

Lupus

Sistemi Distribuiti

A.A. 2023/2024

Andrea Serafini



LUPUS
IN
FABULA

Introduzione



Digitalizzazione del gioco *Lupus in fabula*

Regole:

- Ogni giocatore ha una carta che ne determina il **ruolo**
- Ogni ruolo ha il suo compito e agisce sullo **stato** degli altri giocatori
- Un narratore gestisce le **fasi** del gioco
- Due fasi: diurna e notturna

Requisiti di progetto

Funzionali:

- **Login** utente
- Concetto di **lobby**
- **Personalizzazione** della partita
- Gestione dei **turni** e della fasi di gioco
- **Salvataggio** delle partite

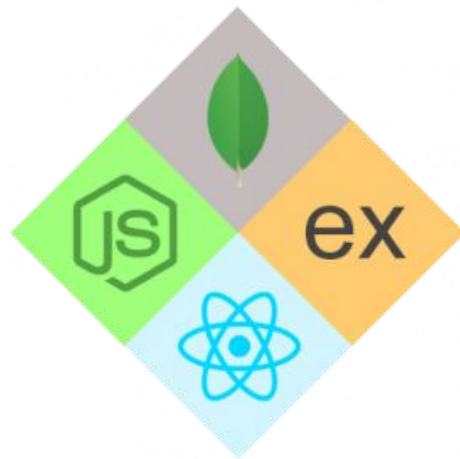
Non funzionali:

- Sistema **scalabile**
- **Architettura** specifica
- Gestione dei **fallimenti**

Tecnologie impiegate

Stack **MERN**:

- **M**ongoDB
- **E**xpressJS
- **R**ead
- **N**odeJS



Deployment:

- Docker
- GitHub Pages

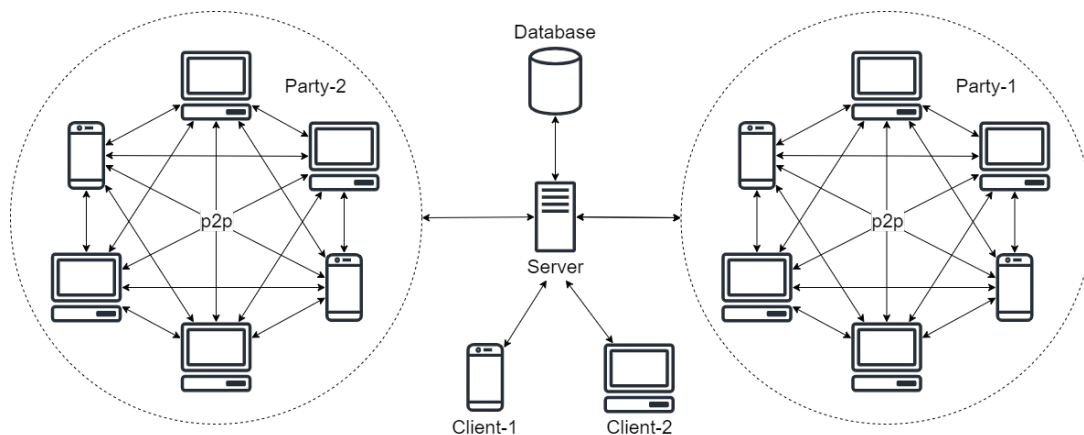
Interazione:

- Socket.IO
- PeerJS



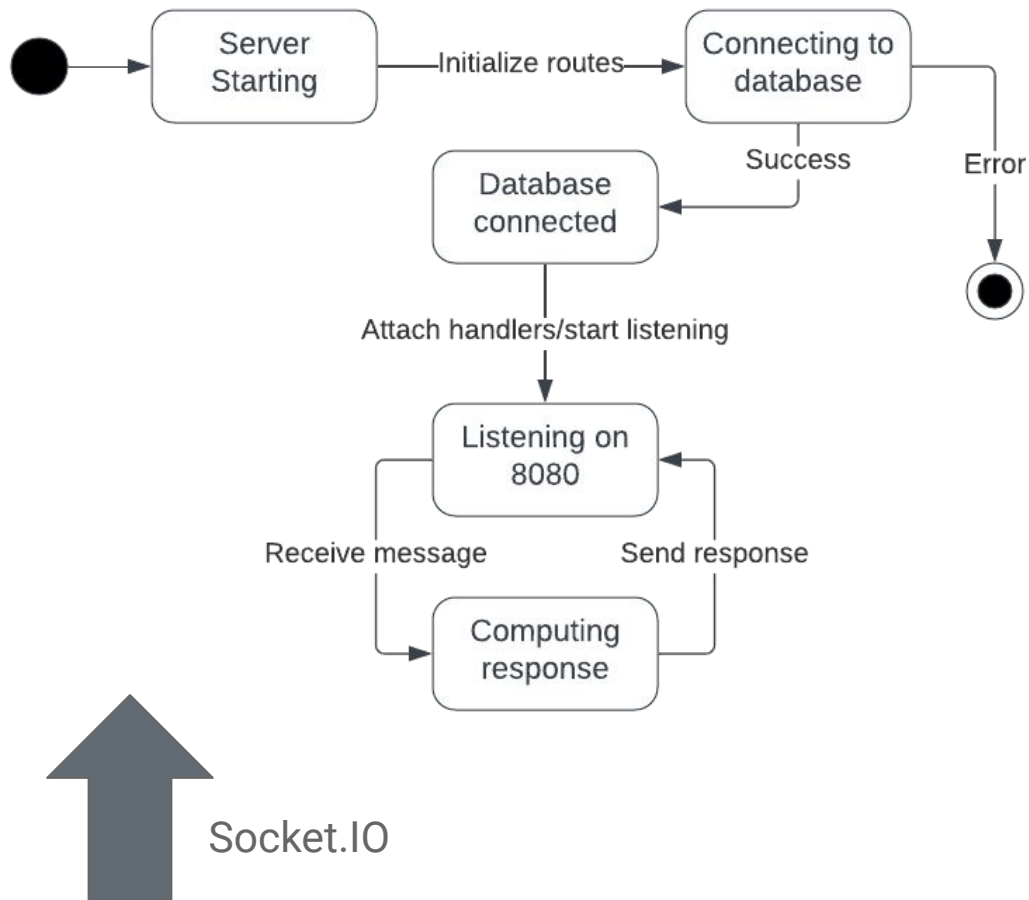
Architettura

- Peer-to-Peer
- Discovery server
- Dispositivi eterogenei



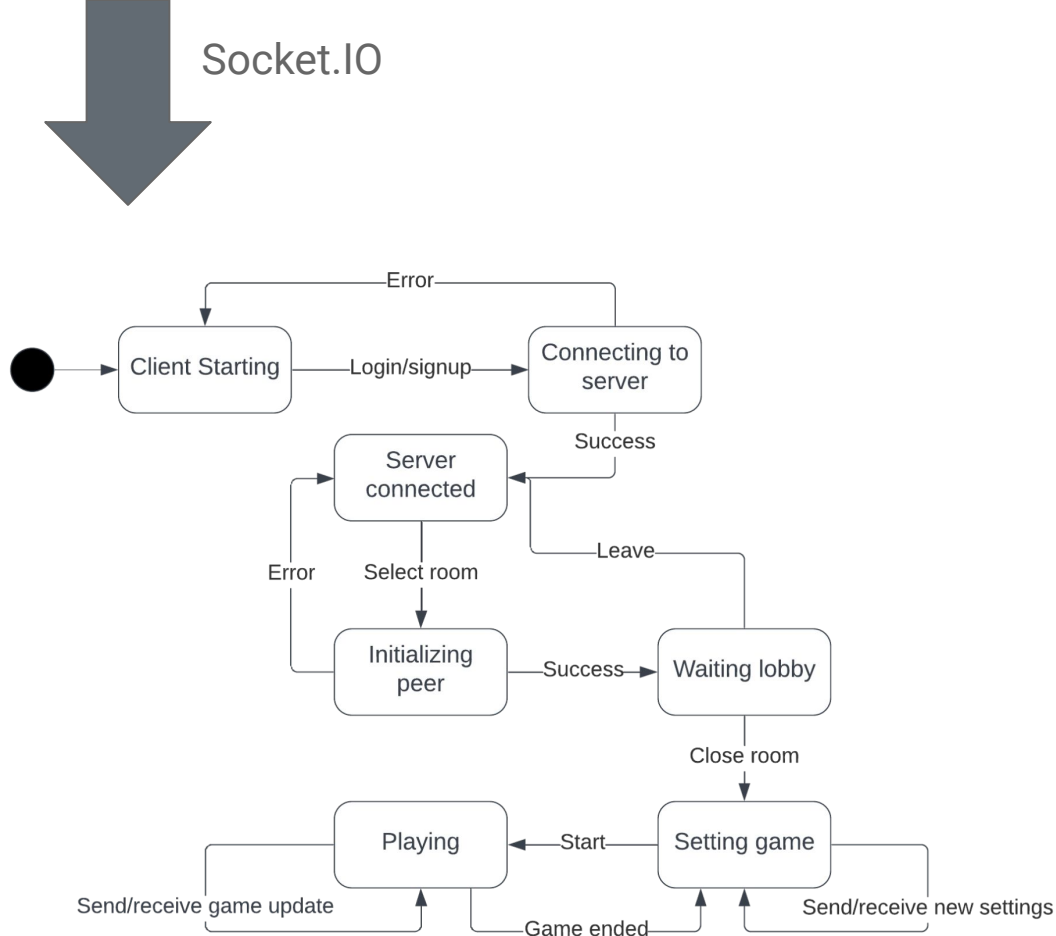
Server side

- Gestione database:
 - Autenticazione
