



PIXIE

COLLABORATIVE REAL-TIME PIXEL ART PLATFORM

Esame di Distributed Systems | 2025/2026

Andrea Zavatta & Matteo Susca



PROSPETTIVA UTENTE E

Funzionalità e Interfaccia Cliente

COS'È PIXIE??

- > Piattaforma collaborativa web-based
- > Creazione di pixel art in tempo reale
- > Interazione simultanea su una tela condivisa
- > Esperienza di navigazione fluida e reattiva



FUNZIONALITÀ CHIAVE



LOBBY

Creazione di lobby
indipendenti con
dimensioni e palette
personalizzate



LIVE SYNC

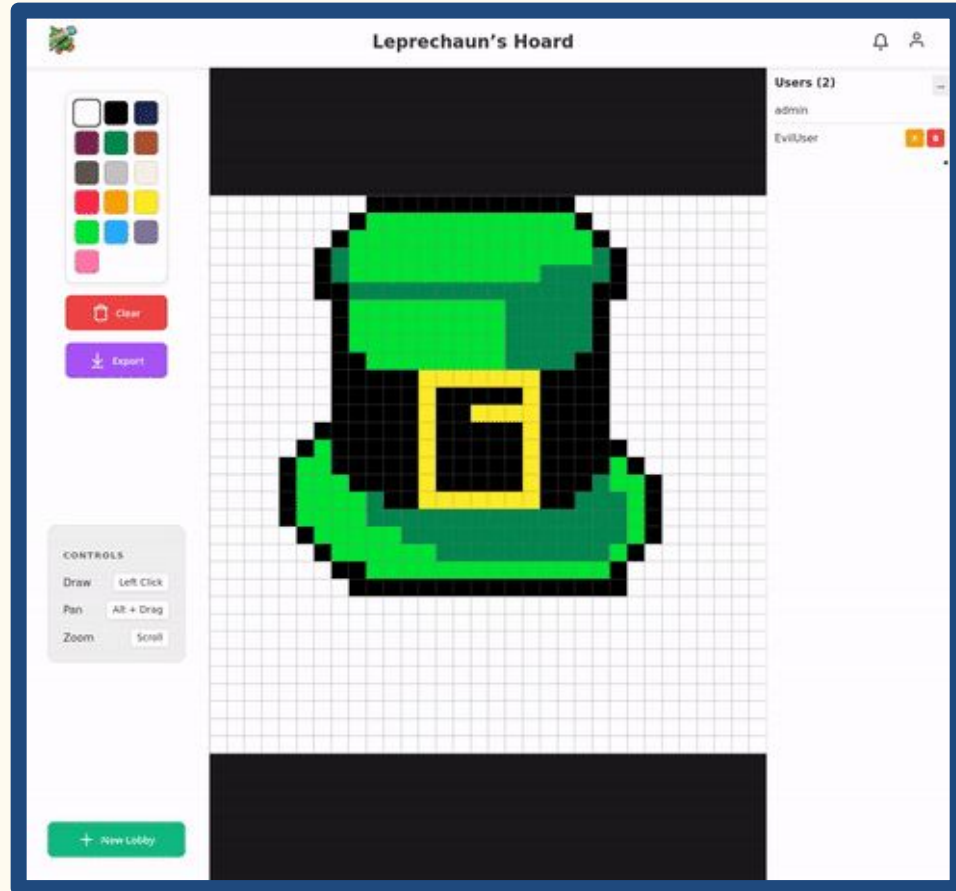
Propagazione in tempo reale
di ogni pixel disegnato
alle persone connesse nella
stessa lobby



NOTIFICHE PUSH

Ricezione istantanea di
notifiche di sistema

RUOLI E MODERAZIONE



TRE TIPOLOGIE DI UTENTE

- > **Base User:** Disegna, si unisce alle lobby e visualizza gli utenti connessi.
- > **Lobby Owner:** Proprietario della lobby, con poteri di kick, ban e pulizia della tela.
- > **Admin:** Privilegi globali per moderare intero sistema e utenti.



