

STEPS Proposal

Alex Groleau



Phase I - Minimum Viable Product

Smart Contract(s)

Determine Contract Requirements with Team 1 week

Core Solidity Contract Code & Test Development 1-2 weeks

- > Use Truffle or Chronicle Solidity frameworks
- > Create the core set of contract(s) that will manage the escrow process
- > Ensure automated testing of all contract functions

Hookup Contract to ChainLink (or Other Oracle Service) & Tests 1-2 weeks

- > Get approved by ChainLink for a triggering service
- > Use Oraclize or other oracle provider (if ChainLink does not meet requirements)
- > Set up MVP courier API service with the contract
- > Deploy to Ropsten for real-world testing of contract

Mobile DAPPs

Determine UI Requirements with Team 1 week

- > Ideas, design, user flow, look and feel, and wire-framing

Core React Native DAPP Prototype 2-3 weeks

- > Support Android & iOS
- > Develop using Reactive Native for easy portability and parity between mobile platforms
- > Coordinate user testing (with mock data and no APIs connected)

React Native Geth Integration (Light Ethereum Client) 1-2 weeks

- > Add Geth to the mobile DAPPs for local wallet management and for signing transactions

Integrate with INFURA Public Ethereum Node Network 1-2 weeks

- > With Geth integrated, INFURA can be used to communicate with the core contract(s)
- > The DAPP should now be in a usable production state, albeit incomplete
- > Add additional automated integration tests with a now connected UI

Phase II - General Availability

Extend Oracle Service

Support Additional Courier APIs 1-2 weeks

Extend Mobile DAPPs

Reach Final UI Design **1-2 weeks**

> Using results of all user testing up until now

Implement Mnemonic Wallet Recovery < **1 week**

> Just in case the private wallet keys are lost or a user is moving between devices

Implement Two-Factor Authentication Scheme < **1 week**

Submit to Apple AppStore and Google Play Store < **1 week**

Phase III - Scaling

Cloud Infrastructure

Choose Cloud Service Provider (Digital Ocean, AWS, Other) < **1 week**

Set up Own Ethereum Node Cluster (Parity & Geth) **1 week**

- > As the user base grows, it might become necessary to break away from INFURA and create a small Ethereum node cluster to better manage contract API response times
- > Setup Docker Swarm cluster with desired replica count of public Ethereum nodes
- > A second datacenter location will allow for proper disaster recovery and redundancy; it would be a mirror of the first