Package 'ShinyBlock'

June 11, 2026

Type Package

Title Multi-Protocol Blockchain Simulator and Enterprise Ledger Framework

Version 0.1.3

Maintainer Isaac Osei <ikemillar65@gmail.com>

Description An interactive framework for simulating blockchain protocols using a hybrid 'R-Shiny' and 'Python' architecture. The package provides tools to visualize peer-to-peer network maps, manage supply chain logistics on-chain, and execute cross-border settlements via smart contract logic. It leverages the 'reticulate' package to perform standardized cryptographic operations, including 'SHA-256' hashing, 'Merkle' Tree construction, and 'ECDSA' (Elliptic Curve Digital Signature Algorithm) key generation. This tool is designed for pedagogical demonstration and rapid prototyping of distributed ledger requirements.

License MIT + file LICENSE

Encoding UTF-8

Imports shiny, reticulate, reactable, networkD3, bslib, jsonlite

Suggests testthat (>= 3.0.0), knitr, rmarkdown

SystemRequirements Python (>= 3.7), ecdsa (Python package)

VignetteBuilder knitr

RoxygenNote 7.3.3

Config/testthat/edition 3

URL https://github.com/ikemillar/ShinyBlock

BugReports https://github.com/ikemillar/ShinyBlock/issues

NeedsCompilation no

Author Isaac Osei [aut, cre],
Yamini Alakunta [aut]

Repository CRAN

Date/Publication 2026-06-11 12:10:18 UTC

Contents

`install_blockchain_deps` *Install Python Dependencies*

Description

Creates an isolated virtual environment and installs the required 'ecdsa' library to support asymmetric key generation.

Usage

```
install_blockchain_deps()
```

Value

No return value, called for side effects to configure the 'Python' environment.

Examples

```
if (interactive()) {
  install_blockchain_deps()
}
```

`launch_blockchain` *Launch the Blockchain Protocol Simulator*

Description

Initializes the Python cryptographic environment and launches the interactive 'ShinyBlock' dashboard application inside the default web browser.

Usage

```
launch_blockchain()
```

Value

No return value, called for side effects to launch the 'Shiny' application.

Examples

```
if (interactive()) {
  launch_blockchain()
}
```

Index