Demo Applicazione



Welcome to Pixie

Enter your name...

Enter



Demo Applicazione





Join a Canvas

 Search Canvases

AllMy Canvases

Create New Canvas

Fablewood Crossroads
by admin
6/9/2026
Join Canvas

Leprechaun's Hoard
by admin
6/9/2026
Join Canvas

tinkerbell
by admin
6/9/2026
Join Canvas



Join a Canvas

 Search Canvases

AllMy Canvases

Create New Canvas

Fablewood Crossroads
by admin
6/9/2026
Join Canvas

Leprechaun's Hoard
by admin
6/9/2026
Join Canvas

tinkerbell
by admin
6/9/2026
Join Canvas



Demo Applicazione



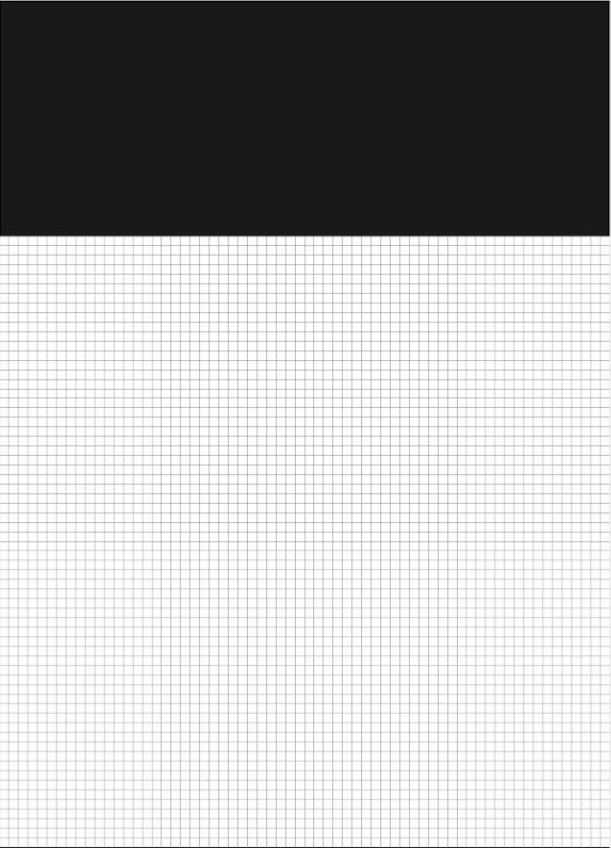


My First Canvas









Users (1)

