

Gianni dApp

Rivista Scientifica nel mondo Web3 Marketplace Decentralizzato

Sistema di vendita di articoli scientifici tramite dApp Web3.

Un'esperienza trasparente, sicura e decentralizzata per una scienza accessibile.

 by Antonino Paternò



Panoramica delle funzionalità

- ☐ **Connessione wallet-Integrazione con MetaMask per l'autenticazione**
- ☐ **Ricerca e visualizzazione articoli-Sistema di ricerca avanzato per titolo**
- ☐ **Acquisto tramite token Eth-Transazioni sicure tramite blockchain**
- ☐ **Verifica del saldo**
- ☐ **Navigazione tra articoli correlati**
- ☐ **Conferma transazioni con ricevuta**

L'utente può acquistare articoli scientifici usando Eth su rete Arbitrum Sepolia, in pochi click.



Stack Tecnologico



Frontend

React + TypeScript + Vite



Blockchain

Ethereum testnet
(Arbitrum Sepolia)



Web3 Librerie

Wagmi + Viem



Styling

CSS Modules



State management

Redux Toolkit



Routing

React Router



Animazioni

Lottie

Wagmi: semplifica l'integrazione di wallet e smart contract.

Viem: libreria TypeScript per interazioni con blockchain EVM, offre controllo granulare su RPC e gestione degli account

Architettura del progetto

Struttura modulare

Organizzazione in cartelle per tipologia (componenti, pagine, stati)

Componenti riutilizzabili

Design system con componenti facilmente estendibili

Separation of concerns

Chiara divisione tra logica, presentazione e gestione dello stato

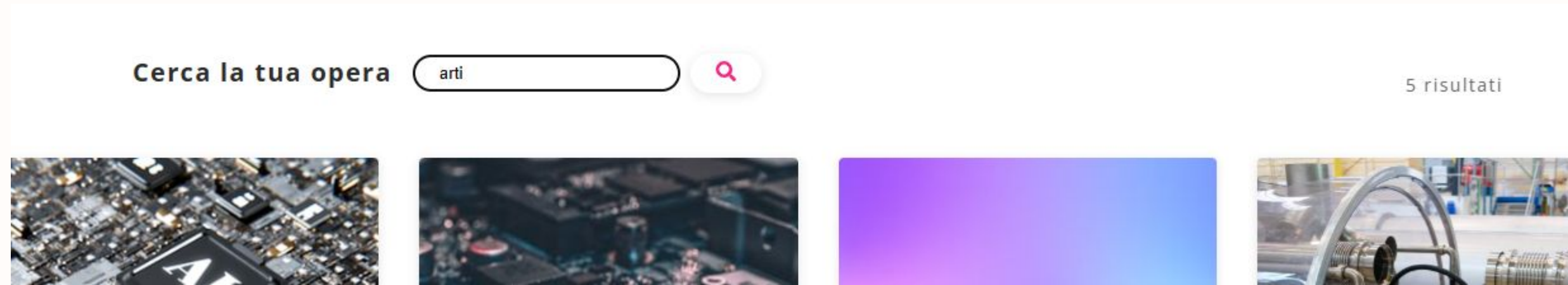
Gestione stati Redux Toolkit

gestione dello stato globale centralizzato migliorando la manutenibilità e la prevedibilità dell'applicazione.

Organizzazione in slice (search, sidebar, scroll, purchase, wallet)

```
src/
├── App.tsx
├── config.tsx
├── index.css
├── main.tsx
├── assets/
├── components/
│   ├── Article/
│   ├── Balance/
│   ├── Articles/
│   ├── Card/
│   ├── Connectors/
│   ├── ErrorMessage/
│   ├── Footer/
│   ├── Hero/
│   ├── Loading/
│   ├── Navbar/
│   ├── SearchForm/
│   └── SideBar/
├── customHook/
│   ├── useFetch.tsx
│   └── useTitle.tsx
├── pages/
│   ├── AboutHero/
│   ├── ContactHero/
│   ├── HomePage/
│   ├── NotFoundPage/
│   ├── SingleArticleScreen/
│   └── SuccessPage/
├── states/
│   ├── scrollSlice.tsx
│   ├── searchBarShownSlice.tsx
│   ├── purchaseSlice.tsx
│   ├── sidebarSlice.tsx
│   ├── walletSlice.tsx
│   └── store.tsx
├── utils/
│   ├── articleMock.tsx
│   ├── links.tsx
│   ├── teamInfo.tsx
│   └── utils.tsx
```


Funzioni di ricerca articoli



Ricerca opera

L'utente può' individuare in base testo digitato gli articoli di interesse i quali verranno mostrati di seguito escludendo quelli non corrispondenti ai criteri di ricerca o mostrando un messaggio di errore nel caso in cui non esistano articoli corrispondenti

