Worcester IS Fab Lab MQP Blockchain and Machine Learning Report

Project Team:

Kelsey Moody, kjmoody@wpi.edu
Mabel Konadu, mkonadu@wpi.edu
Qingbei Shang, qshang@wpi.edu
Cameron Morreale, cmorreale@wpi.edu
Joshua DeBare, jedebare@wpi.edu

Project Advisors

Professor Marcel Blais
Department of Mathematical Sciences
Professor Wilson Wong
Department of Computer Science
Professor Robert Sarnie
Business School

Abstract

Today's FinTech industry combines technologies, such as machine learning, data processing and blockchain. In Worcester MA, the Worcester IS FAB Lab utilizes blockchain and machine learning (ML) techniques to assist internet of things (IoT)-device based start-ups and support social good. The team delivered three different products to sponsors. First, the team utilized ML to help smart energy start-up Embue predict when apartment tenants opened their windows, saving energy by turning off the heat. The team also used ML to predict pavement deterioration for Cyvl, decreasing accidents associated with poor pavement conditions. Last, the team used blockchain to verify IoT devices for Embue.

Executive Summary

Our team worked with our main sponsor, the Worcester IS Fab Lab, to implement machine learning and blockchain to promote social good within two start ups: Cyvl.ai and Embue. Cyvl uses IoT devices and artificial intelligence to determine a road's pavement condition index (PCI) score. Embue uses IoT devices to monitor temperature, humidity, and more in large apartment complexes.

For Cyvl, we created a linear regression model to predict pavement deterioration as a pavement condition index. To do so, we combined data from Cyvl, Massachusetts Department of Transportation, and BETA Group Inc. We then cleaned and organized that data, and implemented both a correlation test and a linear regression model. We tested the accuracy of the model, and determined it had an r-squared value of 0.434. We suggested ways for Cyvl to improve this score, and what data may be relevant to its improvement. Cyvl can use the regression model to help their customers better allocate money for future road repairs. This proactivity prevents roads from falling into complete disrepair, which keeps drivers of these roads safe. This promotes social good in the areas that Cyvl operates in.

For Embue, we implemented an algorithm to determine when units have open windows in residential apartment complexes. We spent significant time cleaning the data, as sensor records with different timestamps needed to be grouped together. After, the team implemented clustering to find units with open windows, by analyzing multiple factors. Finally, the team investigated an approximation of a decision tree to filter readings that could potentially indicate an open window. If accurate, this program could save tenants money on heating and could increase energy efficiency in the building.

For Embue, we also explored how the immutability and decentralization of blockchain could improve the auditability of Embue's IoT network. We met with the head of security at Embue, and he expressed concern that if someone claimed the transfer of one of their devices occurred at an incorrect time, there isn't a good way to disprove this. During our research, we determined that blockchain based storage systems would be a good solution to this problem due to their decentralized nature and immutability. We explored the pros and cons of a wide variety of blockchain based storage systems, and determined Filecoin would be useful for our proof of concept due to its low energy consumption and minimal storage fees. Filecoin differs from a traditional non-blockchain storage system because they have multiple different storage providers storing the data, rather than storing a bunch of files on a centralized network. Filecoin rewards these providers with Filecoin tokens. Additionally, Filecoin uses a cryptographic proof of storage to ensure that your data stays intact over time. Using Filecoin, if a malicious agent tries to claim the contents of the information were dispersed at an incorrect time, you can see the actual timestamp and disprove that. After much trial and error, we successfully gathered a small portion of Embue's sensor data and stored it on the Filecoin blockchain. There is still a lot more data, but we produced a proof of concept that will be a blueprint for both Cyvl and Embue to build on to facilitate auditability within their IoT network. Cyvl and Embue could also leverage the immutability of the data stored on the Filecoin network for machine learning in the future. They would be able to see the timestamps of when the sensor is on and off; and this will help determine what data to filter out to improve the accuracy of the machine learning algorithms used.

To ensure accountability and effectively organize our time throughout the seven week project, we used the Agile Scrum methodology, due its flexible structure. For our project management software, we used Jira. We also utilized GitHub, Colab, Jupyter Notebooks, PyCharm and more to complete this project.

Acknowledgment

We would first like to thank our advisors, Professor Blais, Professor Sarnie, and Professor Wong, for their guidance, support, and positivity throughout the last seven weeks.