DemoUser

CONTROLS

Draw

Left Click

Pan

Alt + Drag

Zoom

Scroll

+ New Lobby



Join a Canvas



AllMy Canvases

+ Create New Canvas

My First Canvas
by DemoUser
6/25/2026
Join Canvas

Fablewood Crossroads
by admin
6/9/2026
Join Canvas

Leprechaun's Hoard
by admin
6/9/2026
Join Canvas

tinkerbell
by admin
6/9/2026
Join Canvas



Demo Applicazione



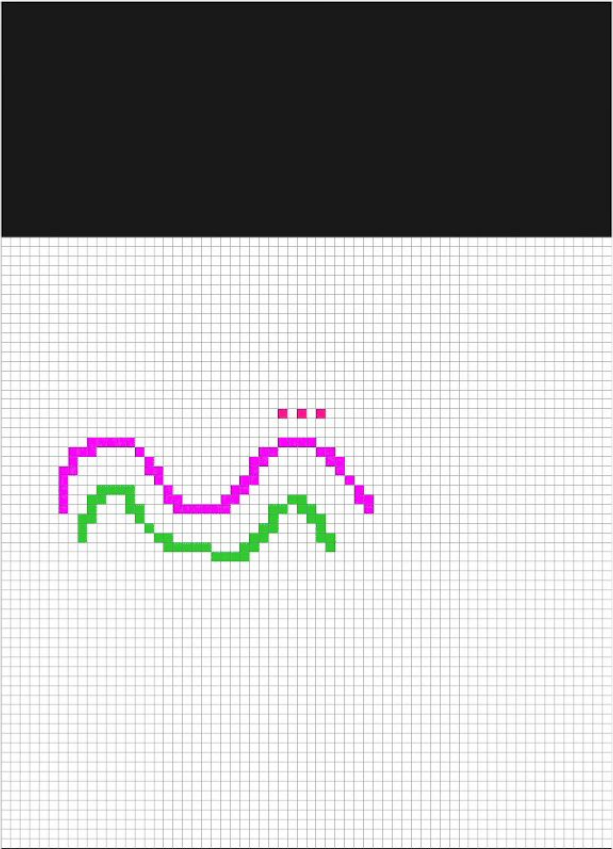


My First Canvas









Users (2)

DemoUser

sysosus  

CONTROLS

Draw 

Pan 

Zoom 



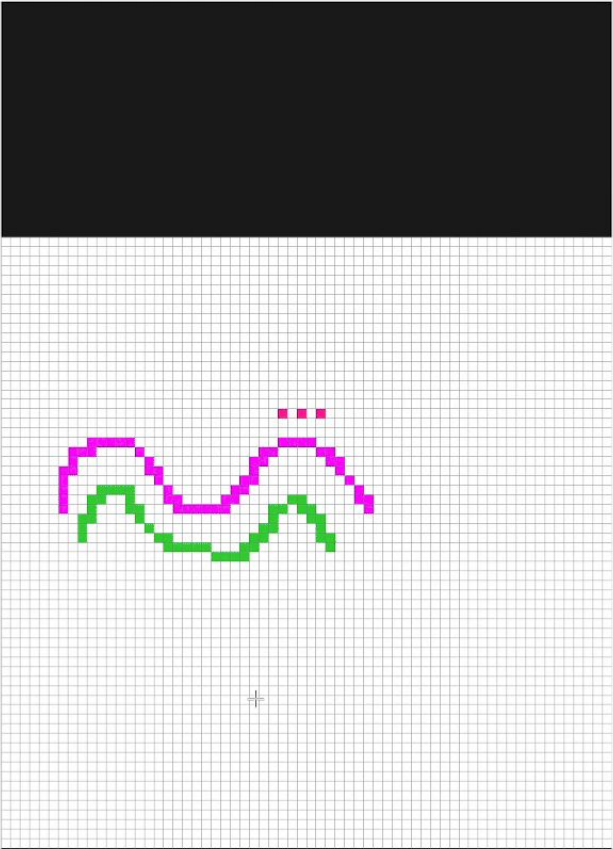


My First Canvas









Users (2)

sysosus

DemoUser

CONTROLS

Draw 

Pan 

Zoom 



Demo Applicazione



My First Canvas

 Clear

CONTROLS

Draw Left Click

Pan Alt + Drag

Zoom Scroll

Users (1)

DemoUser



Join a Canvas



 Search Canvases

All My Canvases

[+ Create New Canvas](#)

My First Canvas

by DemoUser

6/25/2026

Fablewood Crossroads

by admin

6/9/2026

Leprechaun's Hoard

by admin

6/9/2026

tinkerbell

by admin

6/9/2026

Demo Applicazione



The screenshot shows a web-based drawing application. At the top, the title "My First Canvas" is centered. On the left side, there is a color palette with 16 colored squares (white, black, and various primary/secondary colors) and two buttons: "Clear" (with a trash icon) and "Export" (with a download icon). Below the palette is a "CONTROLS" section with three rows: "Draw" with a "Left-Click" button, "Pan" with an "Alt + Drag" button, and "Zoom" with a "Scroll" button. The main area is a large canvas with a white grid on a black background. On the canvas, there are several hand-drawn lines in blue, red, green, and magenta. On the right side, there is a sidebar with the text "Users (1)" and "DemoUser" below it. At the bottom left, there is a green button with a plus icon and the text "New Lobby". At the bottom right, there is a red icon of a person with a slash and the text "Banned Users (1) ►".

