LACE: An Interactive Cluster of Tablet Computers and Kinetic Sculpture to Educate General Audiences on Distributed Blockchain Technologies

Eles Jones

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Science & Applications

Kirk Cameron, Chair
Godmar Back
Margaret Ellis

August 11, 2022
Blacksburg, Virginia

Keywords: LACE, Blockchain, Bitcoin Mining, Data Visualization, Distributed Computing, Visual Art, Education

LACE: An Interactive Cluster of Tablet Computers and Kinetic Sculpture to Educate General Audiences on Distributed Blockchain Technologies

Eles Jones

(ABSTRACT)

Blockchain technologies and cryptocurrency have made a significant impact on today's computing and financial sectors, and the use cases for blockchain applications are increasing day by day. However, there is little understanding of blockchain and cryptocurrencies amongst the general public. In this work, we present LACE, a kinetic sculpture and decentralized ledger created to educate audiences on the complexities of cryptocurrency creation through a visual form. We discuss the design and implementation of LACE as a modular system constructed of 10 kinetic units, each unit containing an array of Microsoft Surface tablets and one delta robot arm to perform touch based operations on each tablet with a modified stylus. Through this structure, we establish a distributed computing system in which each tablet represents blockchain nodes that maintain copies of the blockchain, mine for new blocks and process transactions through visual software interfaces. Additionally, we implement an interactive gaming module to help audiences understand the work of blockchain creation and the mining process. Finally, we evaluate the LACE project's effectiveness to teach audiences through a detailed questionnaire at the 2022 Accelerate Festival in Washington, DC. We found that 73% of visitors agreed they were able to learn something new from LACE and 82% enjoyed their interaction with LACE.

LACE: An Interactive Cluster of Tablet Computers and Kinetic Sculpture to Educate General Audiences on Distributed Blockchain Technologies

Eles Jones

(GENERAL AUDIENCE ABSTRACT)

Global technology, computing, economic and financial sectors are all increasingly influenced by the use of the relatively new technologies known as blockchain and cryptocurrency. A blockchain is a publicly distributed digital ledger that keeps track of transaction data securely through cryptography [47]. This technology is heavily associated with the global economy, following the introduction of the cryptocurrency Bitcoin in 2008. Cryptocurrency has often been compared to fiat currencies which are not backed by a commodity with intrinsic value like gold. Bitcoin is seen as a commodity due to its scarcity, with approximately 19 million bitcoins in existence and can be used as a monetary value to purchase goods and services. Studies have shown that a large segment of the general public has little to no understanding of these concepts, even those who have significant related investments [7]. To help expand the understanding of these topics to general audiences, we present LACE; a kinetic sculpture and digital ledger designed to educate audiences on the complexities of cryptocurrency creation through visual and interactive demonstration. LACE demonstrates the processes of blockchain technologies through physical robotic movement and interactive software visualizations. Consisting of a collection of 10 acrylic hexagon units stacked together like building blocks to mimic a virtual network, each unit interacts with an array of Microsoft Surface tablets through an operating robot arm and modified stylus. These tablets illustrate the work of a blockchain through various visualizations, demonstrating the work of nodes and

miners who operate and maintain a blockchain network. To help audiences understand the work of blockchain creation and the mining process, we implement an interactive gaming module where participants can act as a miner within a blockchain network and assist in the process of mining for new blocks, help maintain the blockchain and process transactions. We evaluate the LACE project's effectiveness to teach audiences through a detailed questionnaire at the 2022 Accelerate Festival in Washington, DC. We found that 76% of visitors had a better understanding of blockchain concepts following their interactions with LACE and 82% enjoyed their interaction with the sculpture overall.

Dedication

To my family who helped me every step of the way.

Acknowledgments

I would like to thank my advisor, Dr. Kirk Cameron, both for his guidance and support. I also would like to thank him for entrusting me with the LACE project and providing me with a great learning experience. I would like to thank Professor Margaret Ellis for being my introduction into the world of undergraduate research and for mentoring and supporting me throughout my entire time at Virginia Tech. I would like to thank Dr. Godmar Back for his technical expertise and guidance throughout my time working in both the SCAPE lab and for the CSG team. I would like to thank my colleagues and friends Skylar Liang and Gregory Bolet for all their hard work and continued support during our time working on this project. This accomplishment is as much yours as it is mine. I'd also like to thank Professor Samuel Blanchard and his artistic team for working alongside of our team and helping us create such a beautiful project. I'd also like to thank the rest of the SCAPE lab and the CSG team for their help and support during our work on this project. I can't wait to see this project grow over time. This material is based upon work supported in part by the National Science Foundation under Grant No. 1838271, 1565314, and 1939076.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The introduction of blockchain and cryptocurrency has had a significant impact on today's financial and computing sectors. The 2008 debut of blockchain and Bitcoin created a massive shift in the global economy and the trust between third-party businesses for financial transactions [53]. Cryptocurrency has often been compared to fiat currencies which are not backed by a commodity with intrinsic value like gold. Bitcoin is seen as a commodity due to its scarcity, with approximately 19 million bitcoins in existence and 91% have already been issued. Following the creation of Bitcoin, there are now more than 1500 different cryptocurrencies, which is more than eight times the amount of fiat currencies that currently exist. Blockchain has also influenced advancements in computing with blockchain technologies being introduced into multiple non-computing focused industries such as the government, energy, and medical fields [39, 45, 69].

While there is much buzz around blockchain and cryptocurrencies, there is little understanding or actual involvement by the average person. A 2021 study interviewing 750 bitcoin investors found that only 16.9% truly understood the value and potential of cryptocurrency. In the same study, 33.5% of investors had little to no knowledge as to what they were actually investing in [7]. While blockchain technology is being further introduced into everyday life, there is little to no knowledge or understanding of the concepts of blockchain to the average person. We believe that the abstract concepts of blockchain and particularly bitcoin should be accessible and digestible to a wider variety of audiences, allowing them to understand its functionality and how it may affect them.

In this work, we introduce LACE, a collaborative project between the Department of Computer Science at Virginia Tech and the Virginia Tech School of Visual Arts to demonstrate the abstract concepts of blockchain computing as an interactive experience. We created an interactive kinetic sculpture and functional, immutable, decentralized ledger to showcase how transactions are processed on the blockchain. We establish a network of 30 Microsoft Surface tablets to function as nodes within a blockchain. With these tablets, viewers are able to see and interact with fictional blockchain nodes within a distributed computing system that are tasked with maintaining a blockchain, mining for new blocks, and processing transactions.These nodes function within 10 kinetic modules which represent the computational work of bitcoin miners through their physical movement and interactions with the tablets. Together, these modules represent the work of an entire blockchain network and the computing capacity needed to mine blocks within the network. Through this demonstration, we aim to educate audiences on the importance of blockchain technology and bitcoin mining, while also demystifying the mining process through a visually engaging kinetic sculpture.

For our work, we determine three key goals for our kinetic sculpture; (1) to entertain and educate audiences on blockchain concepts; (2) personalize the importance of blockchain/cryptocurrency education and(3) inspire participants to learn more. The focus of this project is to unite the beauty of both art and computer science to create a fun and engaging tool to educate the public on the basics of blockchain and how it's used.

1.2 Research Contributions

In order to achieve the goals of this thesis, we make the following contributions.

- We propose, design, and implement an interactive software interface to visualize the process of bitcoin mining.
- We design and create a kinetic structure to model the work of a mining cluster.
- We design and build a distributed computing system to simulate the communication within a bitcoin mining pool.
- We demonstrate the LACE project to audiences to entertain and evaluate LACE's ability to educate viewers on the importance of blockchain technology.

My work on this project focused on the creation of the interactive software interfaces. Additionally, I developed a quantitative and qualitative questionnaire to evaluate the effect of LACE on audiences.

1.3 Organization

The remainder of this document is organized as follows. Chapter 2 reviews previous works related to blockchain, visualization tools for programming education, and integrating art with science. Chapter 3 discusses the methodology for designing LACE, and Chapter 4 presents the implementation method. Chapter 5 evaluates the use of LACE as an educational tool and finally, Chapter 6 provides closing thoughts and potential future work for the LACE project.

Chapter 2

Background and Related Works

In this section, we provide a literature review of existing research regarding the history and architectures of blockchain technologies. We then discuss the effect of visualization tools in teaching computer science concepts. Finally, we demonstrate the importance of integrating art and science in education and the use of kinetic art as a learning tool.

2.1 Blockchain and Bitcoin

2.1.1 History of Blockchain

To understand the significance of blockchain technology, we should first understand its history and rise in popularity. A blockchain [64] is a distributed digital ledger that stores records of data, often referred to as blocks, securely linked together in a chained structure through hash functions. This idea was originally developed in 1991 by Stuart Haber and Scott Stornetta as a method of digitally time-stamping documents securely to avoid tampering or backdating [32]. The technology was rarely used until a new adaptation was implemented in 2008 by an anonymous person/group called Satoshi Nakamoto. In their paper, "Bitcoin: A Peer-To-Peer Electronic Cash System", Nakamoto discussed the current dependency on financial institutions to serve as trusted third parties for electronic transactions. They argued that this type of trust based model is weak and vulnerable, and proposed a new system based on cryptographic security. This new system would allow for two parties to securely create transactions directly between each other without the need of a third party [53]. Using the blockchain application, Nakomoto was able to develop the first digital cryptocurrency, Bitcoin. Since then, the applications of blockchain technologies has expanded drastically, not only in the financial sector, but with applications being developed in multiple fields like the energy sector, healthcare, and in government [39, 45, 69] .

2.1.2 How Blockchain Technologies Work

A blockchain is a publicly distributed and decentralized digital ledger that keeps track of all transaction data securely through cryptography [47]. Each block within the chain holds transaction data, which is determined based on the type of blockchain network. Blocks also consist of a timestamp and a cryptographic hash calculated to represent it. Each block also contains a cryptographic hash of its previous block to verify its identity. If any changes are made to an individual block, a new hash would be generated, invalidating the verification of the block in the chain. This architecture design provides security for each transaction and renders the blockchain immutable.

A transaction can be seen as a data structure that represents the transfer of digital assets between peers on the blockchain network [49]. Because there is no central third party to verify transactions, each transaction must be verified through a consensus algorithm. A consensus algorithm is “a procedure through which all the peers of the blockchain reach a common agreement about the present state of the distributed ledger” [8]. Different algorithms are implemented depending on the network. Three of the most common consensus algorithms are Proof-of-Work (PoW), Proof-of-Stake (PoS) [30] and Practical Byzantine Fault Tolerance (PBFT) [19]. Once the verification process is completed, the data is added to the blockchain as a new block.

2.1.3 Bitcoin and Bitcoin Mining

Bitcoin is the first blockchain based, peer-to-peer cryptocurrency [53]. The aim behind Bitcoin was to establish a system for creating payments within a network of users, without the need for a centralized party. Just like any other currency, Bitcoins can be used to pay for goods and services, be transferred or exchanged, or used as an investment instrument [13]. These payments, also referred to as transactions, are done through the bitcoin network and must be validated through a Proof-Of-Work (PoW) system, before being updated into the blockchain [48]. This PoW verification process is done by what is referred to as bitcoin miners.

Bitcoin miners work within the bitcoin network to verify transactions by solving a computational challenge imposed by the PoW protocol [30]. The challenge for the miners is to generate a target hash that is less than or equivalent to the hash for the current transaction block. This process is also referred to as bitcoin mining. Once a miner has generated a "winning" hash, the hash is verified by others within the network, and the transaction block is added to Bitcoin's blockchain. The motivation behind this process is that it is very difficult for miners to guess the target but easy to verify whether it is correct once found. Miners are incentivized to complete this process by collecting the transaction fees and mining rewards once a transaction is verified [38]. However, the chances of successfully mining a block and receiving bitcoins is very low for the average person which has led to the creation of mining pools [68]. Mining pools are used to distribute the work of mining a block between multiple miners and eventually distribute profits. Within the pool, each miner will attempt mining the block and update other miners when a partial solution has been found. A partial solution is a solution that comes close to being full block, containing a portion of the work needed to find the necessary hash. An example was given by Ken Shirriff stating “if Bitcoin mining requires a hash starting with 15 zeros, the mining pool can ask for hashes starting with 10 zeroes, which is a million times easier. . . . Eventually one of these solutions will start with not just 10 zeros but 15 zeros, successfully mining the block and winning the reward for the pool” [61]. Each partial solution contributes towards the solving of the problem and results in the miners obtaining a portion of the final reward the block is successfully mined [61]. The amount of work contributed to a pool by a miner is known as a share. When a miner correctly solves a part of the problem and contributes to the future solution, this is known as an accepted share. Shares can also be rejected for multiple reasons; if the share is submitted too late, when a miner provides the wrong or invalid calculation, or if a particular share is submitted more than once [5].

Shortly after Nakamoto's paper discussing Bitcoin was published, an open source program was implemented and released for the bitcoin system. The first bitcoin network was created by Nakamoto in 2009 with the creation of the first bitcoins [59]. In 2009, the worth of an individual bitcoin was almost nothing. It was not until February of 2011 when the value of Bitcoin crossed the $1 threshold, and then $5 by the end of the year. From 2013 on, the price of bitcoin skyrocketed, receiving a gain of 6,600% and totaling around $1,100 by the end of the year [62]. Like any currency, the value of Bitcoin has fluctuated over time, but has still had a significant increase since its inception. At the time of writing this, the worth of an individual bitcoin is $21,354.04 [3].

2.1.4 Blockchain Applications

The use of blockchain technologies has expanded dramatically across different areas since its adaptation in the financial sector. In 2019, Jaoude and Saade [10] analyzed the state of blockchain research and found 151 different applications for blockchain technologies across 1500 academic works. Their research showed that the top applications of blockchain technology were Internet of Things (IoT), Energy, Healthcare, Finance and Government, respectively. Blockchain technologies have been used to address security and privacy challenges in IoT devices. Hammi et al. [35] argued the need for an efficient centralized authentication system for IoT devices in order to properly exchange data. They proposed ”bubbles of trust”, a decentralized system which relied on blockchain technology to ensure identification of devices within a network. Within the energy sector, some of the main applications of blockchain are facilitating energy trade, increasing energy grid security and assisting in the proliferation of green energy. Pipattaasomporn et al. [55]. discussed the potential integration of blockchain technologies into the green energy field through the work of solar photovoltaic's (PV) or "solar panels" to create a peer-to-peer exchange of excess electricity. In regard to healthcare, Zaabar et al. [74] proposed HealthBlock, a blockchain-based system for a decentralized management system to ensure secure healthcare management and information sharing between patients and medical staff.

2.1.5 Blockchain-based games in Education

As the importance of Blockchain grows, more efforts have been made to provide proper education on the technology. There has been a large increase in cryptocurrency and blockchain courses available in universities. According to a 2018 study, 42% of the worlds top 50 universities offered at least one course on cryptocurrency or blockchain [21]. Still, the underlying concepts of blockchain can be difficult to digest for beginners , which has led researchers to demonstrate the functions of blockchain through games. Research has shown that the "gamification" [23, 31, 52] of educational curriculum helps develop students' knowledge and skills through collaboration, while also influencing their commitment and motivation to the content [42].

Dettling and Schneider [24] developed a software tool, Bloxxgame, to simulate the workings of a public blockchain, similar to Bitcoin. Bloxxgame stems from a whiteboard-based blockchain game but was adapted into a web-based simulation to teach different scenarios. In the game, users are able to act as a node and create transactions, blocks, and learn about the consensus algorithm. The tool also allows users to experience more abstract mechanisms of blockchain such as hashing, signing, and block building. Bloxxgame was evaluated through an information systems course with students of varying backgrounds and technical skills. Overall, students showed more engagement with content following the demonstration of Bloxxgame and showed interest in learning more advanced topics.

Oktian et al. [54] developed a blockchain-based system to illustrate the complexity of blockchain mining and how coins can be received in the process. The game allows users to store and update user information, and also mimic the transactions made during the mining process. Users are given a reward score when registered with the game and their progress is updated when they have generated a valid block. The game also allows for different difficulty levels to further demonstrate the intricacy of the mining process. Users are also given a performance report at the end of each game to show they performed and what improvements can be made for mining in the future.

2.1.6 Criticisms of Blockchain and Bitcoin

While these technologies and currencies have been praised for their advancements in computing technology, there are many issues that have been addressed since their implementation. The PoW protocol has been criticised for its lack of security and poor environmental impact. In terms of security, PoW blockchains are only truly deemed secure if used in larger networks. In order to manipulate a blockchain network, a user must gain access to a majority of a network, specifically 51% or more. This is additionally known as a 51% attack [28]. Owning this amount of the network would allow for users to manipulate it as wanted, which is much easier to do with smaller networks as there is less to gain control of. While there are new methods being implemented to detect and defend against these types of attacks, this issue has brought concern to many about the true security of these types of networks and encouraging the transition to a different type of consensus algorithm.

The environmental impact of the bitcoin mining process has also been a major concern in the past few years. The task of the PoW protocol is difficult to complete and therefore takes a large amount of computing power in order to successfully complete. This has influenced miners to invest more resources into the mining process. Since Bitcoin is a limited commodity, the more and more people mine for bitcoins, the more competitive the mining process becomes. As a result, people are increasing the amount of power and effort put into the process, making it much more energy intensive. According to a study done by the University of Cambridge, the annual electricity consumption of Bitcoin averages at 0.6% of the global energy usage, which is more than the country of Argentina [4]. Suggestions have been made to combat this issue by changing the consensus algorithm used by bitcoin network from the PoW protocol to algorithms like Proof-of-Stake or Proof-of-Burn, which have been studied and shown to have less environmental impact. While there is no say as to whether this change may occur, it is important to understand the consequences of the bitcoin mining process and how it may affect the world in the long run.

2.2 Visualization Tools For Education

The use of visualizations tools in educational settings has been demonstrated to be an effective method in assisting teaching complex topics.

Fouh et al. [25] analyzed various algorithm visualization tools used in computer science courses and evaluated their effectiveness. They aimed to determine if existing algorithm visualizations were effective in teaching computer science concepts. It was shown that the use of these visualization tools in educational settings had a direct correlation to an increase in student performance. An example of this was shown in Alice [63], a 3D visual programming environment implemented for introductory programmers which allowed users to visualize objects in a 3D world. Moskal et al. [50] evaluated the effectiveness of Alice for underperforming or "at risk" programming students to determine whether the tool could improve retention of the curriculum and their overall performance. The results of the study showed that "at risk" students who used Alice during their time in the course performed better with a 2.98 overall GPA average, while "at risk" students *without* Alice performed with an average of 1.18 GPA. Students who used Alice also had a much higher retention rate, 88%, than those who did not, 15%. These findings led the researchers to theorize the introduction of 'hypertextbooks' would provide a productive addition to the curriculum of introductory programmers by integrating algorithmic visual tools and traditional textbooks [25].

As a result of the previous work, Fouh and additional researchers designed and implemented an interactive eTextbook, OpenDSA [27], for undergraduate computer science students. The focus of OpenDSA is to support students' comprehension of algorithms and help analyze their processes and effects through visualizations, code snippets, text, graphics and various types of exercises. The content of the OpenDSA modules covers intermediate level data structures and algorithms curriculum, varying from sorting algorithms, hash functions and visualizing the process of binary search trees. OpenDSA was evaluated through a survey given to students enrolled in a data structures and algorithms course and were asked to extend their opinions towards using OpenDSA in comparison to the traditional textbook curriculum. Students showed a strong preference for using OpenDSA in comparison to the traditional textbook and lecture [34]. Students who have used OpenDSA also reported spending less time preparing for their midterm test than those who had not used the tool. OpenDSA was also helpful to instructors as they now had more time to focus on activities and programming project design rather than traditional lectures.

In a 2012 study, Ali and Rosminah [58] found that 72% of students within a Fundamental Programming course agreed that the use of visualization tools were helpful aids for beginning programmers. The aim of this research was to understand the difficulties faced by students when learning the basics of programming. Students were provided a questionnaire regarding their programming experience and their overall experience during their time in the introductory programming courses. The results of the survey showed that students struggled most with understanding abstract topics such as visualizing variable position in computer memory and designing programs efficiently [58]. The subjects argued these issues occurred due to a lack of examples and visual aids to help students conceptualize more abstract topics of programming.

Hahn et al. [33] designed a visualization tool, ThreadCity, to assist in the exploration of multi-threaded software systems. The purpose of this tool was to assist in the visualization of systems with concurrent runtime behavior for software maintenance and development. The tool uses a hierarchical aggregation technique to structure the visualization of system components. ThreadCity was evaluated with a small group of users instructed to complete three tasks for two parallel processing datasets. The tool was compared against ViewFusion, another tool used for runtime visualizations [67]. Participants were asked to (1) categorize data-parallel or task parallel behavior, (2) determine the main tasks of a certain thread based on its involved components, and (3) analyze which component in each thread was the most time consuming. Participants found ThreadCity to be the ideal tool to complete tasks 1 and 2 because of the aggregation implemented in the tool, which helped get a faster overview of the systems behavior. Participants found task 3 difficult for both visualization tools, leading the researchers to discuss further improvements for this work.

In 2017, Bart et al. developed CORGIS [14], a web-based tool used to visualize and explore over 40 real-time datasets without programming. Introducing data science and computer science curriculum to those without programming experience can be a daunting task and can often deter non-CS students to participate in these type of courses. CORGIS was implemented to assist non-CS majors in data science lessons and improve their understanding without needing to depend on programming skills. Within the course, students used CORGIS to create visualizations of data trends, assist in understanding programming topics and solve open-ended data science problems using computational techniques. CORGIS was evaluated through a survey using a class of 50 students in a Computational Thinking course, in which the students had no programming experience. The overall response was positive, stating that CORGIS helped students feel successful in the material. Students also indicated that working with data that directly related to their discipline was helpful for their career goals.

Naser [11] implemented an algorithm visualization tool to demonstrate the traversal of Artificial Intelligence (AI) searching algorithms. The tool assists students in visualizing the dynamic behavior of AI algorithms by stepping through each part of algorithms like depth first search, breadth first search and A* search. Ninety undergraduate students were randomly assigned to 3 groups with one group taught conventionally and the additional two taught using different versions of the tool. The study showed a strong positive relationship between students who used the visualization tool and their performance in their course.

Qasem [56] discussed the difficulties computer science students often have with hardware-focused courses. The author argued that multiple factors contribute to this outcome, including a lack of active learning in most system courses and a lack of deeper understanding of the underlying systems of hardware. To address this issue, Qasem developed a pedagogical tool, YODA, that produces Jupyter notebooks used to assist students in learning systems concepts through guided interaction and observation. Through YODA, users can simulate the actions of different hardware systems such as the cache, VM and TLB, and CPU. YODA also includes demos, self paced tutorials for users and assessment material for each simulations learning outcome. The tool was evaluated over four semesters amongst undergraduate students in a computer architecture course. YODA was used in only one section which was labeled the ”intervention” section with an additional section taught by the same instructor without YODA, used as the ”control” section. The results showed that higher percentage of intervention students received A’s and B’s in the course and had an overall higher passing rate than the control group for the last three semesters. The student perception of YODA was also evaluated through an end-of-semester survey and received high rankings amongst students for all semesters, with slight increases as improvements were made to YODA.

