Adam Kasperski
Ecem Alakus
Filippo Chinetti
Sundaresan Karunakaran

DApp Project
Distributed Systems

DApp Project – A decentralized rental marketplace

1. Theme

The project is focused on developing a secure and transparent platform that allows users to rent and list items directly. By utilizing blockchain technology, the platform removes the need for intermediaries, reducing associated costs and building trust between users. Smart contracts are employed to manage rental agreements, payments and deposits automatically and in an immutable way. This ensures a straightforward and dependable rental process. The aim is to revolutionize the rental experience by integrating the benefits of blockchain technology with an accessible and user-friendly web interface.

2. Implementation

Our project combines various technologies to create a decentralized rental marketplace. The focus was on integrating blockchain technology, building an intuitive frontend, setting up a simple yet functional backend, and ensuring seamless operation. Through rigorous review of code and configurations, we designed a flexible and manageable architecture that aligns with the project’s goals.

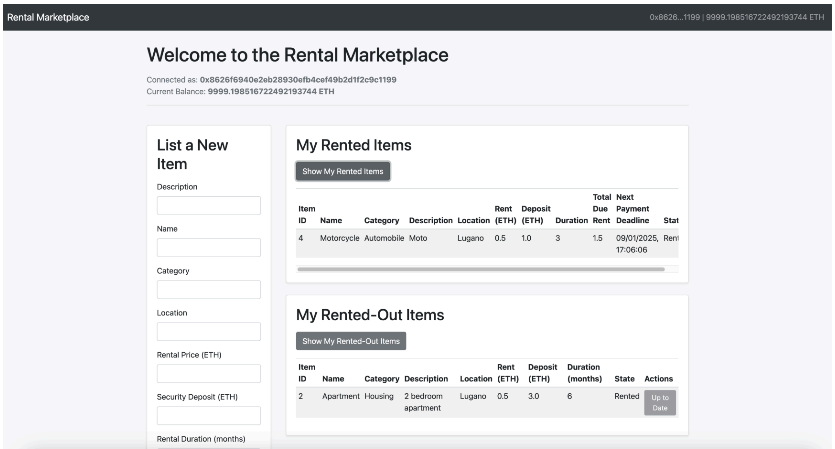
Frontend: The frontend of the application was developed using **React** and **JavaScript**, enabling the creation of an interface that efficiently supports the platform's functionality. The application includes a single page with all the operations that are possible. We firstly connect their Ethereum wallets via MetaMask for blockchain integration. On the Home page, users can browse items for renting and apply filters based on price, category and location to quickly find what they need. The Sell page allows users to list their items for rent by providing essential details like the item’s name, description, rental price, duration and deposit amount. The Rent page is where users can finalize rental agreements, ensuring that all terms are clear and agreed upon before proceeding. This structure ensures a streamlined and intuitive user experience, catering to all essential features of a decentralized rental marketplace.

Smart Contract: The core of our project is the smart contract system, built using Solidity, which ensures that transactions are automatic, reliable and immutable.
We linked the smart contract to the front end using **Web3.js** and **Ethers.js.**
The contract manages the rental process by securely handling details such as:

- Rental Terms: Duration, payment amount, and deposits.
- Agreement Activation: Contracts are triggered when renter agrees to the terms.
- Payment Management: Deposits are required before any exchange and payment timelines are clearly outlined.
- User information: linked to the Ethereum wallet address.

Backend, Data Management and blockchain integration: For the backend, we decided to handle user data and authentication directly through the smart contract. This approach lets us eliminate other third parties like Firebase, providing decentralization and security. Therefore, the smart contract manages:

- **User data**: Users are linked to their Ethereum wallet addresses, which serve as their unique identifiers on the platform. The wallet address is used to associate users with rental agreements and item ownership.
- **Item listings**: Item details (name, description, location, rental price, security deposit and rental duration) are stored directly on the blockchain, ensuring that all data are immutable and securely handled.
- **Rental agreements**: All rental agreements, including payment schedules and deposits, are stored on the blockchain as part of the RentalAgreement struct. This ensures transparency and reliability, as agreements are only activated once the renter agrees to the terms.



Welcome to the Rental Marketplace

Connected as: **0x8626f6940e2eb28930efb4cef49b2d1f2c9c1199**
Current Balance: **9999.198516722492193744 ETH**

List a New Item

Description

Name

Category

Location

Rental Price (ETH)

Security Deposit (ETH)

Rental Duration (months)

My Rented Items

Item ID	Name	Category	Description	Location	Rent (ETH)	Deposit (ETH)	Duration	Total Due Rent	Next Payment Deadline	Stat
4	Motorcycle	Automobile	Moto	Lugano	0.5	1.0	3	1.5	09/01/2025, 17:06:06	Rent

My Rented-Out Items

Item ID	Name	Category	Description	Location	Rent (ETH)	Deposit (ETH)	Duration (months)	State	Actions
2	Apartment	Housing	2 bedroom apartment	Lugano	0.5	3.0	6	Rented	Up to Date

3. Challenges

During the development of our project, we faced several challenges:

1. **Blockchain integration** with the front end

Using Web3.js and Ethers.js, we connected our React application to the Ethereum blockchain, enabling users to interact with smart contracts. Setting up this connection and ensuring secure communication was challenging, especially when integrating MetaMask to let users connect their wallets and complete transactions smoothly.

2. **Designing a user-friendly interface**

We focused on making the platform simple and intuitive. We used Material-UI and custom CSS to create a responsive and clean design. Adding features like filtering items by price, category, and location made it easier for users to navigate the platform. However, balancing simplicity with the advanced features needed for blockchain functionality was a challenge.