What is OpenZeppelin?

OpenZeppelin is a popular open-source framework for writing secure and scalable smart contracts in the Solidity programming language. It provides a library of reusable and secure contract components that developers can use to build their own dApps on the Ethereum blockchain.OpenZeppelin is used by a wide range of organizations and individuals to build dApps like Aave, Balancer and Maker.
While the primary platform that it interacts with is Ethereum, it is definitely compatible with other blockchain platforms such as BSC (Binance Smart Chain).

Some of its features are:

1. It provides a range of reusable contract libraries that developers can use to build dApps and contracts for ERC-20 tokens, crowdsales, voting systems, etc. These contracts are secure and easy to use, and have been thoroughly tested and audited by the OpenZeppelin community.
2. It comes with a comprehensive testing framework that developers can use to write and run tests for their smart contracts. This helps ensure that the contracts are working as intended and are free of bugs and vulnerabilities.
3. It has a user-friendly interface that makes it easy for developers to integrate its libraries into their projects. It also includes extensive documentation,support resources and it has a large and active community of developers which makes it easier for developers to get help with their projects and stay up to date with the latest developments in the field.

OpenZeppelin has a smart contract library with different contract templates that we can use. It also provides tools for automated contract testing and deployment as well as a security-focused development framework called OpenZeppelin SDK. It also provides a range of contract libraries that are relevant to the DeFi space. It is also widely used in other areas such as supply chain management, identity verification, etc.

OpenZeppelin example use cases include (but are not limited to) the following items:

- **Access Restriction:** It allows us to determine who can access particular resources and functionalities in a system. Using this we can restrict voting, minting, sending transactions, etc., to specific entities.
- **Ownership:** Deploying this contract would help specific entities to acquire certain capabilities. We can also transfer ownership from one account to the other and also renounce the previous ownership.
- **Calculations:** “SafeMath” library is useful to prevent operation overflow when coding in Solidity.

- **Tokens:** OpenZeppelin created the required infrastructure for developers to make tradable tokens and allocate them to the participants inside your network. Using this contract, we can monitor the prices, specify token transfer methods, authenticate purchases, etc.
- **Payments:** Using the OpenZeppelin “paymentSplitter” contract, we can send the income from our dApp purchases to different parties.
- **Gas Station Network:** This contract helps us to develop dApps in which we will pay on behalf of our users. Users do not need to hold ETH in their wallets, and we will instead pay for them.