Yang et al. [72] designed a web-based interactive visualization tool, JavalinaCode, to assist in teaching object-oriented programming in introductory courses. Object-oriented (OO) programming is often taught in introductory computer science courses but can be difficult for beginners to comprehend. JavalinaCode is implemented to help students in OO principles and concepts like UML diagrams and program execution through static and dynamic visuals. Mustafa [51] developed an interactive simulator to provide support to both instructors and students of computer architecture and operating system courses. The simulator provides a high-level view of the process of three main components, the compiler, CPU and OS. The simulations mimic the actions of the true components by generating code from the compiler to be run by the CPU through the OS. The simulation also allows users to optimize code,create and run processes, view cache and pipeline simulations and analyze memory.

2.3 Art In Education

2.3.1 Integrating Science And Art

The field of science and engineering has grown vividly for decades, with more disciplines becoming intertwined with each other. As a result of this change, the term STEM was created in 2001 by Judith A. Ramaley, a previous director of the U.S. National Science Foundation's Education and Human Resources division, to describe the collaboration of Science, Technology, Engineering and Mathematics in education [43]. The purpose of this was to help provide a context for the collaboration of disciplines in order to correctly represent the approach to real-world problems and opportunities students may face [70]. STEM initiatives were promoted in educational programs to create a project-focused learning that incorporated all of the subjects. However, there has been much debate as to whether the arts should be also included with these disciplines, introducing a new acronym STEAM, Science, Technology, Engineering, Art and Mathematics [29, 36, 46]. Watson argued that "STEM disciplines require artistic thinking [70]" as engineering requires not only logic but creativity to derive new solutions to complex issues. Connor et al. [22] stated that "creativity and innovation cannot be treated separately from STEM and 'arts' should be an integrating part of the puzzle that combines creativity and innovation into a unified whole.".

Braund and Reiss [15] discuss the relationship between science education and the arts, and how allowing for artistic creativity may make science more complete. They suggest the arts can improve the teaching and learning of science on various levels such as the function of the human brain. Research has shown that the arts can contribute heavily to science education by stimulating the brain in ways that may not be done in traditional science activities [40]. They also argued that a benefit of art is its visualization in comparison to the standard narrative alternatives in education. The ability to visualize scientific concepts can make the curriculum more accessible and allow for more active involvement in the learning process.

Yilmaz [73] discussed the relationship between art and engineering disciplines and the importance of their collaboration. They argued that art often benefits from various engineering disciplines and fields such as mathematics and mechanics, ultimately reconstructing artists into engineers. This dates back to 15^{th} century art, with artists such as Jan van Eyck benefiting from chemical engineering to modify various pigments and oils to formulate oil paints [73]. With the advancement of technology and the expansion of engineering and art, new artistic concepts have emerged as a result of engineering.

Prior to the 19^{th} century, the term kinetic referred to topics relating to motion, often used in the context of physics or chemistry. However, in the 20^{th} century, movement began to be heavily incorporated into art resulting in the creation of the kinetic art. The first kinetic sculpture was demonstrated by Naum Gabo in 1920 titled “Standing Wave”, incorporating his understanding of dynamics principles and engineering with art design to create a movable mechanical structure forming the illusion of a twisting wave [65]. The structure was the pioneer for kinetic art. With the vast amount of technical improvements of engineering and technology, the realm of kinetic art has become more expansive than ever, becoming one of the major gateways for integrating art and engineering.

2.3.2 Kinetic Art For Education

Kinetic sculptures and artwork have been used as great methods of engaging and educating through creativity and engineering. Brunvard and Stout [16, 17] developed a collaborative course to pair together computer science and engineering design into creative studio projects. The focus of the course was to integrate embedded system design into art design to create kinetic art sculptures. The curriculum of the course focused both on engineering topics like programming and electronics fundamentals, while also incorporating art design topics such as art history and material studies. The class consisted of students from multiple disciplines, including art, computer science, electrical engineering, etc. Students in the course were split into teams consisting of both engineers and artists and were tasked in creating a kinetic sculpture. An art gallery of the students' work is then shown at the end of the semester. Student evaluations taken for the course were very positive and resulted in scores above their department's average. Students commented that the interaction with students of different disciplines was a positive experience and helped improve their overall experience in the course.

Chung [20] developed a STEM program, the Global Robotics Arts Festival (GRAF), to allow pre-college students to integrate visual arts and engineering. Students were required to create robots that fell into the visual arts or performing arts category. All robots were required to have a computational component that implemented sensors that were programmed by the students. An example of a visual art robot created by a team displayed kinetic art patterns by using 16 servo motors and distance sensors [20]. These patterns were programmed to change when a spectator was detected viewing the structure. Following the festival, participants were given a survey to evaluate the efficacy of the GRAF program on engaging their interest in STEM. The response was overwhelmingly positive with students stating the program helped motivate them to consider pursuing STEM related classes and careers.

Jee and Hong [41] studied the effects of using kinetic art for teaching STEAM curriculum in elementary aged students. The focus of the curriculum was science and art, focusing on Mobility, Light and Color, Rotation, Energy and Exhibition. 10 classes participated in the study, 5 being the experimental group and the other the control. The results of this study found that the experimental group significantly improved their overall literacy in STEAM concepts compared to the control. Students in the experimental group also acknowledge that the use of kinetic art was helpful in understanding the scientific concepts and engaged their interest in science further.

2.4 Conclusion

In this chapter, we discussed the significance of blockchain technologies and their applications, bitcoin, and the influence of blockchain on education. Additionally, we established that visualization programs are effective tools in teaching complex computer science concepts to novice programmers. Finally, we discussed the use of kinetic art as a tool for integrating science and art in education.

Chapter 3

Design

The purpose of the LACE project is to entertain and educate audiences on the processes of bitcoin mining and blockchain in a simplified manner. In order to do this, we propose a kinetic sculpture that would visualize the work of a blockchain network through an interactive guessing game. In this chapter, we describe the design decisions made for the implementation of LACE. We briefly discuss the purpose of the LACE project, the blueprint behind the LACE sculpture, and the design of the LACE game and its interfaces.

3.1 Design of LACE

The purpose of LACE is to take the abstract concepts of Blockchain computing and visualize them in an interactive experience. We represent the work of blockchain nodes through a distributed computing system of 30 Microsoft Surface tablets, each representing an individual node. Each tablet will be tasked with maintaining copies of the blockchain, mining for new blocks and assisting in processing transactions. This will be represented through a visual software interface on each tablet, and each kinetic unit of the LACE structure will use physical movement to represent the work done in blockchain creation.

LACE is the successor of a previous kinetic art project, SeeMore [44], which was used to visualize the works of parallel computing. SeeMore was originally named after Seymour Cray, an American electrical engineer who designed a series of supercomputers that were the fastest in the world for decades, earning his title as the "father of supercomputing". SeeMore also coined its name through its double entendre of having viewers "see more" of parallel computing. Our team wanted to follow this structure with LACE, naming her after the first computer programmer Ada Lovelace, but also introducing lace imagery into her design. Looking at how lace binds together led us to design the individual modules of LACE into a hexagon shape. The modules can all be stacked together in a honeycomb like structure. This hive-like design also helps visualize the connection of the distributed computing system and how each module is networked together.

We also wanted to try to intersect the very beginnings of computer science and modern computer science into the design of LACE's structure. Ada Lovelace's work on the Analytical Engine alongside Charles Babbage was revolutionary to the field of Computer Science, proposing the first steam-powered programmable computer, shown in Figure 3.1. This machine dates back to the 1800's where machinery was much more physical and its movement and work could be seen easily by the naked eye. This influenced our decision to represent the work of bitcoin miners in a more physical way through moving kinetic units. We also chose to have many of the materials of LACE represent this time which Ada Lovelace worked, mimicking the design of the Babbage machine with anodized gold and green edged glass. Overall, a very sleek design with components that date back to the Victorian era.

Using these ideas, we were able to create a rough design of the overall structure of LACE. Figure 3.2 shows a concept development sketch of the LACE structure to scale. Figure 3.4 shows a close up look of the design for the inside of each kinetic unit.

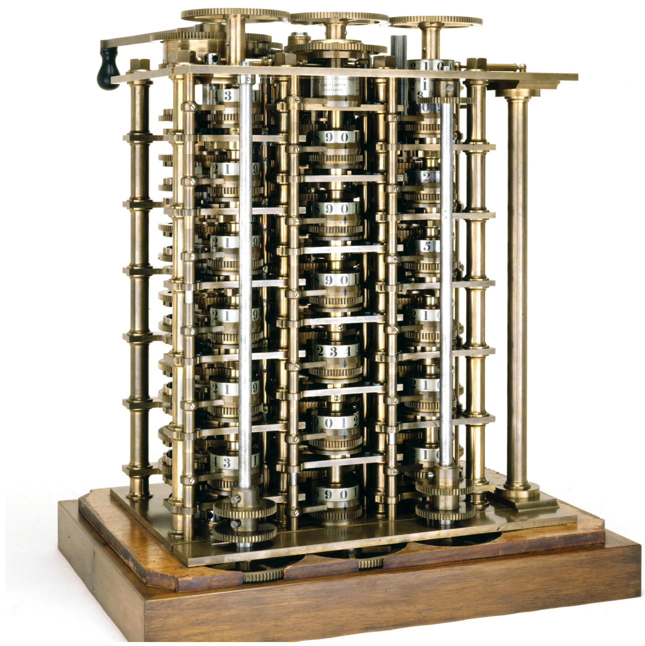


Figure 3.1: Analytical Engine - Created by Ada Lovelace and Charles Babbage in 1842

Figure 3.3: Digital rendering of the cells for the LACE sculpture.

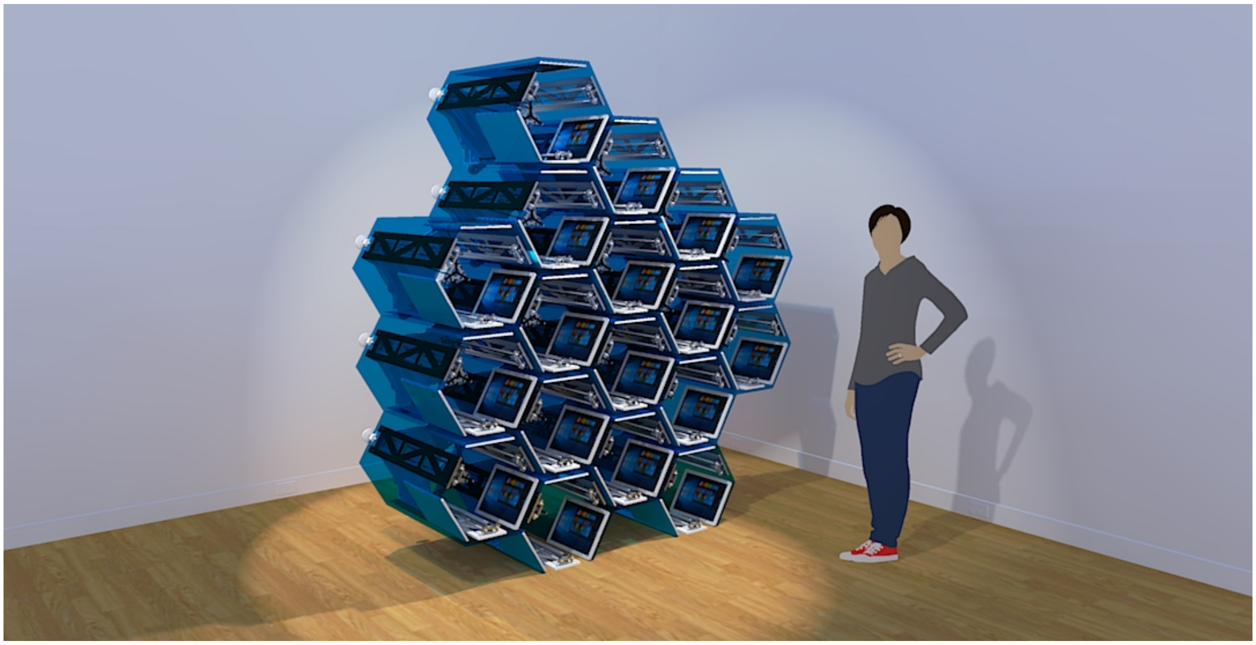


Figure 3.2: Digital rendering of the scale of the LACE sculpture.

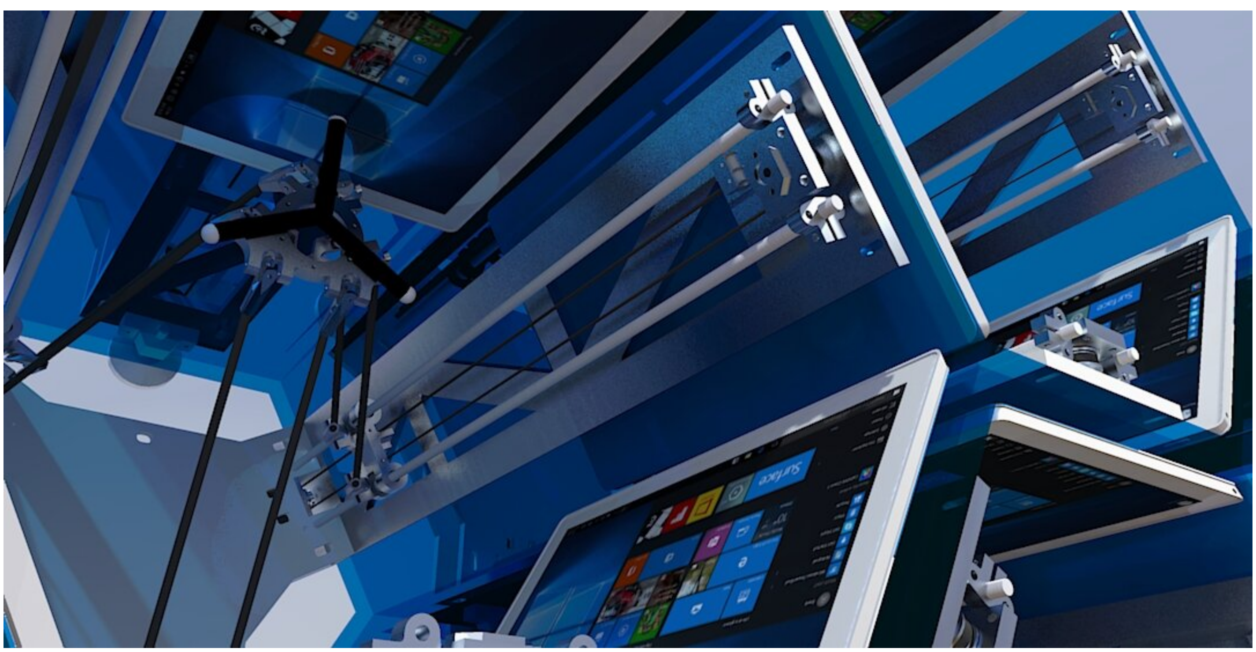


Figure 3.4: Digital rendering of modules of LACE.

3.2 Game Requirements and Specifications

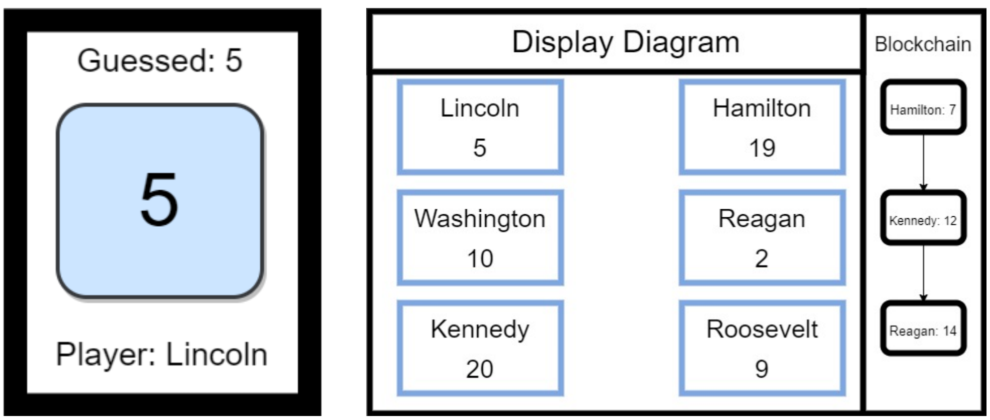
In order to explain the concept of bitcoin mining and blockchain to an audience, we adapted the work of bitcoin miners into a simple guessing game. We considered the following criteria when designing the LACE game.

- LACE must have an interactive game that simulates the behavior of bitcoin mining through a number guessing game.
- The game must be playable through a network of 30 Microsoft Surface Tablets within the LACE structure.
- LACE must have an interactive component to allow audiences to participate in the game.

3.3 Game Concept Idea

To simplify the mining process to audiences, we propose a simple number guessing game to demonstrate the work of bitcoin miners. The work of a bitcoin miner can essentially be boiled down to guessing a hash value that is less than or equal to the target hash. The premise of our game is that each player will represent a different bitcoin miner who is attempting to guess the particular hash for a block. We simulate this work by instead having each player guess a random number through a defined range, looking for a target value, as if they were doing work to obtain or "guess" the winning hash. The players will continue guessing numbers within this range until an individual player has guessed the correct target value. Once this value is found by a player, the remaining players come to a "consensus" stage and validate the work of the player by determining if the number guessed is correct, similar to verifying the hash. Once this player is verified, their name and winning number is added to our game's blockchain for the remainder of the game. To stay in line with our theme of historical figures, we also decided to name the players in our game after historical figures in American history. We chose 50 influential men and women to represent players in our game. Each figure has been represented in one of the Smithsonian museums in Washington, D.C, where we plan to debut the LACE sculpture.

3.4 Game Interface Design



(a) Design for Client Interface (b) Design for Display Interface

Figure 3.5: Designs for Visual Interface

At the start of this project, we envisioned two interfaces that would display the work being done by the clients. Our first interface represents the view of each client, which is shown in Figure 3.5a. We wanted to simulate the guessing of a hash, or for our purposes a number, through clicking a button that would display the current number being guessed. The screen will display the name of the client and what number they had previously guessed. The button will display the clients next guess. When the button is pressed, this will simulate a guess being completed by a client. If the number that had been chosen was wrong, the screen will display a new number to guess and continue until a client guesses the correct number. When a winner is found, the player's screen will alert them that they have won and other players will be alerted that they have lost the game.

We also needed a way to display the work of all the clients simultaneously. This interface will be shown on a larger screen, separate from the client interfaces, presenting the work of all clients at once. In our Display page design, shown in Figure 3.5b, we visualize the work of each client by displaying their name and the number they have guessed. When a player has "guessed" a number or tapped their screen, an animation is shown on the display for the player, which flashes said number. Once a player has guessed the correct number, their icon on the page will display green while the rest will turn red. At the start of a new game, each player's icon will refresh to their new number to guess and continue. We also wished to visualize the blockchain ledger for our game to display the previous winners and how they would be connected in a real blockchain, with previous transactions being attached to the most recent. For our blockchain, we decided to display the name of the player who had won the round, and the number they guessed as the winning number.

3.5 Audience Interaction

To allow for audiences to directly engage with the LACE game, we proposed a strategy to use a customized TwitterBot as an intermediate between LACE and its users. A twitter bot is a type of bot software that controls a Twitter account via the Twitter API [9]. This bot can also communicate with other twitter users and send out automated messages. Using a bot alongside LACE would allow human players to tweet at a specified hashtag, e.x. (#LACE), and play against the 30 clients within the LACE structure. With each tweet, a human player can guess a single number within the specified range. The TwitterBot will send automated messages back to each player whether or not their number is correct. If the correct number is found, a message will be sent to all current players of the winner and their winning number. Once a game is finished, users are asked if they would like to play again and the process will repeat. Creating this automated system will allow for more active participation and a more hands-on experience with LACE.

Chapter 4

Implementation

This section will discuss the methods we chose to implement the LACE project and the reasoning behind them. We will also discuss issues we faced and deviations made from the original design.

4.1 Creating LACE

4.1.1 Module Prototypes

To perfect the overall design of the kinetic unit, our researchers and artistic team developed and redesigned multiple iterations of a singular module. In this section, we will discuss three module prototypes, their issues and the decisions which led to our final 10 module structure.

Module Design

The design for the overall structure would contain a modular system of interlocking hexagonal cells. Each module would contain 3 Microsoft Surface Tablets, acting as nodes on the network, and one delta robot arm to perform touch based operations to each tablet with a modified stylus platform. Figure 4.1 shows an early rendition of the modular system design and the theoretical placement of the three tablets.

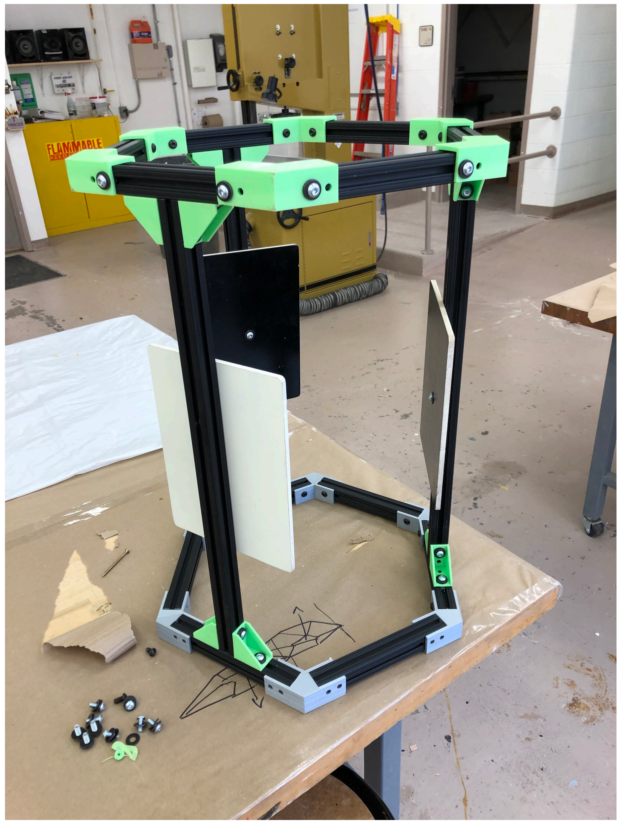


Figure 4.1: Image of our starting module design.

Mark I

The basis of this design was formed around the idea of having some type of physical movement within each of the modules to demonstrate the work of the computing system. Our artistic team theorized the use of delta-style robot arms within each module in order to visualize this type of work. Delta robots are a type of parallel robot with a base connected to jointed parallelograms, which are often seen and used in more industrial settings for automation packing processes. These types of robots are also often seen in modern 3D printers due to their quick and precise movement and capabilities. Using these types of robots would allow for physical movement within each unit and have the resolution to interact with the tablets within each unit.

We began by prototyping basic 3D printer models to understand the movement we required for the LACE structure. The Mark I prototype used the Anycubic Linear Plus Kossel V2 3D Printer model as our main source for movement with additional parts customized by our artistic team. Within each kinetic unit, we added three additional towers to place three Microsoft Surface tablets. We also customized the base of the delta-robot arm into a triangular platform that attaches three separate styluses to tap on the corresponding tablet shown in Figure 4.3. These customized pieces were all 3D printed by our team using the Ultimaker S5 3D printer.

