida del giorno X. **50 punti**
- Come giocatore vorrei poter partecipare ad una sfida giornaliera che presenta una configurazione da battere nel minor numero di mosse possibile. **44 punti**
- Come utente vorrei poter scegliere se continuare o chiudere una partita in corso quando ne inizio un'altra. **33 punti**
- Come utente vorrei poter abbandonare una partita inserendo il comando resign sul bot di telegram. **21 punti**
- Come giocatore vorrei durante la partita poter vedere entrambi i timer (il mio e quello del mio avversario). **20 punti**
- Come utente vorrei poter giocare contro il bot anche attraverso un bot di telegram. **20 punti**
- Come giocatore voglio poter condividere il link di accesso alla partita su dei social. **20 punti**
- Come giocatore, voglio poter salvare i risultati della partita appena conclusa. **20 punti**

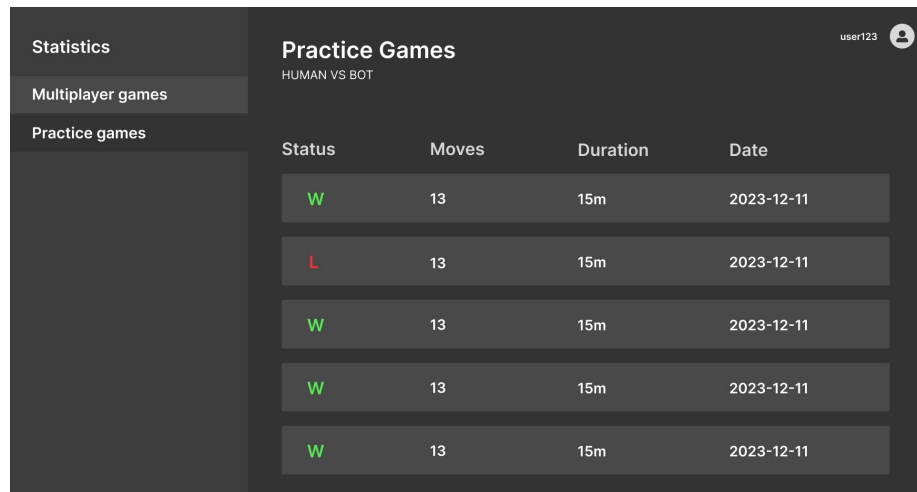
# Definition of Done

- **Implementazione:** L'implementazione del codice è completata, rispettando gli standard di sviluppo e le linee guida dell'organizzazione.
- **Test di Unità:** Il codice (back-end) è stato sottoposto a test di unità per verificare che le singole unità di software funzionino correttamente.
- **Integrazione:** Il codice è stato integrato con successo nell'ambiente di sviluppo
- **Revisione del Codice:** Il codice è stato sottoposto a una revisione da parte dei membri del team per garantire la qualità e la coerenza.
- **Approvazione del Product Owner:** Il Product Owner ha esaminato l'incremento e lo ha accettato come completo rispetto ai requisiti concordati.
- **Distribuzione:** La feature é pronta per essere unita al branch principale e distribuita in produzione.

