

Description

LDK (Lightning Development Kit) is powered by [rust-lightning](#), a lightning implementation designed to be modular, flexible & customizable. It allows you to build a lightning node without having to worry about all of the low-level lightning logic like the state machine or routing.

The lightning network has multiple implementations including c-lightning, lnd, ldk, electrum, eclair etc. This is great and helps keep the network decentralized but we need to make sure they are all interoperable and work together. The [BOLT](#) specification defines a standardized technical specification for the implementation of the Lightning Network.

How can we ensure all implementations are following the specifications?

Vigorous testing. Inprotest is a test suite for the Lightning Network Protocol created by Rusty Russell. It was designed to plug into any lightning specification and verify that it correctly follows the spec. This project would aim to integrate LDK into that testing framework.

Expected outcomes

- Learn about some of LDK's core API's and how to use them
- Create a script similar to [this](#) in the Inprotest repo that exercises the LDK implementation

Resources:

- <https://github.com/lightningdevkit/rust-lightning>
- <https://github.com/rustyruessell/inprotest>
- <https://github.com/lightning/bolts>

Skills required:

- Python
- Rust
- Git

Mentors: Conor Okus (<https://github.com/ConorOkus>, <https://twitter.com/ConorOkus>)

Difficulty: Medium