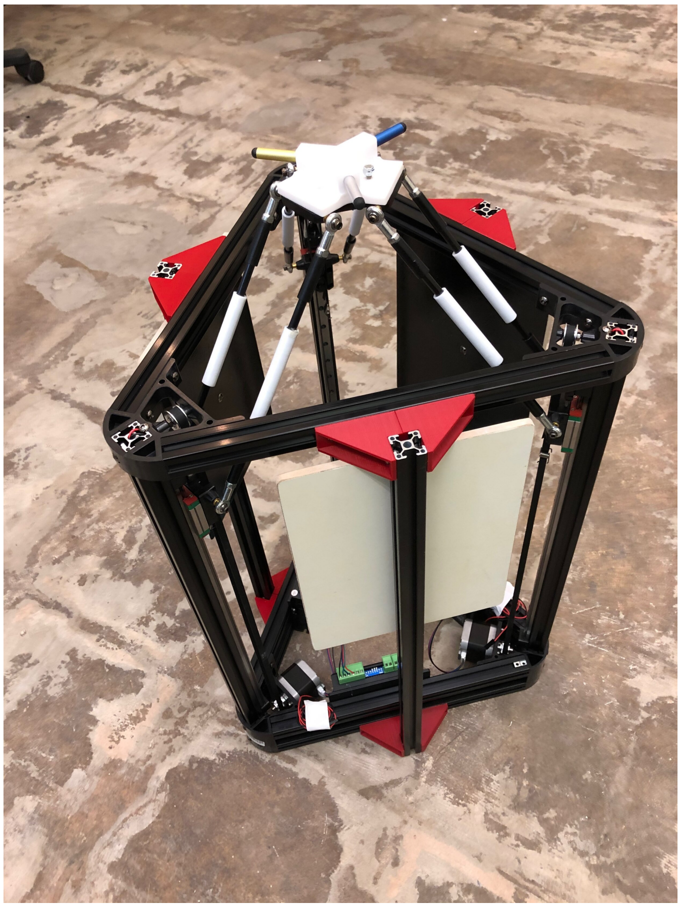


Figure 4.2: Mark I prototype

The control mechanism of the kinetic unit consists of three separate stepper motors mounted

Figure 4.3: Custom Base for Mark I Prototype

to the frame of the module, which attach to timing belts to operate a linear carriage vertically along a linear rail or tower, shown in Figure 4.4a. Each stepper motor is controlled by an individual stepper motor driver, which receives input from our Arduino control unit. We use the P-Series Nema 12 Stepper Motor and Closed Loop Stepper Drive for Nema for this prototype which provided the necessary functionality for a reasonable price.

We use the Arduino MEGA 2560 model as the control unit for motion. Each Arduino is secured to the frame of the unit and connects to the stepper motor drivers. We used the 54 digital I/O pins and 16 Analog input pins to control the stepper motor(rephrase this). Additionally, we implement the Arduino AccelStepper library, which provides an object-oriented interface for 2, 3 or 4 pin stepper motors and motor drivers to support the acceleration and deceleration of the motors. We eventually transitioned to the Arduino UNO WiFi Rev2 model for simplicity, which is further discussed in section 4.3.1.

In order to program the movement of the module, we look to the work of our fellow researchers, Greg Bolet and Skylar Liang, who implemented a kinematics library based on the work from "The Delta Parallel Robot: Kinematics Solutions" which provides geometric equations to simulate the movement of the delta robot [71]. We focus on the implementation of the Prismatic-Input Delta model as it best reflects our model. For the nature of

(a) Stepper Motor (bottom) and Linear Carriage (top)

(b) Limit Switch

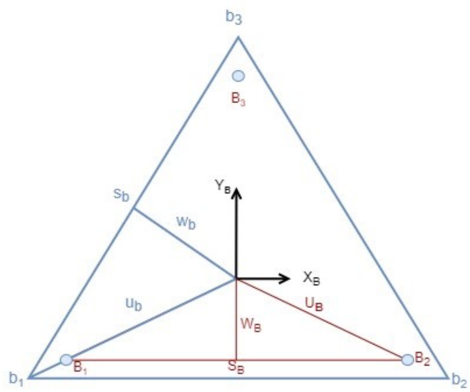
(c) Three Stepper Drivers and Arduino MEGA 2560

Figure 4.4: Movement Components of Kinetic Unit

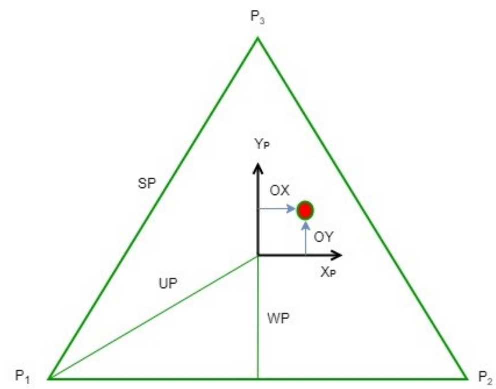
our design, we invert the kinematics of this system such that the base is at the bottom and the movement of each tower's carriage is upwards. Through this library, we calibrate the movement of each motor to a distinct set of locations, ultimately moving our main platform in a triangular motion we call "Tri-Taps" in order to touch each of the Microsoft Surface tablets. Once the motion is completed, the motors are instructed to return each carriage to their home location, notated through limit switches on the end of each run.

To establish the kinematics for a module, a set of precise measurements were needed from each module to determine how far to move the carriage of each tower. The prismatic-input Delta Robot consists of three identical prismatic-universal-universal (PUU) legs in parallel between the top fixed base and the bottom moving end-effector platform. Figures 4.5 and 4.6 show a geometric diagram of the Delta Robot measurements and Figure 4.7 and 4.8 show examples of the measurements in the context of our module. Using the measurements provided, the kinematics library is then used to calculate the exact position needed to move to and instruct each motor to do so. A guide for the geometric measurements details is shown in Appendix A.1.

We also added limit switches to the end of each tower to be able to detect when a motor had reached the bottom position of each run. These switches were used to establish a "home" position for each motor, and were used to indicate when a module's movement had ended. This device can be seen in 4.4b.



(a) Prismatic-Input Delta Robot Fixed Base Details, adapted from [71].



(b) Prismatic-Input Delta Robot Moving Platform Details, adapted from [71].

Figure 4.5: Geometric diagrams for Delta Robot models.

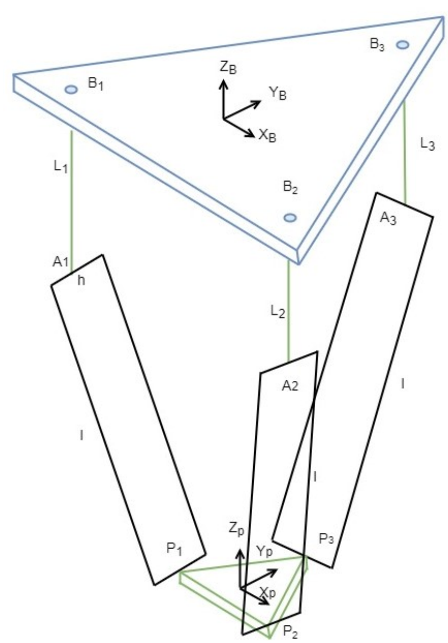
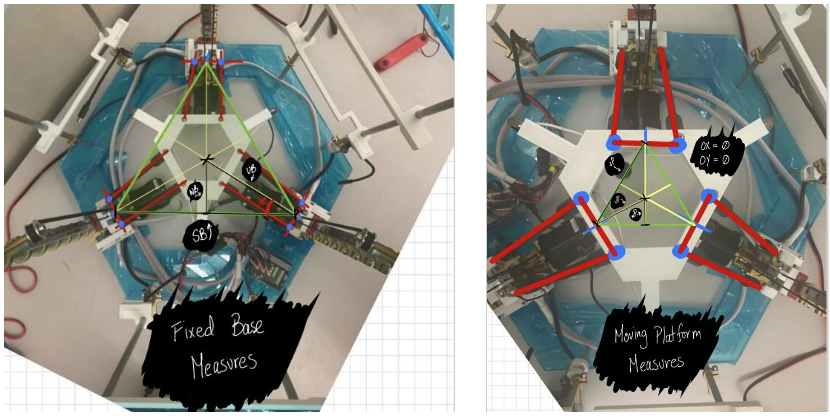


Figure 4.6: Delta Robot Kinematic Diagram - Prismatic Inputs, adapted from [71].



(a) Reference for Fixed Based measurements. (b) Reference for Fixed Platform measurements.

Figure 4.7: References for prototype measurements from final prototype.

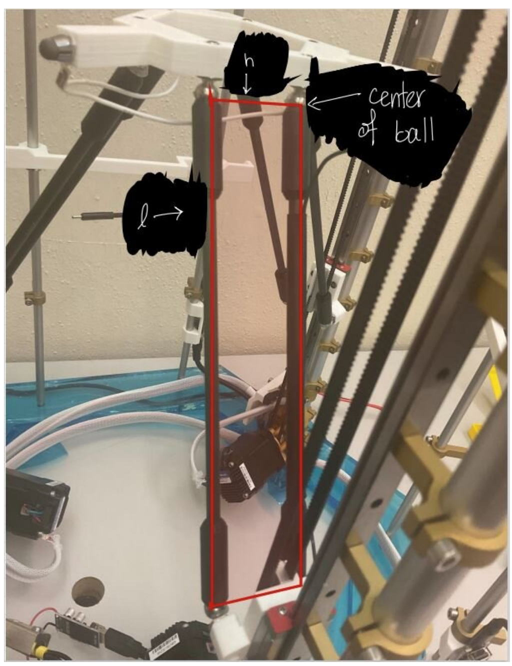


Figure 4.8: Reference for Leg measurements based on the final prototype.

Mark II

Once we understood the functionality of the delta robot arm movement, we then moved on to perfecting the design of the module and its functionality with the Microsoft Surface

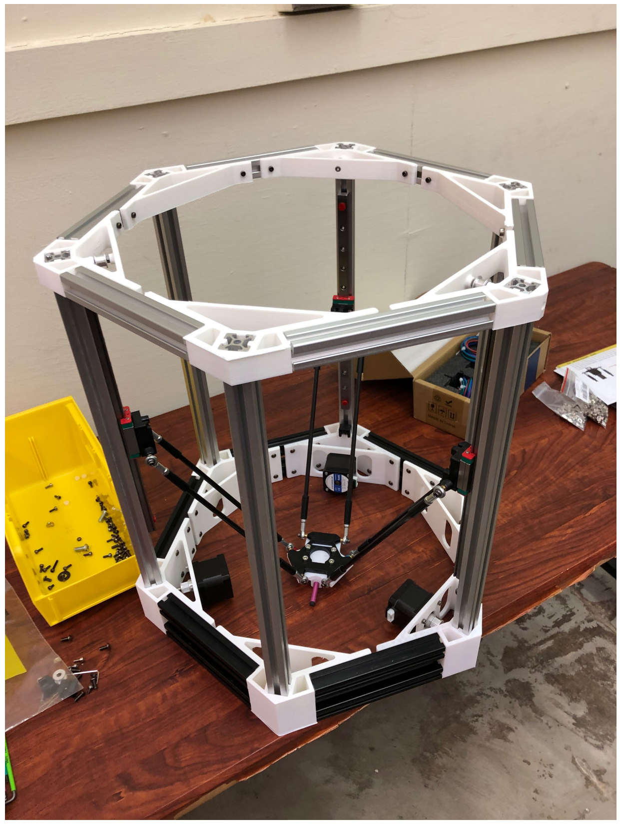


Figure 4.9: Image of the Mark II prototype prior to new design changes.

tablets.

For the Mark II design, we moved to a honeycomb like structure for the base of the prototype. This was done to improve the overall visibility of the tablets within the module and relate to our design of lace imagery, discussed in Chapter 3. We also customized the delta-robot arm structure with magnetic ball joints and Carbon Fiber and Delrin arms.

An issue noted in the Mark II prototype was the unresponsiveness of the tablets to the styluses' touch. During early testing, we realized that the stylus would not trigger the touch screen on contact. We learned that this problem occurred due to the lack of electricity conducted through the stylus. A majority of touch screens today are capacitive screens, which use an electrostatic field to register when the field is contacted by a conductor [26].

Styluses are able to operate due to the human body being a natural conductor of electricity, and use the electric properties charged within our fingertips to make contact. Without this current, the styluses were not able to make a conductive touch on each screen. This issue resulted in our artistic team attaching a grounding wire to each stylus. In addition, we purchased hybrid stylus replacement fiber tips that were better conductors of electricity than our original rubber ones. To accommodate the grounding issue, we also changed the structure of the moving platform to conduct the grounding wires to each stylus. Figure 4.10 shows the structure of the new platform and the new stylus.

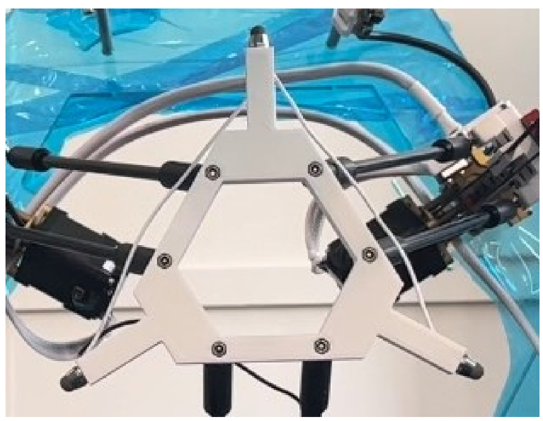


Figure 4.10: New Platform Design for Mark II prototype, including new hybrid fiber stylus nibs and grounding wire.

Mark III

In Mark III, we adapted on some of the issues from the previous prototypes. The Arduino MEGA 2560 models were transitioned into Arduino UNO Wifi Rev2 models to remove the need for multiple Ethernet cord connections to our main server, which is further discussed in the Network section, 4.3.

This prototype was also built with a wider honeycomb design to increase visibility of an individual tablet for a viewer. Figure 4.11 shows a rendering of the Mark III prototype's

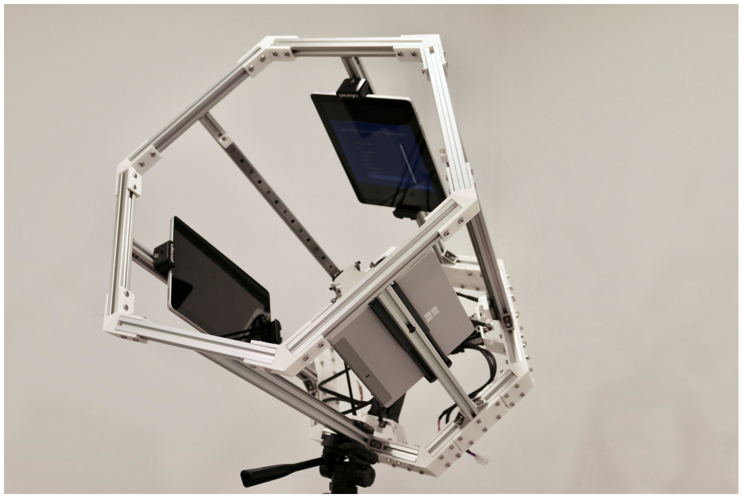


Figure 4.11: Digital rendering of the Mark III prototype.

design.

Final Prototype

After the in-depth testing of the three previous prototypes, we were able to create our final prototype which adapted on all of the previous issues. For the final design, the wider hexagonal design from Mark III was removed and returned back to its design from Mark II as we found the Mark II design to be more aesthetically pleasing for the design and worked better with our kinematics library. The prototype measures at approximately 24” in diameter and 24” deep. We updated the motor model to the Nema 17 Stepper Motor and the driver to the Integrated Stepper Motor Driver ISD04. The module was also customized with an attached USB-C charger for all three Microsoft Surface tablets in order to help them remain charged during the duration of a game.

We also observed the motors of our prototype would tend to overheat after about an hour of non-stop motion. We found this issue to be due to the motors constant work. Even while a stepper motor may be holding a position, the driver is going to be constantly applying

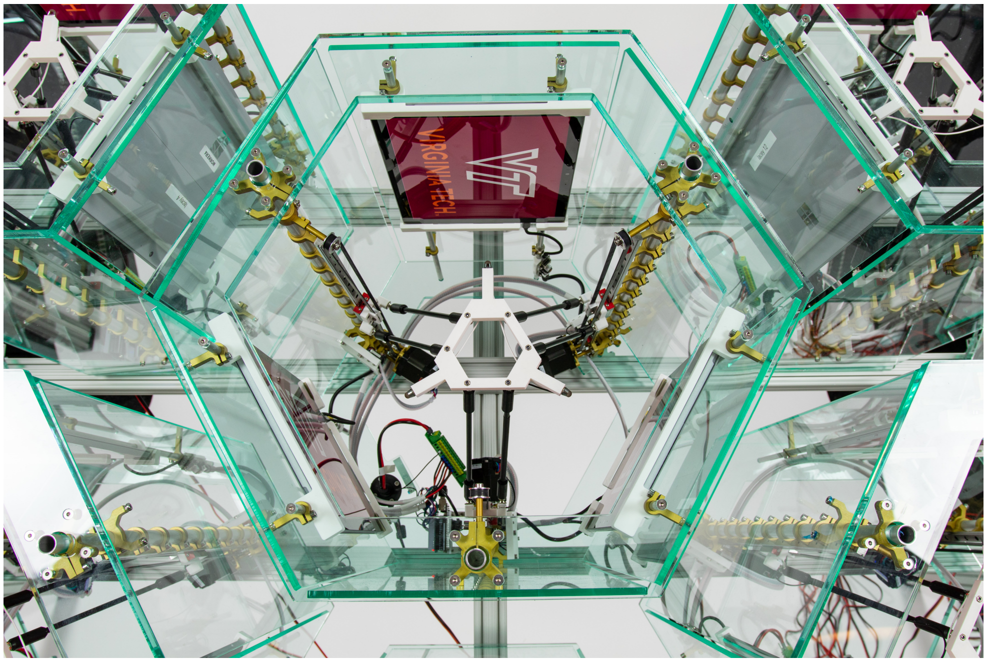


Figure 4.12: Final Prototype with Acrylic Bases. Photographed by Nikita Shokhov, 2022.

current to the motor and heating it. This ongoing current caused our motors to become hot to touch after about an hour of running. This resulted in a redesign of our kinematics library to enable and disable each of the motors after every move. Once a motor had moved to its determined location, it would immediately be disabled to avoid the constant current application. It is only re-enabled once it has been instructed to move again. Doing this allowed us to maintain the temperature of the motors within its insulation class of 180 degrees Celsius or 356 degrees Fahrenheit.

Once the functionality of this prototype was successful, we focused on improving the overall design of the module artistically. To create a more visually alluring experience, the artistic team added translucent-green acrylic glass covers to surround the covers of each module. This allows viewers to see the precise movement of the modules and interaction with the tablets closely, while also giving the structure a beautiful glow illuminated by the tablets. We also customized the color and design of the LACE to be similar to Ada Lovelace's previous work on the Babbage Engine, by anodizing the mechanisms inside each module with gold finishing shown in Figure 4.14.

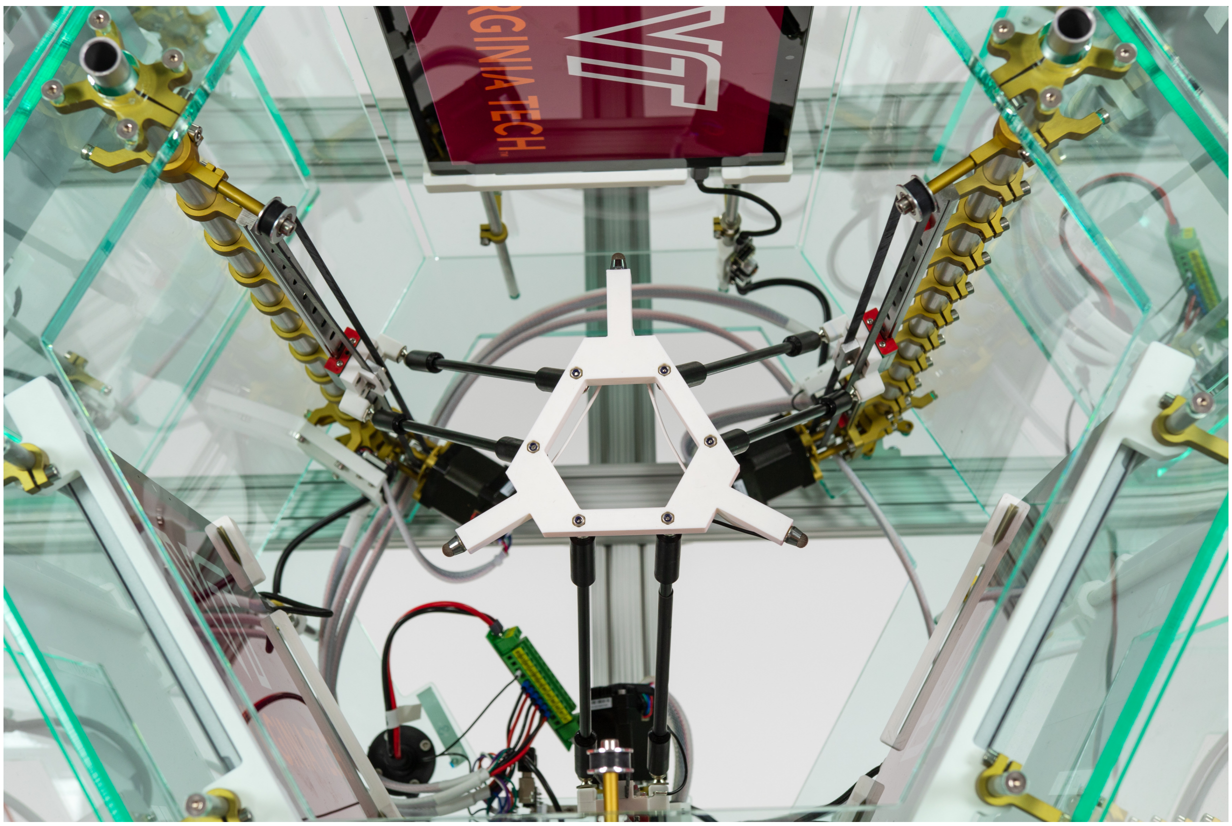


Figure 4.13: Closeup look of the finalized prototype for the LACE modules. Photographed by Nikita Shokhav, 2022.