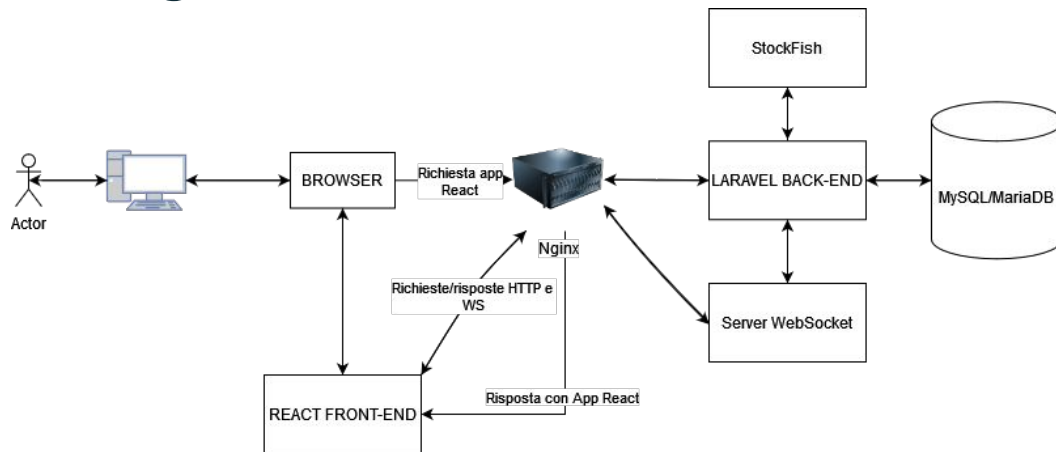
# Mockup - 1



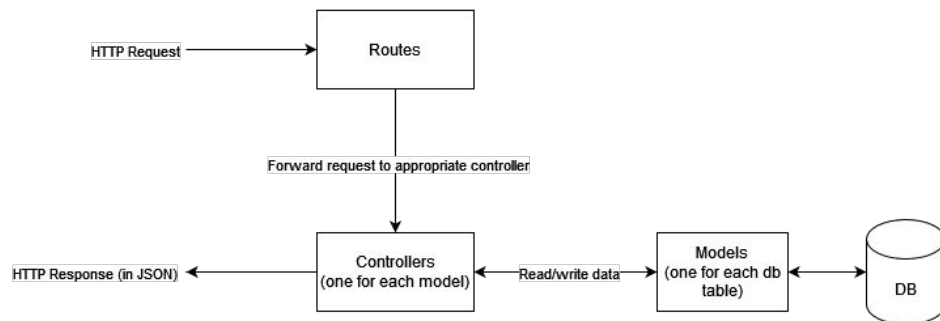
# Mockup - 2

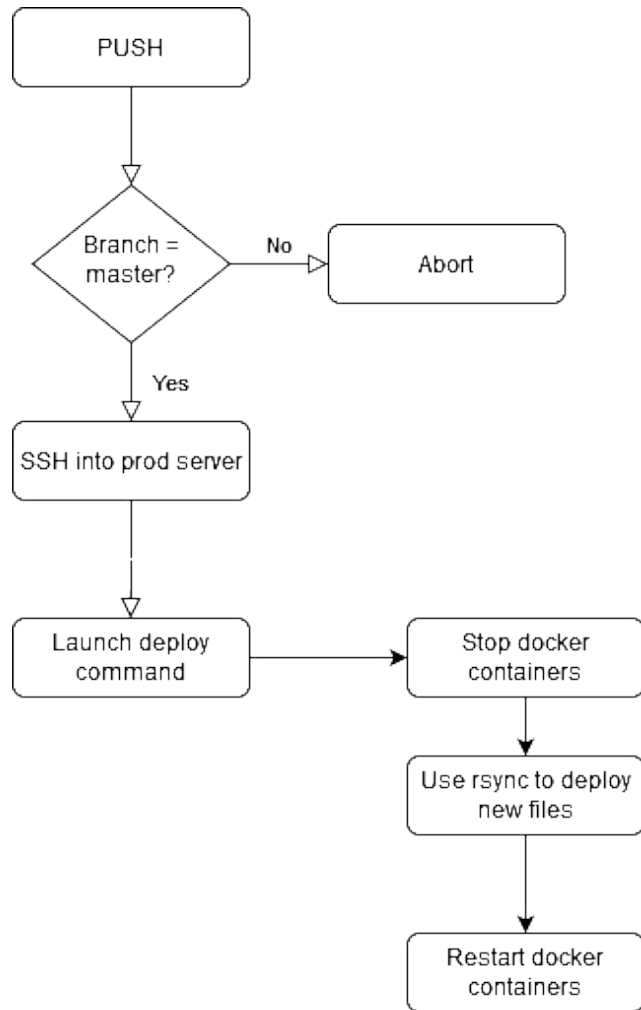


# Diagramma dell'architettura



Laravel Back-end





## Diagramma CI-CD

# Ambiente Locale e Ambiente di Produzione

Attraverso l'utilizzo di Docker abbiamo reso l'ambiente locale di sviluppo e quello di produzione pressoché identico. Questo ha permesso di ridurre di molto tutte le potenziali problematiche che potevano sorgere durante la pubblicazione.

Oltre all'applicazione, anche tutti i servizi di processo e di sviluppo (Git, Taiga, Mattermost, Sonarqube) sono situati su un server privato di proprietà di uno dei membri del gruppo.

# Diagramma ER

