Blacklist / Whitelist

Create and deploy a token completely owned by yourself - you are the issuer.
Assume there are 5 wallets in this network. All wallets on the network are initially whitelisted.

Whitelisted wallets can transfer tokens to other whitelisted wallets. But blacklisted wallets do not have the privilege to send or receive tokens. Create functionality to blacklist wallets. If a wallet is blacklisted, then all peers that have directly interacted with this wallet also get blacklisted. If a blacklisted wallet is re-whitelisted, then peers that were blacklisted because of it, also get whitelisted.

The state of any wallet can either be blacklisted or whitelisted. By default, all addresses are blacklisted. You initially start by whitelisting 5 addresses.
What happens when you increase the number of wallets on the network to 1000? Does a smart contract only approach still work? If not, what alternative solution do you propose? Highlight the changes required to implement the additional functionality on the application side and relevant smart contract changes if any.

Objectives

- Create a UML diagram (or equivalent) to showcase how you would architect such a system (on-chain and off-chain components)
- Create smart contracts for the on-chain components of the system described above
- List out any edge cases that you think may be pertinent and how you would solve them
- Create a private GitHub repository for your code and share it with satish-xalts and reuvab.
- Email a link of the GitHub repository to satish.prasad@xalts.io and reuven.aboulker@xalts.io.

Timeline
This assignment must be submitted within 48 hours of receiving it