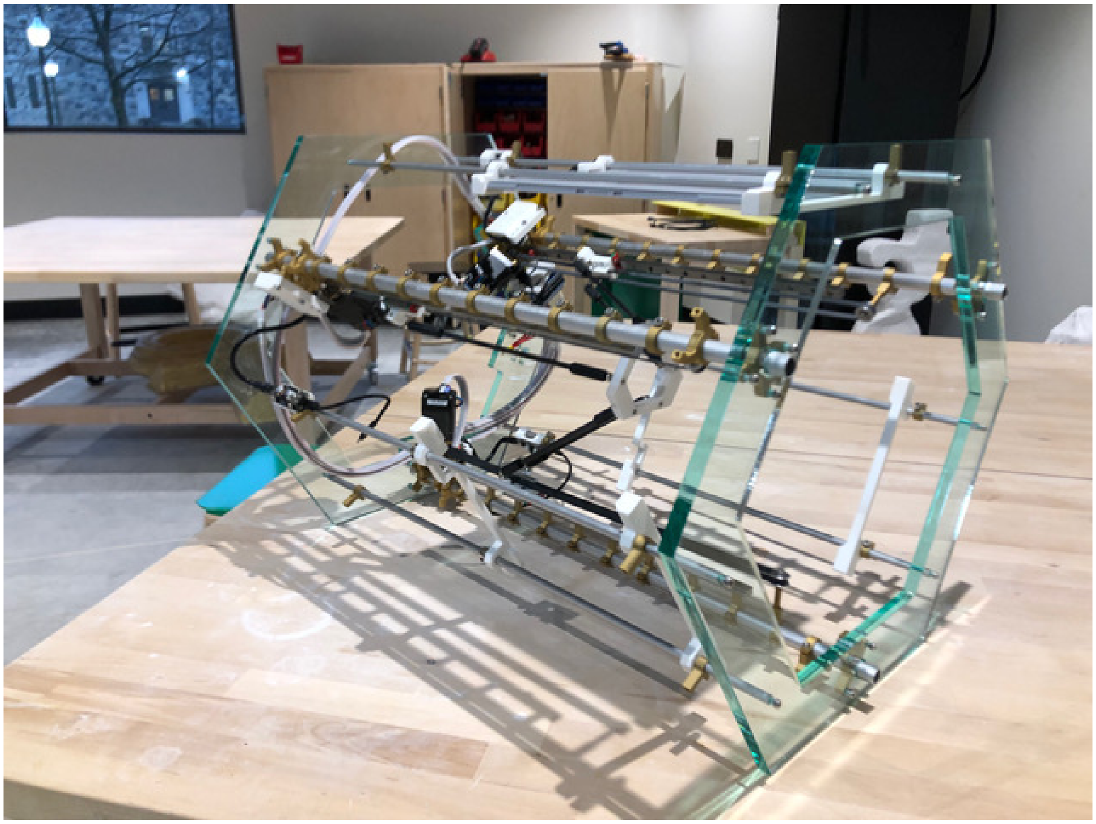


Figure 4.14: Final kinetic module without acrylic paneling.

4.1.2 LACE

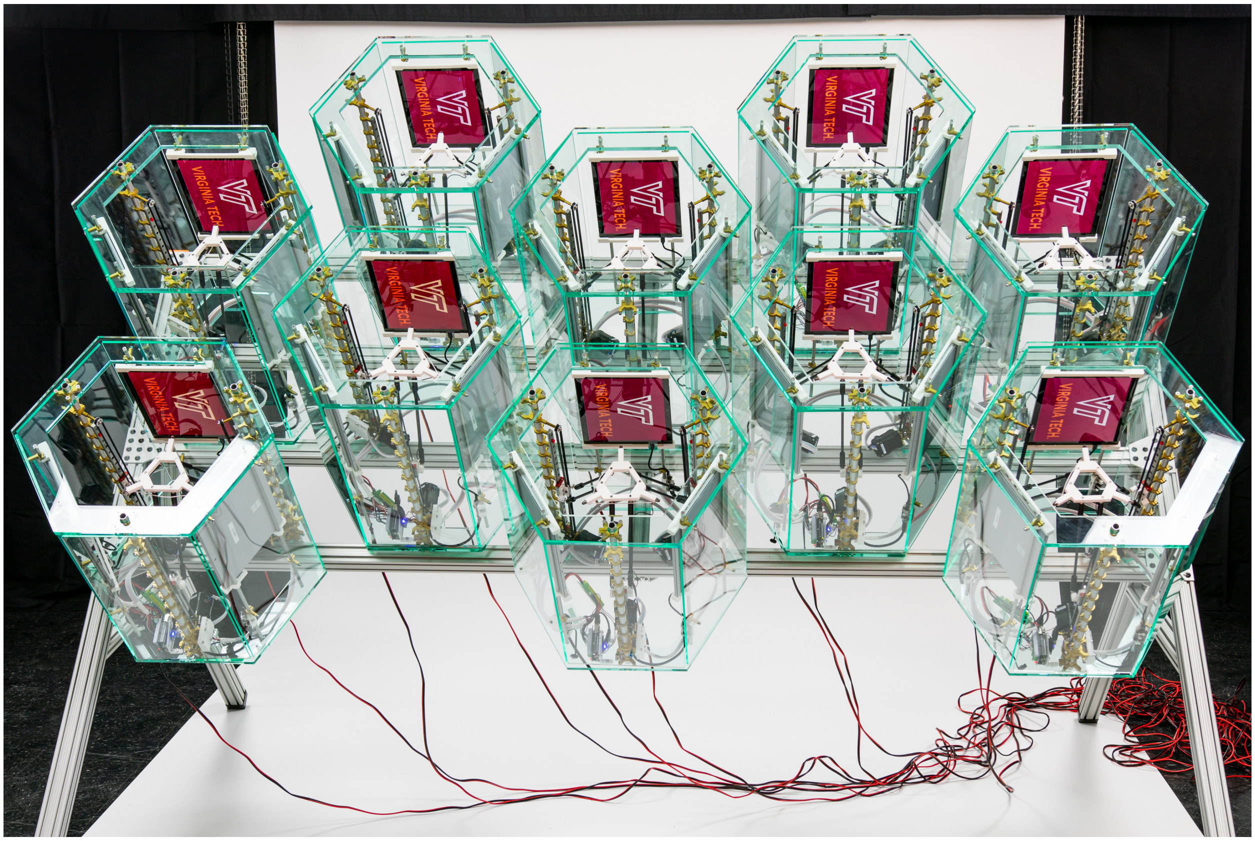


Figure 4.15: Completed LACE sculpture with 10 kinetic modules, each containing three Microsoft Surface tablets functioning as nodes within the blockchain network.Photographed by Nikita Shokhav, 2022.

Once the design of an individual kinetic unit was finalized and tested, we set out to scale the design to a 10-unit cluster of kinetic units. We assembled 9 additional kinetic units from our final prototype design. All 10 modules were mounted onto an aluminum extrusion grid which was developed by our artistic team. The modules are structured in a honeycomb-like pattern. The aluminum grid was also built to lean the LACE sculpture at a 30 degree angle, in order to ensure better visibility for viewers both close and far away. This angle ensured better freestanding ability for the structure as well. Figure 4.15 shows the final design of the 10 module sculpture.

We use a Mac Mini as the central server for LACE and an Archer AX11000 Tri-Band WiFi 6 Router for our network. This server communicated with both the game application and the Arduino's to operate the movement of the sculpture. Each module is powered by a Mean Well LRS-350-24 Power Supply. We used a single 15A120 circuit to power the entire structure. A central power server was developed by our artistic team with overload and surge protection in order to protect the components, shown in Figure 4.16. We estimate the weight of an individual module being about 25 lbs while tablets are installed, making the entire structure roughly over 250 lbs. We estimate about 2000 individual pieces of hardware are used within LACE, including screws, nuts, washers and additional materials. We estimate about 80ft of aluminum extrusion for the entire structure as well.

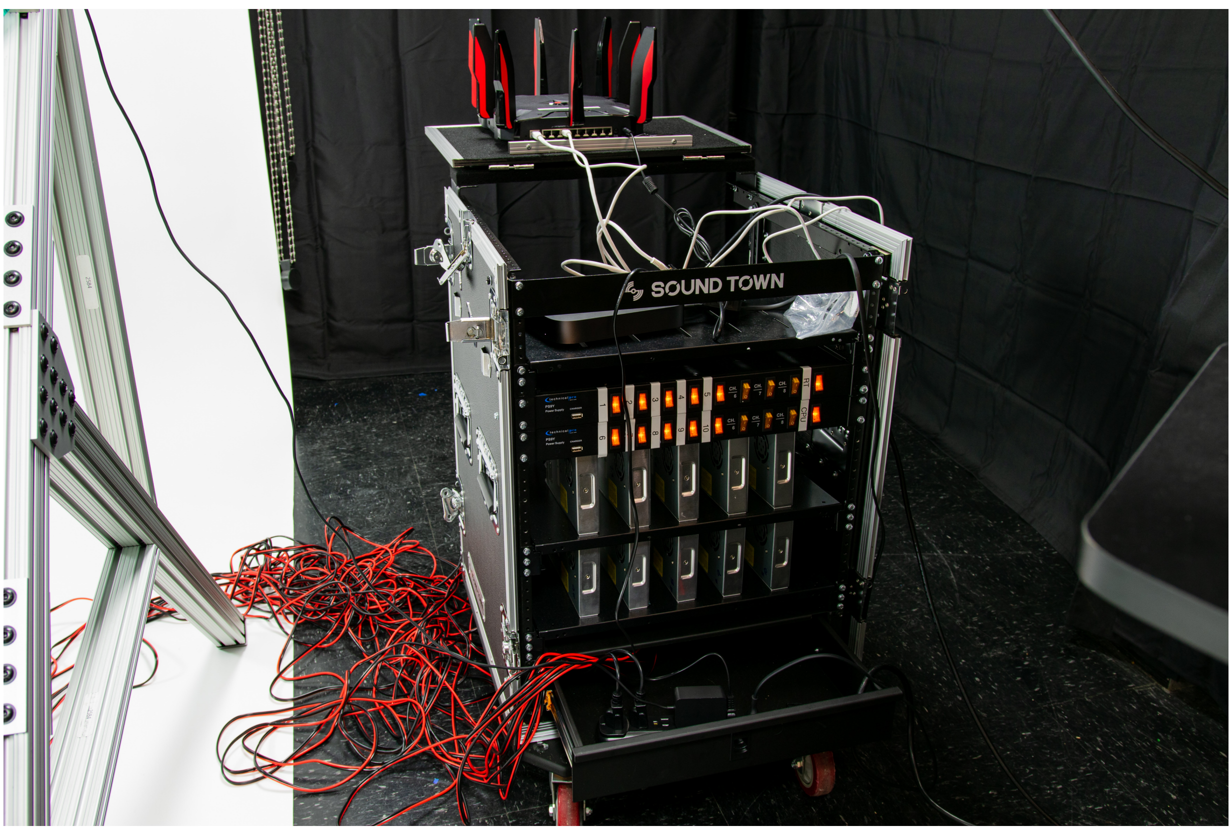


Figure 4.16: Power structure for LACE sculpture. Contains 10 individual controls for each kinetic module to be turned on and off. Additionally powers LACE's server and routing component. Photographed by Nikita Shokhav, 2022.

An additional feature we added to LACE was a visual representation of the effects of different types of real-world hardware on the mining process. For each demonstration, the speed of each modules' movement would be adjusted to represent the type of hardware a miner may use. Units with faster movement would demonstrate the work of more improved and efficient hardware miners like those used in specialized bitcoin mining farms. Slower modules would represent the mining process of the average individual who may not have access to such advanced/expensive equipment. To mimic this hardware disparity amongst miners, our server communicates with the controller Arduino within the modules to determine the speed of movement for each unit. At the start of the game, a list of speeds is randomly generated and is given to each individual unit to determine how fast or slow their movement will be.

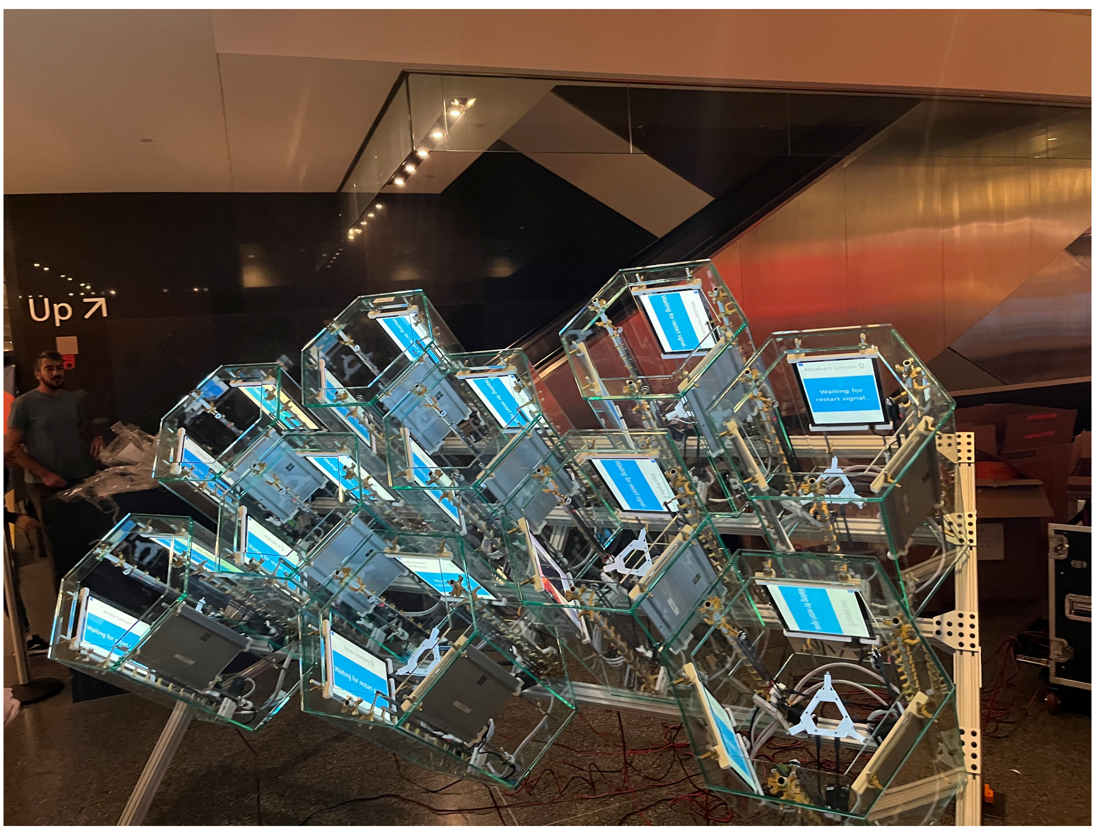


Figure 4.17: LACE structure displayed at the 2022 ACCelerate Festival in Washington, D.C.
.

4.2 Creating the Game

4.2.1 Underlying Technologies

In our early design, we believed that the use of a python-based program would work best to create the structure of our game. Python is a very powerful programming language with a vast library of tools to use and provides the ability to rapidly produce prototypes. Python also supports multiple libraries and frameworks specifically for game development. For this work, we attempted to use PyGame [60], a cross-platform set of Python modules designed for writing video games and multimedia applications.

However, we realized early on that the use of PyGame would not satisfy our requirements. Since our network of clients consisted of a minimum of 30 Microsoft Surface tablets, our application would need to be able to communicate effectively throughout this cluster. We also wanted the ability for our server to receive information from each client and allow messages to be sent back and forth from our server to various clients simultaneously. This type of interaction would allow the game server to be continuously updated on the work of each client and display such work in real time. We attempted to implement a multi-threaded program that would allow for clients to continuously listen for and receive messages from a main server. This method was not successful as we encountered an issue regarding Python's Global Interpreter Lock (GIL). The GIL is a "mutex or lock that allows only one thread to hold the control of the Python interpreter" [12]. This means that only one thread can be executed or in a state of execution at a time, switching the control between individual threads continuously, which became a major performance bottleneck for our program. This also prevented the ability to continually listen on a port for incoming messages. These issues could potentially be resolved by using a multiprocessing approach, rather than threading, but this would have ultimately changed our client-server architecture design and over-complicate our work. Finding this issue, we decided to look into an asynchronous approach, which would allow for requests from clients to be handled concurrently. While Python does support asynchronous programming, it is not a default option of the language and is not the optimal framework for the type of messaging system we were looking for in our work. As a result, we decided to explore different options for applications to support the needs of our game.

We then decided to explore different web-app multiplayer gaming frameworks that would allow us to model the client/server architecture from our original design. To do so, we investigated three technologies, WebSockets, Socket.IO and Node.js.

WebSockets [37] is a bidirectional correspondence between a client and server through a TCP connection. The protocol defines an API that creates a socket connection to allow for multi-way communication. Unlike the traditional HTTP request/response format, this protocol allows for the server to communicate with and send information to the client without the need for a request by keeping the client-server connection open after the initial connection. Communication is conducted through the same initial channel until the channel is terminated on either end.

Socket.IO [57] is a JavaScript library that uses the WebSocket protocol to provide an event-driven communication between the server and client. The library consists of a Node.js server API and a JavaScript API library for the browser or client. The Socket.IO library also has advantages over the original protocol as it supports additional functionality such as broadcasting, fallback options, and multiplexing. The use of this technology allows for real-time, two way connections in which both client and server can initiate communication and freely exchange data [18].

Node.js [2] is an asynchronous event-driven JavaScript run-time platform that is designed to build scalable network applications by using non-blocking, event-driven I/O to efficiently share data across clients. Node.js is able to manage several concurrent clients through their "Single Threaded Event Loop" design. Within this framework, all incoming requests are handled by a single event loop thread which assigns the requests to an event queue. The event loop continuously receives and processes non-blocking requests through callback functions associated with the event and send the response back to the client. Blocking requests are processed through a separate thread pool maintained by the event loop thread. The event loop assigns available threads within the pool to read and process the clients blocking request before sending back to event loop. The event loop then sends the response back to the respective client and process repeats. Because of this event loop set up, we can initiate the requests of multiple clients immediately after each other without waiting for a response to the request, eliminating the need for multi-threading. Since fewer threads are used, Node.js employs fewer resources and results in faster task execution overall. This model also solved our previous issue with PyGame regarding threading, since all of this work would be handled by the Node.js framework.

Together, these frameworks and libraries are often used to implement client/server communication modules. We found these technologies to be effective with our requirements as it handled multi-way communication between both servers and clients internally.

Additionally, we needed a resource to host our application. For initial testing, we turned to Heroku. Heroku [1] is a container-based cloud Platform as a Service (PaaS) that is used by developers to develop, manage, and deploy apps. Heroku allows users to build and deploy Node.js applications for free while also managing their overall metrics such as CPU usage, memory and throughput. Once we had an understanding of the application, we transitioned to using the Express.js framework for hosting our game application through our central server during production, listening on port 6000 for connections. Express.js is a back-end web application framework that provides broad features for web applications and supports the use of Socket.IO [6]. Express handles HTTP requests coming into the application and the Socket.IO module listens for incoming websocket connections on the same port. Because each websocket connection has a unique identifier, we use these to track our connections throughout a game. Figure 4.18 shows a diagram of the architecture of our final application adapted from [66]. Using these technologies, we implemented an interactive web application to represent the work of bitcoin miners and the blockchain.

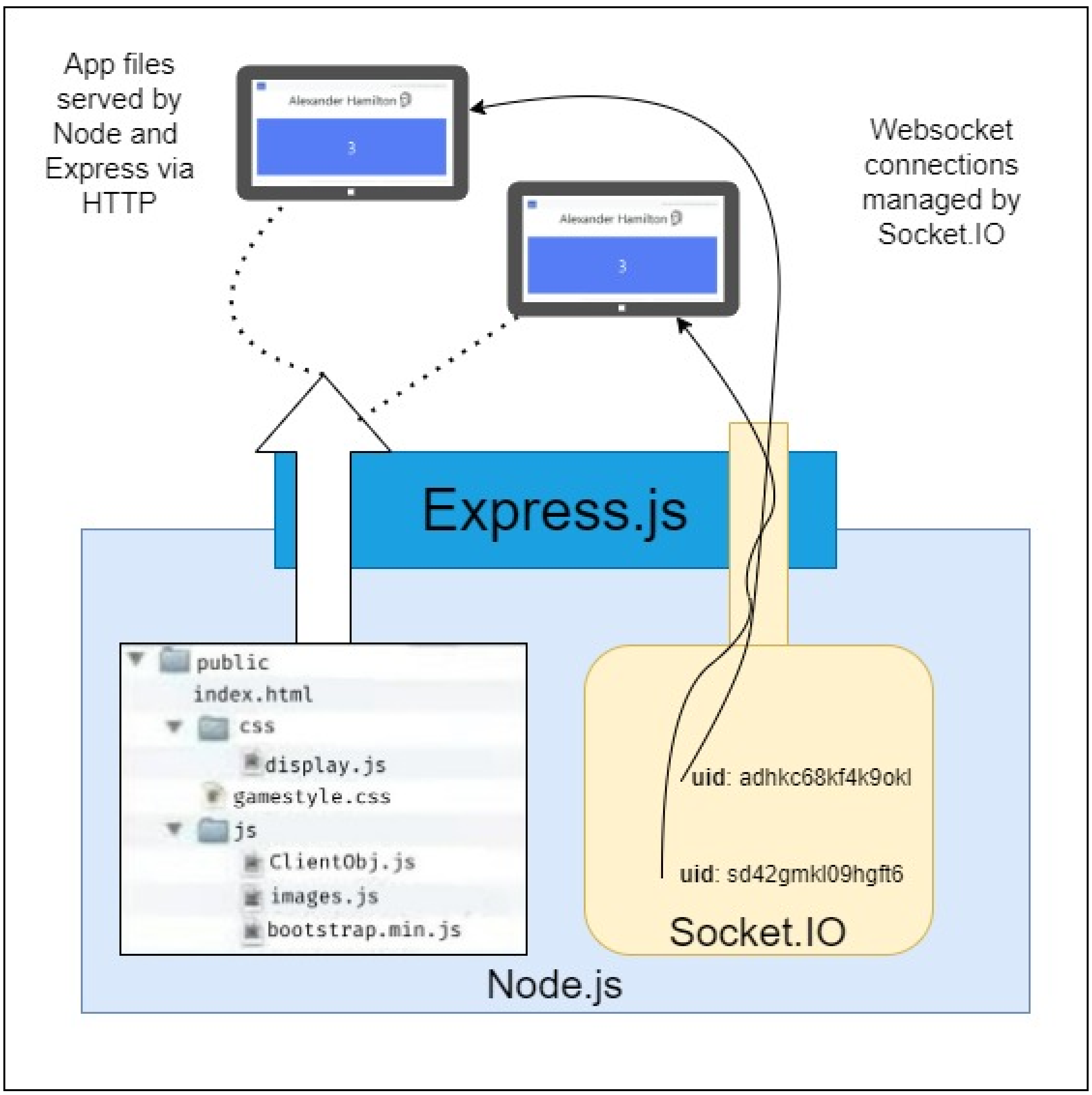


Figure 4.18: Architecture of Game Design, adapted from [66].

4.2.2 Mining Algorithm

In order to simulate the work of bitcoin miners, we had to simplify the process into using a simple guessing game. To make the work of our players as realistic to the mining process as possible, we mimic the work and communication of miners within a bitcoin mining pool for our mining algorithm. In this section we will describe the backend operations of the mining process for our algorithm.

Each game starts by partitioning the exploration space. This mimics the distribution of work done in mining pools, discussed in Chapter 2. At the start of the game, a range of numbers to guess from within the problem space and the size of an individual share is determined. We define a share to be a list of numbers that the client will guess, which represents their completed work. These values are then used to generate non-overlapping shares of the determined size from this space. For example, if the exploration space range was 1-10, with a share size of 4, these are examples of the potential shares: [1,3,5,6], [9,2,4,8], [7,10]. The remaining share will contain the remainder of numbers and may not be the predetermined size as shown. The given range must only contain numbers greater than or equal to one.

