# **Testing and Test automation in Web 3.0 (e.g. hardhat testing for etherium, web3js testing)**

Web 3.0 is a new way of building applications that uses decentralized networks like blockchain instead of centralized servers. Ethereum is a popular decentralized network for building decentralized applications (DApps) using smart contracts. Testing and test automation are important in building reliable DApps on the Ethereum network.

Hardhat is a well-known testing and development environment for Ethereum DApps. It lets developers write and run tests in a simulated Ethereum environment, which is a fast and safe way to test smart contracts without using real Ether. Hardhat Network is a built-in network that comes with Hardhat and can be used for testing.

Web3.js is a widely used JavaScript library for interacting with the Ethereum network. It provides APIs that allow developers to send transactions, read smart contract data, and listen to events on the Ethereum network. Web3.js also offers testing utilities that help developers write tests for their smart contracts using popular testing frameworks like Mocha and Chai.

In summary, testing and test automation are crucial for building reliable DApps on the Ethereum network. Hardhat and Web3.js offer powerful tools for testing and development, making it possible to write and run tests in a simulated environment, interact with the Ethereum network through APIs, and use popular testing frameworks to write smart contract tests.

# **Testing and Test Automation in AI and ML (e.g. deepchecks, or checklist)**

AI and ML are increasingly used in software applications for better automation, data analysis, and prediction capabilities. However, testing AI and ML models can be challenging due to their complexity and unpredictable outputs. Thorough testing and test automation are essential for ensuring the reliability and effectiveness of AI and ML models.

DeepChecks is a testing framework that helps developers test their AI models for common errors, such as overfitting, underfitting, and data leakage. It also provides visualizations and explanations of the model's predictions to aid in debugging. DeepChecks works with popular machine learning frameworks like TensorFlow and PyTorch.

Checklist is another tool used for testing and debugging ML models. It provides best practices and guidelines for testing models, including robustness, reliability, and fairness. Checklist also has a set of automated tests that developers can run to ensure their models meet these best practices. It is compatible with several machine learning frameworks, including TensorFlow, PyTorch, and Scikit-learn.

To summarize, testing and test automation are crucial for ensuring the reliability and effectiveness of AI and ML models. DeepChecks and Checklist are tools that can help developers test their models and meet best practices for robustness, reliability, and fairness. These tools offer automated tests and visualizations to simplify debugging and provide insights into model predictions.