The image is a screenshot of the 'Join a Canvas' interface in Canvas LMS. At the top, there's a header with the Canvas logo on the left, the title 'Join a Canvas' in the center, and notification and user icons on the right. Below the header is a search bar labeled 'Search Canvases' and two tabs: 'All' and 'My Canvases'. A blue button labeled '+ Create New Canvas' is positioned to the right of the tabs. The main content area displays four canvas cards. Each card has a title, a creator name, a date, and a 'Join Canvas' button. The cards are: 'My First Canvas' by DemoUser (dated 6/25/2026), 'Fablewood Crossroads' by admin (dated 6/9/2026), 'Leprechaun's Hoard' by admin (dated 6/9/2026), and 'tinkerbelle' by admin (dated 6/9/2026). A mouse cursor is visible near the bottom center of the page.



Demo Applicazione





My First Canvas

🔔 👤

🔲

🔳

🔴

🔵

🟡

🟢

🟠

🟣

🟤

🟡

🟢

🟠

🟣

🟤

🟡

🟢

🟠

🟣

🟤

🗑️ Clear

📄 Export

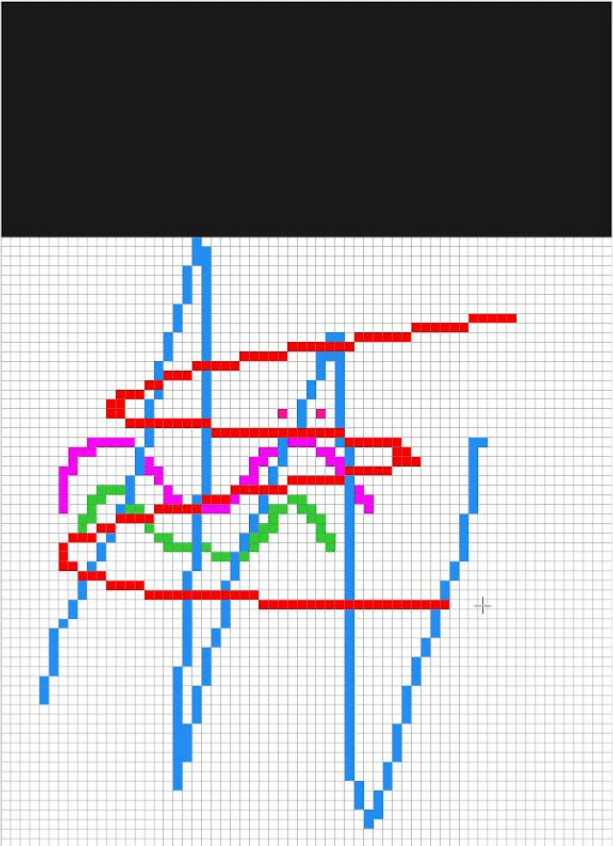
CONTROLS

Draw Left Click

Pan Alt + Drag

Zoom Scroll

+ New Lobby



Users (1)

DemoUser

🚫 Banned Users (1) ▶



Join a Canvas

🔔 👤

All My Canvases

+ Create New Canvas

My First Canvas

by DemoUser

🕒 6/25/2026

Join Canvas

Fablewood Crossroads

by admin

🕒 6/9/2026

Join Canvas

Leprechaun's Hoard

by admin

🕒 6/9/2026

Join Canvas

tinkerbell

by admin

🕒 6/9/2026

Join Canvas



DIETRO LE QUINTE

Architettura & Sistemi Distribuiti



ARCHITETTURA 3 TIER

1. PRESENTATION TIER

SPA costruita in Vue.js (TypeScript) con pattern MVVM. Gestisce input e off-screen canvas rendering per performance ottimali.

2. APP & DATA TIER

Cluster Node.js/Express stateless. Usa Redis come In-Memory Data Store (Pub/Sub) e MongoDB per persistenza duratura.



