Noah Grant Application

The story of Blixt

Since I got into Bitcoin I have always been interested in the idea of using it for every day payments. We have the ability to build something from the ground up that is more efficient and user friendly for people.

Back in 2019 I started working on Blixt Wallet, a self-custodial Lightning wallet that has an open-source ethos, with good defaults but with the flexibility to customize if the user wants to, including opening channels to any Lightning node you want.

Blixt has grown and stands today as a powerful wallet with a strong, albeit niche, community. We now have a small team of contributors and community members.

We have learned a lot over the years and encountered some of the obstacles of Lightning for end-user wallets. Particularly related to channel management, liquidity, backup and offline payments. This is why we are building a new wallet codenamed “Noah”.

Researching Ark

With the experience we have gained from Blixt, we started researching and looking into Ark. Ark is a scaling solution for end-user wallets that depends on the Lightning Network. We believe it has the potential to improve the user-experience for self-custodial wallets.

This is because Ark does not have the concept of payment channels, but instead has the concept of VTXOs (Virtual Transaction Outputs). You are able to receive payments without first opening a new payment channel which normally would require an onchain transaction as in the case of Lightning. Additionally you don’t have the obstacle of inbound/outbound liquidity thresholds.

However, in the end the Ark protocol is still dependent on the greater Lightning Network to be able to send and receive payments. This is important because Lightning today is an established and mature payment channel network that we do not wish to compete head-on against. Ark works well for end-user wallets.

Where we are and the road ahead

Blixt dev Nitesh Balusu started working on a new wallet codenamed "Noah" using the Ark protocol. The wallet uses Second's Ark client Bark under the hood.

The goal here is to focus primarily on ease-of-use while still being a self-custodial wallet, taking inspiration from wallets such as Aqua and Muun. These wallets were able to successfully compete against custodial wallets such as Wallet of Satoshi.

We want to provide a wallet with a single balance that can seamlessly send and receive via Lightning and onchain.

We currently have a working wallet for Android and iOS that can send and receive payments. However, we are still working on base functionality related to the Ark protocol itself, as in Ark, you have to refresh your VTXOs.

As this is a new wallet we have the opportunity to rethink the user experience from the ground up. Here are the primary features and functionalities we would like to focus on:

Ark round syncing

A proper Ark wallet currently requires refreshing your VTXOs. We are working on a solution for this via silent push notifications. Care must be taken so that this procedure works reliably.

Lightning Address and/or BIP353 support

Lightning Address (and potentially BIP-353 in the future) will be a common way to receive payments in the wallet.

Having Lightning Address support in a self-custodial wallet is uncommon, so this is a valuable feature to have.

We have basic support for this feature already in the wallet, but we would like to expand on this by making it production-ready, having a donation webpage as well as supporting more of the related LNURL protocols such as LUD-18.

Easy backup management

We are planning to have a backup system that is solely based on the seed phrase, that would restore all funds in the wallet. This would encrypt the backup with the user's seed phrase, making it trustless.

”Offline” payments

We would like to have a way to receive payments while the app is not running. We plan to do this via silent push notifications, in order to wake up the device to receive the payment. This would work together with the Lightning Address support.

Auxiliary Lightning and Nostr ecosystem protocols

We intend to support Lightning-related protocols such as the LNURL protocol suite, WebLN, Nostr Wallet Connect and more where applicable.

Additionally, we are looking into receiving and sending payments via Nostr profiles and potentially developing this into an open standard.

For privacy matters we will add Tor support to the wallet.

Continuing Blixt Wallet work

While our primary focus will be on Noah, Blixt is not going away. Work will be dedicated to continuing its development. It is a versatile wallet liked by tens of thousands of users.

Blixt will be shifted to be a power-user wallet and we plan to work on features such as BOLT12-support (coming in an upcoming lnd release), offline-receive, faster chain sync, desktop versions (Windows and Linux), LN/onchain swap, LSP-spec support, some GUI-revamping and more.

Potential Impact

The potential impact of Noah and the Ark protocol is to make self-custodial wallets just as user friendly as custodial wallets. It would be able to support Lightning Address, easy backups, offline payments and more.

It would simplify onboarding by letting users instantly receive payments using Ark’s concept of VTXOs. Users can send and receive payments without liquidity thresholds.

Overall we wish to demonstrate that self-custodial wallets don’t have to be complicated. Lightning helped Bitcoin scale and provide fast payments and low fees. However, it added an extra layer of friction for end-user wallets. We believe that Ark and Noah help on the last mile by removing this friction.

We believe that this will support Bitcoin’s evolution from a Store of Value to a Medium of Exchange.

Ending thoughts and financials

We are excited about the future of Lightning and the Ark protocol. We believe Ark has the potential to provide a user experience that is close to custodial Lightning wallets, where contemporary self-custodial Lightning wallets sometimes fall short.

If you would like to see what we have accomplished thus far, we can invite you to download the app for Android and iOS.

To keep going with our project we are seeking funding for the following:

Hampus Sjöberg

Programmer
Will work on developing the wallet and auxiliary services and servers.
Additionally responsible for guiding overall project direction.
Email: hampus.sjoberg at protonmail.com
Nostr: `npub1v4v57fu60zvc9d2uq23cey4fnwvxlzga9q2vta2n6xalu03rs57s0mxwu8`
Telegram: @hsjoberg
GitHub: hsjoberg
$80k

Nitesh Balusu

Programmer
Will work on developing the wallet and auxiliary services and servers.
Email: niteshbalusu at protonmail.com
Nostr: `npub1qgwhaaa2lsp54rl0hfx7qa3z678ax6wlre0em475rhpvl7n54cpqgg7y7n`
Telegram: @nitesh_btc
GitHub: niteshbalusu11
$80k

Salman Khan

Project Manager and Support
Will create actionable development milestones, coordinate with external partners including UI/UX designers, manage project timelines, handle user support and maintain project documentation.
Email: salmanzrul at gmail.com
Nostr: `npub1ssqkc9yk22lxacfs4cj2p46zcgah4nh33xeqxzqk5f4e23z8gp6qace082`
Telegram: @SalmanKhanBTC
$60k

Total: $220k

Contact

- Email: hampus.sjoberg at protonmail.com
- Hampus Telegram: @hsjoberg
- Telegram group: @BlixtWallet (we release signet builds of Noah here)
- GitHub project: BlixtWallet/noah
- Source-code for this page: hsjoberg/grant

Thank you for considering our grant application.