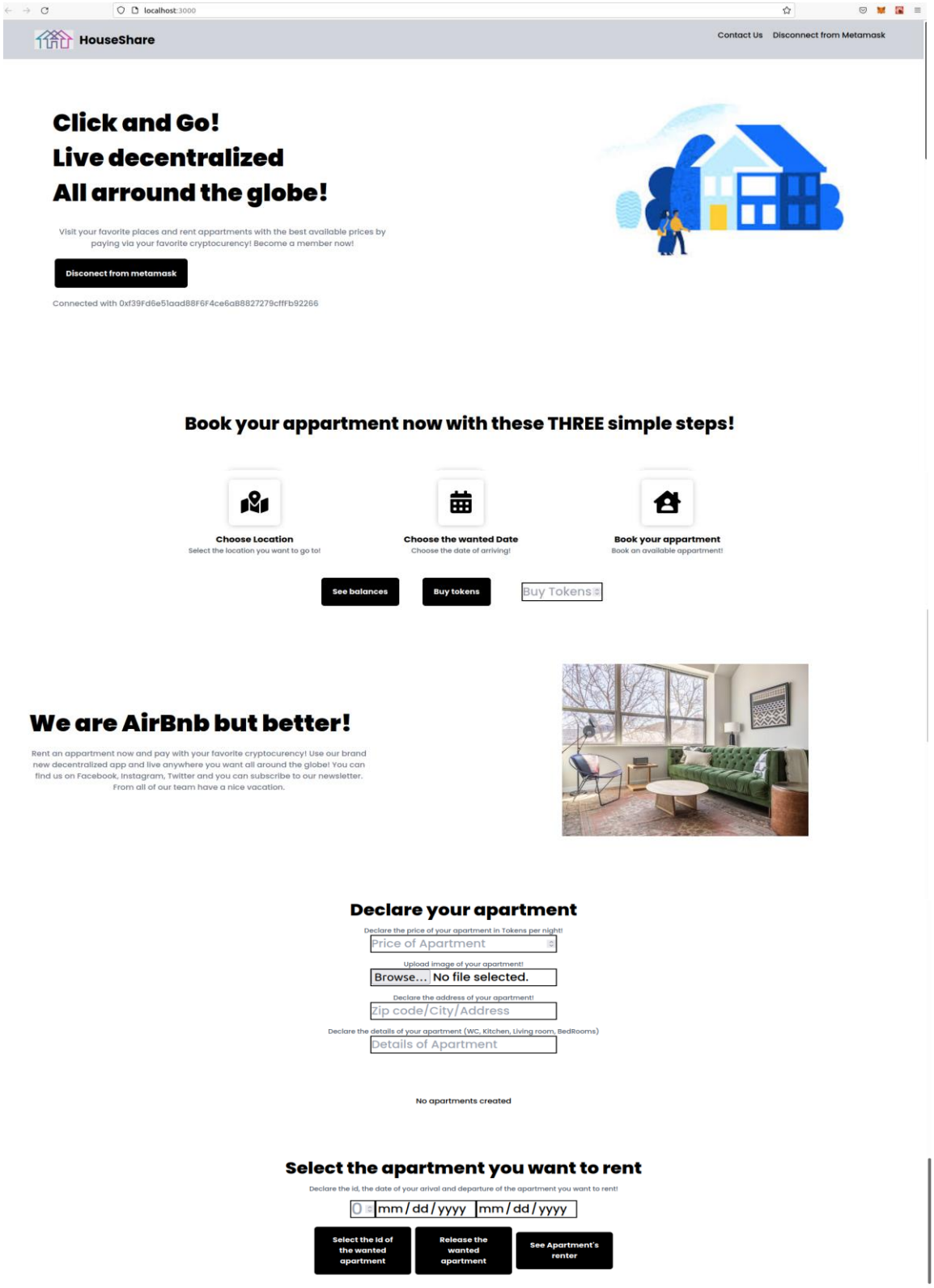
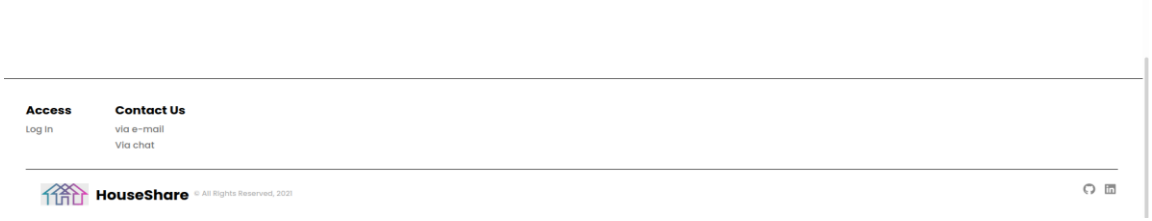
