

[Home](#)

[Intro](#)

[Content](#)

[Demo](#)

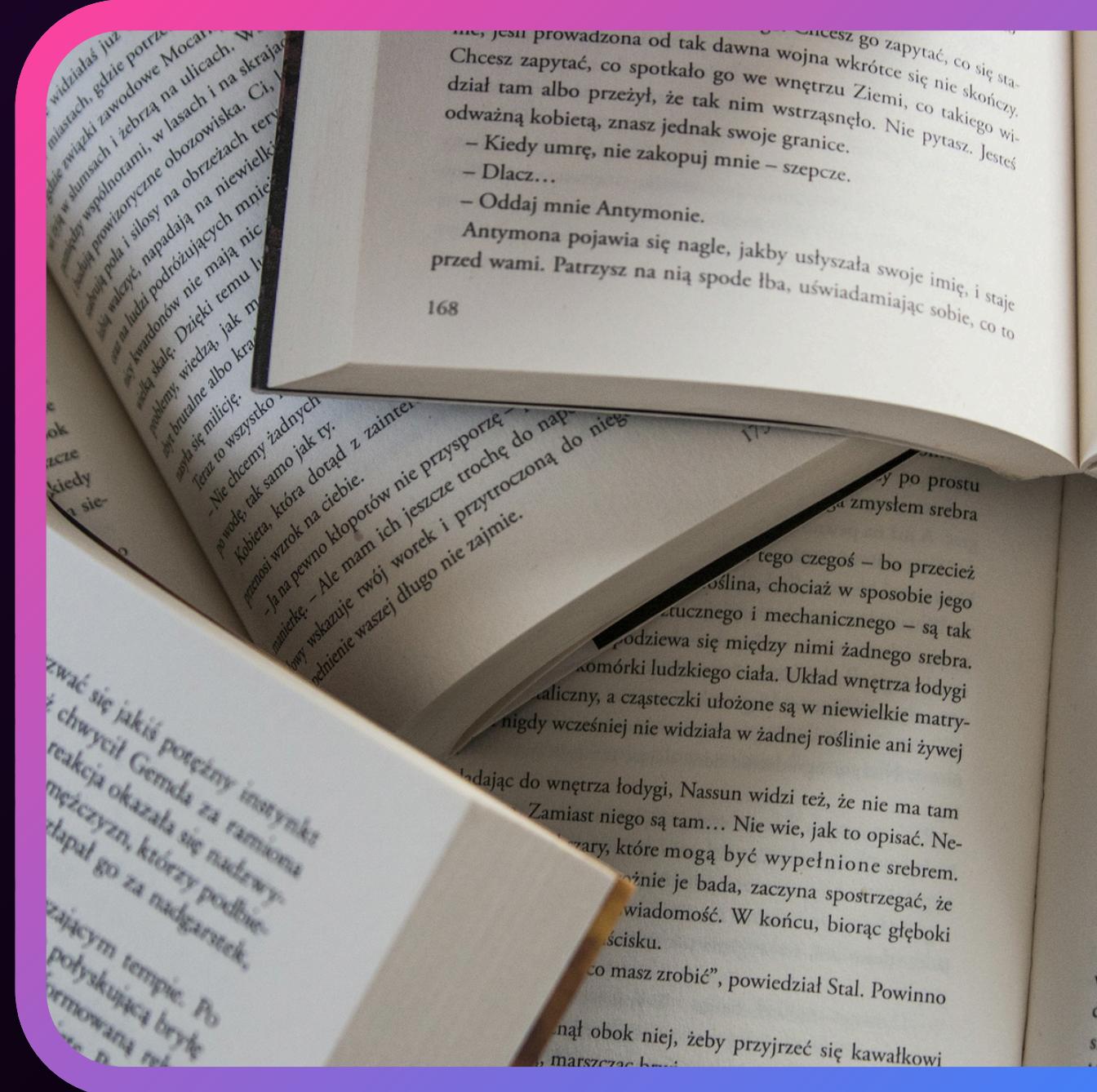
BIBLIOCHAIN

Blockchain course project

Martina Galasso

Contesto

- Sviluppo di un sistema bibliotecario
- Realizzato usando react
- Implementazione prestiti & vedita



