

W3F Grant Proposal

This document will be part of the terms and conditions of your agreement and therefore needs to contain all the required information about the project. Don't remove any of the mandatory parts presented in bold letters or as headlines! Lines starting with a > (such as this one) can be removed.

See the [Grants Program Process](#) on how to submit a proposal.

- **Project Name:** Expandable Substrate Module and Storage Network Compatible w/ BitBadges
- **Team Name:** BitBadges
- **Payment Address:** [0xC8F72de05110352684A1e9e4acB7AEF112220698](#) (ETH)
- **Level:** 2

⚠ The combination of your GitHub account submitting the application and the payment address above will be your unique identifier during the program. Please keep them safe.

Project Overview

This isn't in response to a RFP or a request for a follow-up grant.

Overview

We are working on creating a storage network module on substrate compatible w/ the [BitBadges](#) blockchain. This will allow for substrate to interact in a meaningful way with the BitBadges ecosystem, which follows an experimental PoCP or Proof of Computation proof.

Other projects like [Crust](#) or [Subspace](#) are within the same realm of trying to create some form of storage network. The difference is how BitBadges goes about accomplishing this and what a substrate module as an integration will provide. BitBadges as a whole is working on offline-centric networks and integrating variations of distributing sharding algorithms. Their data is transmitted w/ CouchDB being an integral part of its core for node synchronization. BitBadges is also integrating w/ various third party networks viewed as either pegs or applications.

We believe that the substrate ecosystem can provide another integration w/ our network and we might even be able to create something similar to what Crust is doing on their mainnet, but w/ BitBadges as the storage method or a mirrored peg for the data. Our team, which is currently just two people, are passionate and have years of blockchain development experience under our belts. We think Substrate can work great as one of our third party integrations or pegged chains.

Project Details

Currently, we have built a variety of technologies in relation to BitBadges (1.0) for the preparation of this proposed release and its other sidechains. Among those are, building the initial [database schema](#), and launching the MVP, getting hundreds of early users, and having migrated to a more decentralized system and our own blockchain.

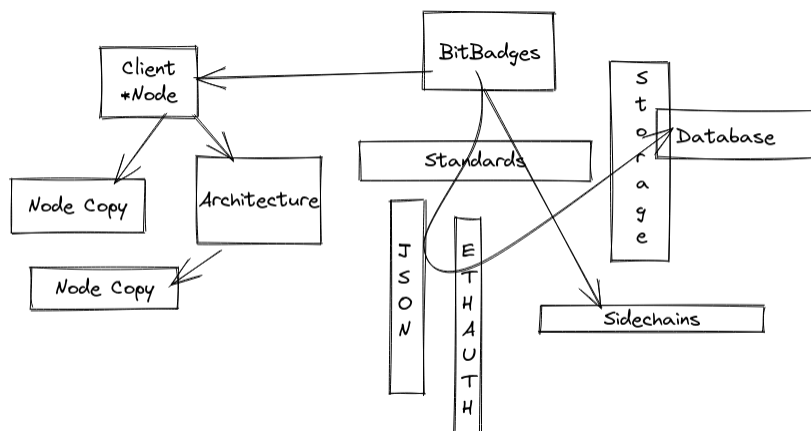
BitBadges (2.0) or BitBadges has a [paper](#) on IACR preprints introducing its architecture. Its architecture is centered around PoCP proofs i.e Proof of Computation. BitBadges is validated through synchronized nodes from those who want to host daemons, as more and more nodes are interconnected, the network becomes more and more decentralized. This allows compute through an offline-centric network, and it utilizes a variety of technologies such as the [decentralized-internet SDK](#) and [CouchDB](#) to perform this.

The use-case of BitBadges is centered around identity, specifically within the privacy-preserving realm. The first application of BitBadges was utilizing its database schema to issue badges in relation to different users on the blockchain. For permanence, BitBadges in its new form will utilize [Blake3](#) and its own blockchain for the core hashing integration. This is as opposed to IPFS which may now be looked at as potentially more of a future sidechain integration.

