Installation

The best way to access Request Network is using the Request Network SDK with a Request Node. In this way, the Request Node operator pays the protocol fee for creating requests and reduces the number of transactions signed by the end user resulting in a better user experience.

The Request Network SDK is split into multiple packages so that Builders can pick and choose the features they need.

SDK Packages

These are the packages that we think would be most commonly used by Builders to build applications.

Package	Description
`@requestnetwork/request-client.js`	Create, update, and retrieve requests.
`@requestnetwork/web3-signature`	Sign requests using web3 wallets like Metamask

Package	Description
`@requestnetwork/epk-signature`	Sign requests using Ethereum private keys
`@requestnetwork/data-format`	Standards for data stored on Request, like invoice format
`@requestnetwork/epk-decryption`	Decrypt encrypted requests using Ethereum private keys
`@requestnetwork/payment-processor`	Pay a request using a web3 wallet
`@requestnetwork/request-node`	Web server that allows easy access to the Request system

For a list of internal SDK packages, see Internal SDK Packages.

Component Packages

These packages offer pre-built components for quickly integrating certain Request Network features.

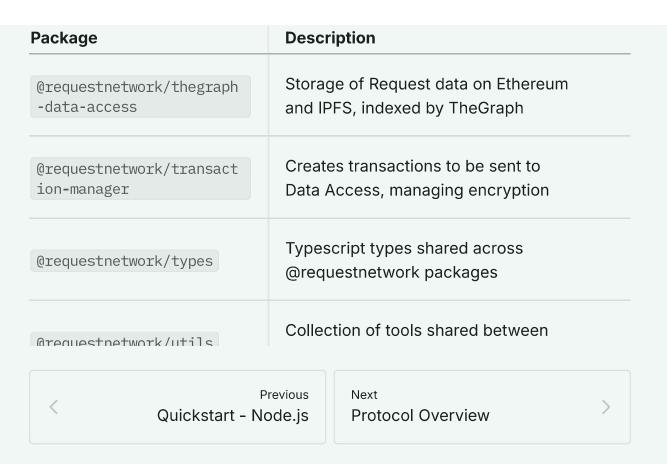| Package	Description
`@requestnetwork/create-invoice-form`	A form for creating invoices in Request Network
`@requestnetwork/invoice-dashboard`	A dashboard for viewing and paying invoices in Request Network
`@requestnetwork/add-stakeholder`	A dialog box for granting third-party access to an encrypted invoice created via Request Finance

Import the packages

The Request Client library can be imported as ES6 or CommonJS modules.

ES6	CommonJS

```
import { RequestNetwork } from '@requestnetwork/request-client.js
import { Web3SignatureProvider } from '@requestnetwork/web3-signa
import { payRequest } from '@requestnetwork/payment-processor';
```

Internal SDK Packages

These packages are published publicly but contain functions that are considered internal to the SDK Packages. It is less likely that a Builder would need to use these packages.

Package	Description
`@requestnetwork/advanced-logic`	Extensions to the protocol
`@requestnetwork/data-access`	Indexing and batching of transactions
`@requestnetwork/ethereum-storage`	Storage of Request data on Ethereum and IPFS, with custom indexing
`@requestnetwork/multi-format`	Serialize and deserialize object in the Request Network protocol
`@requestnetwork/payment-detection`	Payment detection, to compute the balance.
`@requestnetwork/request-logic`	The Request business logic: properties and actions of requests
`@requestnetwork/smart-contracts`	Sources and artifacts of the smart contracts

Package	Description
`@requestnetwork/thegraph-data-access`	Storage of Request data on Ethereum and IPFS, indexed by TheGraph
`@requestnetwork/transaction-manager`	Creates transactions to be sent to Data Access, managing encryption
`@requestnetwork/types`	Typescript types shared across @requestnetwork packages
`@requestnetwork/utils`	Collection of tools shared between

Last updated 5 months ago