REQUISITI

BASE

LIBRI COME ASSETS

PRESTITO TRA UTENTI

GESTIONE SCADENZE

REGISTRO STORICO

BONUS

VALUTAZIONE LIBRI

DEPOSITO CAUZIONALE

MARKETPLACE VENDITA

EXTRAS

MULTI SIGNATURE

PAUSABLE

USER TRUST SYSTEM

ANTI-SPAM

BAN SYSTEM

MULTIPLE OWNERS GOVERNANCE

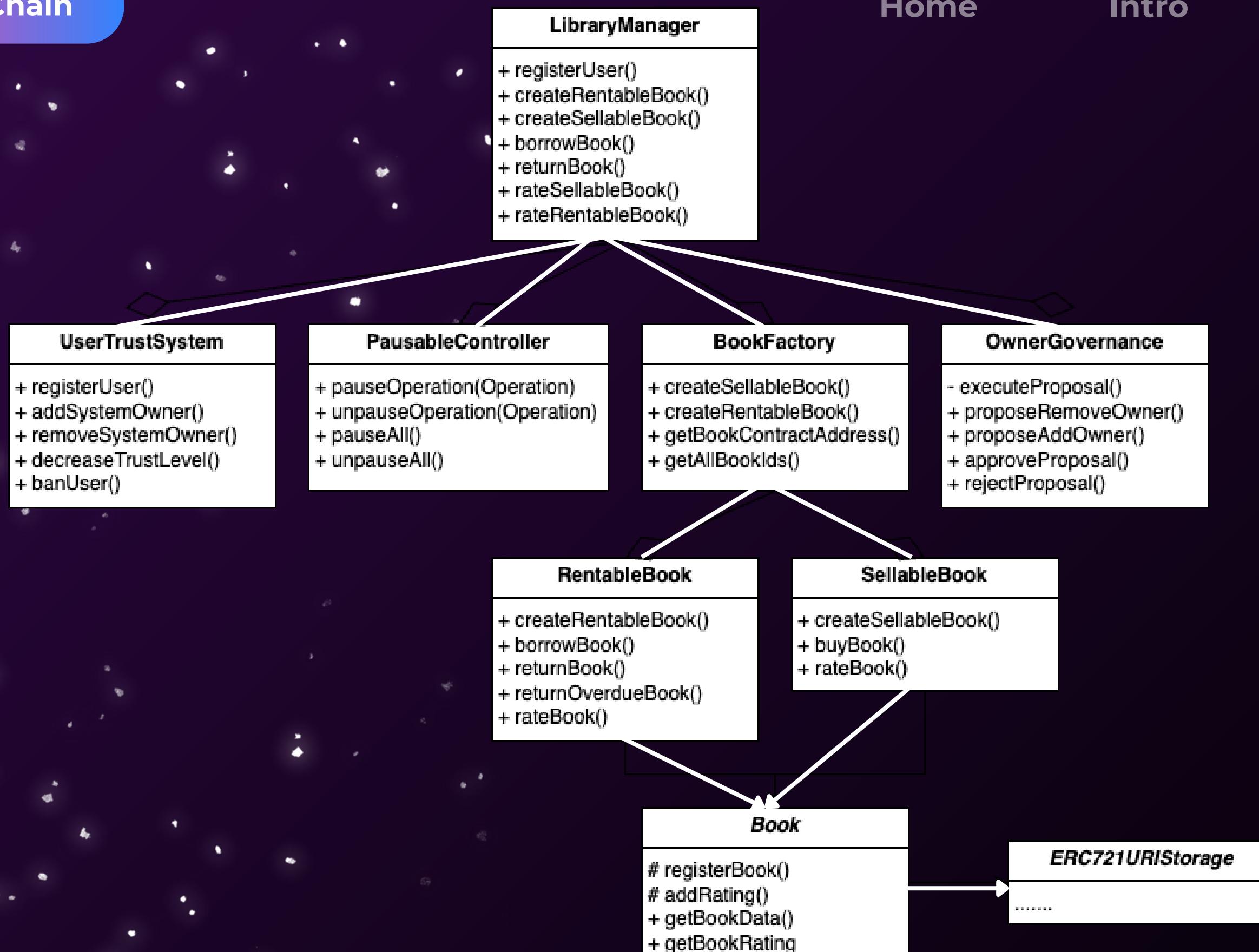
BiblioChain

Home

Intro

Content

Demo



Deploy e ownership

DEPLOYER

Il deployer è di default l'unico admin dopo il deployment.
È tenuto a trasferire l'ownership dei contratti al Facade

FACADE

il facade funge da ingresso per tutte le funzionalità.