We believe that BitBadges will be a great fit for the Substrate ecosystem. Substrate allows for quick scalability and even the ability to easily make custom RPC calls. It also has compatibility w/ EVM or the Ethereum Virtual Machine. While BitBadges already integrates w/ rollup layers including [CloutContracts](#), we think an EVM integration separate from directly requiring deployments to Ethereum's network is a plus.

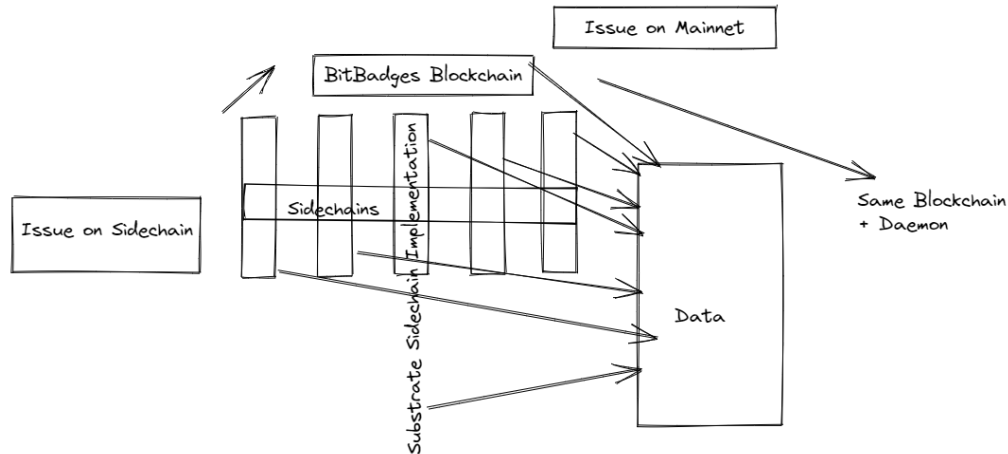
Outside of wanting to utilize the EVM module w/ substrate, we want a substrate chain for other reasons as well. We are looking to potentially deploy as a sidechain integration for expandability [something like what Crust is doing w/ maxwell or its mainnet (as mentioned)]. We also believe that many people who want to develop or integrate w/ BitBadges may want to code in Rust, Wasm, or Solidity.

The current UML to demonstrate the flow of BitBadges is:



BitBadges at its core contains various parts. First, is the BitBadges blockchain which includes the storage method and the CouchDB database. You have different standards in regards to badges being issued (JSON structures and similar to ERC721), and there are many different sidechains being integrated with the BitBadges ecosystem. The core of BitBadges is the client node, and more and more people can host copies of BitBadges and create P2P networks within each other.

The UML in regards to the substrate sidechain is:



In regards to the Substrate sidechain, one needs to keep into consideration how sidechains are structured. Sidechains still issue within the same blockchain, but are interoperable. This means that if someone issues on the Substrate sidechain for example, the data is still synced along w/ what was issued on the BitBadges mainnet (regular BitBadges blockchain). That same data can still be hosted and synced across nodes.

Ecosystem Fit

We are targeting many developers who want to build on top of BitBadges as well as further expanding our options and integrations w/ the Substrate sidechain (as seen in the target audience). Other cryptocurrency networks such as [Skynet](#) also benefited from integrating Substrate. We believe many likeminded developers will build on top of BitBadges as a result of the Substrate sidechain, and this allows both networks to benefit as a result.

Target Audience:

- We are targeting developers who want to build on top of BitBadges
- Devs who use our Substrate sidechain will also be expanding the Substrate ecosystem (given its a substrate sidechain)
- This should also open up more [Wasm](#) developers and those with experience w/ Ethereum's EVM to want to do further dev work

Team

Team members

- Andrew M. K. Nassief
- Trevor Miller

Contact

- **Contact Name:** Andrew M. K. Nassief
- **Contact Email:** andrew@cloutpool.net
- **Website:** <https://bitbadges.web.app>

Legal Structure

- **Registered Address:** P.O. Box #100. Washington Township, MI 48094
- **Registered Legal Entity:** BitBadges (Becoming a sub-entity LLC.)

