
FutureTech Industries, LLC

Entity: Name (or first name if applying as a natural person)

FutureTech Industries, LLC

Entity: E-mail address

ceo@fti.bi

Team member introductions

Our team comprises of experts who have built and scaled some incredible solutions for equally incredible organizations. Some examples include:

--

Alex has designed and operated planet-scale automation systems for Tinder, MIT, and Tsunami.gov.

Madison has architected, built, deployed, and maintained similarly planet-scale solutions for NOAA, MIT, and Tinder to name a few.

Austin has contributed substantially to Mozilla open source software, Tsunami.gov, NOAA, and a variety of others.

Introduce your Entity/Company

FutureTech Industries architects, designs, deploys, and manages planet-scale solutions for its clients, backed by modern yet stable technologies that have been battle-tested. We've been responsible for and successfully managed critical systems for a variety of organizations in the most difficult of situations, including air-gapped deployments and kernel-level implementations, with a level of intricacy and accuracy that can only be described as graceful. We're proud to maintain 99.999% uptime and 100% SLA in the majority of systems we maintain. We love Factom, our clients' solutions backed by Factom love Factom, and we're excited to continuously contribute to the Factom Protocol. In fact, we've already put out one series on Medium about running Factom at scale, and there are many more to come.

How many nodes do you envision to run on the Factom Community Testnet (or other testnets?)

2

What type of legal structure does your team use?

LLC

Will you operate as an incorporated company?

Yes

If yes to the above, where are you/will you incorporate? Are you already incorporated?

We're already incorporated in Delaware.

Are there any major shareholders (>10%) other than the members in your application?

No

Do you and/or your team members have previous experience running a business or managing large capital? If so, make a short writeup of your experience below.

Running businesses is what we do best! We're happy to have built our organization from the ground up over the last decade, moving from producing 1-2 applications a year to a handful a month on a consistent basis for major players in the software space. Additionally, we work comprehensively with our clients to bring their concept into reality, and ultimately assist them in turning their MVP and eventually final software into a stable product with millions of users. In terms of managing large capital, it's a regular occurrence with clients and we have rigid protocols in place to ensure auditability. We build software, but so much more importantly, we create and enable organizations to do better for everyone.

What has your team done to ensure a proactive approach to managing the financial aspects of your business?

We have strict protocols in place for anything involving money changing hands, including auditing/continuous review, analysis, reporting, and anything else we can do. Working with major and government organizations, that level of responsibility is implicit. To say we take our finances and the security involved around it seriously is to make an understatement.

Clarify your commitment to the Factom protocol? What have you brought to the table already, and what will you bring in the future?

We're very much committed to the Factom Protocol and furthering it. We enjoy contributing to the open source community, and in a natural conclusion, we authored a series (<https://articles.fti.bi/factom-ha-immutable-audit-logs-part-one>) on deploying highly available Factom on Kubernetes, garnering a substantial volume of traffic on all social channels. Part Two of that series, coming out shortly, will provide the first plug-and-play Factom deployment available for Kubernetes, which has been taking the majority market for container deployment as of late. We've also designed and deployed solutions for our clients that directly and indirectly integrate with Factom, and our team is actively contributing to a product coming soon that uses Factom at its core for auditability for enterprise clients. All in all, Factom is incredible and we want to give back in any way we can.

What is your/your entity's motivation for applying for hosting Factom Authority servers?

Simply put, we love Factom, giving back is very important to us, and we believe that for such an important platform, stability and security is critical. We've managed critical systems like this before, and not only is it a fun set of challenges, we truly believe we'd be a great fit on the ANO team. We have a team ready to go to monitor any nodes we operate 24/7 and, Since we're running solutions that require Factom, we have a vested interest in the stability and success of the network. We also love giving back to any community we can.

What vision do you/your entity have of the future of Factom?

We envision Factom kind of the way we've seen it work for our clients. Factom becomes a perfect quorumed protocol that allows developers to simply integrate, while introducing an unprecedented degree of immutability which entire organizations benefit from. When you need or just might benefit from an audit log, Factom's there, allowing an integration in a matter of minutes for minimal constant cost. We especially see a use for Factom in the operations and application logging spaces. We believe Factom can be used to protect organizations against malicious developers, and as evidence that software logs haven't been altered. In addition, we believe that Factom can be used with software as a service companies such as ourselves, to prove to clients that events we claim took place actually did. It is our goal to expand Factom's role in this space.