We would also like to thank our main sponsor, the Worcester IS FAB Lab, and Professor Sarkis, for giving us the opportunity to work on this project and helping us learn and Professor Treku for providing additional support with blockchain. Additionally, we would like to thank Daniel Pelaez and the team at Cyvl, and both Marc Printz and Nathan Rosenberg along with the team at Embue, for their guidance and help throughout this project. Additionally, we would like to thank Sam Nathans from BBA for his guidance; Sam's knowledge of blockchain is unmatched, and his help was invaluable to us.

We would also like to thank Professor Elkorchi and Professor Sabuncu, for their help and expertise in civil engineering and thermodynamics, respectively. We would also like to thank BETA Group Inc. for providing data to us!

Lastly, we would like to thank all of our parents for supporting us throughout this entire process, we couldn't have done it without you!

Table of Contents

List of Figures

List of Tables

Authorship

Section	Main Author(s)	Main Editor(s)
Abstract	Kelsey Moody	Joshua DeBare
Executive Summary	Kelsey Moody, Joshua DeBare, Cameron Morreale	Kelsey Moody
1. Introduction	Kelsey Moody, Joshua DeBare	Kelsey Moody
1.1 Background	Kelsey Moody	Kelsey Moody
1.2 Embue	Josh DeBare	Kelsey Moody
1.3 Cyvl	Kelsey Moody	Kelsey Moody
1.4 Blockchain	Kelsey Moody	Kelsey Moody
2. Background	Kelsey Moody	Kelsey Moody
2.1 Worcester IS Fab Lab and WorcLab	Kelsey Moody	Kelsey Moody
2.1.1 Worcester IS Fab Lab	Kelsey Moody	Kelsey Moody
2.1.2 WorcLab	Kelsey Moody	Kelsey Moody
3. Project Management Methodology	Mabel Konadu	Kelsey Moody, Joshua DeBare
3.1 Scrum Agile Methodology	Mabel Konadu	Kelsey Moody
3.2 Epics and Stories	Mabel Konadu	Kelsey Moody, Joshua DeBare
3.3 Team Roles	Kelsey Moody	Kelsey Moody
3.4 Daily Scrum Meetings	Mabel Konadu	Kelsey Moody, Joshua DeBare
3.5 Sprint Planning	Mabel Konadu	Kelsey Moody
3.6 Sprint Retrospective	Mabel Konadu	Kelsey Moody, Joshua DeBare
3.7 Sprint Review	Mabel Konadu	Kelsey Moody
4. Project Management Software	Mabel Konadu, Qingbei Shang, Kelsey Moody	Joshua DeBare
4.1 Git and GitHub	Qingbei Shang	Joshua DeBare
4.2 Google Drive	Mabel Konadu	Joshua DeBare
4.3 Discord	Mabel Konadu	Joshua DeBare
4.4 Jira	Qingbei Shang	Joshua DeBare
4.5 Slack	Kelsey Moody	Joshua DeBare
5. Software Requirements	Kelsey Moody, Joshua DeBare, Qingbei Shang	Joshua DeBare, Kelsey Moody