To track the shares being worked on, we divided them into three lists; incomplete shares, in-progress shares, and completed shares. All shares will start off as incomplete until the game has started. A winning number is also selected at random between the range and is used for the verification of determining the winner.

The shares are then assigned to each client and the game may commence. Clients are also passed the winning number for the round in order to verify their share. In a realistic scenario, miners would be unaware of the target hash value and would use a function to compare their guess to the hash of the given block. To avoid overloading our server with messages to check each clients guess, we pass the winning number of the round to the client when a new share is requested to allow for internal verification by our clients. Clients will compare their current guess to the winning number and if a match is found, a signal is sent to the server for verification. However, the work or guesses of the client is entirely random and does not benefit from knowing the winning number.

Each client will then "guess" the first number within their given share and check to see if it is the predetermined winning number. If they have not guessed the winning number, the client will continue to guess numbers within their share until that share is completed. This process is demonstrated in Figure 4.19 in cells 1 - 6.

Once a share is completed, a client will report said share back to the server. The share will then be processed to see if its work has already been completed. If the share exists within the list of completed shares, the client's work is then rejected. Otherwise, the work of this client will be accepted and placed within the list of completed shares and removed from the in-progress shares. This is done because multiple clients may have or receive the same share and will simulate previous work of other clients. Shares are accepted if they have made progress towards the goal of finding the winning number and rejected if the work has already been done. Cells 7 - 9 demonstrate this process in Figure 4.20. This process reflects the work of bitcoin miners accepting and rejecting shares, discussed in Chapter 2.

After this process, the client will request for a new share to work on. Preference is given to incomplete shares that have not been worked on yet. If none are available, the client will select a share from the in-progress shares in a round-robin style. This results in clients duplicating work, but in hopes that faster workers will mine the remaining exploration space faster.

This process will continue until a client encounters said winning number. When this occurs, the client will send a signal back to our server and request for verification. The server will check if the client has the winning share and will then halt the progress of the remaining clients. The winning client will then be added to our game's blockchain with their player name and winning number. The blockchain of winners is continuously updated with the most recent winner added following a game. This process can be seen in cell 10 in Figure 4.20. At the end of a game, all clients will wait for a restart signal for a new game.[1]

To assist audiences in understanding the mining process of our game, we do not display the transfer of work or shares between clients to audience members. This idea of sharing work may be too complex for beginning audiences to understand, so we omit it from our visualization. Instead, the number a client is planning to guess within their share is displayed on the button, allowing clients to repeatedly click and guess numbers within their shares. This process continues until a winner is found and the game has ended.

[1]Description of algorithm provided from work from and discussion with Gregory Bolet.

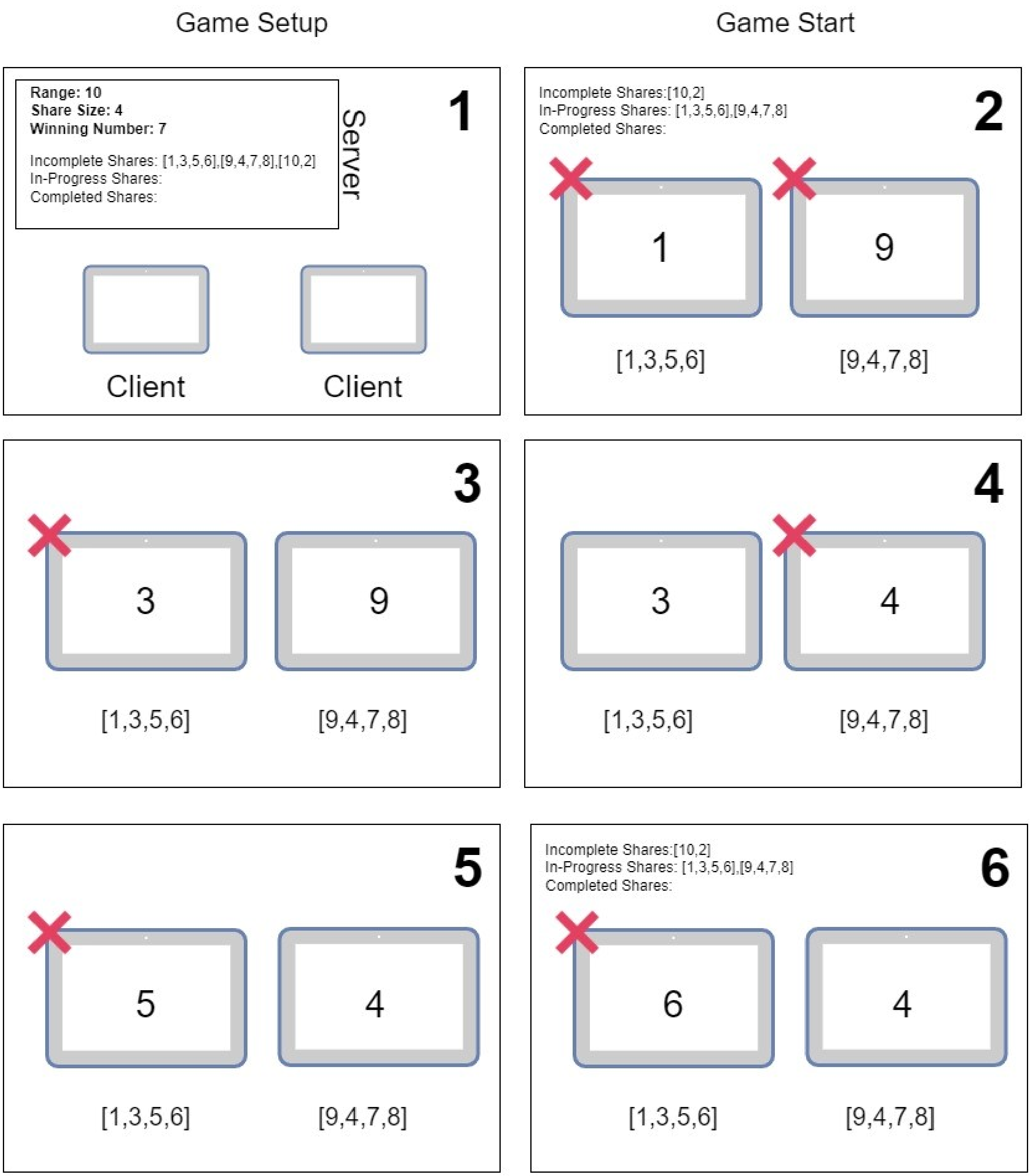


Figure 4.19: Diagram of Mining Algorithm with two clients. Both clients are given a share on game start by the server and then guess numbers with their share, one after the other. Continued in Figure 4.20.

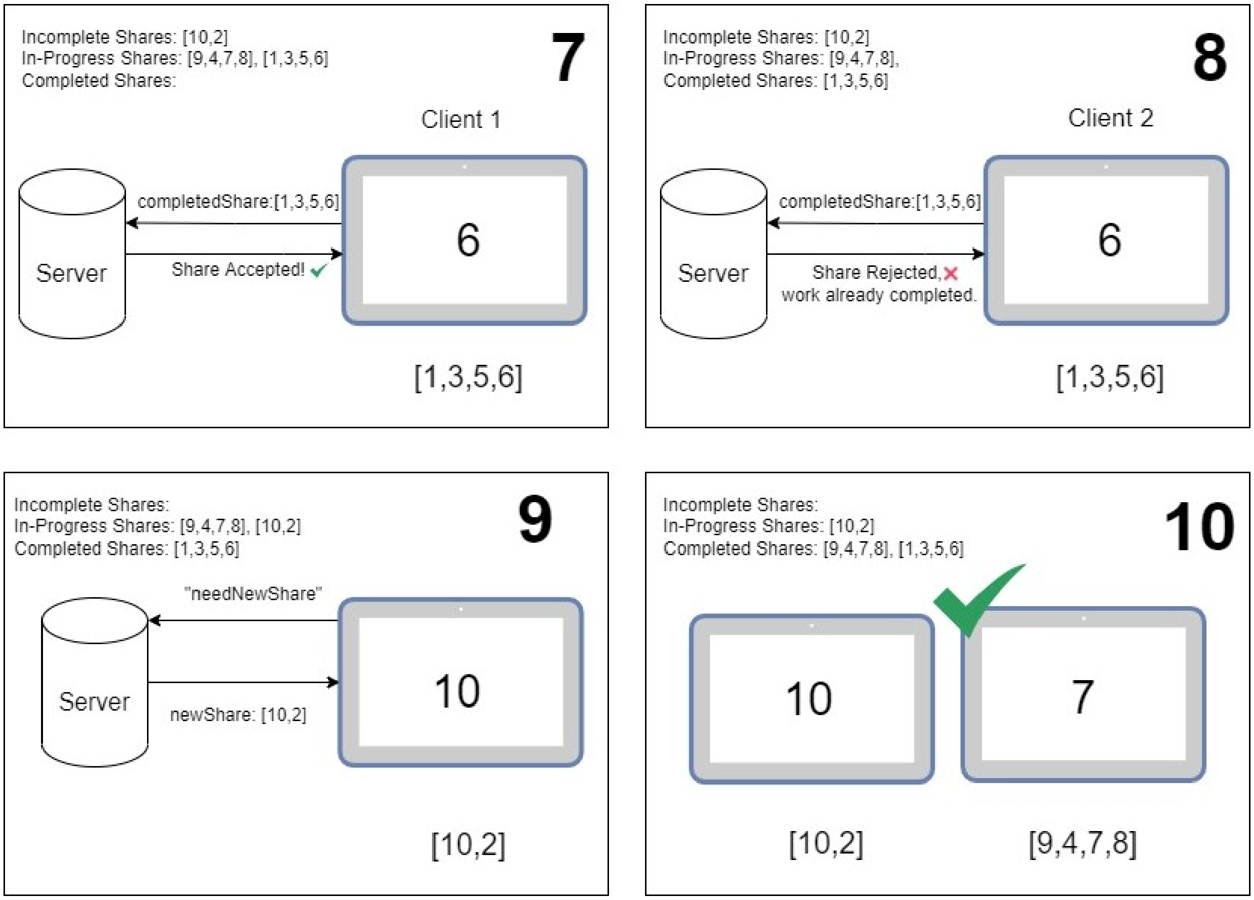


Figure 4.20: Continued diagram of Mining Algorithm with two clients. Client with a completed share verifies their work through the server to be accepted or rejected. Winning clients also verify work with the server and update list of complete shares.

4.2.3 Software Interfaces

The goal of LACE is to educate on bitcoin mining while also creating a visually appealing experience for our audience. In order to demonstrate the work being conducted, we created three visual interfaces, the Client Page, Display Page, and Administrative Page. Each interface was implemented using a combination of HTML, CSS, and JavaScript. This application is hosted through a Mac Mini and communicates with all 30 Microsoft Surface tablets and 10 Arduino's through websocket connections over our private network.Our program consists of four main components, the Server, Client, Display and Administrative entities.

Server

The architecture of our application is client-server based. While our server does not have its own visual interface, it acts as the central command for our application. The server listens for incoming socket connections from client tablets and once found, a client connection is created. On connection, each client is registered as a player and receives their player information. To maintain organization of all clients, we create custom Client objects which contain identifying information regarding each client including their unique socket ID, assigned player name, image, and number to guess. We maintain this information to keep track of the updates with each client and pass the updated information to our display page when changes occur. The server contains a mapping of all client objects using their unique socket id. This map contains information regarding each client including their character name, and number they plan to guess. Once a client guesses a new number, a message is emitted to the main server that a change has occurred and the mapping is updated. The server then emits a message to the display page to update the client's new number. The server additionally handles the creation and division of shares between clients mentioned in section 4.2.2. If a client disconnects from the game at any point, the server removes said client from the mapping. The server also monitors the connections and events of our Administrative interfaces and Display interfaces. In order to separate the work of the Client, Display and Administrative interfaces, we create three separate namespaces for each interface. In Socket.IO, a namespace is a communication channel which allows you to split the logic of your application over a single shared connection, essentially assigning different paths for sockets [57]. This feature allows us to introduce separation between communication channels for the respective pages. Through these separate namespaces, we can customize the event messages sent to each namepsace based on the needs of that interface. This feature would allow us to create restrictions for specific pages, such as the Administrative page, for only researchers to have access to if necessary. It also minimizes the number of TCP connections as multiple namespaces share the same WebSocket connection and save socket ports on the server [57]. Figure 4.21 shows a diagram representation of the communication between client and server with multiple namespaces. Finally, the server also communicates with our Arduino's to update their kinematic data, assign movement speed and update the current status of the game.

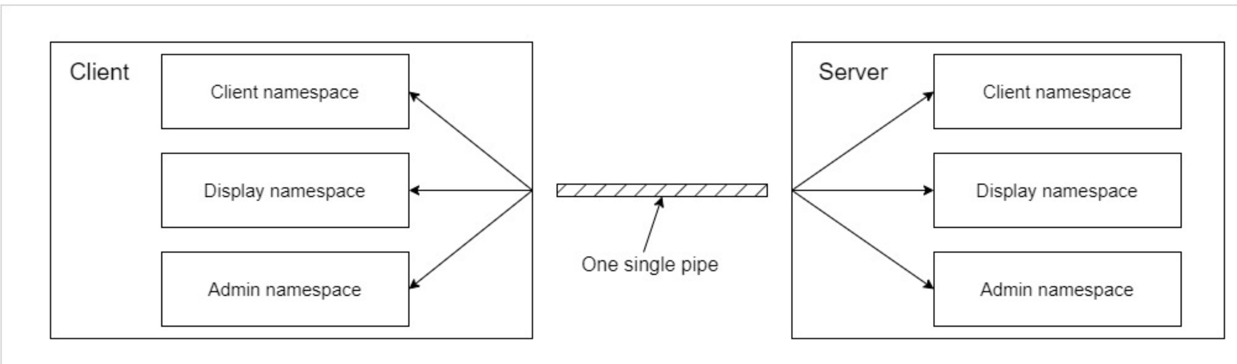


Figure 4.21: Diagram of Client/Server connection with namespaces for game application, adapted from [57].

Client Interface

This interface displays the work of an individual client or player. Each client's interface consists of the player's name, an image of said player, and an interactive button containing the client's current guess, shown in Figure 4.22. To simulate a guess, each client must tap the button. If the client's current number is incorrect, the client will be alerted that they have chosen a wrong number, as shown in Figure 4.23a, and move on to the next number within their share. If the client has chosen the winning number, the client's interface will then turn green with a green check mark, highlighting they are the winner, Figure 4.23b. All remaining players will be alerted that a winner has already been found, and to wait for the restart of a new round.

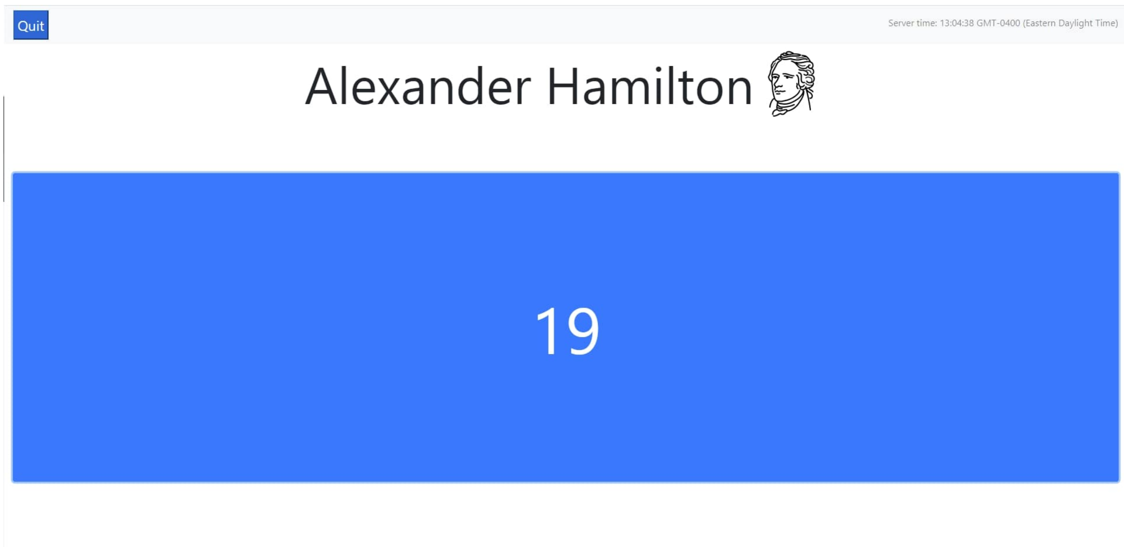
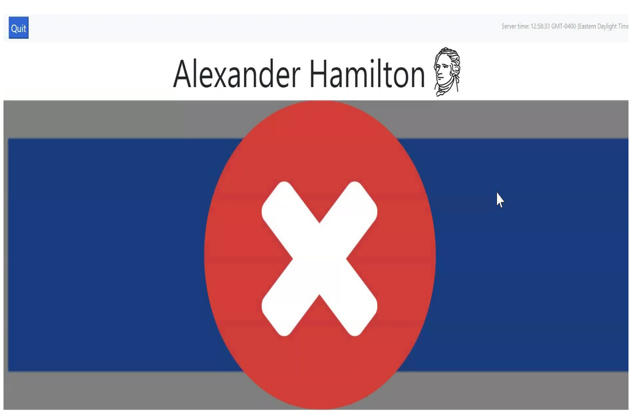
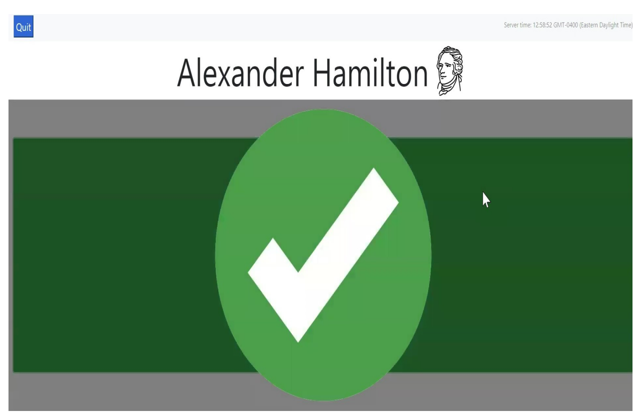


Figure 4.22: Interface of Client during game play.

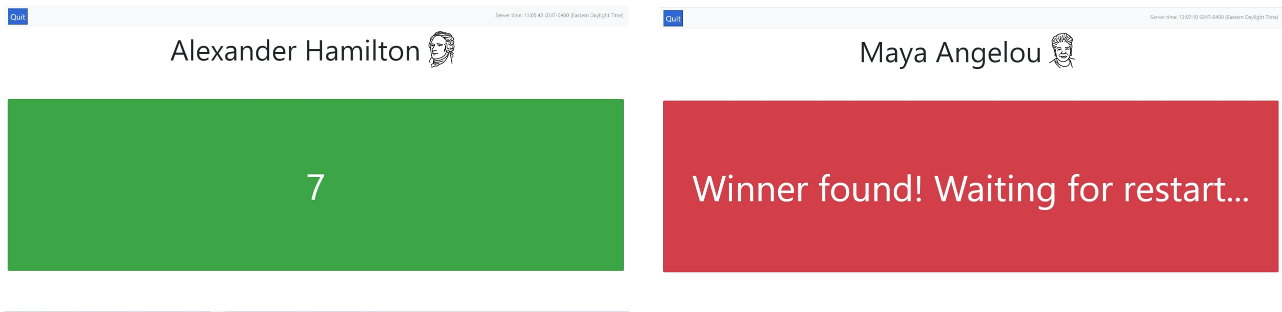


(a) Client interface when a incorrect number is guessed.



(b) Client interface when the correct number is found.

Figure 4.23: Examples of Client Interaction



(a) Winning client screen

(b) Losing client screen

Figure 4.24: End Game Visualizations

We also developed an animation to visually represent the "consensus" period. Once a potential winner has been found, we simulated a "consensus period", in which each client would determine the validity of said winner's guess, similar to the Proof-of-Work algorithm discussed in Chapter 2. To represent this, each client would show either an "ACCEPT" or "REJECT" visual on their screen, as shown in Figure 4.25, to determine whether they agreed the transaction of the winner is valid. We programmed our simulation to have a majority of clients accept the work of the winner, about 90%, with the remaining 10% rejecting the work. Only tablets within the LACE structure would receive this animation. While the verification of the winner is ultimately done by the server rather than the individual clients, we created this visualization to demonstrate what would happen in a real network scenario.

(a) Client acceptance (b) Client rejection

Figure 4.25: Consensus Algorithm Visualizations

Following each game, players outside of the LACE structure were prompted with the option to participate in an optional survey, as seen in Figure 4.26. If they chose “Yes”, players would be redirected to a Google Survey link which provided questions regarding their experience with LACE and their overall understanding of blockchain and bitcoin concepts.

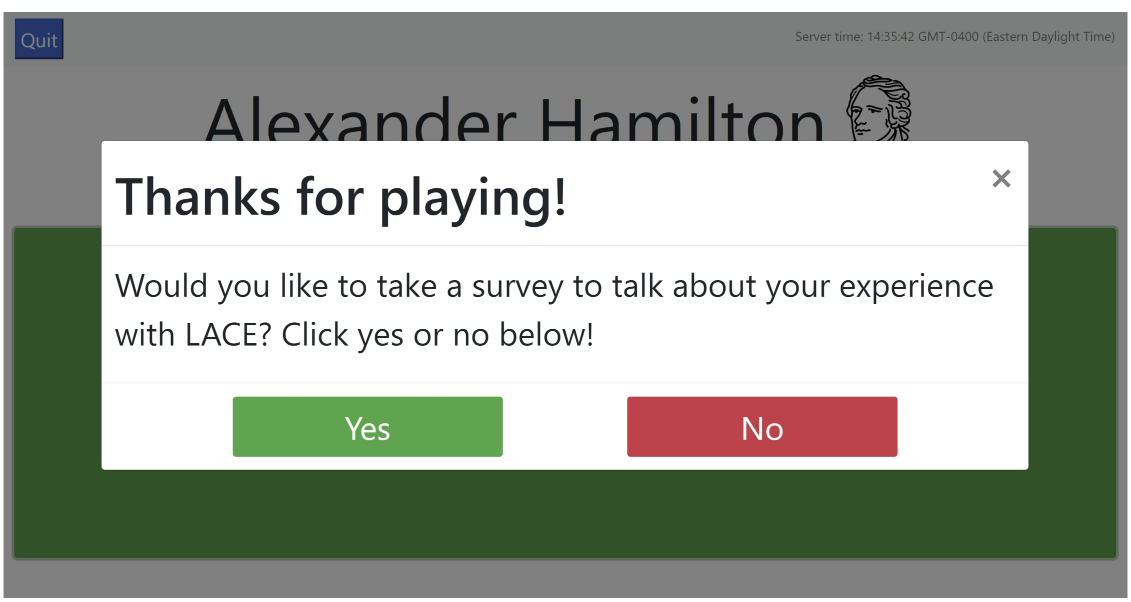


Figure 4.26: Survey request interface, displayed at the end of the game to all participants.