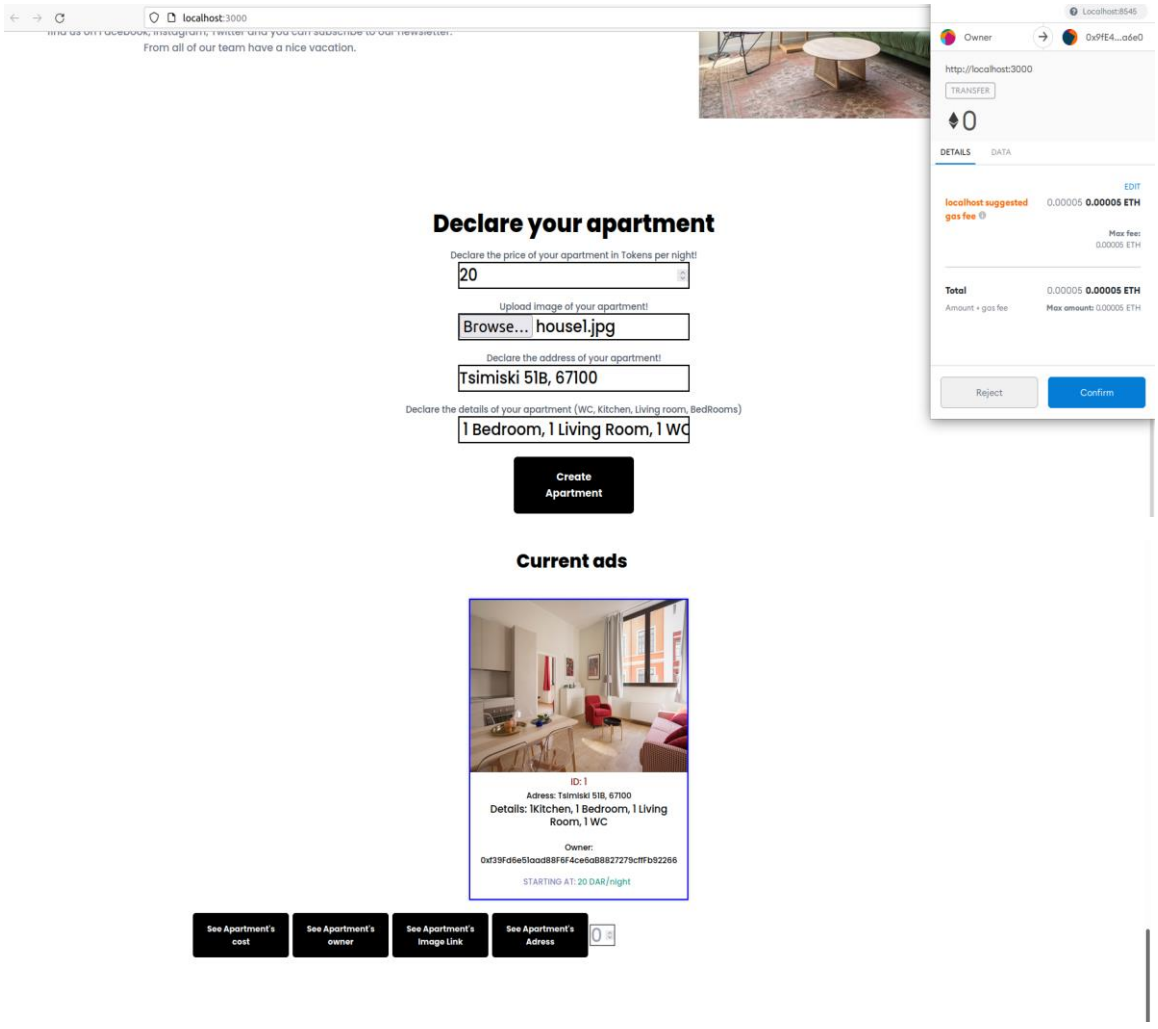
DECENTRALIZED APARTMENT RENTING APP