	5.1 Software Requirement Gathering Strategy	Kelsey Moody	Joshua DeBare
	5.2 Functional and Non-Functional Project Requirements	Kelsey Moody, Joshua DeBare, Qingbei Shang	Joshua DeBare, Kelsey Moody
	5.2.1 Worcester IS Fab Lab	Kelsey Moody	Kelsey Moody
	5.2.2 Embue	Joshua DeBare	Kelsey Moody
	5.2.3 Cyvl	Kelsey Moody	Joshua DeBare
	5.2.4 Blockchain	Qingbei Shang	Kelsey Moody
	5.3 User Stories and Epics	Kelsey Moody	N/A
	5.4 Use Cases	Kelsey Moody, Joshua DeBare, Qingbei Shang	Joshua DeBare, Kelsey Moody
	5.4.1 Embue	Joshua DeBare	Kelsey Moody
	5.4.2 Cyvl	Kelsey Moody	Joshua DeBare
	5.4.3 Blockchain	Qingbei Shang	Kelsey Moody
6. Business and Project Risk Management		Mabel Konadu	Mabel Konadu
	6.1 Risk Vs Reward	Mabel Konadu	Mabel Konadu
	6.2 Risk Culture	Mabel Konadu	Mabel Konadu
	6.3 Additional Risk	Mabel Konadu	Mabel Konadu
	6.3.1 Operational Risk	Mabel Konadu	Mabel Konadu
	6.3.2 Financial Risk	Mabel Konadu	Mabel Konadu
	6.3.3 Reputational Risk	Mabel Konadu	Mabel Konadu
	6.3.4 Innovation and Change Management Risks	Mabel Konadu	Mabel Konadu
7. Software Development		Kelsey Moody, Mabel Konadu	Kelsey Moody, Mabel Konadu
	7.1 Agile Scrum Schedule and Setup	Kelsey Moody	Mabel Konadu
	7.2 Sprint 1	Kelsey Moody	Mabel Konadu
	7.2.1 Overview	Kelsey Moody	Mabel Konadu
	7.2.2 Retrospective	Kelsey Moody	Mabel Konadu
	7.3 Sprint 2	Kelsey Moody	Mabel Konadu
	7.3.1 Overview	Kelsey Moody	Mabel Konadu
	7.3.2 Retrospective	Kelsey Moody	Mabel Konadu
	7.3.3 Documentation	Kelsey Moody	N/A
	7.4 Sprint 3	Kelsey Moody	Mabel Konadu
	7.4.1 Overview	Kelsey Moody	Mabel Konadu

	10.1.2.3 Blockchain in Infrastructure	Cameron Morreale	Qingbei Shang
	10.1.2.4 Blockchain and IPFS	Qingbei Shang	Cameron Morreale
	10.2 Software Development Environment	Qingbei Shang, Cameron Morreale	Qingbei Shang, Cameron Morreale
	10.2.1 Node.js (v18.121)	Qingbei Shang	Cameron Morreale
	10.2.2 NPM (v8.19.2)	Qingbei Shang	Cameron Morreale
	10.2.3 Faker.js (v7.6.0)	Qingbei Shang	Cameron Morreale
	10.2.4 Ubuntu	Cameron Morreale	Qingbei Shang
	10.2.5 WSL	Cameron Morreale	Qingbei Shang
	10.2.6 Storj	Cameron Morreale	Qingbei Shang
	10.2.7 Docker	Qingbei Shang	Cameron Morreale
	10.3 Different Platforms	Qingbei Shang, Cameron Morreale	Qingbei Shang, Cameron Morreale
	10.3.1 Filecoin	Cameron Morreale	Qingbei Shang
	10.3.2 Web3.Storage	Cameron Morreale	Qingbei Shang
	10.3.3 Skynet	Qingbei Shang	Cameron Morreale
	10.3.4 Polkadot	Qingbei Shang	Cameron Morreale
	10.3.5 Substrate	Qingbei Shang	Cameron Morreale
	10.4 Blockchain Design for Embue	Qingbei Shang, Cameron Morreale	Qingbei Shang, Cameron Morreale
	10.4.1 Architectural Design	Cameron Morreale	Qingbei Shang
	10.4.2 Generalized Process Flow	Qingbei Shang	Cameron Morreale
	10.4.3 Use Case: Data Request Smart Contract	Qingbei Shang	Cameron Morreale
	10.5 Findings and Discussion	Qingbei Shang, Cameron Morreale	Qingbei Shang, Cameron Morreale
	10.5.1 Filecoin	Cameron Morreale	Qingbei Shang
	10.5.2 Web3.Storage	Cameron Morreale	Qingbei Shang
	10.5.3 Substrate	Qingbei Shang	Cameron Morreale
	10.5.4 Blockchain Applications for Cyvl	Qingbei Shang	Cameron Morreale
	10.5.5 Blockchain Applications for Embue	Joshua DeBare	Joshua DeBare
	10.6 Future Work	Cameron Morreale	Qingbei Shang
11. Learning Assessment		All	All
	11.1 Accomplishments	Mabel Konadu	Kelsey Moody
	11.2 Business Learnings	Mabel Konadu / Joshua DeBare	Kelsey Moody