What will your efficiency be with only one node?	0,100000000000000001
What will your efficiency be with two nodes?	0,25

Node #1 Type	We deploy to Google Cloud's VMs, which have varying degrees of hardware and software isolation far above the norm while technically being a VPS.
<i>Node #1 Location (VPS: Provider, Region // Other: Country, City, Datacenter)</i>	northamerica-northeast1-c
<i>Node #1 CPU, Number of cores</i>	8+
<i>Node #1 CPU, type & clock-speed</i>	Skylake Xeon 2.0GHz
<i>Node #1 RAM, amount in GB</i>	28
<i>Node #1 RAM, scalable if < 24 GB</i>	Not applicable (>= 24 GB)
<i>Node #1: Storage, RAID type</i>	Regional PDs backed by RAFT
<i>Node #1: Storage, Disk type</i>	SSD
<i>Node #1: Storage, Free Size in GB for Factom</i>	200
<i>Node #1: Storage, Do you have a separate factom volume/disks?</i>	Separate disk(s), Separate volume(s)
<i>Node #1: Connection & uplink speed (not just your NIC speed)</i>	10Gbps down/16Gbps up
Node #2 Type	We deploy to Google Cloud's VMs, which have varying degrees of hardware and software isolation far above the norm while technically being a VPS.
<i>Node #2 Location (VPS: Provider, Region // Other: Country, City, Datacenter)</i>	asia-south1-c
<i>Node #2 CPU, Number of cores</i>	8+
<i>Node #2 CPU, type & clock-speed</i>	Skylake Xeon 2.0GHz
<i>Node #2 RAM, amount in GB</i>	28
<i>Node #2 RAM, scalable if < 24 GB</i>	Not applicable (>= 24 GB)
<i>Node #2: Storage, RAID type</i>	Regional PDs backed by RAFT
<i>Node #2: Storage, Disk type</i>	SSD
<i>Node #2: Storage, Free Size in GB for Factom</i>	200
<i>Node #2: Storage, Do you have a separate factom volume/disks?</i>	Separate disk(s), Separate volume(s)
<i>Node #2: Connection & uplink speed (not just your NIC speed)</i>	10Gbps down/16Gbps up

Add any other information relevant to server specifications and hosting, including planned availability of your maintenance team and how you would propose to handle an unscheduled restart.

As of today, we're planning to utilize bare nodes, but in the near future, we'd like to move to Kubernetes, which gives us a higher degree of granularity of control while handling a big chunk of maintenance for us. We're particularly excited for Google's 10Gbps down/16Gbps up per node as well as direct peering as the Factom network grows.

Which date did you join the Factom community testnet (approximate date is ok)?

7/1/2018

How does your team administer the nodes (more options possible)?

By more than 1 team member, By known and non constantly rotating directly hired personnel, Fallback by a specialized company

How many people in your team are able to operate the servers (including direct hired personnel, but excluding hired fallback companies)?

5+ depending on uptime requirements

How many years of combined experience does your team have on running production servers?

100

Could you elaborate on the production servers your team has managed (amounts, OS-types, purpose)?

We've managed blocks of nodes backed by terabytes upon terabytes of RAM and disk, including over thousands of servers at once for a single client and tens of thousands of servers over the years spanning every OS imaginable. We enjoy the fact that a sizable number of those nodes have been for critical, client-/public-facing services of the aforementioned orgs, and of those, they've upheld their SLAs/required uptime 100% of the time. Our developers have designed and built many autoscaling production-grade systems: simple web backends, custom workflow engines and their processors, geohashing solutions, clustered and sharded PostgreSQL and Redis configurations with failover, and a wide variety of other intricate solutions for our clients. We can handle every level of the stack and have the ability to monitor and maintain the most complex of setups.

Have you run follower nodes outside the qualified-node pool on the testnet? If so to what effect? Other contributions to the testnet?

We've run a small handful of testnet nodes for when we've designed and tested solutions for our clients. We've had much success, and our Kubernetes chart that's coming soon will provide one-click testnet node deployment.

Have you run any mainnet nodes? If yes, please elaborate why, and for how long

Like the testnet, we've run a handful of nodes for our clients. Unlike the testnet configuration, they're deployed for maximum availability, and have done exactly that. Our nodes have yet to go down in the time we've been running them; for more information check out our aforementioned article.

How are you going to make sure your nodes operate securely?

We do regular and continuous audits and monitored updates/maintenance, including pentesting, load testing, container analysis, binary verification, access logging, network load and policy reviews, as well as deploy a handful of network introspection tools in front of our traffic. If applicable and permitted, we also run honeypots, and for HTTP traffic, we have end-to-end introspectability.

How are you going to make sure you are able to respond quickly?

With our extended experience operating complex configurations, we have a suite of tools in-house to manage SLAs and escalation. Their policies become increasingly aggressive in terms of contact, and don't stop until incidents have been acknowledged and start being worked on. In summary, we have clients with ridiculously tight SLAs, and we have the supporting software to make sure we meet those SLAs every time.

Could you provide a picture on how you would see your ideal auth node infrastructure?

In the ideal world, we see our node running on Kubernetes with high availability running 3+ nodes in multiple zones and regions. With that configuration, we could manage dozens of Factom nodes with fairly constant complexity.

Free-text. Add any additional information deemed relevant.

We're incredibly excited for Factom, and we'd love to be ANOs. If we end up being an ANO, and even if not, we're prepared to dedicate a small team to furthering Factom in a variety of ways, the same team that's designed and deployed a variety of Factom-based solutions for clients, and focuses on R&D. No matter what,

we want to give back to the Factom community however we can, and whether or not we become an ANO we'll still be using Factom in our solutions.

[Add any application supporting files here](#)