

PROPOSTA

A proposta deste trabalho é desenvolver um Decentralized App (DAPP) baseado na Web para a denúncia de crimes ambientais. Para tal, será utilizada uma abordagem de desenvolvimento de DAPP, onde as informações fornecidas pelo denunciante serão armazenadas dentro de uma blockchain, evitando assim que haja adulteração ou exclusão das informações por pessoas mal-intencionadas.

Com a finalidade de atingir uma gama maior de usuários, o aplicativo deverá contar com uma interface amigável e de fácil interação. Ou seja, para este trabalho será necessário o desenvolvimento de um front end. Já para tratar e trabalhar com as informações fornecidas pelo denunciante, será necessário o desenvolvimento de uma outra parte da aplicação, um back end.

A blockchain da Ethereum (BC) será utilizada neste trabalho por possuir uma gama muito alta de ferramentas, além de uma linguagem Turing completa para o desenvolvimento de Smart Contracts (SC). Ela deve manipular e armazenar o DAPP por pura descentralização, porém, devido ao gargalo de desempenho da blockchain, será escolhido uma implementação onde somente algumas partes do sistema estará na blockchain.

DA ARQUITETURA DA APLICAÇÃO

A solução proposta conta com uma abordagem de desenvolvimento de um DAPP indireto, tendo serviços de back end rodando em um servidor centralizado, enquanto o cliente interage com o SC através destes. Como falado anteriormente, e na seção de Fundamentação Teórica, a blockchain possui um desempenho que deixa a desejar para alguns contextos, porém não para o nosso.

Parte da arquitetura da aplicação será desenvolvida utilizando tecnologias mais difundidas e que não terão o papel principal na descentralização da informação. Essa última necessidade estará a cargo da blockchain, que para o nosso desenvolvimento será utilizada via rede de teste.

Para o front end será feito uso da principal linguagem de desenvolvimento para a web da atualidade, o JavaScript. Pode-se ser feito o uso dos principais frameworks ou bibliotecas da linguagem, como o Angular, React ou Vue, dentre outras menos difundidos. Também será feito o uso de um navegador de internet.

No desenvolvimento do back end pode-se fazer uso de bibliotecas, API's e frameworks que agilizam a integração com a BC da Ethereum e que permitem que o programador não precise implementar do início as principais funcionalidades de uma BC. É possível acessar uma quantidade grande dessas ferramentas através da própria documentação da Ethereum.org, a fonte primária e online para a comunidade da Ethereum, de acordo com o repositório do próprio projeto aberto do site no github.

A imagem a seguir fornece uma visão geral da arquitetura da aplicação. Nela podemos ver a divisão entre as camadas mencionadas; front end, back end e API de conexão com a BC da Ethereum.

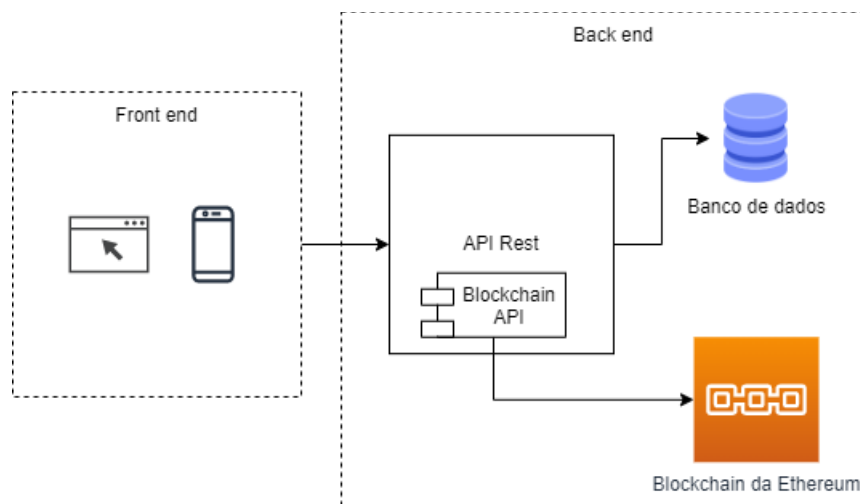


Figura 1: Arquitetura básica da aplicação

Um banco de dados também será utilizado para armazenar as informações de usuário, como dados de autenticação, identificação ou demais informações que não precisam ser armazenadas utilizando a BC. Será feito o uso de um banco de dados não relacional, pois as tabelas do banco de dados podem apresentar características dinâmicas, o que é interessante para aplicação de uma tecnologia que permita esquemas de tabela mais flexíveis, como bancos de dados orientados a documentos.