



# Bitcoin UX Research Kit

## Building in public

*First draft by Christoph on March 23, 2023.*

Building in public means to co-create with your community to help ensure that you are building something that resonates with them.

### **Benefits**

[Transparency](#) is a key principle of bitcoin. The software itself is open-source. The development process is carried out in the open. Anyone with an internet connection can use the network, and transactions are stored publicly for anyone to see. In bitcoin design, we embrace the term [Open Design](#) to port this mentality to design practices.

Most bitcoin applications address financial use cases, so users need to be able to trust them (and their creators). By building in public, you give users the opportunity to evaluate your practices and then decide whether to trust or not (“don’t trust, verify”). This is especially important in open source, where projects are

often provided as-is, and there's no support team or company to help when something goes wrong.

Building in public provides an important co-creation dynamic. Instead of only receiving feedback after your big launch, you receive it continuously. This provides many more chances to adjust along the way and build something that is genuinely wanted and useful. You are not just building for others, but *with* them.

A practical side effect of this approach is also that you are building an audience, and potential user base, before your project is ready for the spotlight.

## **How-to**

A good mental model for this tool is to think of a project as an idea that grows over time, like a small seed that turns into a towering tree. In the Bitcoin Design Community, we have created a helpful [page](#) that outlines this process and the steps involved for community projects. Consider how each step serves to validate a growth phase through interaction with your audience or community. How can you adopt this idea for your own project, whether you're starting from scratch or working on a specific feature?

## **Share the big picture**

Clearly articulate what problem you are looking to solve and make it crystal clear to anyone coming in contact with your work. Build

on this by having a public roadmap where users can see what you are working towards. Invite them to help define and prioritize these goals.

A good example is the [BTCPay server roadmap](#), which allows anyone to see at all times what is coming up in the next few versions, as well as tasks that may get picked up later. Milestones are not top-down imposed, but simply a matter of logically grouping issues that the community has identified and deemed important.

## **Work in open repositories**

We know [GitHub](#) as an open code repository for public collaboration, but this practice can also apply to other areas, for example:

- Publish your main design in Figma community ([Bitcoin UI Kit](#))
- Create dedicated design documentation ([Alby](#), [Bitcoin Core App](#))
- Create a [UX research repository](#) ([Bitcoin Core App](#))
- Set up a localization portal ([BTCPay](#))

Each repository should be optimized for its specific use case and the workflows that fit best for participants.

## **Share work-in-progress**

The process to a finalized design can be long and winding. Much thinking and work happens that never sees the light of day, yet it can be extremely valuable for collaboration and ongoing feedback. Some practices to consider:

- Record calls and keep notes of discussions and put them in a central place (Wallet Scrutiny [videos](#) and [notes](#))
- Write a weekly update sharing loose and unfinished bits of your progress. Don't take it too serious or spend much time on it ([Christoph's updates](#))
- Record short videos of work-in-progress and explain your thought process. You may be happy to come back a year later to remind yourself why you did something a certain way ([Saving Satoshi update](#))

Sharing the process is not just helpful for taking others along, but also helps clarify your own thought process. If you are hesitant about sharing something, consider doing it anyway.

## **Invite continuous feedback**

As you make all these things public, feedback may come in across many different channels and in many different forms. This can quickly become overwhelming, so it is important to provide clear pathways for your audience to respond in just the right way. This allows you to respond appropriately, which in turn ensures users feel taken serious and that their input enters the product direction. Some feedback channels to consider:

- Social media accounts for questions and short conversations
- An open online community (BOLT.FUN, Discord, Telegram, Reddit...)
- In-person feedback at meetups, conferences and other events
- Comments on your weekly updates and videos
- GitHub discussions & issues
- A feedback form

You want open communication channels, but you also don't want to be overwhelmed or answer the same questions 100 times. Think through how you are using each channel, what users prefer each one and what type of communication works best for them. How can you use this feedback to inform the roadmap?

Pair this with seeking out specific feedback, from moderated usability tests to simply reaching out to specific users directly. You may not automatically get input on the areas you want it on. Being close to a project means you have a [deeper understanding](#) than most others. Think about how you can ensure the project as a whole gets the feedback it needs (without imposing your own opinions).

## **Wrapping up**

Building in public is fun because it is such a social activity. We are not used to it from traditional workspaces, and designers often have a harder time with this than developers. By switching from

“private by default” to “public by default,” we open ourselves up to a new way of collaborating that allows us to build things we otherwise could not. We also invite much more conversation which can sometimes be tricky to manage. If you’re new to this way of working, be bold and be patient. It’s worth it.

## **Additional reading & resources**

- [Working in Public: The Making and Maintenance of Open Source Software](#) (Book)
- [5 Reasons for Building in Public](#) (Blog post)
- [#learninpublic](#) (Blog post)