CANALI DI COMUNICAZIONE

REST API

Modello: Request-Response

Uso: Operazioni CRUD, login e gestione lobby. Ideale per scambi di dati puntuali.

SOCKET.IO

Modello: Event-Driven

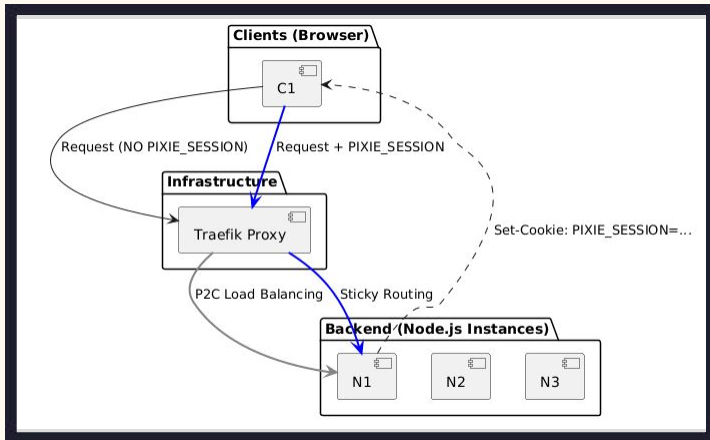
Uso: Sincronizzazione real-time del canvas e messaggistica ad alta frequenza tra client e server.

SSE

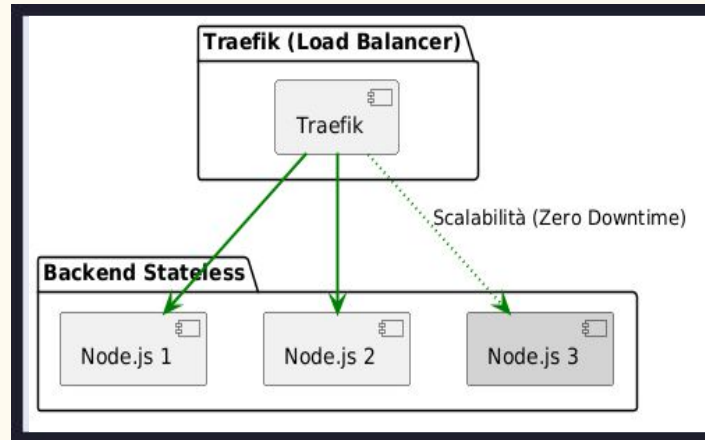
Modello: Streaming

Uso: Notifiche push asincrone. Gestito via Redis Pub/Sub per scalabilità orizzontale.

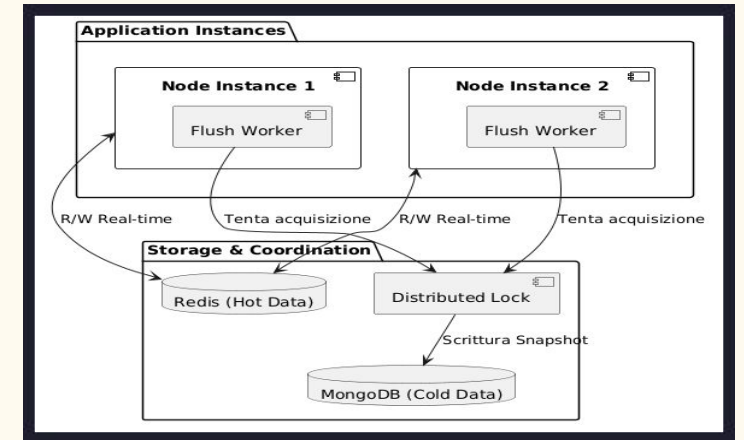
INFRASTRUTTURA SCALABILE



Traefik gestisce Load Balancing (Power-of-Two-Choices) e Sticky Sessions.



Istanze Node.js stateless, scalabili dinamicamente senza downtime.