Display Interface

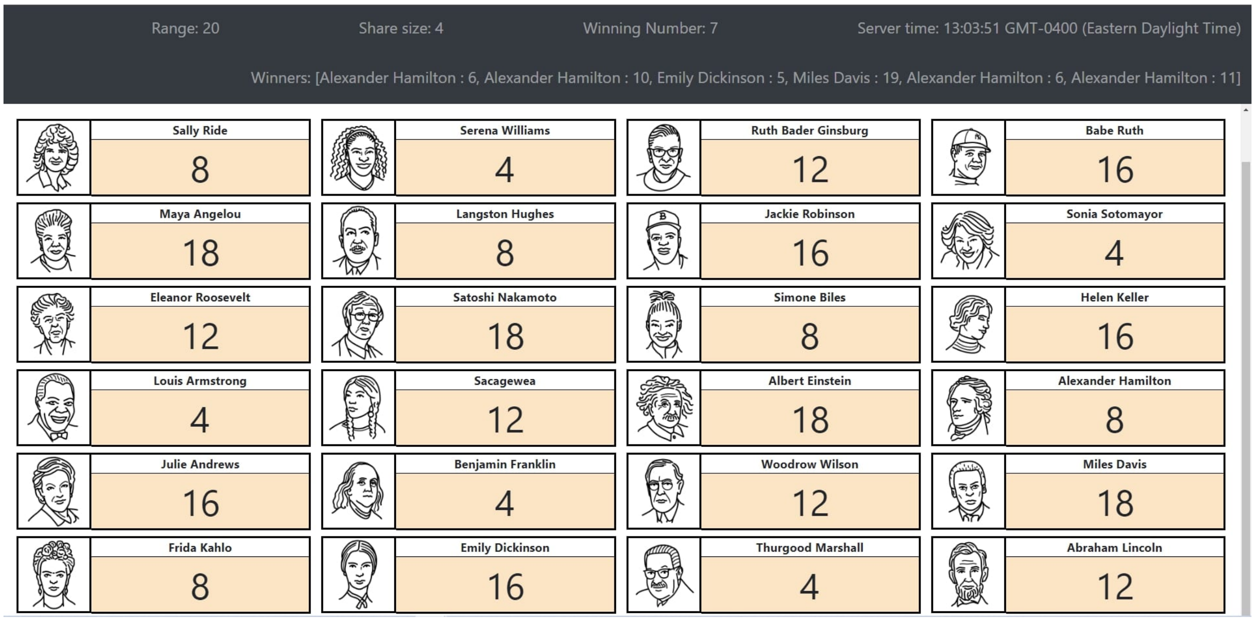


Figure 4.27: Display page during game play with 24 clients. Displays the range, share size, winning number of the game, and the blockchain with all past winners.

The Display interface shows the work of each client within the game. Figure 4.27 shows an example of the display page during a game with 30 players. On this page, each playing client is shown with their name, image, and the number they plan to guess. We also display the range of numbers the client's are guessing through and the size of the shares they are receiving at the top of the screen. To simplify understanding for viewers, we also chose to include the winning number for the current round at the top of the screen so audience members could understand who won and why. In an actual mining scenario, the winning hash would be unknown to the individual miners. During game run time, when a client has guessed a number, the client's component on the display will animate, signifying a guess. If their guess is incorrect, their component on the display page will then switch to the next number they plan to guess and the process repeats. Once a client has been found to be the winner of that round, its component will highlight green with all the remaining clients highlighted red. This is demonstrated in Figure 4.28. Once a player has confirmed a win, they are added to our game's blockchain or ledger with their name and winning number, which is shown on the second bar of Figure 4.28.

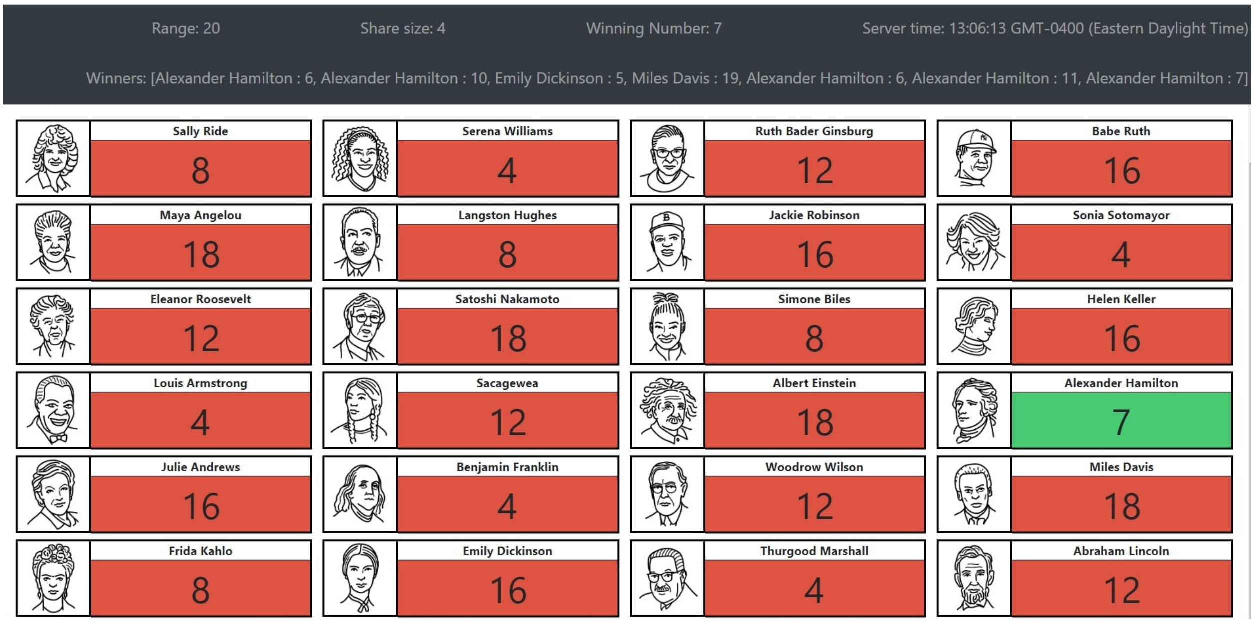


Figure 4.28: Updated display page when a winner is found. Winning client is highlighted green with losing clients remaining red.

Administrative Interface

The purpose of the Administrative interface was to allow for the researchers to start and stop the demonstration for audiences, and control the parameters of the game. Each component of Figure 4.29 is a button that controls the operation of LACE. The Start Game function is used to start and restart games after each round. The current status of the game will be displayed underneath the button, stating either "not started" or "playing". This button also starts the movement of the kinetic units on start. The Start Arduino function allows the researchers to start and stop the movement of the kinetic unit without affecting the game play. The status of the Arduino's is also displayed as either "waiting" or "ready". The

Emergency Stop function is used in case of any emergency or major malfunction with the LACE structure and shuts down movement of each kinetic unit immediately. The Set Game Params function allows for researchers to restart the current game to change the range, share size, and winning number.

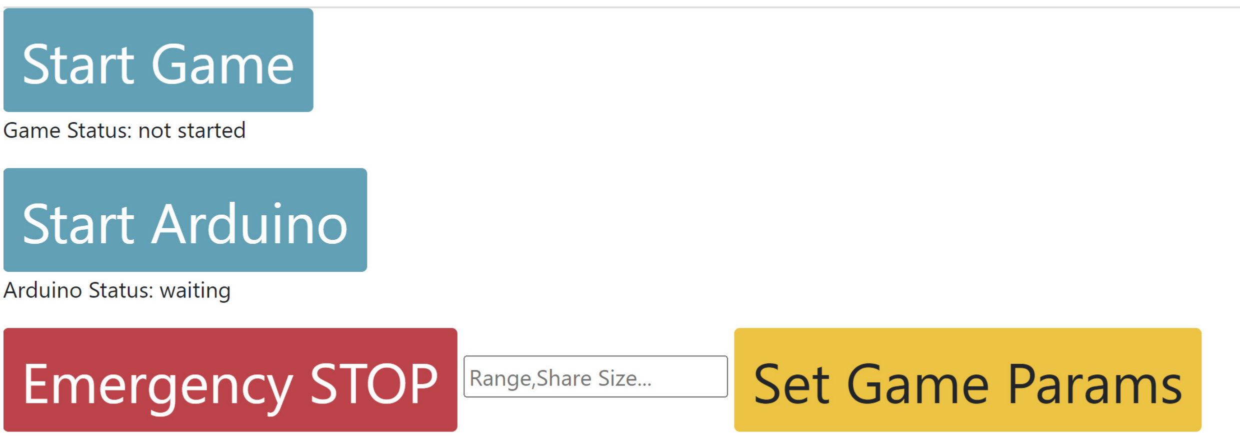


Figure 4.29: Administrative Page

4.2.4 Character Design

Since LACE's debut would take place at the Smithsonian's Museum of American History, we decided to represent each client within our game as a different figure in American History. We researched 50 different influential people in American history to represent the players in our game. Each of these figures have been represented in one or more of the National Smithsonian Museums over time. Our group of characters includes figures like Abraham Lincoln, Martin Luther King Jr., Maya Angelou, Dolly Parton, Sonia Sotomayor, and especially Ada Lovelace. Figure 4.30 shows images of all of our figures included in our demonstration. Each client is randomly assigned one of the 50 figures as their character for the game until a new game is started. We hoped using recognizable faces as players within our game would not only engage the interest of our audience members but help solidify the idea of each player as actual persons competing against each other.

Figure 4.30: Game Characters - Illustrations by Marina Mayflowers

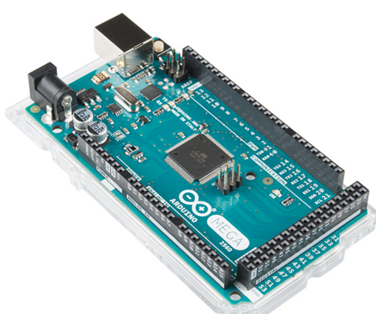
4.3 Network

In this section we will discuss the hardware used to set up the network of our distributed computing system on both the client and server side. We show an overview of the interconnection of all parts of this network, and discuss any issues faced during this process.

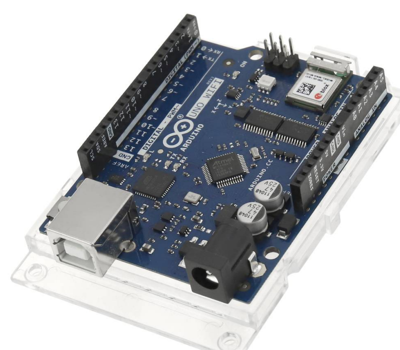
4.3.1 Client Hardware

Arduino

To control each node's movement, we initially used the Arduino MEGA 2560 model as the control unit. Each Arduino would be connected to our central server through an Ethernet switch and receive instructions on the module's movement. However, during the work on the SeeMore project [44], which consisted of a 256-node Raspberry Pi cluster, the researchers found that the amount of direct connections from the Pi's to their direct server was somewhat of a detriment to the beauty and function of the sculpture. Since a major part of LACE's purpose is to be a visually appealing while robust display, we transitioned to the Arduino UNO WiFi Rev2 model which allows for a WiFi network connection to our server and would eliminate the need for additional Ethernet cords. The Arduino UNO WiFi Rev2 consists of 14 digital input/output pins, 6 analog inputs, a USB connection, a power jack, an ICSP header and a reset button.



(a) Arduino MEGA 2560 model



(b) Arduino UNO WiFi Rev2 model

Figure 4.31: Arduino Models

The function of the Arduino's depends on two main parts, the kinematics library and its WiFi manager. For our WifiManager, we use the WifiNINA library, which allows users to establish a network connection with the UNO Wifi Rev2 and instantiate servers, clients and send/receive UDP packets through said connection.

On set up, each Arduino is assigned an individual IP address, and passed the IP address of both the router and our main server. Each Arduino then attempts to connect to our network. If the Arduino fails to connect, the Arduino instructs the main component of the module to move to the middle position, then disables all motors and attempts to reconnect. Once successful, the Arduino is then provided with the kinematic data needed for the module from our server. As mentioned in section 4.1, we produced a kinematics library that establishes the movement of a module based on given measurements. We establish a default measurement for the module based on those taken from the final prototype. While the additional 9 modules are based on the measurements of the original prototype, we found there to be measuring errors when using the default kinematics. To ensure contact with the Microsoft Surface tablet, we added additional offsets to the kinematic data which were specific to an individual module. These offsets would either add or subtract from the default kinematic data based on what was necessary for the module. This custom data is then sent to each Arduino module as its updated kinematic measurement. If an Arduino is not able to connect to our network, it will revert to its default kinematic data, which often would cause a module to not be able to correctly interact with the tablets.

Once the kinematic data is uploaded, each Arduino sends recurring GET requests to the server to determine the status of the game. The default status for each Arduino is stated as “waiting”. When a game is started or in progress, this status is updated to “ready” and the Arduinos are instructed to enable all motors and begin motion. The order of the tri-tap movement for each module, which is discussed in section 4.1, is then randomly assigned for each game through the Arduino kinematics library. These GET requests are sent consistently throughout game play to determine the status of the game. Once a winner has been determined by our server, the status of the game will be returned back to "waiting", which instructs the motors to return to a resting home position until the status is updated. This process continues for the remainder of each game.

Windows Tablets

Figure 4.32: Microsoft Surface Pro 7 Tablet

We use 30 Microsoft Surface Pro 7 tablets within the LACE sculpture as our visual representation of the distributed computing network. Each tablet works as a node within our system that assists in maintaining the blockchain, mining for new blocks, and processing transactions. All 30 tablets communicate through our established network with our main server and each other during game play. To keep track of our client connections and avoid potential DHCP issues, we statically assigned IP addresses for each tablet to be whitelisted to our network. During a demonstration, each tablet is directed to our web application which defaults to the client page and waits to begin game play. To simplify our setup process, we implemented a PowerShell script that would control the features of the windows tablets for the demonstration. On startup, each tablet is logged in and set to automatically run our

PowerShell script. This script was implemented to set the orientation of each tablet to Landscape and automatically attempt to connect to our network. Once connected, an Internet Explorer browser is launched and directed to our web application. Within the script, each tablet was also programmed to maintain full brightness regardless of power levels and lower their volume to mute. All automatic updates were disabled for each tablet as well to avoid issues during demonstration.

4.3.2 Server Hardware

Router & Mac Mini

(a) NetGear AC1000 R6080 WiFi Router

(b) Archer AX11000 Tri-Band WiFi 6 Router

Figure 4.33: Routing Equipment used for LACE's network

To establish our distributed computing system, we started by choosing the hardware for our central server. We used a M1 Mac Mini to host our application and handle the connections between the clients and Arduinos. The Mac Mini is used to provide internet sharing from a public network to our router in order to create our private network. The router then broadcasts the private network to the clients and Arduinos. To establish our network, initially we used the NetGear AC1000 R6080 WiFi Router to establish a network between our clients and server, seen in Figure 4.33a. This router was able to handle up to 15 devices at a time with an overall range of 1000 sq ft and speeds up to 1000 Mbps. This router was used for testing our single prototype and understanding the communication between Arduino and our server.

Once we had an understanding of a single module, we upgraded our router model to the Archer AX11000 Tri-Band WiFi 6 Router. We selected this model because it could handle the 40+ default connections (30 Microsoft Windows tablets, 10 Arduinos) needed for a decent price point. An additional advantage of this model was it offered the use of dual bands for connections. We decided to separate the tablet and Arduino connections onto the 5GHz and 2.4GHz band respectively to ensure the best performance. Since the Microsoft tablets needed to consistently update their pages based on the context of the game, we wanted to ensure high performance of our visual application through the 5GHz connection. The Arduinos were not supported on the 5GHz connection resulting in us moving them to the 2.4GHz band. Also, since the Arduinos were consistently spamming GET requests to our server, it was important to separate them from the tablets to avoid any connection issues.

Additionally, we created a guest WiFi network for participants to connect to our application. This allowed us to add additional connections without interfering with the network of the tablets and Arduino's.

Because we planned to give public access to our network, we wanted to be able to ensure the connections of the 30 tablets and reduce risks of unknown connections. To achieve this, we whitelisted the static IP addresses of each of our tablets into our router by creating a mapping to their given MAC address. This process was also done for the individual Arduino's. Figure 4.34 shows a diagram of the network interconnect.

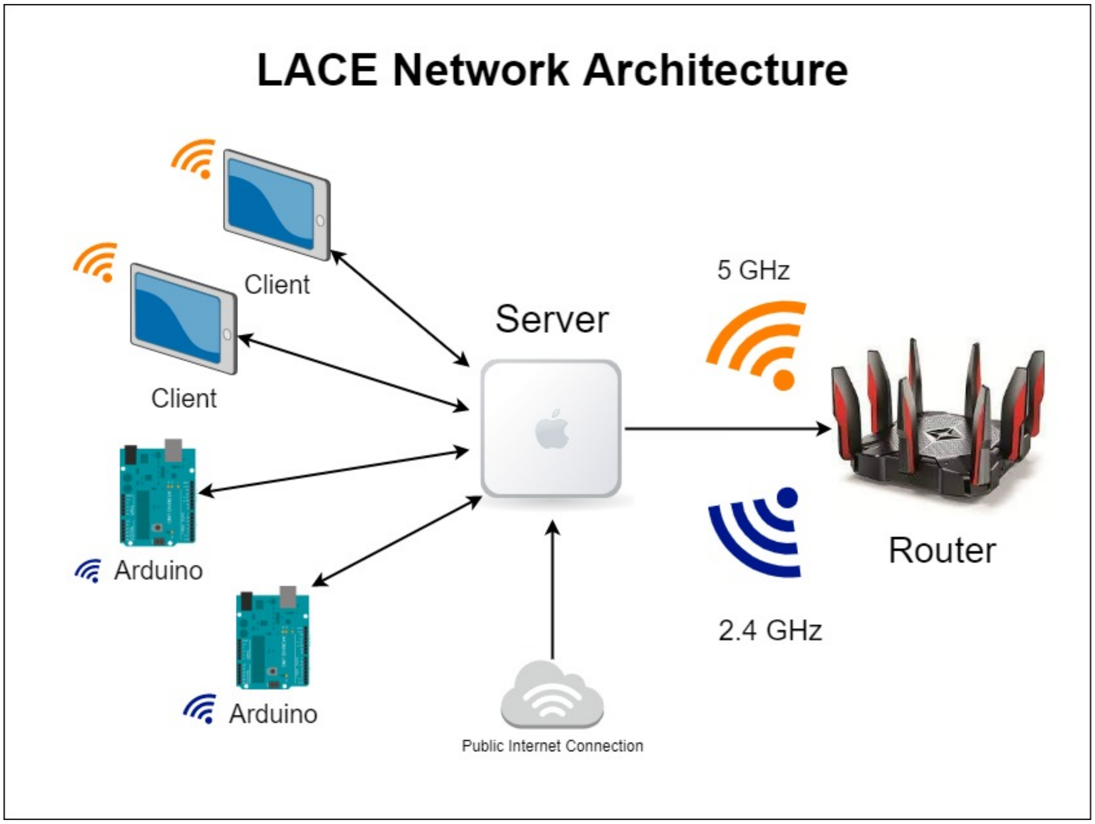


Figure 4.34: Architecture of the LACE Network

4.4 Audience Participation

4.4.1 Twitter

As mentioned in Section 3.5, we attempted to create a TwitterBot that would allow for interaction with LACE. The Twitter API is a set of programmatic endpoints that can be used to navigate and use Twitter in advanced ways. In order for developers to gain access to the API, they must sign up to the Twitter Developer Portal and present a proposal of their potential project to be reviewed by a committee. We found that our proposed use case violated one or more of the Developer Agreement Policy and/or Automation Rules, although it was never specified to us as to which section was violated. Due to time constraints, we were unfortunately unable to implement this factor into our main presentation and moved onto additional methods of engaging our audience. We discuss ideas for future presentations with this concept in section 6.2.

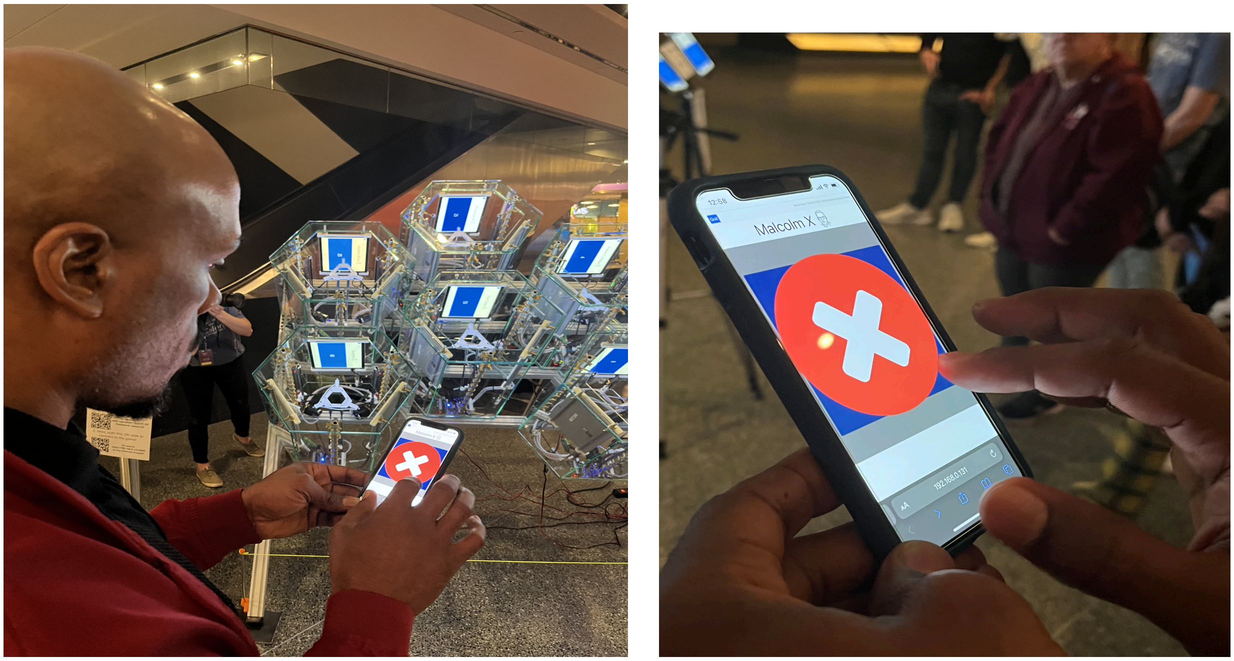
4.4.2 Tablet and Phone Interaction

Since our game now functioned as a web application, we could now host multiple connections to our game through our own network. We wanted to encourage the participation of audience members as much as possible in order to get a better understanding of how LACE worked and have an overall fun experience.

To allow for more direct interaction with LACE, we provided two Microsoft Surface mini tablets that audience members could interact with alongside LACE. These tablets were displayed during LACE's presentation to allow passersby to participate. Audience members were instructed to "guess" along with LACE and even attempted to beat LACE by trying to guess the winning number first. Figure 4.35 shows two audience members competing against the machine to win.