 - Storico partite
- Interazione con i client
 - Socket.IO
 - Discovery server



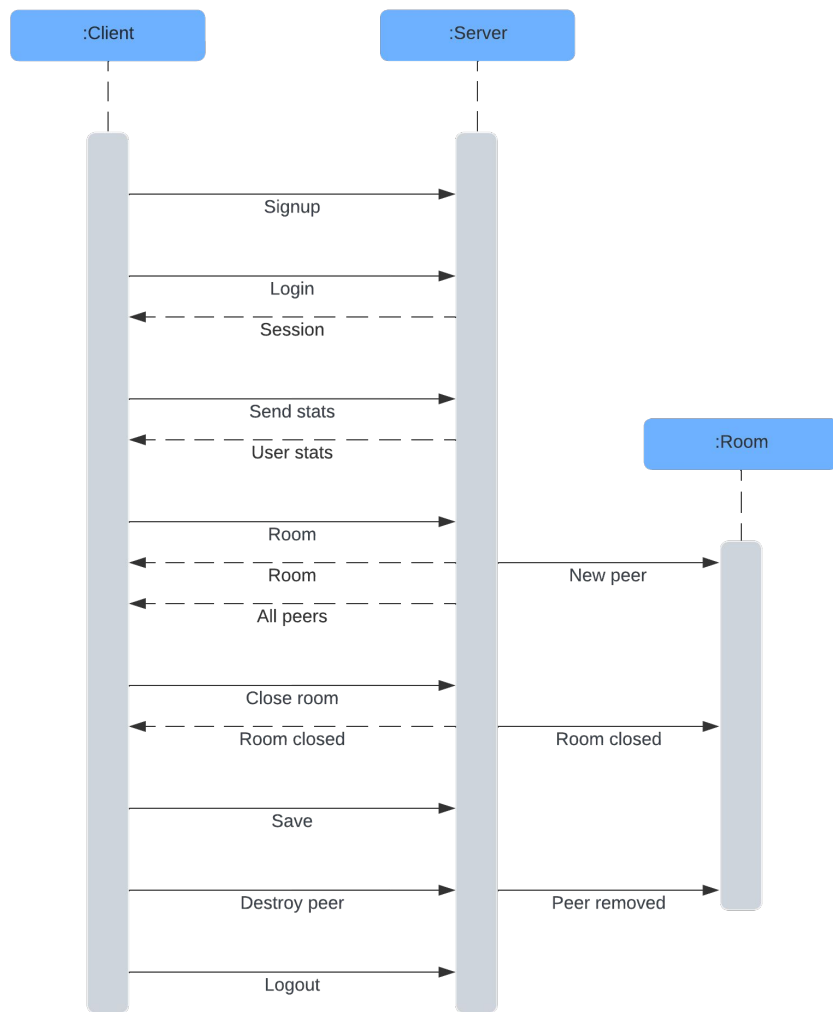
Client side

- Architettura peer-to-peer
- Incorpora la logica di gioco
- Gestione degli stati con Redux