ALTRI SMART CONTRACT

Gli altri smart contract accettano le chiamate solo dal Facade (owner), o in casi eccezionali dal Governance.
Obbligando gli utenti normali a passare attraverso il facade

FIDUCIA

GLI ALTRI CONTRATTI SI FIDANO DELL'OWNER. QUINDI SUPPONGONO CHE I CONTROLLI D'ACCESSO SIANO EFFETTUATI DALL'OWNER.

in questo modo la logica di access control attraverso i modificatori è centralizzata nel facade

LIBRI

- Assets digitali
- Solo dati rilevanti salvati nello smart contract, gli altri su IPFS (come titolo, copertina etc...)
- Il deposito viene trattenuto dal contratto fino alla restituzione
- La vendita trasferisce l'ownership, il prestito no
- Lo storico è implementato attraverso gli eventi
- Una sola recensione per utente

RESTITUZIONE

IN TEMPO

- L'utente che noleggia riceve interamente il deposito

PRESTITO SCADUTO

- Il libro viene inserito in una lista pubblica
- Chiunque può restituire il libro
- In questo caso il 30% del deposito va all'utente che l'ha restituito, il restante 70% all'owner per il risarcimento del danno
- Viene decrementata la fiducia dell'utente

MULTI SIGNATURE

QUORUM

Si hanno quorum diversi in base alla proposta
aggiungi admin: 50% + 1
rimuovi admin: unanimità

REJECTION VOTE

Si può scegliere se accettare o rifiutare la proposta. Se il rifiuto non permette di raggiungere il quorum, la proposta viene disabilitata

ESECUZIONE

La proposta viene eseguita automaticamente quando raggiunge il quorum, quindi all'ultimo voto necessario

*L'implementazione di openzeppelin non permetteva quorum diversi ne reject vote

PAUSABLE

PAUSA

Si può scegliere se pausare tutti i servizi in una transazione unica o servizi specifici (come prestito, vendita o creazione)

PREVENZIONE

L'uso del pause permette di previnire abusi e minacce nel caso in cui si scoprono vulnerabilità, quindi non richiede più firme per effetto immediato

OWNERSHIP

Solo gli admin (owners) possono pausare i servizi. Questa operazione essendo

*L'implementazione di openzeppelin non permetteva il pause di servizi specifici e pause parziale

USER TRUST & SPAM

USER TRUST

- Sistema a punti
- 2 tipi di infrazioni (Low & High)
- A 0 punti viene bannato automaticamente
- Livello gestito automaticamente
- Il sistema non decrementa la fiducia degli admin

