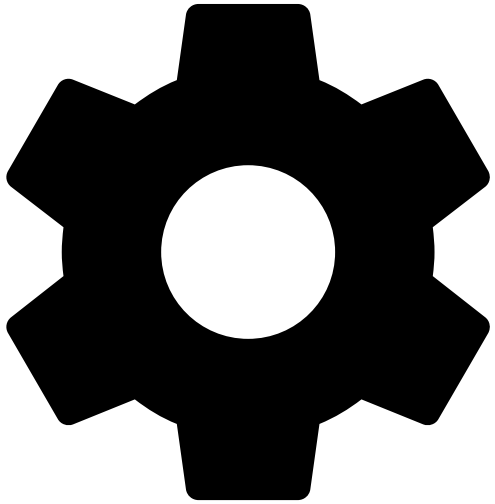
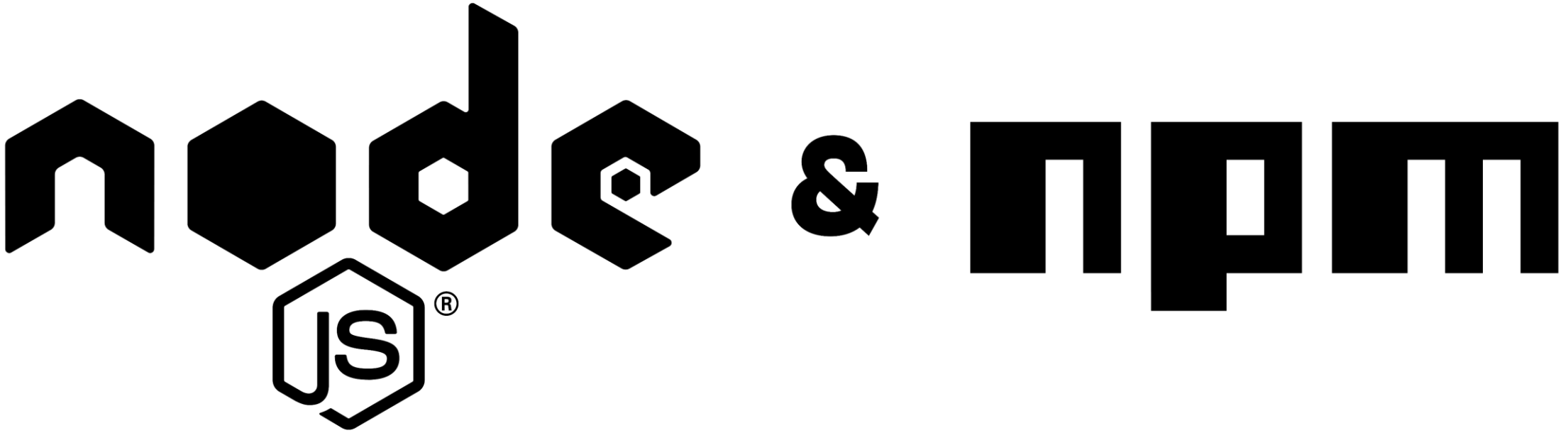
GETTING STARTED



Setup steps for EasyFHE usage

PRE-REQUIREMENTS



https://nodejs.org/en/download/

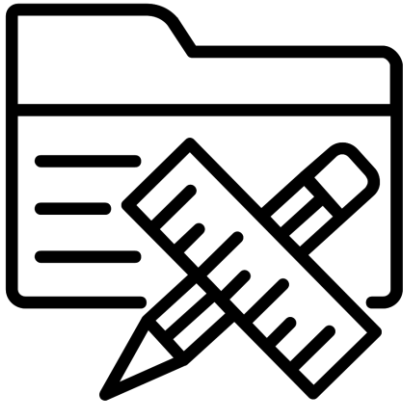
INSTALLING

Quasar

Open the terminal and type the following command:

```
npm i -g @quasar/cli
```

CREATING THE QUASAR PROJECT



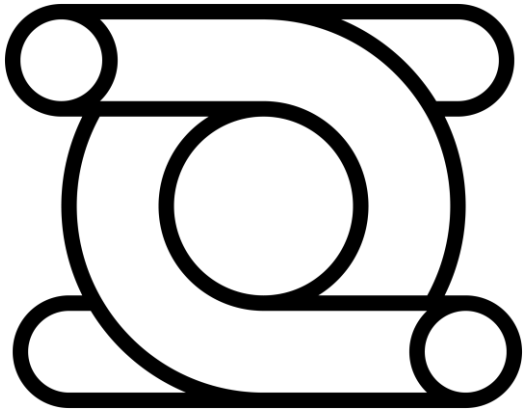
Open the terminal and type the following command:

```
npm init quasar
```

Go in *'quasar.config.js'* and add to the build property the following:

```
extendWebpack(cfg) {
    cfg.experiments = {
            asyncWebAssembly: true,
    };
},
```

CREATING THE QUASAR PROJECT



Open the terminal and type the following command:

git clone https://github.com/mihailpreda/fhe-module-typescript-wrapper.git easyFHE

cd easyFHE

npm install

Then go to Quasar project's *'package.json'* file and add EasyFHE to dependencies like:

easyFHE : file:../easyFHE

USAGE EXAMPLE

```
<script lang='ts'>
import getEasyFHE, {
  EasyScheme,
  EasySecurity,
  EasySpeed,
  EasyFHE,
} from 'easyFHE';
(async () => {
    // Get the WebAssembly module
    const easyFhe: EasyFHE = await getEasyFHE();
    // Initialize it
    await easyFhe.Setup.initialize();
    // Set your preferred encryption parameters
    easyFhe.Setup.fastSetup(
      EasyScheme.BFV,
      EasySecurity.TC128,
      EasySpeed.FAST,
    );
    // Generate a keypair
    const [publicKey, secretKey] = easyFhe.generateKeys();
     // ciphertext + ciphertext
     //-------------------------------------
     const leftValue0 = easyFhe.encrypt(Int32Array.from([3]), publicKey);
     const rightValue0 = easyFhe.encrypt(Int32Array.from([6]), publicKey);
     const encResult0 = easyFhe.Cipher.add(leftValue0.save(), rightValue0.save());
     const result0 = easyFhe.decrypt(encResult0.save(), secretKey);
     //-------------------------------------
    easyFhe.deallocateLibrary();
    publicKey.delete();
    secretKey.delete();
})()
</style>
```

CONTACT

Preda Mihail Irinel

Github Email LinkedIn