Elementi decorativi: Lottie



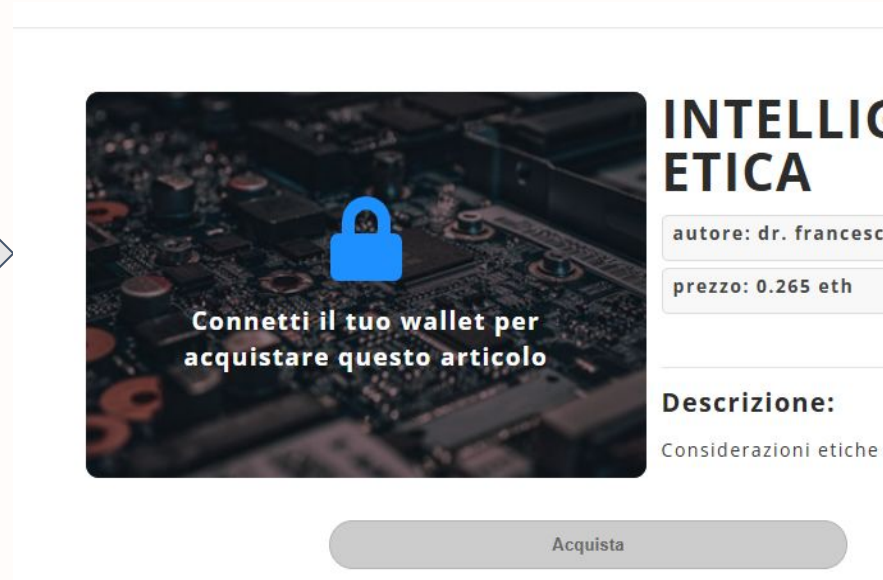
PAGINA NON TROVATA



Animazioni dinamiche

Per rendere gradevole l'esperienza utente, il Marketplace presenta diverse animazioni come all'interno dell'Hero Component in Homepage o ancora la pagina di errore 404 in caso di indirizzamento URL sconosciuto.

Processo di selezione ed Acquisto (1/2)



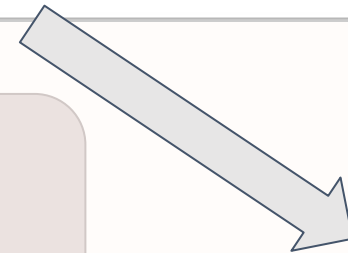
Gestione UX per flusso utente

Al passaggio del mouse su ognuno degli articoli renderizzati in base al testo ricerca, l'utente potrà accedere alla pagina specifica dell'articolo che mostra dettagli quali Autore, Descrizione e Prezzo espresso in Ether.

Se l'utente non è connesso, l'articolo è coperto da un'icona "lucchetto".

La connessione avviene tramite apposito menù a discesa presente sulla NAVBAR che abilita la visione del Saldo disponibile oltre che disabilitare la visione dell'icona "lucchetto" mostrando in modo completo l'immagine sottostante.

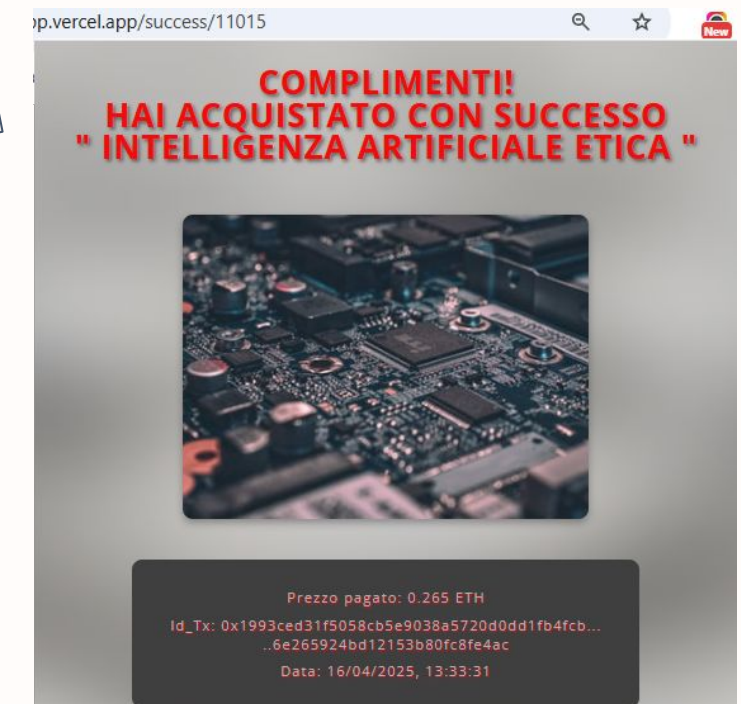
Processo di selezione ed Acquisto (2/2)



Firma Tx per conferma acquisto

L'utente , una volta aver visionato i dettagli dell'articolo , potrà confermare l'acquisto firmando al Transazione su Metamask , o ricevendo un messaggio di errore "TransactionExecutionError" in caso di anomalie o saldo insufficiente.

Al termine del processo di acquisto, una success Page fornirà un collegamento ai dettagli della Tx su block explorer.



Perché IPFS?

1

Decentralizzazione

Anche dei media.

2

Integrità

Garantisce integrità del contenuto.

3

Visione Web3

Coerenza con i principi Web3.

Ogni immagine raffigurante l'articolo è stata caricata su IPFS per rafforzare la visione "priva di intermediari" del progetto.



```
articleMock.tsx ×
TypeScriptReactWeb3diAntoninoPaterno > EcommerceWeb3dapp > src > utils > articleMock.tsx > mockArticleData
export const mockArticleData = [
{
  idArticle: "11000",
  strArticle: "La via Lattea",
  strArticleThumb: https://bafybeigavj3at52a3bvvr6wurv5caqmz7asuanuyiqx55uvowgbrmejbse.ipfs.dweb.link
}
```

Gianni Dapp – Web3 Scientific Journal Marketplace



Repository GitHub

<https://github.com/antopat1/EcommerceWeb3dapp>

URL deploy Vercel

<https://ecommerce-web3dapp.vercel.app/>