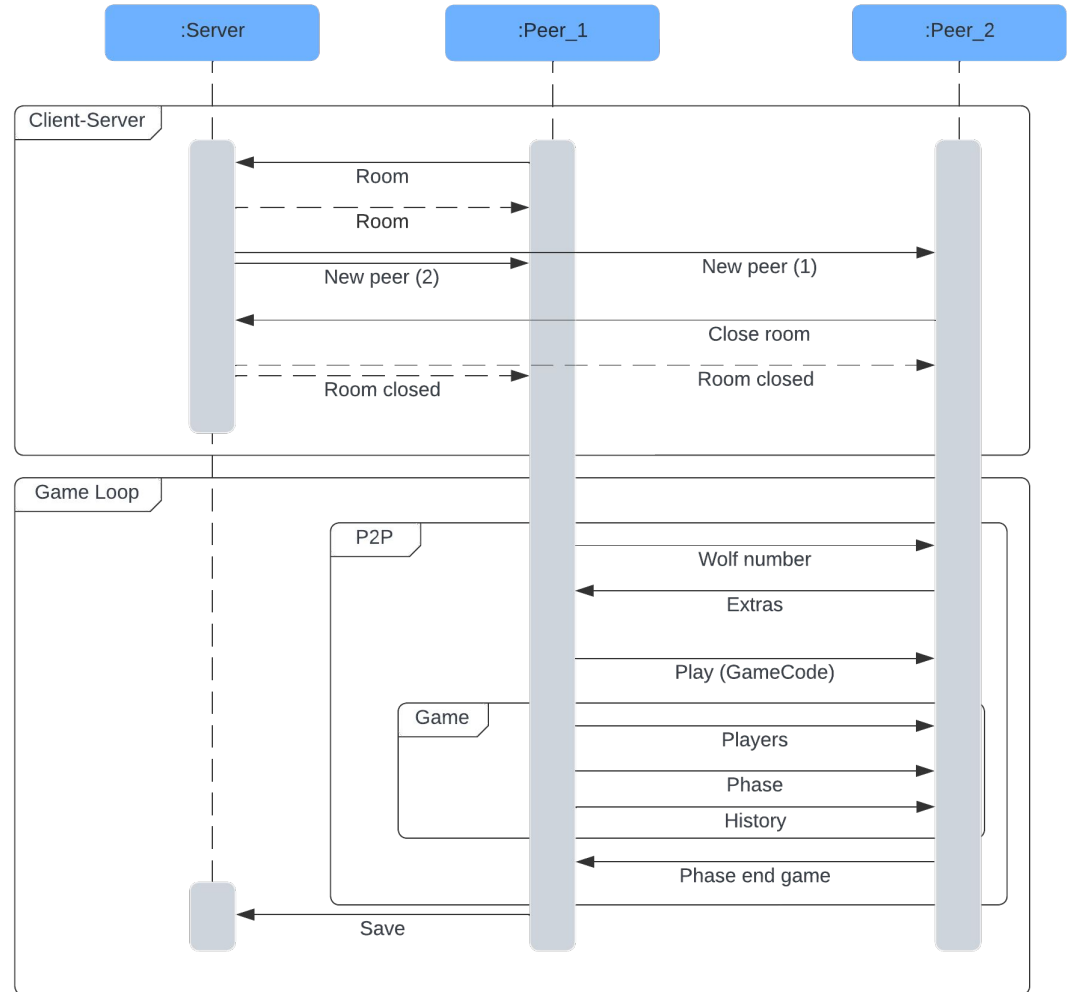
Interazione client-server

- Operazioni utente
- Interazioni database
- Socket.IO rooms
- Gestione fallimenti con **sessionID**



Interazione peer-to-peer

- Messaggi in broadcast
- Gestione dei fallimenti con ripristino stato



Deployment

Locale:

- npm
- vs.
- Docker

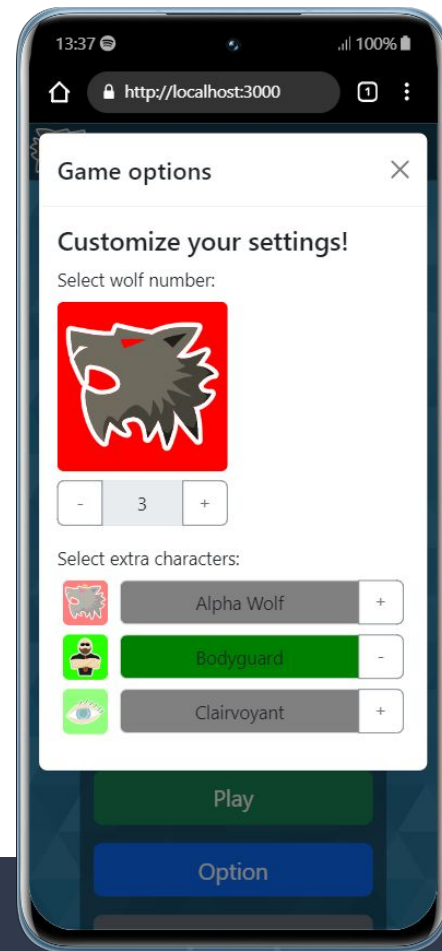
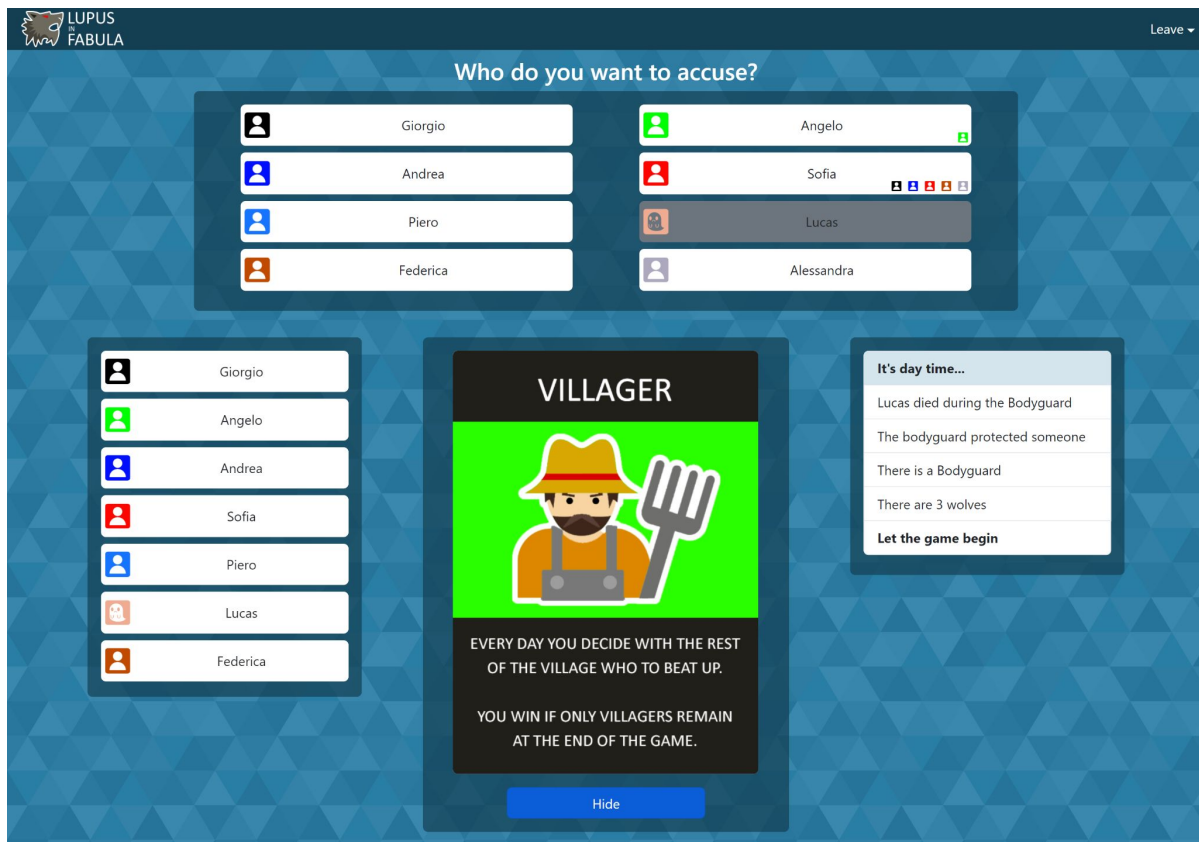
Distribuito:

- Ngrok
- GitHub Pages

Serving!

- Local: `http://localhost:3000`
- Network: `http://192.168.10.1:3000`

Copied local address to clipboard!



Grazie per l'attenzione

