

Wallet Selector

An easy-to-navigate modal that allows users to select their preferred wallet to easily interact with the NEAR protocol.

Launched in March 2022 by the NEAR Foundation, this simple modal will appear whenever users are given the option to "Connect Wallet" to the NEAR blockchain.

Initial screen of [Wallet Selector](#)

Framework agnostic

[React /Next.js](#) and [Angular](#) variations of the [Guest Book](#) dApp can be found in the [examples](#) directory. Developers can use these to gain a concrete understanding of how to integrate NEAR Wallet Selector into their own dApp.

Unlocking the wallet ecosystem

Wallet Selector makes it easy for users to interact with dApps by providing an abstraction over various wallets and wallet types within the NEAR ecosystem.

You can check the current list of supported wallets in the [README.md](#) file of near/wallet-selector repository. Thanks to NEAR's open and inclusive approach, other wallet developers can contribute to the NEAR ecosystem by following the documentation and instructions on the [NEAR Github repository](#) on how to add a new wallets to the Wallet Selector.

To learn more on how to include new wallets for Wallet Selector you can check the listing criteria for third party wallets on this [link](#).

Install

The easiest way to use NEAR Wallet Selector is to install the core package from the NPM registry, some packages may require near-api-js v0.44.2 or above check them at packages.

npm install near-api-js@^0.44.2 npm install @near-wallet-selector/core Next, you'll need to install the wallets you want to support:

```
npm install \ @near-wallet-selector/near-wallet \ @near-wallet-selector/my-near-wallet \ @near-wallet-selector/sender \
@near-wallet-selector/nearfi \ @near-wallet-selector/here-wallet \ @near-wallet-selector/math-wallet \ @near-wallet-selector/nightly \
@near-wallet-selector/meteor-wallet \ @near-wallet-selector/ledger \ @near-wallet-selector/wallet-connect \
@near-wallet-selector/nightly-connect \ @near-wallet-selector/default-wallets \ @near-wallet-selector/coin98-wallet
```

Setup Wallet Selector

Optionally, you can install our [modal-ui](#) or [modal-ui-js](#) package for a pre-built interface that wraps the core API and presents the supported wallets:

npm install @near-wallet-selector/modal-ui Then use it in your dApp:

```
import
```

```
{ setupWalletSelector }
```

```
from
```

```
"@near-wallet-selector/core" ; import
```

```
{ setupModal }
```

```
from
```

```
"@near-wallet-selector/modal-ui" ; import
```

```
{ setupNearWallet }
```

```
from
```

```
"@near-wallet-selector/near-wallet" ;
```

```
const selector =
```

```
await
```

```

setupWalletSelector ( { network :
"testnet" , modules :
[ setupNearWallet ( ) ] , } ) ;

const modal =
setupModal ( selector ,
{ contractId :
"test.testnet" , } ) ;

modal . show ( ) ; Required CSS To integrate the Wallet Selector, you also need to include the required CSS:
import "@near-wallet-selector/modal-ui/styles.css"

```

API Reference

The API reference of the selector can be found [here](#)

Wallet API

Sign in

```

// NEAR Wallet. ( async
( )
=>
{ const wallet =
await selector . wallet ( "my-near-wallet" ) ; const accounts =
await wallet . signIn ( { contractId :
"test.testnet"
} ) ; } ) ( ) ;

```

Sign out

```

( async
( )
=>
{ const wallet =
await selector . wallet ( "my-near-wallet" ) ; await wallet . signOut ( ) ; } ) ( ) ;

```

Get accounts

```

( async
( )
=>
{ const wallet =
await selector . wallet ( "my-near-wallet" ) ; const accounts =
await wallet . getAccounts ( ) ; console . log ( accounts ) ;
// [{ accountId: "test.testnet" }] } ) ( ) ;

```

Verify Owner

```
// MyNearWallet ( async
( )
=>
{ const wallet =
await selector . wallet ( "my-near-wallet" ) ; await wallet . verifyOwner ( { message :
"Test message" , } ) ; } ) ( ) ;
```

Sign and send transaction

```
( async
( )
=>
{ const wallet =
await selector . wallet ( "my-near-wallet" ) ; await wallet . signAndSendTransaction ( { actions :
[ { type :
"FunctionCall" , params :
{ methodName :
"addMessage" , args :
{ text :
"Hello World!"
} , gas :
"3000000000000000" , deposit :
"1000000000000000000000000000000" , } , } , ] , } ) ; } ) ( ) ;
```

Sign and send transactions

```
( async
( )
=>
{ const wallet =
await selector . wallet ( "my-near-wallet" ) ; await wallet . signAndSendTransactions ( { transactions :
[ { receiverId :
"guest-book.testnet" , actions :
[ { type :
"FunctionCall" , params :
{ methodName :
"addMessage" , args :
{ text :
"Hello World!"
} , gas :
"3000000000000000" , deposit :
```

"1000000000000000000000000", }, },], },], },) ; }) ([Edit this page](#) Last updated on Feb 9, 2024 by [Damián Parrino](#) Was this page helpful? Yes No

[Previous NEAR CLI RS](#) [Next Explorers](#)