	11.2.1 Leadership Learnings	Mabel Konadu	Kelsey Moody
	11.2.2 Culture Learnings	Mabel Konadu	Kelsey Moody
	11.2.3 Time Management Learnings	Mabel Konadu	Joshua DeBare
	11.2.4 Team Management Learnings	Mabel Konadu	Joshua DeBare
	11.3 Technical Learnings	Mabel Konadu	Kelsey Moody
	11.3.1 Machine Learning	Joshua DeBare	Kelsey Moody
	11.3.2 Blockchain	Cameron Morreale, Qingbei Shang	Cameron Morreale, Qingbei Shang
	11.4 Joshua DeBare - Embue Lead	Joshua DeBare	Joshua DeBare
	11.5 Kelsey Moody - Cyvl Lead	Kelsey Moody	Kelsey Moody
	11.6 Cameron Morreale - Blockchain Lead	Cameron Morreale	Cameron Morreale
	11.7 Qingbei Shang - Blockchain Lead	Qingbei Shang	Kelsey Moody
	11.8 Mabel - Business Lead	Mabel Konadu	Mabel Konadu
	11.9 Overall	Kelsey Moody	Kelsey Moody
12. Conclusion		Kelsey Moody	Kelsey Moody
	12.1 Embue	Kelsey Moody	Kelsey Moody
	12.2 Cyvl	Kelsey Moody	Kelsey Moody
	12.3 Blockchain	Kelsey Moody	Kelsey Moody
References		Joshua DeBare	Kelsey Moody

1. Introduction

1.1 Background

Today's FinTech industry combines machine learning, data science, blockchain, and many more technologies. In Worcester MA, the Worcester IS FAB Lab utilizes blockchain to allow various Internet of Things (IoT)-based enterprise startups to provide security of data for social good. This data is then used for machine learning and AI purposes to promote social good. The Worcester IS FAB Lab sponsored this project and provided two Iot-based startups for us to work with: Embue and Cyvl. Our project had three sub-project goals: implementation of blockchain with Embue and Cyvl for social good, implementation of machine learning with Cyvl for social good, and implementation of machine learning with Embue for social good.

1.2 Embue

Landlords' energy bills for large apartment complexes can be expensive. Embue tracks temperature, humidity, and many other factors for large apartment units to increase energy efficiency in the buildings. Units tend to lose energy anytime a tenant opens a window while the central heating system simultaneously runs. If Embue could turn off the heating system in units when windows are open, they could save energy. This mindfulness around energy consumption can lead to less energy usage, which lowers tenants' energy bills, making housing more affordable. In addition, energy savings can lead to less energy emissions which, if applied on a large scale, can make renewable energy more feasible, lessening the impacts of climate change.

The group used Python, Pandas, and Sci-Kit Learn to implement a k-nearest neighbors model that determines when a window is open using humidity and temperature data from Embue. This model will help tenants save money and energy in the buildings where Embue's Iot devices reside.

1.3 Cyvl

Municipalities must keep infrastructure and road systems safe; however, local governments often have limited funds. This limitation restricts their ability to repair roads, so municipalities must prioritize which roads to fix. Therefore, municipalities want to predict future road conditions, so they can determine when they should allocate resources to specific roads. Currently, Cyvl uses Iot devices to determine the pavement condition index (PCI) for municipalities (Cyvl.ai, n.d.). The PCI scores are then displayed onto an interactive dashboard map, so that municipalities can track pavement conditions. However, Cyvl does not have the data to predict these PCI scores. Therefore, the municipalities only know the current conditions of the roads; they do not know when the roads will need to be repaired.

Our team used Python, Sci-Kit Learn, and Pandas to create a linear regression model to predict PCI scores. We used data about specific factors surrounding each road, such as pavement type and traffic conditions, and analyzed how these factors affect PCI scores. We then used these factors to predict PCI scores; our model had a r-squared value of 0.434. Lastly, we suggested multiple ways for Cyvl to improve this score in the future, and other potential regressions to investigate.

1.4 Blockchain

Blockchain is a public access, anonymous digital immutable ledger that can verify transaction data between two individuals. Currently, blockchain is most commonly used to ensure the transparency and security of transaction data between two or more parties.