- Page preview when there is no apartments created yet:



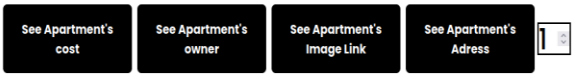


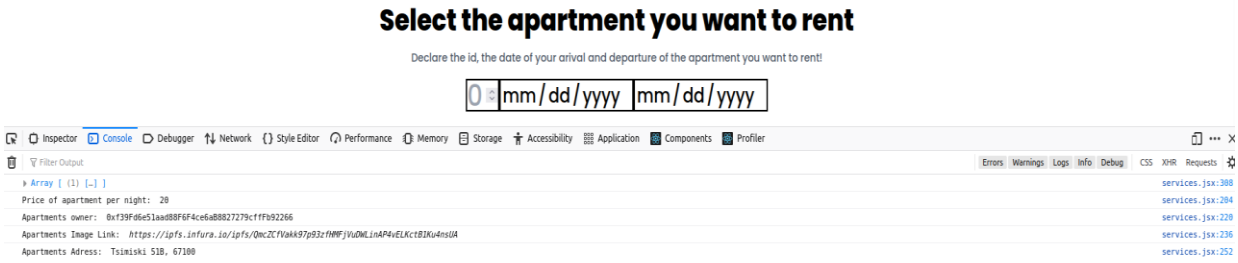
- Preview during and after the creation of an apartment:



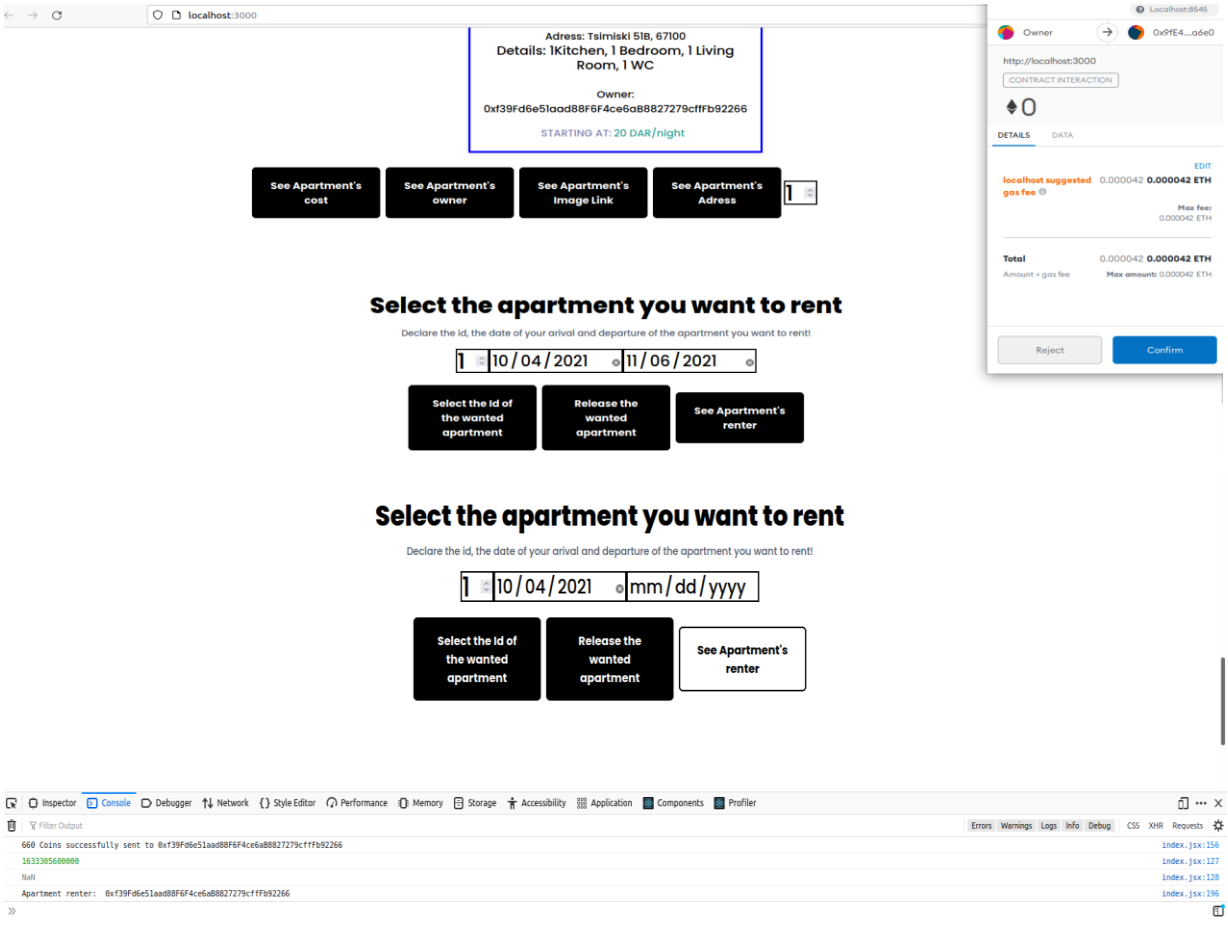
Apartments are created dynamically, the image is stored on the ipfs server and the app retrieves the hash so it can be displayed on the frontend. The price and the details of the apartment are all given by the user and are stored on the blockchain with the hash , the owner's address and the apartments Id.

