

Documentation utilisateur

Cette documentation décrit la procédure pour utiliser ce projet dans un environnement de développement.

- [Pérequis](#)
- [Installation](#)
- [Déploiement](#)
 - [Environnement de développement](#)
 - [Environnement de test](#)
- [Utilisation](#)
 - [Environnement de développement](#)
 - [Environnement de test](#)

Pérequis

- [Node.js](#)
- [Truffle](#)
- [Ganache](#)
- [MetaMask](#)

Installation

Les composants sont à installer dans l'ordre.

Node.js

Merci de se reporter à cette [documentation](#). L'installation de la version v13.X est recommandé, mais le projet reste compatible avec toute version supérieur ou égale à la v10.X.

Truffle

Ouvrez un terminal et executer la commande suivante:

```
npm install -g truffle
```

En cas de problème avec la commande précédente sur Windows, se référer à cette [documentation](#).

Ganache

Se référer à la [documentation officielle](#).

MetaMask

Se référer à la section *Installing MetaMask de cette [documentation](#).

Déploiement

Environnement de développement

1. Lancer Ganache
2. Cliquer sur le bouton **New workspace**
3. Dans l'onglet *Workspace*, indiquer:
 - dans le champ **Workspace name**: `monkey-voting`
 - dans le champ **Truffle projects**: le chemin complet vers le fichier `truffle-config.js`.
4. Ouvrir un terminal à la racine du projet et taper la commande `truffle migrate`
5. Si la commande précédente ne conduit pas à un déploiement en succès, forcer le déploiement avec la commande `truffle migrate --reset --compile-all --skip-dry-run`
6. Suivre ce [tutorial](#) pour configurer MetaMask

Environnement de test

Installation initiale

Dans un terminal, taper la commande `geth --goerli account new` afin de créer un compte dans le réseau de test.

Afin d'obtenir des `eth` pour nos tests, il faut utiliser le site <https://goerli-faucet.slock.it/> et taper l'adresse précédemment obtenue dans le champ associé.

Après avoir obtenu des `eth` sur le compte goerli, il faut l'importer dans MetaMask. Pour ce faire, il faut ouvrir l'extension et choisir le réseau `Goerli testnet`. Cliquer ensuite sur `import account, choose from JSON file`, et choisir le fichier généré par geth avec la commande précédente. Enfin, entrer la passphrase utilisée à la création du compte.

Upload du contrat sur le réseau Goerli testnet

Il faut créer un compte sur <https://infura.io/>. Un accès à un wallet et à un call API pour communiquer avec Goerli sera mis à disposition. Infura est la méthode préférée pour communiquer avec la blockchain Ethereum (en février 2019, 63% des Dapps étaient postées dans ce réseau).

Une fois le compte obtenu, il faut créer un fichier `.env` avec votre `MNEMONIC` et votre clé d'API.

```
MNEMONIC="some word that compose your mnemonic"
INFURA_API_KEY="xxxx"
```

Réaliser la migration Truffle

Dans un nouveau terminal, exécuter la commande:

```
truffle migrate --network goerli
```

La migration va débuter et va déployer les contrats sur le réseau Goerli. Quand elle sera terminée, l'adresse associée au contrat devra être sauvegardée dans le fichier ABI, et la MonkeyElectionDapps trouvera l'adresse

correcte correspondant au réseau choisi dans MetaMask.

Configuration de Swarm

Afin de déployer la webapp sur la blockchain, il faut utiliser Swarm. Dans un terminal, exécuter la commande suivante en remplaçant **<ACCOUNT>** par l'adresse lors de la création du compte dans le réseau Goerli:

```
swarm --ens-api "" --bzzaccount <ACCOUNT>
```

Pour uploader la webapp, se rendre à la racine du projet **MonkeyVotingDapp** et exécuter la commande suivante:

```
swarm --recursive up ./
```

Un hash sera retourné. Il établit le lien à Swarm. Il faut conserver ce hash.

Utilisation

Environnement de développement

1. Lancer Ganache dans le workspace **monkey-voting**
2. À la racine du projet, ouvrir un terminal et taper la commande **ng serve**
3. Dans un navigateur, se rendre à l'adresse <http://localhost:4200>

Environnement de test

1. Dans un terminal taper la commande **geth attach http://127.0.0.1:8545**
2. Dans un navigateur, se rendre à l'adresse http://localhost:8500/bzz:<swarm_upload_hash>/index.html en remplaçant **<swarm_upload_hash>** par le hash obtenu à la suite de l'upload avec Swarm