Figure 4.35: Two audience members playing against the LACE structure through our personalized tablet interaction.

We also introduced the ability for attendees to play alongside LACE using their own cellular device. In order to connect to our game, participants were required to connect to our networks guest WiFi connection, and then access the URL address of our game. To play, audience members were instructed to scan a provided QR code for WiFi connection and a following QR code with a link to our game application. Once connected, participants were able to play against LACE to guess the winning number and were also included on our display page. Figure 4.36 shows a participant playing our game through their phone along with LACE. Unfortunately, due to the large demand of players, we found that network failures occurred with the introduction of mobile players. We assume this issue to be a result of our router being unable to handle too many connections at once. We discuss further solutions for this in our future work.



(a) Audience Member participating with LACE on his phone.

(b) Phone Interaction closeup

Figure 4.36: Examples of Phone Interaction

Chapter 5

Analysis and Results

The objective of the LACE project is to create an interactive exhibit to:

- O1: Entertain and educate all audiences on blockchain concepts;
- O2: Personalize the importance of blockchain/cryptocurrency education;
- O3: Inspire participants to learn more.

In order to determine LACE's effectiveness in engaging audiences in blockchain education, we decided to conduct an evaluation of LACE during demonstrations.

5.1 Pre-Assessment: Single Node LACE

The final prototype node of LACE was exhibited in the Moss Arts Center at Virginia Tech for the 2021 Virginia Tech Science Festival on November 11^{th}, 2021. The focus of the festival was to promote science and creative thinking to young audiences. Our participants consisted of over 100 young students ranging from ages 7 to 9.

This early study was done to help the researchers practice their demonstration of LACE to audiences and evaluate what changes may need to be made. While we did not take formal evaluations, we did use this event to help us better prepare and evaluate what changes may need to be made for future presentations. In dealing with a younger audience, adjustments had to be made to our original description of the LACE project. We focused on explaining the game of the project to the audience and emphasized the process of guessing a number in order to win. The students were able to grasp the concept of a guessing game fairly quickly and even attempted to guess along with LACE.

We concluded, through our demonstration, that the proximity of LACE to audiences may be an issue as many participants were inclined to touch or grasp some of LACE's mechanisms. In addition to the final prototype, we also brought the Mark I prototype to show the progression of our work. During the exhibit, there were issues with participants attempting to reach into the LACE prototype. Unfortunately, this did result in some damage to our Mark I prototype. While these kinds of issues are more likely with a younger audience, we made note of creating distance between viewers and the structure for future presentations.

Attendees also had difficulty viewing the inside of the node structure from a distance. Only about 1-3 people were able to properly view the screens of the tablets during a presentation. The acrylic structure of the prototype did help visibility for those not as close to the structure. While the singular node was not designed for large audiences, we noted this issue for future expansive designs.

5.2 Evaluations of LACE at Scale

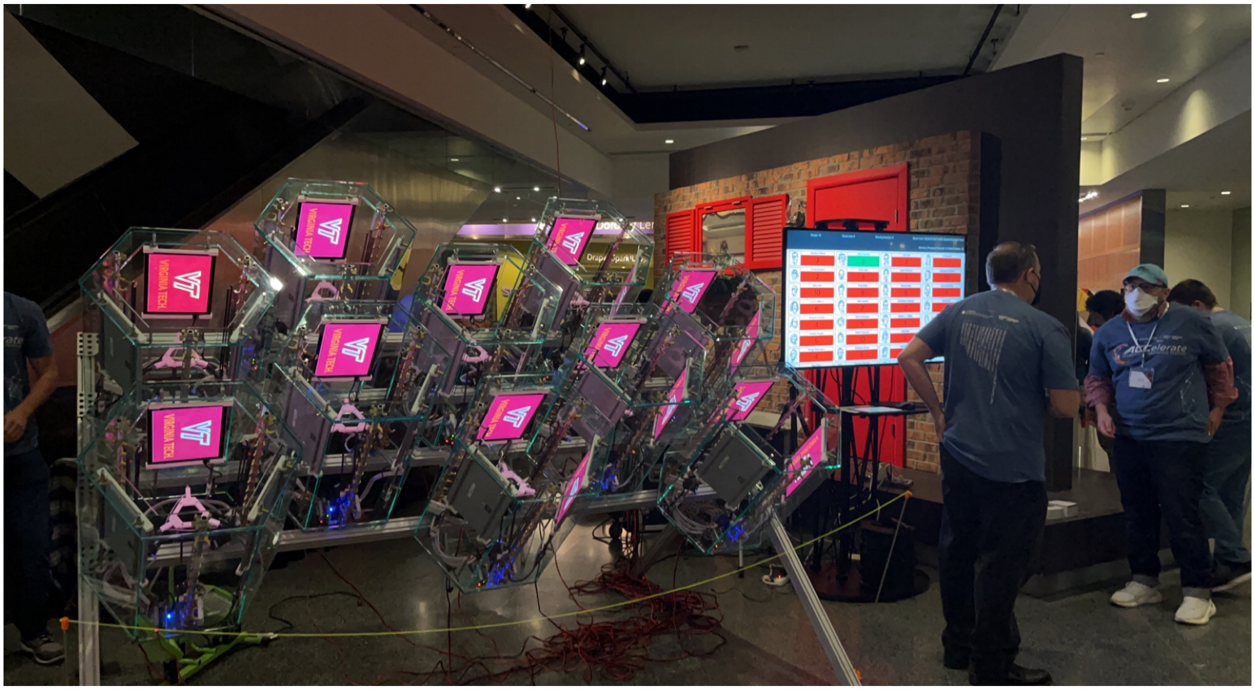


Figure 5.1: LACE demonstration setup during the 2022 ACCelerate Festival at the Smithsonian Museum of American History in Washington, D.C.

LACE officially debuted at the Smithsonian National Museum of American History during the 2022 ACCelerate Festival from April 8th-April 10th in Washington, D.C. This festival was hosted by Virginia Tech's Institute for Creativity, Arts, and Technology and the Smithsonian's Lemelson Center for the Study of Invention and Innovation. 12 institutes across the Atlantic Coast Conference (ACC) came together and presented various projects that focused on science, engineering, history and creativity.

Exhibit audience members viewed a short 1-3 minute live demonstration of LACE along with a verbal description of LACE's work conducted by researchers. Figure 5.1 shows the setup of the demonstration, with the LACE sculpture and a wide screen TV to the right of the structure, showing the Display page with the playing clients. Each presentation was followed by a short Q&A session with members of the research team. Following the demonstration, participants were also asked to complete a survey to gauge their experience with LACE and their overall understanding of blockchain and bitcoin mining concepts. Audience members who played alongside LACE through our interactive setup were also prompted to participate in our survey. Table 5.1 shows the survey questions that were provided to both groups of participants and how each question correlated to our objectives. The questions and objectives for this survey were sampled from the study of LACE's successor, SeeMore [44].

Table 5.1: ACCelerate Festival 2022 Survey Questions. Questions were provided to audience members at the 2022 Accelerate Festival in the Smithsonian Museum of American History, April 8th - 10th, 2022 (31,500 attendees) . Participants were asked to fill out surveys after a short 1-3 minute demonstration and Q&A with the LACE design team. *O1 = Entertain and educate all audiences on blockchain concepts; O2 = Personalize the importance of blockchain and cryptocurrency education; O3 = Inspire Participants to learn more.*

Questions	**LACE Project Objectives**		
	O1	O2	O3
1. Before interacting with LACE, how familiar were you with the concept of blockchain?		X	
2. After interacting with LACE, how would you rate your level of understanding with blockchain?	X	X	
3. Indicate whether you (1) strongly disagree, (2) disagree, (3) neutral, (4) agree or (5) strongly agree with the following statements:			
a. I now better understand how things work in blockchain. (1)(2)(3)(4)(5)	X		
b. I would like to learn more about blockchain. (1)(2)(3)(4)(5)			X
c. Having background knowledge of computer science concepts (e.g. cryptocurrency) is valuable in and of itself. (1)(2)(3)(4)(5)		X	X

d. I learned something new from LACE. (1)(2)(3)(4)(5))	X		
e. I enjoyed my interaction with LACE. (1)(2)(3)(4)(5)	X		
f. I would like to see future exhibits with LACE. (1)(2)(3)(4)(5)	X		
g. LACE would get students interested in STEAM careers (Science, Technology, Education, Arts and Mathematics). (1)(2)(3)(4)(5)	X	X	X
4. I was interested to interact with LACE because (Select one or more options)			
a. I like to learn about computer related topics		X	X
b. I did not know what blockchain was	X		X
c. The sculpture looked cool and I just wanted to know what it is	X		X
d. I was with a friend and came along	X		
5. Tell us what you learned or would like to share about LACE.	X	X	X

5.3 Analysis of ACCelerate Festival Interviews and Survey

The ACCelerate Festival was estimated to have over 31,500 attendees over the course of three days. We estimated that 10,000 of those participants attended our exhibit. Of these, 102 participated in our survey. A summary of the survey results is provided in Table 5.2. The participating audience for the ACCelerate Festival has a relatively equal division of ages, skewing towards a younger audience with 22.5% of participants being below the age of 18 [Table 5.2, Q1] and 26.5% being current K-12 students[Table 5.2, Q2]. About 70% of all participants had some form of higher education, ranging from some college education, a bachelors degree, master's or Ph.D.

The overall division of the survey resulted with 69.6% of the audience identifying as white, 10.8% as Asian, 6.9% as Hispanic or Latino, 5.9% as Black or African American, and the remainder as either Biracial or Native Hawaiian or Pacific Islander[Table 5.2, Q4]. Overall, 47.1% of the audience identified as female, and 46.1% as male[Table 5.2, Q3].

Table 5.2: ACC 2022 Survey Results. Survey summary results for 102 participants from the 2022 Accelerate Festival at the Smithsonian of American History, Washington, D.C., April 8-10, 2022 (31,500 attendees). Participants were asked to fill out surveys after a short 1-3 minute demonstration and Q&A with the LACE team.

[102 PARTICIPANTS SURVEYED]								
Q1. What age range do you fit into from the following?								
Below 18	**18-24**	**25-34**	**35-44**	**45-54**	**55-64**	**Above 65**		
23	15	12	13	14	14	11		
22.5%	14.7%	11.8%	12.7%	13.7%	13.7%	10.8%		
Q2. What is your highest degree or level of education you have completed?								
a.Current student K - 12; b. High School degree; c. Some College/Higher Education; d. Bachelors Degree; e. Masters Degree; f. Ph.D. or Higher; g. Trade School;								
a	**b**	**c**	**d**	**e**	**f**	**g**		
27	2	15	29	20	8	2		
26.5%	2.0%	14.7%	**28.4%**	19.6%	7.8%	2.0%		
Q3. To which gender identity do you most identify?								
Male	**Female**	**Non-binary**	**Prefer Not to Answer**					
47	48	3	4					
46.1%	**47.1%**	2.9%	3.9%					
Q4. What is your ethnicity?								
a. White or Caucasian; b.Hispanic or Latino; c. Black or African American; d. Asian; e. American Indian or Alaskan Native; f. Native Hawaiian or Pacific Islander; g. Multiracial or Biracial; h. Other (please specify)								
a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	
71	7	6	11	0	1	5	1	
69.6%	6.9%	5.9%	**10.8%**	0.0%	1.0%	4.9%	1.0%	
Q5. Before interacting with LACE, how familiar were you with the concept of blockchain?								
a. Not Familiar at All, b. Slightly Familiar, c. Somewhat Familiar, d. Moderately Familiar, e. Extremely Familiar								
a	**b**	**c**	**d**	**e**				
41	28	17	9	6				
40.6%	**27.7%**	16.8%	8.9%	5.9%				
Q6. After interacting with LACE, how would you rate your level of understanding with blockchain?								
a. Not Familiar at All, b. Slightly Familiar, c. Somewhat Familiar, d. Moderately Familiar, e. Extremely Familiar								
a	**b**	**c**	**d**	**e**				
7	15	34	28	14				
7.1%	15.3%	34.7%	28.6%	14.3%				
Q7. I now better understand how thing s work in blockchain.								
a. Strongly Disagree, b. Disagree, c. Neutral, d. Agree, e. Strongly Agree								
a	**b**	**c**	**d**	**e**				
6	11	26	39	15				
6.2%	11.3%	**26.8%**	**40.2%**	**15.5%**				

Continued on next page

Table 5.2 – *Continued from previous page*

Q8. I would like to learn more about blockchain.					
a. Strongly Disagree, b. Disagree, c. Neutral, d. Agree, e. Strongly Agree					
a	**b**	**c**	**d**	**e**	
4	7	15	39	31	
4.2%	7.3%	15.6%	**40.6%**	**32.3%**	
Q9. Having background knowledge of computer science concepts (e.g. cryptocurrency) is valuable in and of itself.					
a. Strongly Disagree, b. Disagree, c. Neutral, d. Agree, e. Strongly Agree					
a	**b**	**c**	**d**	**e**	
2	3	22	35	33	
2.1%	3.2%	23.2%	**36.8%**	**34.7%**	
Q10. I learned something new from LACE					
a. Strongly Disagree, b. Disagree, c. Neutral, d. Agree, e. Strongly Agree					
a	**b**	**c**	**d**	**e**	
4	4	17	33	37	
4.2%	4.2%	17.9%	**34.7%**	**38.9%**	
Q11. I enjoyed my interaction with LACE.					
a. Strongly Disagree, b. Disagree, c. Neutral, d. Agree, e. Strongly Agree					
a	**b**	**c**	**d**	**e**	
3	1	12	24	53	
3.2%	1.1%	12.9%	**25.8%**	**57.0%**	
Q12. I would like to see future exhibits with LACE.					
a. Strongly Disagree, b. Disagree, c. Neutral, d. Agree, e. Strongly Agree					
a	**b**	**c**	**d**	**e**	
2	2	11	34	43	
2.2%	2.2%	12.0%	**37.0%**	**46.7%**	
Q13. LACE would get students interested in STEAM careers (Science, Technology, Education, Arts and Mathematics).					
a. Strongly Disagree, b. Disagree, c. Neutral, d. Agree, e. Strongly Agree					
a	**b**	**c**	**d**	**e**	
2	3	7	33	49	
2.1%	3.2%	7.4%	**35.1%**	**52.1%**	
Q14. I was interested to interact with LACE because: (Select one or more options)					
a. I like to learn about computer related topics, b. I did not know what blockchain was, c. The sculpture looked cool and I wanted to know what it was, d. I was with a friend and came along, e. Other					
a	**b**	**c**	**d**	**e**	
51	28	38	12	4	
55.4%	30.4%	**41.3%**	13.0%	4.4%	

Prior to viewing our demonstration, over 67% of participants stated they were either “slightly familiar” or “not familiar at all” with the concept of blockchain[Table 5.2, Q5]. Following their interaction with LACE, over 76% of participants agreed they now had a “Somewhat to Extremely” familiar understanding with blockchain concepts[Table 5.2, Q6], which can be seen in Figure 5.2. In addition, over 55% of participants agreed that LACE provided a better understanding of blockchain concepts, and 73.6% had learned something new as well[Table 5.2, Q7,Q9]. These results directly support our first objective; to entertain and educate all audiences on blockchain concepts[O1]. Another focus of the LACE project is to affirm the importance of blockchain and cryptocurrency education to viewers. 72% of participants agreed that knowledge of cryptocurrency and computer science topics alone is very valuable [Table 5.2, Q8]. This data supports our objective of personalizing the importance of blockchain education to our participants [O2].

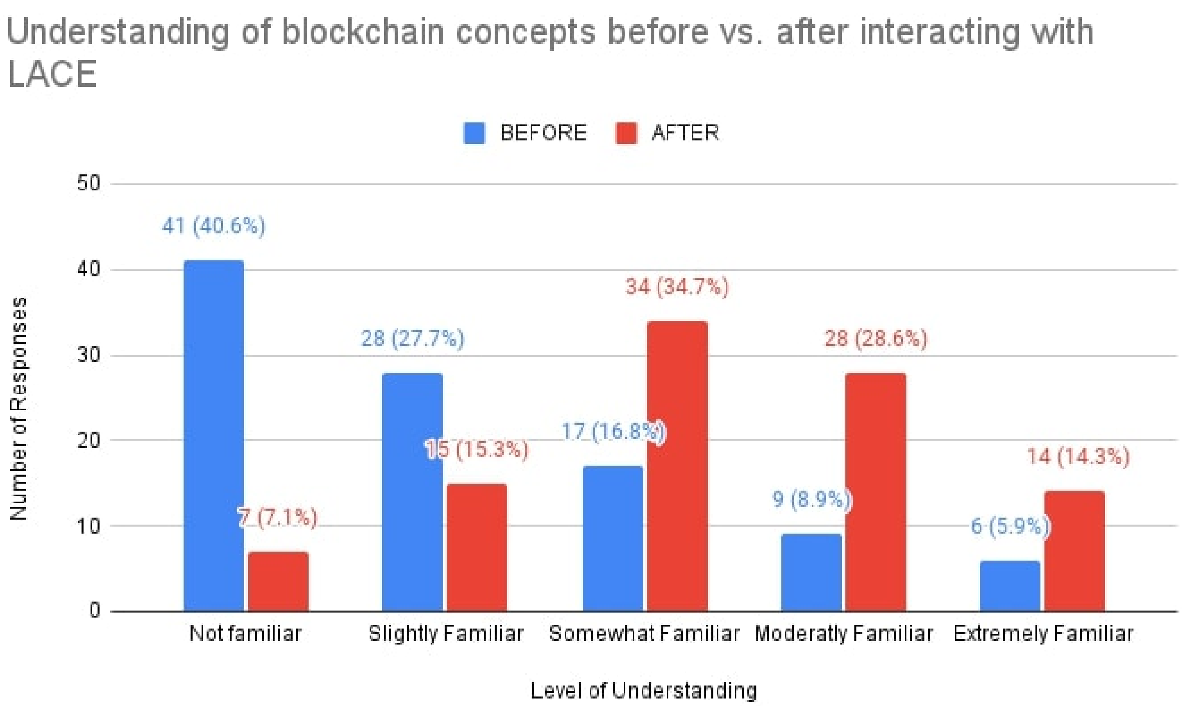


Figure 5.2: Comparison of participants understanding of blockchain concepts before vs. after interacting with the LACE sculpture.

We also want to encourage our viewers to leave wanting to learn more about cryptocurrency and its value today. 82% “agreed or strongly agreed” that they enjoyed their interaction with the LACE structure[Table 5.2, Q10] and over 83% of participants stated they were also interested in seeing future exhibits of LACE [Table 5.2, Q10]. Survey results also indicate that 72% of viewers were inspired to learn more [O3] about blockchain during their visit with LACE, supporting our third project objective[Table 5.2, Q7]. Importantly, over 87% of participants agreed that LACE would be an effective tool in engaging students' interest in STEAM careers [Table 5.2, Q11]. This indicates that the use of LACE within education would benefit younger audiences and be a good introduction into such abstract concepts, supporting our third and final objective.

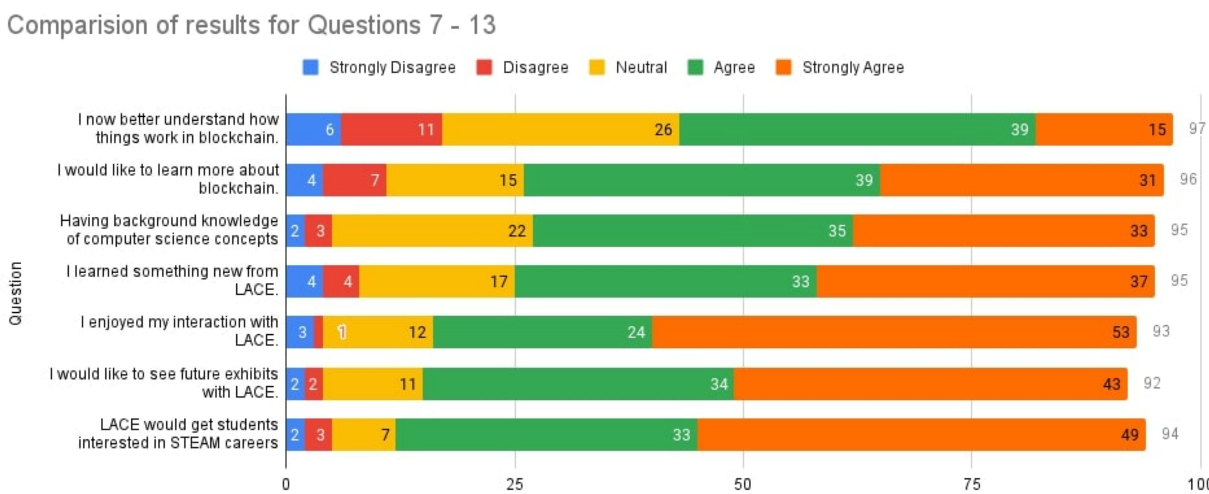


Figure 5.3: Comparison of survey results for questions 7 - 13.

Participants were asked what interested them to view LACE during their visit. 55.4% stated they liked learning about computing related topics, while 30% stated they had little knowledge of blockchain at all. Even with different demographics of overall knowledge on blockchain, LACE was able to capture the attention of an array of participants and engage them in a fun interactive experience. LACE's focus on computing was not the only point of intrigue for viewers. 41.3% of participants stated they were intrigued to view LACE because it "looked cool" and were interested to understand what it was [Table 5.2, Q13]. One participant also stated that they liked "interactive art and tech", which led them to view our display. This shows that the use of art and creativity in education is a great tool to introduce advanced topics and spark interest in complex fields.

Participants were also given the opportunity to give comments about their experience learning about LACE (see Table 5.1). There were 26 responses to this question from participants from a diverse set of backgrounds. Table 5.3 shows a record of a portion of the recorded responses from participants. Participants were able to personalize their experience with LACE to their own education, commenting that the demonstration "gave me a solid background of how blockchain functions"[O2]. Subjects also expressed interest in the artistic side of LACE, stating "It was fascinating to see a visual and artistic representation of distributed ledger technology". Some participants also offered suggestions for future exhibits of LACE. "I would love to see a DNS blockchain example", one participant stated. Another subject discussed the possibilities of discussing and visualizing potential climate related issues with mining through LACE. These interactions provide evidence to support our projects goal of entertaining and educating audiences [O1] while also inspiring them to learn more about these concepts [O3].