- Retriving data from the apartment struct and displaying them on the console:

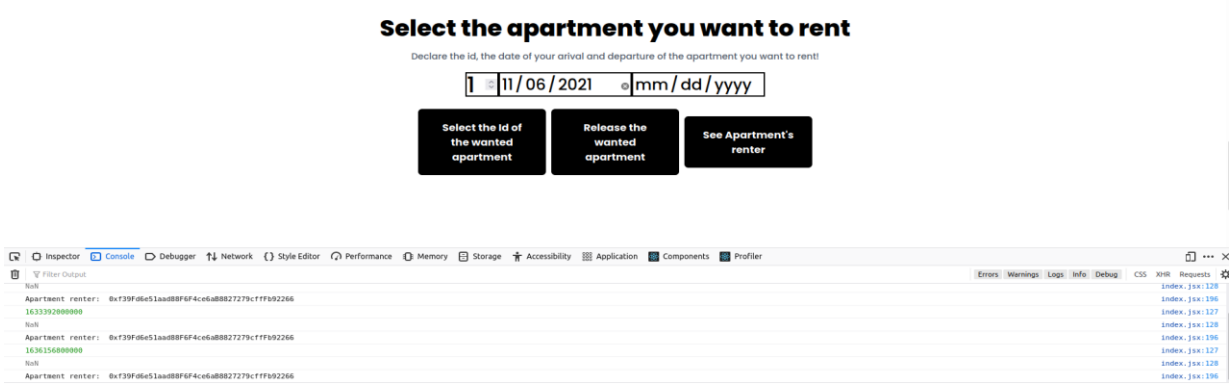




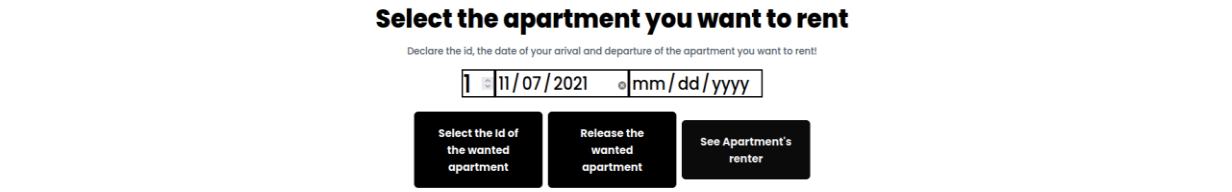
- Renting the apartment for a certain date (the apartment is available for all the other dates):

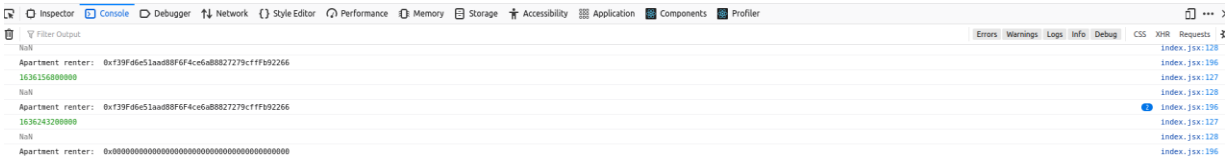


- Checking that the apartment is rented for all the given dates:

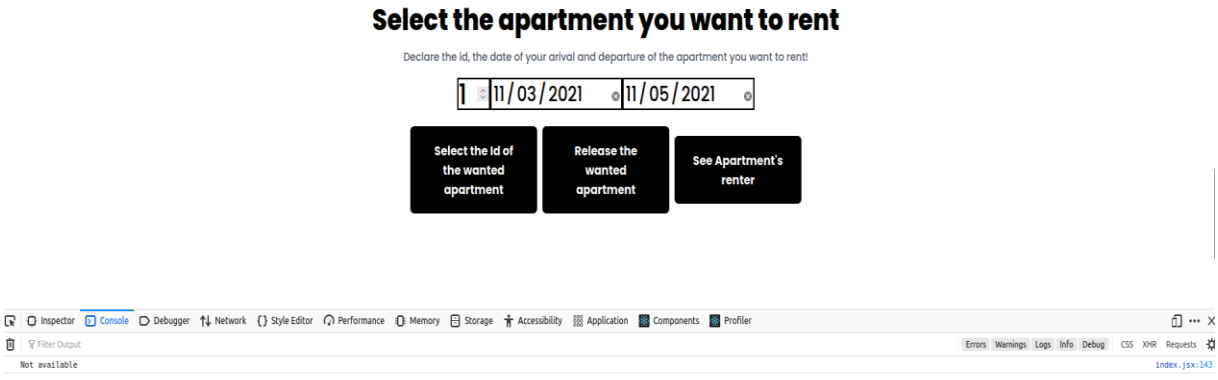


- Checking the apartment's renting status oustide of the previous renting dates:

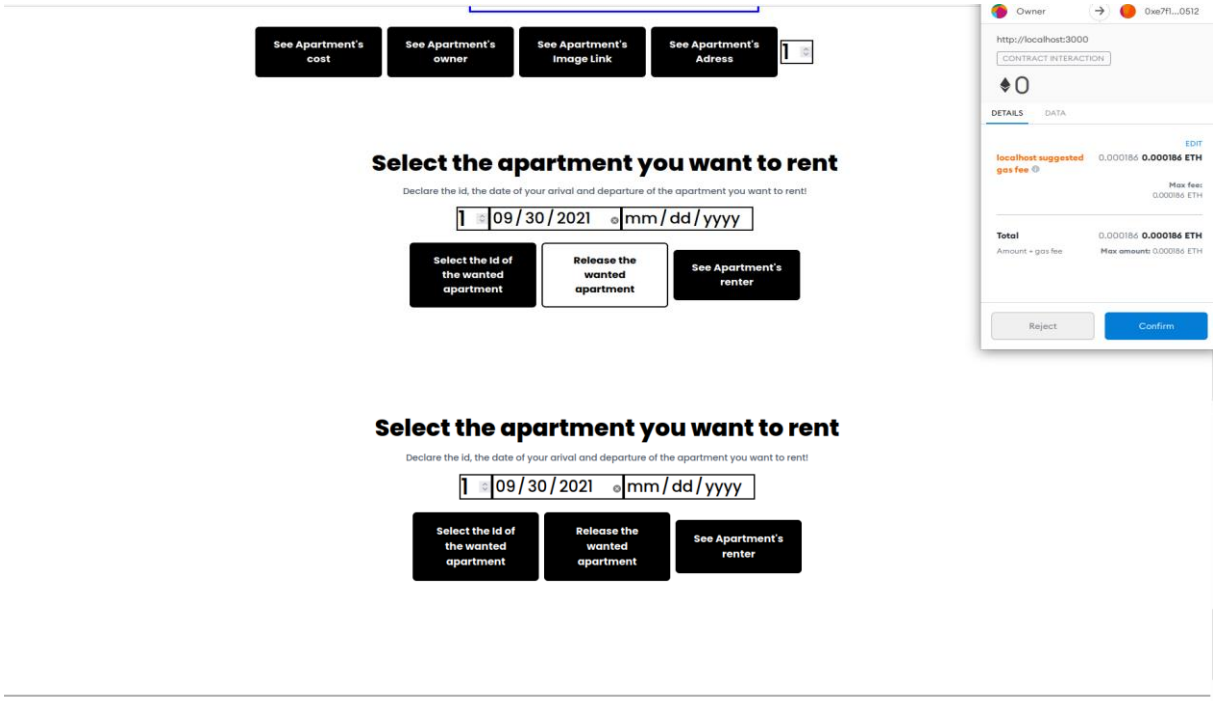


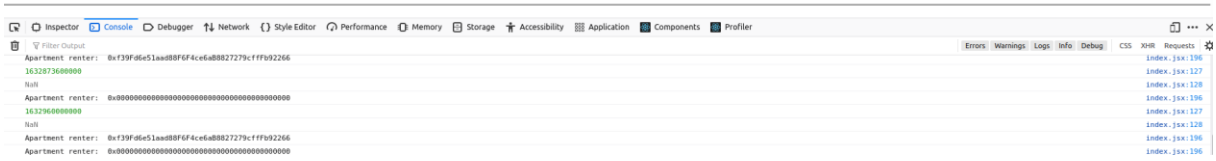


- Output when the user tries to rent an already rented apartment on a certain date:



- Canceling the reservation by the owner given a certain starting date, the rest of the days of the reservation will also get deleted (owner must provide the starting date of the reservation):





A solidity Red-Black Tree binary search library is used to store and access a sorted list of rival intervals (apartment's renting dates). Rival intervals can't overlap.

The Red-Black algorithm rebalances the binary search tree, resulting in O(logn) insert, remove and search time

For the currency i use an ERC20 token named DAR (Decentralized apartment renting) and it's purpose is just financially

TODO:

ERC721 token for ownership and renting validation