Jessica Herron
Legislative Clerk
Subcommittee on Innovation, Data, and Commerce
House Committee on Energy and Commerce
2125 Rayburn House Office Building
Washington, DC 20515-6115

Re: Jessica Herron Response to Additional Questions for the Record

Dear Ms. Jessica Herron,

I would like to express my appreciation to the Subcommittee for inviting me to appear before it on June 7, 2023, titled, ""Building Blockchains: Exploring Web3 and Other Applications for Distributed Ledger Technologies."

Pursuant to the Rule of the Committee on Energy and Commerce, I am attaching my answers to additional questions for the record, in the required format.

Thank you very much. Please let me know if you have any further questions.

Best regards,

Hasshi Sudler

Hasshi Sudler

Chairman & CEO

Internet Think Tank, Inc.

Attachment – Additional Questions for the Record

The Honorable Russ Fulcher

1. **You noted the connection between blockchain and AI is very powerful, such as in the use of Smart Contracts. I really appreciated your testimony and the way you laid out smart contracts, and the ways that they are often misunderstood. Can you expand on the ways that smart contracts can be used? How is it that software developers are able to build applications on top of a blockchain?**

A smart contract is a software program that executes a blockchain transaction when a specific condition is satisfied. The conditions for triggering the contract are programmed into the smart contract software and can receive polled data into the smart contract to evaluate if it meets the smart contract's condition to trigger a transaction.

Smart contracts were originally inspired by financial trading contracts such as spot transactions, futures and options contracts. Spot transactions execute immediately on demand at the current "spot" rate. Futures contracts perform a payment transaction now for a transaction to be performed at a specific point in the future. For futures contracts, time becomes the condition that triggers the smart contract transaction. And options contracts executes when a specific condition is met that triggers a transaction, otherwise, the options contract expires after a specified period of time.

The capability for smart contracts to autonomously execute transactions based on the above contract types provides a powerful way for both financial and information transactions to be performed through software in a reliable and immutable manner.

Below are some examples of smart contracts in applications that sit on top of a blockchain (referred to as Distributed Applications or DAPPS).

- Example 1:
    - COVID contact tracing application

When two people come into close proximity, the smart contract detects that the two users' phones came within 6 feet of each other for a specified period of time (i.e.: 10 min). The proximity and time duration triggers the smart contract to record that the two anonymous individuals came in contact with each other for a significant period of time. If one of the individuals updates his/her status on the mobile app to COVID positive, the smart contract then triggers a transaction that sends a notification to the non-infected individual that an infection event may have occurred and to get a COVID test.

- Example 2:
    - Blockchain satellite example for collision avoidance

If two satellites have orbital paths that come close to one another (a conjunction) or are on a collision path, the smart contract on the blockchain would notify the other satellite of a

conjunction or collision event. Through negotiation between the two satellites, the smart contract would determine which satellite should adjust its orbit to avoid the collision, while the non-moving satellite would appropriately compensation the moving satellite for using fuel to perform the avoidance maneuver.

- Example 3:
    - Energy trading in an energy microgrid

Homes in an energy microgrid can be made up of units that each have solar and wind turbines to create clean energy. If one has an excess of energy (in kilowatt hours), other homes with energy needs can purchase that energy in a direct, peer-to-peer transaction over a blockchain. A smart contract would detect which homes have excess discretionary energy to sell and which homes have energy needs. Based on transaction conditions, the smart contract would trade a reasonable amount of energy from one home (so as to not deplete the home of needed reserves) and sell it to another home in need of energy.

These are a few examples of blockchain smart contracts and how they can play an important role in executing autonomous transactions between multiple parties.

DApps, or distributed applications, are integrated software programs that have a traditional web application using HTML, CSS and JavaScript exchange data with a backend blockchain platform. The data exchange is performed through an API called Web3.JS. Web3.JS moves information from the web application to the blockchain smart contracts for evaluation. Web3.JS also moves data from the blockchain smart contracts to the web application for further processing and presentation to the end user. Creating Web applications on a blockchain must take into consideration performance, security, and scalability across numerous IoT devices.