Redis (Hot Data & Sync) e MongoDB (Cold Data Snapshots).



SICUREZZA & MIDDLEWARE

STATELESS AUTH (JWT)

Autenticazione stateless tramite token **JWT** su ogni richiesta HTTP, WebSocket e SSE.

Le repliche backend validano i token in modo indipendente tramite chiave condivisa, eliminando sessioni condivise.

RBAC AUTHORIZATION

Controllo accessi basato su tre ruoli distinti: **Base User**, **Creator** e **Admin** .

I middleware verificano i permessi a livello di rotte REST ed eventi Socket, bloccando comandi non autorizzati.

GENTLE ERROR HANDLING

Gestione globale delle eccezioni sul backend tramite middleware Express centralizzato.

Gli errori sono intercettati sul client e notificati all'utente in tempo reale tramite popup e toast.

IL TEOREMA CAP

- > Pixie predilige **Availability** (A) e **Partition Tolerance** (P).
- > Scegliamo *Availability*: la latenza minima del disegno è prioritaria.
- > Eventual consistency garantita grazie a Write-behind caching e *Last-Write-Wins*.

FLUSSO DI SINCRONIZZAZIONE

2. SCRITTURA REDIS

Scrittura atomica
(SETRANGE) filtrando
ridondanze.

4. RENDERING

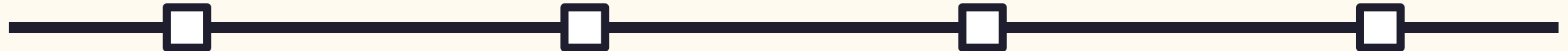
Client riceve `PIXEL_UPDATE`
e aggiorna il canvas.

1. EVENTO

Client invia `DRAW_BATCH` via
WebSocket.

3. PUB/SUB

Redis Adapter trasmette a
tutti i nodi Node.js.



TOLLERANZA AI GUASTI I



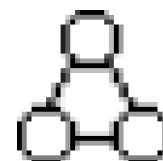
DISTRIBUTED LOCK

Worker coordinati via Redis
Lua Script prevengono
salvataggi multipli
concorrenti.



WRITE-BEHIND

Cache state salvato
periodicamente su DB; dati
persi limitati a un solo
flush interval.



STATELESS SERVER

In caso di indisponibilità di
un server, i client connessi
vengono reindirizzati verso
un'altra istanza attiva grazie
a Traefik.

OTTIMIZZAZIONI CORE

- > **Due Canvas Separati:** Il rendering separa il buffer dei dati dal display. Zoom e pan non ricalcolano i pixel, assicurando 60 FPS costanti.
- > **Double-Pipeline Batching:** Le chiamate Redis confrontano i valori precedenti prima di scrivere, azzerando le scritture ridondanti (risparmio di rete).
- > **Indici a 1-Byte:** I dati pixel sono salvati come indici di palette. Il consumo di banda è ridotto di 7x rispetto a formati RGBA.



TEST SUITE

UNIT TESTS

Validazione isolata dei singoli moduli (Canvas, Lock, Redis Adapter) tramite mock, senza richiedere servizi esterni attivi.

INTEGRATION TESTS

Verifica delle interazioni tra moduli reali con istanze Redis e MongoDB in-memory per testare flussi di disegno e socket.

INFRASTRUCTURE TESTS

Test a livello di Docker Compose per verificare la scalabilità dinamica e il bilanciamento di Traefik su più repliche.

SVILUPPI FUTURI

- > **Rimozione Single Point of Failure:** Migrazione a Redis Sentinel o Cluster per eliminare l'attuale Single Point of Failure
- > **Lobby private:** Lobby accessibili solo tramite invito, whitelist o password
- > **Canvas Infinito:** Sostituzione di un singolo buffer con un sistema di chunk dinamico
- > **Misc QoL:** Undo/Redo, Replay, Righello, Riempi etc...



DOMANDE??

Grazie per l'attenzione.

Andrea Zavatta | Matteo Susca