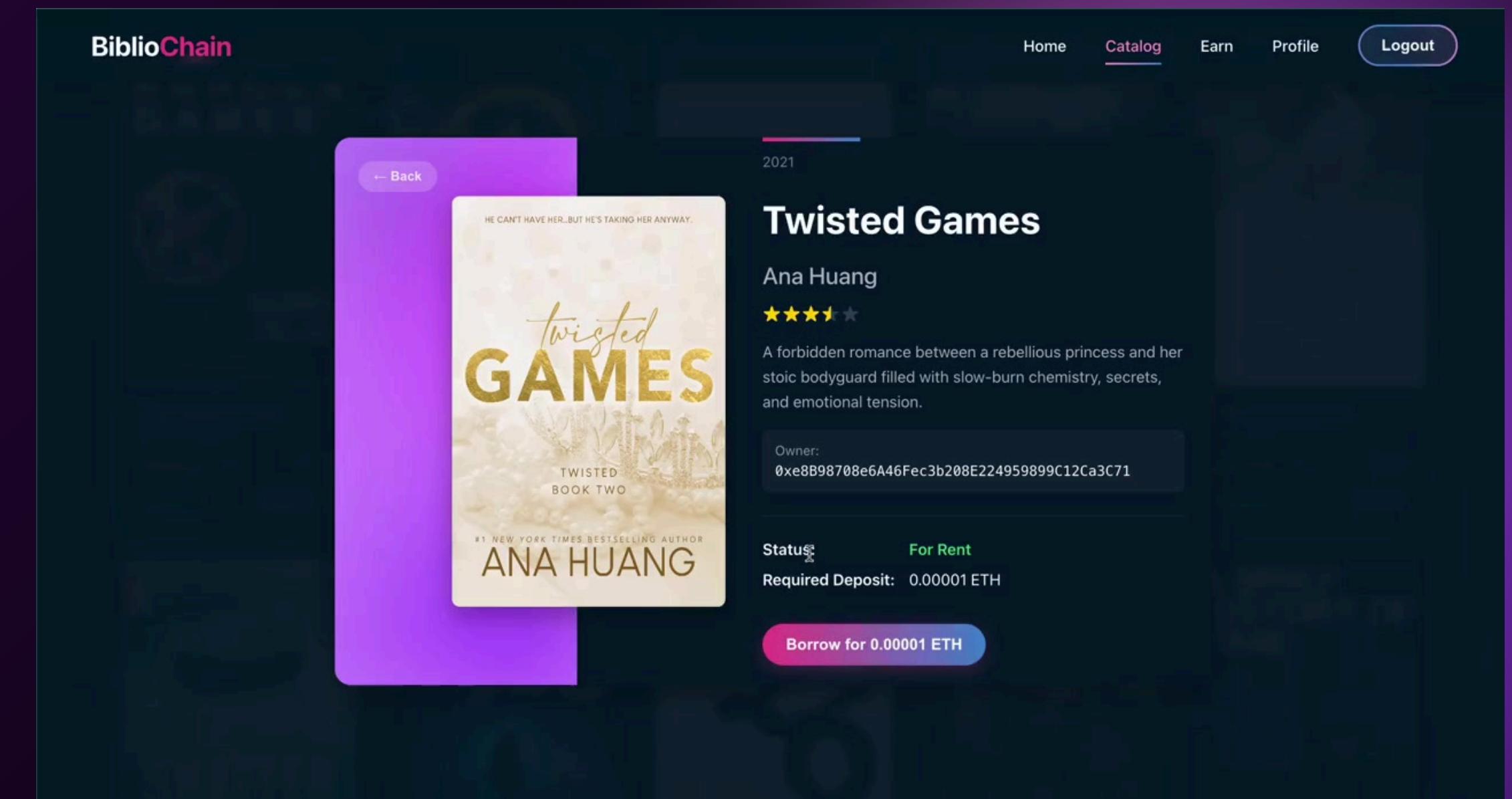
SPAM

- Viene controllata se la transazione è spam
- Si ha un threshold di 10 transazioni all'ora, dopo le quali si viene bannati a causa di una High infraction
- Ogni ora, se l'utente invia transazioni, viene resettato il limite
- Gli admin non hanno l'anti spam

UI

Sviluppato usando CSS puro avanzato

- Animazioni & Transizioni
- Pseudo classi ::before & ::after
- Glass, blur & shimmer effects
- Interceptor
- Carosello con z index
- Modello 3D
- Bordi animati
- Animazioni 3D
- Gestione gerarchia elementi
- Cubic Bezier



231 TEST

Comprensivi di test d'unità e di integrazione

Tests - Coverage

all files contracts/

100% Statements 309/309 100% Branches 302/302 100% Functions 112/112 100% Lines 409/409

File	Statements	Branches	Functions	Lines
Book.sol	<div style="width: 100%;"> </div>	100%	13/13	100%
BookFactory.sol	<div style="width: 100%;"> </div>	100%	25/25	100%
LibraryManagerContract.sol	<div style="width: 100%;"> </div>	100%	101/101	100%
OwnerGovernance.sol	<div style="width: 100%;"> </div>	100%	63/63	100%
PausableController.sol	<div style="width: 100%;"> </div>	100%	11/11	100%
RentableBook.sol	<div style="width: 100%;"> </div>	100%	40/40	100%
SellableBook.sol	<div style="width: 100%;"> </div>	100%	20/20	100%
User.sol	<div style="width: 100%;"> </div>	100%	36/36	100%

DEMO

CONSIDERAZIONI & CONCLUSIONI