Team's experience

Both members of the team have robust development experience. Trevor is currently pursuing a BS/MS dual degree and has experience developing a variety of open-source projects including BitBadges. Andrew, the team leader has years of development experience behind his belt and experience with a vast array of research across a variety of topics. His research across a variety of industries is quite extensive.

Past experiences include work within the realms of Computational and Theoretical Physics, Quantum Engineering, and Financial Engineering. He has experimented with high performance computing in the past, and built quant trading algorithms. He also developed an obsession w/ superconductivity and Qubit processing mechanisms. He has experience contributing to the development of technology within a variety of startups.

This is the first time that this team is applying to a grant from the Web3 Foundation.

Team Code Repos

- <https://github.com/BitBadges>
- <https://github.com/BitBadges/bitbadgesbackend>

- <https://github.com/BitBadges/bitbadges.github.io>

Please also provide the GitHub accounts of all team members.

- <https://github.com/Mentors4EDU>
- <https://github.com/trevormil>

Development Status

Right now in regards to the substrate sidechain, we have the initial idea planned and the BitBadges blockchain ready to start integrating sidechains. In regards to BitBadges development, we are near the final phase of BitBadges 2.0 which integrates CouchDB and its proposed offline-centric capabilities. A lot has been done such as working on the API to CouchDB integration, integrating EthAuth, doing the database migration, and having a MVP. We are essentially trying to really push development for the release of the BitBadges 2.0 software update.

In regards to public-facing code, the links are as follows:

- API: <https://github.com/BitBadges/bitbadges.github.io>
- Backend (Original): <https://github.com/BitBadges/bitbadgesbackend>
- Frontend (Original): <https://github.com/BitBadges/bitbadgesreactfrontend>
- TNB Proposal: <https://github.com/BitBadges/TNB-Proposal>
- Whitepaper: <https://github.com/BitBadges/Whitepaper>

RE: Blake 3: <https://hackernoon.com/2b-or-to-b3-that-is-the-question-blake-2b-vs-3-for-siacoin>

RE: TNB Proposal: <https://hackernoon.com/my-development-proposal-for-the-new-boston-payment-gateway-sidechain>

What is worth noting is that these links are for older versions of the code (*BitBadges 1.0*), and more work has been done in regards to the second version. Lots of the same database schema however, is also utilized in version 2.0 which is more decentralized in regards to technological implementation and architecture.

Although BitBadges haven't applied for any other grants (*as of time of submission*), BitBadges is actively seeking out partnerships as well as planning on competing in the Digital Learning XPrize (working on gamified e-learning badge solutions). The scope of the technology BitBadges can create and provide is very diverse, all of which is centered around its core ecosystem.

Development Roadmap

Outside of the sidechain integration, we also want to build another expansion on Substate. This is oriented towards more of the frontend and UX phase. Phase 2 as we will call it, will be one of the applications that can be built on top of that chain. The deliverance of phase 2 will be a badges based social network centered around timestamps and issued badges within the Substate/Polkadot sidechain.

Users will be able to load up the sidechain, connect to BitBadges core, and then issue badges to each other. Badges issued on top of the sidechain integration will appear within a nice UI. Other features such as profile, pictures, bios, etc. can all be implemented through badges. Likely, the user interface that can load all this data as well as potentially categorize/tag it, will be built w/ Node.js + React and may integrate the Bulma CSS framework.

Regarding the Milestones:

We have a specific set of tasks that will need to be completed as each phase is finished.

Milestone 1: Sidechain Development w/ Substrate

- Build out sidechain and work on deployment
- Different IT configurations in relation to development
- Add and integrate the EVM module (pallet)
- Create compatible chainspec file
- Create consensus layer and demo client
- Integrate contracts API and may add keyring package

Milestone 2: Sidechain Frontend

- Develop UI frontend
- Build w/ Node.js + React
- Consider utilizing the Bulma CSS framework
- Dockerize a sidechain + frontend package

Milestone 3: Social Network

- Integrate core BitBadges features
- Build off of Substrate sidechain
- Make sure network is fully responsive
- Consider making the social network modular
- Create user guide + dev guide (documentation)
- Consider making an Identity API guide
- Consider publishing plans for a secondary phase

The deliverables will likely be in the form of links to our code repos and a discussion (update) on our implementation. The scope of work in regards to each milestone, require a variety of different things. In regards to specification and the overview regarding the development roadmap, please turn to the overview section. For milestone #1, building out the sidechain will have alot of integrations to take into consideration. Some of the implementation will include not just integrating w/ BitBadges, but EVM compatibility, the Contracts API, the keyring package, and a well written chainspec file. In regards to milestone #2, we will have the frontend, and milestone #3 will include a usecase built on top of the sidechain (*the identity network*).

