The zkSync Era Remix Plugin: A How-To Guide

[Bohdan Ohorodnii](#)

[Follow](#)

Nethermind.eth

--

Listen

Share

Introduction

The Remix Project, an open-source and community-led developer toolset, is frequently utilized in the field of Ethereum smart contract development. It stands out as a highly configurable environment that allows developers to write, test, debug, and deploy Ethereum smart contracts directly in their browser. This guide introduces an important addition to the Remix toolset: the zkSync Era Remix Plugin, developed by Nethermind and Matter Labs.

The zkSync Era Remix Plugin provides a smooth and user-friendly interface for developers of all skill levels to engage with the zkSync ecosystem. The plugin simplifies the writing and deployment of zkSync's smart contracts, making it accessible to both newcomers and experienced users.

We'll show you the key features of the plugin, and take you through a step-by-step tutorial.

Benefits of the zkSync Era Remix Plugin

1. Seamless transition between different zksolc versions

You have the flexibility to choose the version that best fits your project needs — upgrade to the most recent release for the latest features, or continue using an older or preferred version for consistency. At the time of writing, the plugin supports 1.3.13

,

and the latest release, with a roadmap to include more versions.

2. Simplified compilation

The plugin offers a simplified compilation process for contracts, with an isolated Hardhat environment under the hood. You won't have to install various software on your PC — you can code from any device with an internet connection!

3. Flexible deployment

You have the freedom to experiment with 4 different environments and deploy contracts on:

- A remote devnet:

A hosted devnet zkSync Era node, shared among everyone to speed up development time. This setup includes 10 well-funded accounts, simplifying the deployment and interaction with contracts.

- A local devnet:

A devent that enables you to run a node locally using [zksync-cli](#) and connect to it from your browser. It also features funded accounts for testing but offers a more isolated and faster environment compared to the remote devnet.

- A wallet:

Select and use as an account to interact with mainnet or testnet. You can use most of the widely-used wallet options available today.

- A manual account:

Allows creating new accounts that can be used as a temporary connection to the zkSync Era Testnet.

4. Easy interaction

Interact directly with your deployed contracts in the same environment you used for deployment. The results of each transaction or query are shown in the terminal logs. This speeds up development time by removing the need for additional scripts to interact with contracts. Simply press the button, and you're set!

5. Effortless transaction tracking

Easily access your transaction details on the block explorer with a simple click on the transaction ID or the sender's address (From:0x…). This is applicable for transactions from manual accounts or wallet-type environments.

How to Use the zkSync Era Remix Plugin

Step 1: Install the plugin

1. Visit [The Remix Project](#).

2. Access the plugins: look for the 'Plugins' section located at the bottom left corner.

3. Find "zkSync" and enable it.

4. Once enabled, you'll notice the zkSync logo on the left sidebar. Click on it to interact with opened Solidity files.

Step 2: Compile Solidity

Click Compile with a valid Solidity file open. After the compilation, you'll have your artifacts in the artifacts folder near your contracts for further inspection. This will also allow the deployment of said contracts.

Step 3: Deploy contracts

For this, you'll need to decide on an environment.

1. You can choose to use our remote devnet, from which you'll have access to 10 accounts to play around with.

2. Alternatively, you can do the same with a devnet you run on your machine! Use zksync-cli for this purpose.

3. You can also select the wallet you want to use as an account and interact with mainnet or a testnet. If you're not interested in using your account for this, create a test account, fund it from the faucet, and deploy it.

4. Select your desired environment, copy your accounts, and play around! After that, it's a matter of going to the Deploy tab, selecting your compiled contract, and clicking Deploy!

Step 4: Calling contract functions

After your deployment is complete, navigate to the Interact tab where you'll find a list of methods from your contract. Here, you'll be able to use both read and write methods to interact with your contract.

Step 5: Explore transactions

Here are the steps to explore transactions in zkSync Remix IDE plugin:

1. After interacting with your contract in the Interact tab, you can view the transaction history.

2. Click on the 'Transactions' tab to see all the transactions you have sent.

3. You can also click on each transaction to view more details or check them on the block explorer if they were on the testnet.

That's a wrap, have fun Remixing on zkSync!

Contributing

The zkSync Era Remix Plugin is currently in alpha so there are numerous ways to get involved! Spotted any issues? Head on over to our [good first issues](#) or read through our [Contribution Guidelines](#) to get started.

About Nethermind

Nethermind is a team of world-class builders and researchers. We empower enterprises and developers worldwide to access and build upon the decentralized web. Our work touches every part of the web3 ecosystem, from our Nethermind node to fundamental cryptography research and infrastructure for the Layer-2 ecosystem.

If you're interested in solving some of blockchain's most difficult problems, visit our

[job board

](https://www.nethermind.io/open-roles)!