Therefore, for Embue, we explored how the immutability and decentralization of blockchain could improve the security of Embue's IoT network. After exploring a variety of different blockchain storage systems, the team decided to use Filecoin, due to its decentralized nature, immutability, low energy consumption and minimal storage fees. Using Filecoin, if a hacker gets access to one sensor, they won't be able to access other sensors. Embues current network has all the IoT device ID signatures stored on one central database, so a hacker who gets access to one sensor can access every sensor. Once data is stored on chain, it cannot be altered. The benefit of this immutability is that the IoT data stored on the blockchain cannot be altered by potential hackers, and it cuts out the need to have a middleman verify that the integrity of the data. Since Filecoin doesn't encrypt data, the team decided to encrypt the IoT sensor data using Gnupg before storing the data on the Filecoin blockchain. The team also used the Web3.storage API, which manages the data flow process involving the encrypted IoT data, the IPFS network, and the Filecoin blockchain behind the scenes ("Better Storage. Better Transfers. Better Internet." n.d.).

We successfully gathered a small portion of Embue's sensor data, encrypted those files using gnupg, and stored it on the Filecoin blockchain using the Web3.storage API. By encrypting the data, storing it on the Web3.storage API, and retrieving the data from the IPFS network using a private key, we produced a proof of concept that can be leveraged by both Cyvl and Embue. The high level diagram we developed to illustrate the process of how we successfully integrated gnupg encryption, Filecoin, and IPFS within our proof of concept gives Cyvl and Embue a foundation they can refer to when trying to implement blockchain for future use cases.For a future stakeholders, Cyvl can use the blockchain based platform to give them access to only certain elements of the data to have edit access to using a private key.

2. Background

The following sections describe the preliminary research about WorcLab and Worcester IS Fab Lab that we needed to complete this project.

2.1 Worcester IS Fab Lab and WorcLab

Worcester IS Fab Lab sponsored and supported this project. WorcLab, a coworking space in Worcester MA, provided the necessary space to collaborate and meet with stakeholders. The following sections describe the backgrounds of these two companies.

2.1.1 Worcester IS Fab Lab

The rise in Internet of Things (IoT) devices generated enormous data-driven opportunities. However, this came with significant security risks and data-related issues. Much of this new IoT innovation comes from startups who typically lack the resources and expertise to address these issues.

Blockchain technologies are effective tools for data validation, incentivization, tracing and more. Artificial intelligence and machine learning solutions are effective tools for data analytics. Many universities conduct significant research to understand which blockchain and data analytics tools to use and develop for different business needs. However, these universities historically lack deployed Iot devices and users to test these solutions.

Worcester IS Fab Lab, the sponsor of this project, utilizes blockchain platforms to allow "various IoT-based social enterprise startups to provide secure, reliable, transparent, traceable, and efficient networks to share data amongst key stakeholders." The lab partners startup Iot companies with local universities, to support the use of data for "AI and social good research and data analyses" ("Worcester IS FAB Lab for Social Good," n.d.).

The founding partners of the Lab include Worcester Polytechnic Institute, Worcester State University (WSU), Worclab, the city of Worcester, and the start-ups Embue and Cyvl. This collaboration brings together "community leaders, government officials, business owners, researchers, entrepreneurs, and students" ("Worcester IS FAB Lab for Social Good," n.d.).

The purpose of the Worcester IS Fab Lab is to create a collaboration and research program for the development of IoT and Smart Technology that leverage fintech, AI, and blockchain to promote social enterprise and economic development.

WPI houses the research arm of the lab, WSU houses the education and outreach arm, and WorcLab houses the enterprise development arm.

2.1.4 WorcLab

WorcLab describes themselves as "Worcester's Premier Business Incubator & Coworking Space of Passionate Innovators" ("WorcLab" n.d.). Their 10,000 square foot building in downtown Worcester, MA is the largest work lab, collaboration space, and start-up incubator in the Worcester area; consequently, their customers are local start-ups and small businesses who use the WorcLab building to collaborate and create. WorcLab is part of StartUp Worcester, an initiative "to launch and to provide support as well as co-working space for the region's budding entrepreneurs ("WorcLab" n.d.)."

While WorcLab has many local competitors – including Worcester Idea Lab, Technocopia Makerspace, coffee shops, and other event spaces in the area – their focus on supporting local startups differentiates them.

For this project, WorcLab provided a meeting and working space.

