**Zatonet el blockchain**

**Solidity** is a programming language specifically designed for developing smart contracts on the Ethereum blockchain. Smart contracts are self-executing contracts with the terms of the agreement directly written into code. Ethereum, being a decentralized platform, allows developers to deploy smart contracts to the blockchain, enabling trustless and automated execution of code.

“El language eli bnkteb beha el smart contarct eli howa el back end “

**Ganache** is a personal blockchain for Ethereum development that you can use to deploy contracts, develop your DApp, and run tests. It is part of the Truffle suite of tools and is often used in conjunction with Truffle for Ethereum smart contract development. Ganache provides a local blockchain environment that allows you to simulate the behavior of the Ethereum network without the need for an actual network connection.

**Local Blockchain:** Ganache runs a personal blockchain on your local machine, which is useful for testing and development purposes. It allows you to interact with the blockchain without the need for an internet connection.

“Mn el akher ganache da bydeek local blockchain on you host tgarab 3aleha “

**Truffle is a Blockchain Development Framework:**

* In the context of blockchain technology and development, "Truffle" refers to a popular development framework for Ethereum. Truffle simplifies the process of developing, testing, and deploying decentralized applications (DApps) on the Ethereum blockchain. It provides tools for smart contract compilation, testing, deployment, and management.

“Truffle de eli btkhalek trfa3 el smart contract bta3ak 3la ganache “

Steps to create a blockchain Project

1-write your smart contract (solidity🡺 programming language)

**(all variables , roles, functions and inputs from user) back end**

2-Deploy contract to ganache network using

**(truffle compile –reset and truffle migrate –reset) 🡺 output contract address**

3-using contract address and ip address of Ganache network in web3 library using (react or angular) to output the data from blockchain on user interface

**(truffle compile –reset and truffle migrate –reset)**