In regards to testing, the code for this sidechain will be fully open source along w/ its packages and instructions on running it. We can also integrate automated unit and integration testing or code testing w/ GitHub actions (*as needed*).

Overview

The total we are asking for regarding development for the first initial phases, is \$50k USD. This will cover costs related to deployment (hosting, potential gas fees, miscellaneous), marketing, and the whole design process. We know this may seem like much, but this is just to get the initial phases completed. Perhaps when this is expanded, further grant phases could follow to build integrations on top of this as a base. There are various considerations that go into this when looking at it through the scope of work perspective. By developing this project, we are not only both expanding our usecase and reaching more Substrate users, but we are opening up more dev opportunities on top of both platforms as well (*as mentioned*).

Milestone 1 ~ Sidechain Development w/ Substrate

- **Estimated duration:** 2 Month
- **FTE:** 1
- **Costs:** \$12500

Number	Deliverable	Specification
0a.	License	MIT, OPNL 2.0 and Apache-2
0b.	Documentation	Documentation on the sidechain and different modules
0c.	IT Configs	Different IT configs related to the sidechain and deployment/run-time
0d.	EVM Module	EVM module for Solidity compatibility and further developer onboarding
0e.	Chainspec	The required chainspec file and all its different configurations
1.	Contracts API	Integrate the contracts API module and contract pallet
2.	Demo Client	Build and package together a compiled demo client w/ consensus layer

Milestone 2 ~ Sidechain Frontend

- **Estimated Duration:** 2 Month
- **FTE:** 1
- **Costs:** \$10500

Number	Deliverable	Specification
0a.	License	MIT, OPNL 2.0 and Apache-2
0b.	Documentation	Create guide/documentation regarding the frontend
0b.	Dockerize	Dockerize the sidechain and frontend into one package
1.	Develop Frontend	Styling, UI, and UX (Develop the User Interface) using preferred tools

Milestone 3 ~ Social Network

- **Estimated Duration:** 3 Month
- **FTE:** 2
- **Costs:** \$27000

Number	Deliverable	Specification
0a.	License	MIT, OPNL 2.0 and Apache-2
0b.	Documentation	Create user and dev guide for network build
0c.	Testing Guide	Create a testing guide w/ runtime instructions
1.	Design	Work on the design, core features, and identity API integration
2.	Client Core	Make network modular, design network responsively, and compile client

We go over what needs to be done in the development roadmap. We are conservative in regards to the estimated duration for each milestone, but aim to finish way before that. If a delivery is finished prior, a status update can be provided early and we can receive the costs reimbursements and go on to the next milestone early. This is for sake of simplicity in regards to making sustainable progress.

Future Plans

Outside of updating dependencies and actively maintaining this project, our team has lots of plans. Many of these plans are centered around expansion in regards to our usecases and projects built around it. This will include cross-chain expansions such as TNB (TheNewBoston) and other crypto integrations as well as onboarding many developers to build on top of BitBadges and perhaps on top of its substrate sidechain as well. In regards to promotion, there are alot of things we want to do such as dev or crowdsourced innovation challenges, and focusing on constant improvement of our documentation or tutorials. BitBadges also plans on integrating a bug bounty program. We also want to focus on establishing various usecases and a large community of early adopters.

Additional Information

We heard about this grant in particular through an open grants website. We thought this matched well w/ what we are trying to do. Currently, we haven't applied for grants in the past. There is work *as mentioned* already done w/ BitBadges 2.0 and in regards to the substrate sidechain. We are also well aware of substrate's core functionality and how lots of the modules work. So far, there hasn't been anybody else who has contributed financially towards this project and we wish for careful consideration given we believe what we are developing has the potential to be massively scaled.