Table 5.3: ACCelerate Festival 2022 Survey Question. Participants were asked to provide additional comments about what they learned or would like to share about their interaction with LACE. Quotes were evaluated in relation to our project objectives. *O1 = Entertain and educate all audiences on blockchain concepts; O2 = Personalize the importance of blockchain and cryptocurrency education; O3 = Inspire Participants to learn more.*

	Q: Tell us what you learned or would like to share about LACE.	**LACE Project Objectives**		
		O1	O2	O3

Age 35 - 44, Male, Black or African American, Bachelors Degree	"It was fascinating to see a visual and artistic representation of distributed ledger technology."	X		
Below 18, Female, White or Caucasian, Current student K - 12	"Great interaction of art and tech. Very digestible!"	X	X	
45 - 54, Female, White or Caucasian, Masters Degree	"Super interesting and cool to look at!"	X		
25 - 34 ,Female, Hispanic or Latino, Bachelor's Degree	"Really great display, gave me a solid background of how blockchain functions. Can educating the public in this way help build movements to counteract/bring awareness to climate related issues with mining. Really cool!"	X	X	
45 - 54, Male, White or Caucasian, Master's Degree	"I would love to see a DNS blockchain example."		X	X

Above 65, Male, Asian, Some College/ Higher Education	“Great layman understanding visually of bitcoin and blockchain.”	X	X	
45 - 54, Female, White or Caucasian, Female, Bachelor’s Degree	“Love the name - always interested to see women in science/math to be “represented””	X	X	
55 - 64, Male, Hispanic or Latino, Master’s Degree	“Wonderful presentation!”	X		

Chapter 6

Conclusions

6.1 Conclusion

In this paper, we describe the design, implementation and evaluation of the kinetic sculpture and decentralized ledger, LACE. LACE consists of a network of Microsoft Surface tablets, functioning as nodes within a blockchain network working to mine for new blocks and maintain the blockchain itself. We represent the computing work of these nodes through physical movement between 10 kinetic units, using delta-robot arms to interact with an array of tablets in order to mine a block. We simplify the process of bitcoin mining into a simple number guessing game, using interactive software interfaces on our tablets to demonstrate the work of an individual miner and the competition between the network. The aim of this work was to entertain and educate audiences on blockchain topics; personalize the importance of blockchain and cryptocurrency education; and inspire viewers to learn more. We evaluated these objectives during our debut demonstration of LACE through a detailed questionnaire at the 2022 ACCelerate Festival in Washington DC. From approximately 10,000 viewers of LACE, we surveyed 102 participants who stated an overwhelmingly positive experience with the kinetic model. The results of our data show that we exceeded our objectives in educating and inspiring our audience to learn about blockchain, bitcoin mining and their importance. Not only were participants enamored by the beauty and creativity of the LACE sculpture, but also developed a better understanding of blockchain concepts with over 76%

of participants stating LACE provided them with a better understanding of blockchain technologies. LACE also inspired over 72% of participants to learn more about blockchain and 87% agreed LACE would be an effective tool to introduce students to careers in STEAM fields. This work demonstrates the effectiveness of kinetic visualizations as a tool for imparting knowledge of complex topics to broad audiences. We look forward to expanding on the scale of LACE and introducing LACE to even larger audiences, and hope to encourage others to expand their knowledge on computer science topics.

6.2 Future Work

We were unable to implement an interactive Twitter application with the LACE structure due to developer restrictions and time constraints. While we were not given explicit details as to why our use case was denied during the application process, we attribute the issues with our plans of automated messaging and potential issues with consent and permissions of users' information being used. Future work on this project should look to create a solution which adheres to the Twitter Developer guidelines while maintaining the nature of our game. This functionality will make the game more accessible to those not directly with LACE and also emphasize the role of distributed computing in blockchain.

We were also limited in the amount of external connections that could connect to our network during game play. Because this issue occurred during a live demonstration, we were unable to do an in-depth investigation into the cause. We speculate that this occurred because of our network's inability to handle more than the 40 connections needed for the LACE structure. For future presentations, we look to upgrade to a more powerful routing system that would allow for individuals to interact with LACE through their phones.

We also look to expand the size and scale of the LACE structure to include more modules.

Because of LACE's design, there are no bounds to the shape and form of the overall structure. We are considering creating more fantastic dome or hive-like structures to emphasize the beauty of LACE as a sculpture, while also demonstrating the large scale of these blockchain networks and their consistent growth.

In the future, we will also consider introducing LACE into educational curriculum and help engage students in further understanding blockchain technologies. LACE could also be used to help students explore the bridge between art and engineering, and encourage the creation of even more creative and innovative computing projects like this one.

Bibliography

[1] [n.d.]. About Heroku. Retrieved July 8, 2022 from https://www.heroku.com/about

[2] [n.d.]. About Node.Js. Retrieved July 18, 2022 from https://nodejs.org/en/about/

[3] [n.d.]. Bitcoin. Retrieved June 8, 2022 from https://www.coindesk.com/price/bitcoin/

[4] [n.d.]. Cambridge Bitcoin Electricity Consumption Index (CBECI). https://ccaf.io/cbeci/index

[5] [n.d.]. Leading cryptocurrency platform for mining and trading. https://www.nicehash.com/blog/post/what-are-accepted-and-rejected-shares

[6] [n.d.]. Node.js web application framework. Retrieved July 20, 2022 from https://expressjs.com/

[7] 2021. All aboard the crypto train - covid-19 trend report. https://www.cardify.ai/reports/crypto

[8] 2022. Consensus algorithms in Blockchain. https://www.geeksforgeeks.org/consensus-algorithms-in-blockchain/#:~:text=A%20consensus%20algorithm%20is%20a,state%20of%20the%20distributed%20ledger.

[9] 2022. Twitter bot. https://en.wikipedia.org/wiki/Twitter_bot

[10] Joe Abou Jaoude and Raafat George Saade. 2019. Blockchain Applications – Usage in Different Domains. *IEEE Access* 7 (2019), 45360–45381. https://doi.org/10.1109/ACCESS.2019.2902501

[11] Samy Abu-Naser. 2008. Developing Visualization Tool for Teaching AI Searching Algorithms. *Information Technology Journal* 7 (02 2008). https://doi.org/10.3923/itj.2008.350.355

[12] Abhinav Ajitsaria. 2021. What is the python global interpreter lock (gil)? https://realpython.com/python-gil/

[13] Shivam Arora. 2022. What is Bitcoin Mining: How Does it work, proof of work and more: Simplilearn. https://www.simplilearn.com/bitcoin-mining-explained-article

[14] Austin Cory Bart, Ryan Whitcomb, Dennis Kafura, Clifford A Shaffer, and Eli Tilevich. 2017. Computing with corgis: Diverse, real-world datasets for introductory computing. *ACM Inroads* 8, 2 (2017), 66–72.

[15] Martin Braund and Michael Reiss. 2019. The 'Great Divide': How the Arts Contribute to Science and Science Education. *Canadian Journal of Science, Mathematics and Technology Education* 19 (06 2019), 219–236. https://doi.org/10.1007/s42330-019-00057-7

[16] Erik Brunvand. 2013. Arts/tech collaboration with embedded systems and kinetic art. https://doi.org/10.1145/2504459.2504488

[17] Erik Brunvand and Paul Stout. 2011. Kinetic Art and Embedded Systems: A Natural Collaboration *(SIGCSE '11)*. Association for Computing Machinery, New York, NY, USA, 323–328. https://doi.org/10.1145/1953163.1953263

[18] Tomislav Capan. 2013. Why the hell would I use node.js? A case-by-case tutorial. https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js

[19] Miguel Castro and Barbara Liskov. 2002. Practical Byzantine Fault Tolerance and

Proactive Recovery. *ACM Trans. Comput. Syst.* 20, 4 (nov 2002), 398–461. https://doi.org/10.1145/571637.571640

[20] Chan-Jin Chung. 2014. Integrated STEAM education through global robotics art festival (GRAF). *ISEC 2014 - 4th IEEE Integrated STEM Education Conference*, 1–6. https://doi.org/10.1109/ISECon.2014.6891011

[21] Coinbase. 2019. The rise of crypto in Higher Education. https://blog.coinbase.com/the-rise-of-crypto-in-higher-education-81b648c2466f

[22] Andy Connor, Sangeeta Karmokar, and Chris Whittington. 2015. From STEM to STEAM: Strategies for Enhancing Engineering Technology Education. *International Journal of Engineering Pedagogies* 5 (05 2015), 37–47. https://doi.org/10.3991/ijep.v5i2.4458

[23] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. 2011. From Game Design Elements to Gamefulness: Defining Gamification. *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek 2011* 11, 9–15. https://doi.org/10.1145/2181037.2181040

[24] Walter Dettling and Bettina Schneider. 2020. Bloxxgame–A simulation game for teaching blockchain. In *International Conference on Games and Learning Alliance*. Springer, 169–178.

[25] Monika Akbar Eric Fouh and Clifford A Shaffer. 2012. The role of visualization in computer science education. *Computers in the Schools* 29, 1-2 (2012), 95–117.

[26] Tulie Finley-Moise. 2019. What are the benefits of using a touch screen pen (or stylus)?: HP® Tech takes. https://www.hp.com/us-en/shop/tech-takes/

benefits-of-using-touch-screen-pen#:~:text=Capacitive%20stylus,contact%20via%20electrostatic%20field%20disruption.

[27] Eric Fouh, Ville Karavirta, Daniel A. Breakiron, Sally Hamouda, Simin Hall, Thomas L. Naps, and Clifford A. Shaffer. 2014. Design and architecture of an interactive eTextbook – The OpenDSA system. *Science of Computer Programming* 88 (2014), 22–40. https://doi.org/10.1016/j.scico.2013.11.040.

[28] Jake Frankenfield. 2022. What is a 51% attack? https://www.investopedia.com/terms/1/51-attack.asp

[29] Xun Ge, Dirk Ifenthaler, and J Michael Spector. 2015. Moving forward with STEAM education research. *Emerging technologies for STEAM education* (2015), 383–395.

[30] Julija Golosova and Andrejs Romanovs. 2018. The Advantages and Disadvantages of the Blockchain Technology. In *2018 IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE)*. 1–6. https://doi.org/10.1109/AIEEE.2018.8592253

[31] Fabian Groh. 2012. Gamification: State of the art definition and utilization. *Institute of Media Informatics Ulm University* 39 (2012), 31.

[32] Stuart Haber and W Scott Stornetta. 1990. How to time-stamp a digital document. In *Conference on the Theory and Application of Cryptography*. Springer, 437–455.

[33] Sebastian Hahn, Matthias Trapp, Nikolai Wuttke, and Jürgen Döllner. 2015. Thread City: Combined Visualization of Structure and Activity for the Exploration of Multi-threaded Software Systems. In *2015 19th International Conference on Information Visualisation*. 101–106. https://doi.org/10.1109/iV.2015.28

[34] Simin Hall, Clifford A Shaffer, Eric Fouh, Mai Hassan ElShehaly, and Daniel Breakiron. 2013. Evaluating online tutorials for data structures and algorithms courses. In *2013 ASEE Annual Conference & Exposition*. 23–549.

[35] Mohamed Tahar Hammi, Badis Hammi, Patrick Bellot, and Ahmed Serhrouchni. 2018. Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. *Computers & Security* 78 (2018), 126–142.

[36] Anne Harris and Leon de Bruin. 2017. STEAM Education: Fostering creativity in and beyond secondary schools. *Australian art education* 38, 1 (2017), 54–75.

[37] Ian Hickson and Michael Carter. [n.d.]. The Websocket API. https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API

[38] Binbing Hou and Feng Chen. 2020. A Study on Nine Years of Bitcoin Transactions: Understanding Real-world Behaviors of Bitcoin Miners and Users. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. 1031–1043. https://doi.org/10.1109/ICDCS47774.2020.00091

[39] Heng Hou. 2017. The application of blockchain technology in E-government in China. In *2017 26th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 1–4.

[40] Paul Howard-Jones, Wayne Holmes, Skevi Demetriou, Carol Jones, Eriko Tanimoto, Owen Morgan, David Perkins, and Neil Davies. 2014. Neuroeducational research in the design and use of a learning technology. *Learning, Media and Technology* 0 (09 2014), 1–20. https://doi.org/10.1080/17439884.2014.943237

[41] Kyoungjun Jee and Eunju Hong. 2015. “The Development and Application of STEAM

Education Program based on Kinetic Art,”. *Journal of Korean Elementary Science Education* 34 (2015), 276–287.

[42] Gabriela Kiryakova, Nadezhda Angelova, and Lina Yordanova. 2014. Gamification in education. Proceedings of 9th International Balkan Education and Science Conference. https://doi.org/10.4018/978-1-5225-5198-0

[43] Shelley Latham, Lion, Gearge Garton, and Iteea Account. [n.d.]. Defining stem. https://www.linkengineering.org/Explore/LE_Blog/58003.aspx#:~:text=STEM%20weaves%20those%20elements%20of,oF%20education%2C%E2%80%9D%20Ramaley%20said.

[44] Bo Li, John Mooring, Sam Blanchard, Aditya Johri, Melinda Leko, and Kirk W Cameron. 2017. SeeMore: A kinetic parallel computer sculpture for educating broad audiences on parallel computation. *J. Parallel and Distrib. Comput.* 105 (2017), 183–199.

[45] Xueping Liang, Juan Zhao, Sachin Shetty, Jihong Liu, and Danyi Li. 2017. Integrating blockchain for data sharing and collaboration in mobile healthcare applications. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. 1–5. https://doi.org/10.1109/PIMRC.2017.8292361

[46] Christine Liao. 2016. From interdisciplinary to transdisciplinary: An arts-integrated approach to STEAM education. *Art Education* 69, 6 (2016), 44–49.

[47] Arunmozhi Manimuthu, Raja Sreedharan V., Rejikumar G., and Drishti Marwaha. 2019. A Literature Review on Bitcoin: Transformation of Crypto Currency Into a Global Phenomenon. *IEEE Engineering Management Review* 47, 1 (2019), 28–35. https://doi.org/10.1109/EMR.2019.2901431

[48] Arunmozhi Manimuthu, Raja Sreedharan V., Rejikumar G., and Drishti Marwaha. 2019. A Literature Review on Bitcoin: Transformation of Crypto Currency Into a Global Phenomenon. *IEEE Engineering Management Review* 47, 1 (2019), 28–35. https://doi.org/10.1109/EMR.2019.2901431

[49] Ahmed Afif Monrat, Olov Schelén, and Karl Andersson. 2019. A Survey of Blockchain From the Perspectives of Applications, Challenges, and Opportunities. *IEEE Access* 7 (2019), 117134–117151. https://doi.org/10.1109/ACCESS.2019.2936094

[50] Barbara Moskal, Deborah Lurie, and Stephen Cooper. 2004. Evaluating the Effectiveness of a New Instructional Approach. *SIGCSE Bull.* 36, 1 (mar 2004), 75–79. https://doi.org/10.1145/1028174.971328

[51] Besim Mustafa. 2013. YASS: A system simulator for operating system and computer architecture teaching and learning. *European Journal of Science and Mathematics Education* 1 (01 2013), 34–42. https://doi.org/10.30935/scimath/9385

[52] Fiona Fui-Hoon Nah, Qing Zeng, Venkata Rajasekhar Telaprolu, Abhishek Padmanabhuni Ayyappa, and Brenda Eschenbrenner. 2014. Gamification of education: a review of literature. In *Proceedings of the International conference on hci in business*. Springer, 401–409.

[53] Satoshi Nakamoto. 2009. Bitcoin : A Peer-to-Peer Electronic Cash System. *Decentralized Business Review* (2009), 21260.

[54] Yustus Eko Oktian, Ivan Kristianto Singgih, and Friska Natalia Ferdinand. 2019. Serious Game for Blockchain Education Purposes (using Proof-of-Work Consensus of Bitcoin). In *2019 5th International Conference on New Media Studies (CONMEDIA)*. 177–183. https://doi.org/10.1109/CONMEDIA46929.2019.8981820

[55] Manisa Pipattanasomporn, Murat Kuzlu, and Saifur Rahman. 2018. A Blockchain-based Platform for Exchange of Solar Energy: Laboratory-scale Implementation. In *2018 International Conference and Utility Exhibition on Green Energy for Sustainable Development (ICUE)*. 1–9. https://doi.org/10.23919/ICUE-GESD.2018.8635679

[56] Apan Qasem. 2022. YODA: A Pedagogical Tool for Teaching Systems Concepts. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 1* (Providence, RI, USA) *(SIGCSE 2022)*. Association for Computing Machinery, New York, NY, USA, 613–618. https://doi.org/10.1145/3478431.3499322

[57] Guillermo Rauch. [n.d.]. Socket.io. Retrieved July 27,2022 from https://socket.io/

[58] Siti Rosminah, siti rosminah md derus, Ahmad Zamzuri Mohamad Ali, and Mohamad Ali. 2012. Difficulties in learning programming: Views of students. https://doi.org/10.13140/2.1.1055.7441

[59] Simanta Shekhar Sarmah. 2018. Understanding blockchain technology. *Computer Science and Engineering* 8, 2 (2018), 23–29.

[60] Pete Shinners. 2011. PyGame. http://pygame.org/

[61] Ken Shrriff. [n.d.]. Bitcoin mining the hard way: The algorithms, protocols, and bytes. http://www.righto.com/2014/02/bitcoin-mining-hard-way-algorithms.html

[62] SoFi. 2022. Bitcoin price history: Price of bitcoin 2009 - 2022. https://www.sofi.com/learn/content/bitcoin-price-history/

[63] Edward Sykes. 2007. Determining the Effectiveness of the 3D Alice Programming Environment at the Computer Science I Level. *Journal of Educational Computing Research - J EDUC COMPUT RES* 36 (05 2007), 223–244. https://doi.org/10.2190/J175-Q735-1345-270M

[64] Pinyaphat Tasatanattakool and Chian Techapanupreeda. 2018. Blockchain: Challenges and applications. In *2018 International Conference on Information Networking (ICOIN)*. 473–475. https://doi.org/10.1109/ICOIN.2018.8343163

[65] Tate. [n.d.]. 'kinetic construction (standing wave)', Naum Gabo, 1919–20, replica 1985. https://www.tate.org.uk/art/artworks/gabo-kinetic-construction-standing-wave-t00827

[66] Eric Terpstra. 2021. Building multiplayer games with Node.js and Socket.io. https://modernweb.com/building-multiplayer-games-node-js-socket-io/

[67] Jonas Trümper, Alexandru Telea, and Jürgen Döllner. 2012. ViewFusion: Correlating Structure and Activity Views for Execution Traces. In *Theory and Practice of Computer Graphics*, Hamish Carr and Silvester Czanner (Eds.). The Eurographics Association. https://doi.org/10.2312/LocalChapterEvents/TPCG/TPCG12/045-052

[68] Florian Tschorsch and Björn Scheuermann. 2016. Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. *IEEE Communications Surveys Tutorials* 18, 3 (2016), 2084–2123. https://doi.org/10.1109/COMST.2016.2535718

[69] Qiang Wang and Min Su. 2020. Integrating blockchain technology into the energy sector—from theory of blockchain to research and application of energy blockchain. *Computer Science Review* 37 (2020), 100275.

[70] Andrew D Watson and Gregory H Watson. 2013. Transitioning STEM to STEAM: Reformation of engineering education. *Journal for Quality and Participation* 36, 3 (2013), 1–5.

[71] Robert L Williams. 2016. The delta parallel robot: kinematics solutions. *Mechanical Engineering, Ohio University* (2016), 1–15.

[72] Jeong Yang, Young Lee, David Hicks, and Kai H Chang. 2015. Enhancing object-oriented programming education using static and dynamic visualization. In *2015 IEEE Frontiers in Education Conference (FIE)*. IEEE, 1–5.

[73] Barış Yılmaz. 2014. Art engineering and kinetic art. *Journal of Arts and Humanities* 3, 12 (2014), 16–21.

[74] Bessem Zaabar, Omar Cheikhrouhou, Faisal Jamil, Meryem Ammi, and Mohamed Abid. 2021. HealthBlock: A secure blockchain-based healthcare data management system. *Computer Networks* 200 (2021), 108500.

Appendices

Appendix A

Delta Robot Measurement parameters

Prismatic-Input Delta Robot Parameters

name	meaning
s_b	base equilateral triangle side
w_b	planar distance from $\{0\}$ to near base side
u_b	planar distance from $\{0\}$ to a base vertex
H	frame height
S_B	P joints (B_i) equilateral triangle side
W_B	same as w_b, for P joints equilateral triangle
U_B	same as u_b, for P joints equilateral triangle
s_P	platform equilateral triangle side
w_P	planar distance from $\{P\}$ to near platform side
u_P	planar distance from $\{P\}$ to a platform vertex
o_X	nozzle X offset
o_Y	nozzle Y offset
L_{min}	i = 1,2,3 minimum prismatic joints lengths
L_{max}	i = 1,2,3 maximum prismatic joints lengths
l	lower legs parallelogram length
h	lower legs parallelogram width

Figure A.1: Parameters for Prismatic-Input Delta Robot measurements

Appendix B

Text of Survey

ACCelerate Festival 2022 LACE Survey

1. What is your age?

☐ Below 18 ☐ 18-24 ☐ 25-34 ☐ 35-44 ☐ 45-54 ☐ 55-64 ☐ 65+

2. What is the highest degree or level of education you have completed?

☐ Current student K-12 ☐ High school degree

☐ Some College/Higher Education ☐ Bachelor's Degree

☐ Masters' Degree ☐ Ph.D. or higher

☐ Trade School ☐ Other (please describe): ________________________

3. To which gender identity do you identify?

☐ Male ☐ Female ☐ Non-Binary

☐ Prefer Not To Answer ☐ Other (please specify):__________________________

3. What is your ethnicity?

☐ White or Caucasian ☐ Black/African American ☐ Asian

☐ Hispanic or Latino ☐ Native Hawaiian or Pacific Islander ☐ Bi-racial

☐ American Indian or Alaska Native ☐ Other (please specify):__________________________

<u>For questions 5-12 please indicate your level of agreement by checking one box.</u>

5. Before interacting with LACE, how familiar were you with the concept of blockchain?

☐ Not familiar at all ☐ Slightly Familiar ☐ Somewhat familiar

☐ Moderately familiar ☐ Extremely familiar

6. After interacting with LACE, how would you rate your level of understanding?

☐ Not familiar at all ☐ Slightly Familiar ☐ Somewhat familiar

☐ Moderately familiar ☐ Extremely familiar

6. I now better understand how things work in blockchain.

☐ Strongly disagree ☐ Disagree ☐ Neutral ☐ Agree ☐ Strongly agree

7. I would like to learn more about blockchain.

☐ Strongly disagree ☐ Disagree ☐ Neutral ☐ Agree ☐ Strongly agree

8. Having background knowledge of computer science concepts (e.g cryptocurrency) is valuable in and of itself.

☐ Strongly disagree ☐ Disagree ☐ Neutral ☐ Agree ☐ Strongly agree

9. I learned something new from LACE.

☐ Strongly disagree ☐ Disagree ☐ Neutral ☐ Agree ☐ Strongly agree

10. I enjoyed my interaction with LACE.

☐ Strongly disagree ☐ Disagree ☐ Neutral ☐ Agree ☐ Strongly agree

11. I would like to see future exhibits with LACE.

☐ Strongly disagree ☐ Disagree ☐ Neutral ☐ Agree ☐ Strongly agree

12. LACE would get students interested in STEAM (Science, Technology, Engineering, Art, and Mathematics) careers.

☐ Strongly disagree ☐ Disagree ☐ Neutral ☐ Agree ☐ Strongly agree

13. Please select which of the following statements apply to your experience with LACE.

I was interested to interact with LACE because.

☐ I like to learn about computer related topics.

☐ I did not know what blockchain was.

☐ The sculpture looked cool and I just wanted to know what it is.

☐ I was with a friend and came along.

☐ Other (please explain):____

____

14. Please use the space below to share any additional comments or questions you have about LACE.

____

____

____

Thank you for your participation!