3. Project Management Methodology

3.1 Scrum Agile Methodology

The scrum agile methodology is a project management tool used to ensure accountability and manage the many tasks throughout the course of a project. Agile is a sprint-based approach, where the team creates tasks each week and distributes them throughout the group. This methodology gives developers the flexibility to adjust to changing conditions or new discoveries, and the sprints make projects more manageable ("What Is AGILE? - What Is SCRUM?" n.d.).

3.2 Epics and Stories

User stories utilize the following form to describe what tasks need to be worked on and its importance: "As a (type of user), I want to (some goal) because/so that (reason for goal)." Furthermore, stories give an estimation of how much time it will take to complete (in hours); time is tracked using a point system determined by the team. User stories are complete when the tasks are fully completed without defects, and the team has accepted the user story.

Epics consist of a grouping of user stories, which combine into a larger section of the project that form a substantial deliberate goal, covering multiple sprints. In short, an epic allows a team to break a project into several major parts, before breaking the epics down into user stories. Sometimes teams can work on epics simultaneously when there are few overlapping parts between epics.

3.3 Team Roles

The agile process requires many roles that the team must fill, including the Scrum Master and the Product Owner. The Scrum Master must ensure cooperation between team members, and assist in removing any blockers. They also organize the user stories and scrum meetings. The Product Owner drives the business part of the project: they define and prioritize the project requirements. They are also responsible for maintaining the sprint backlog, where they store all future user stories.

3.4 Daily Scrum Meetings

The scrum portion of the Agile Scrum methodology refers to brief daily meetings, where each team member shares the stories they worked on and blockers that were in the way. The team members also detailed what they planned to complete that day. Overall, scrum meetings focus on 3 main questions:

1. What have you worked on?
2. What are you working on today?
3. Do you have any blockers/ have any concerns?

These meetings ensure that all team members are on the same page with the direction of the project.

3.5 Sprint Planning

Additionally, in order to divide work on user stories and tasks during sprints, a planning session is held before each sprint to create and organize work being done. While this is led by the product owner, the whole team plans and creates stories for the future sprint. By the end of sprint planning, the epics will have user stories that will be assigned to individual developers and the sprint will be ready to start (University of Phoenix n.d.).

3.6 Sprint Retrospectives

Sprint retrospective enables all team members to reflect on the good and bad of previous sprint to improve workflow and team collaboration. During these meetings, the team focused on each individual member answering three questions:

1. What went well?
2. What needs improvement?
3. What are our action items?

Action items focus on what the team will continue working on to improve during the next sprint. If there were any cases that were not completed in the previous sprint, and are still needed, the team will add those to the following sprint. The action items should represent both the things that went well and the things that need improvement (University of Phoenix n.d.).

3.7 Sprint Review

Agile also contains weekly sprint reviews at the end of each sprint. During the sprint review, the scrum team updates the product manager on the progress made, and they show the work they have completed throughout the sprint. These reviews tend to be more casual than the retrospectives, but they are just as vital .

4. Project Management Software

Our project requires many project management tools to organize the team, the stakeholders, and the project. Major pieces of the software that we used include communication software, collaborative writing software, agile project management software and code management software. These tools were very important to our project and development process, and they allowed the team to work efficiently.

4.1 Github

Git and Github were our primary code-base management software. Github repo feature allowed us to track our collaborative progress, which included the code script, graphs, CSV files, and more. Github's "branch" feature allowed us to work independently and merge our independent branches to the main repository.

4.2 Google Drive

We used Google Drive to organize all our files throughout the project, together with Google Docs, and Google Slides. We chose Google Drive to allow us to work synchronously on reports, presentations and to have our advisors give us feedback on our paper. Other alternatives such as Outlook and Microsoft Suite have similar features; however, they are not well integrated for simultaneous editing like Google Drive.

4.3 Discord

The team utilized discord as the team's main communication platform. Discord was a convenient way to message team members or send direct messages to them, with support for both smartphones and desktops. Compared to its alternatives such as regular text messages and Microsoft Teams, our team had a good experience using Discord. Discord was free for our uses and we could create various channels to help us organize our work.

4.4 Jira

Jira was the team's project management software. Firstly, Jira has a Scrum-style and Kanban-style dashboard that allowed the team to keep track of workflow and break projects into manageable pieces of work. Since the team has three different epics for two start-up company stakeholders, Jira helped us prioritize tasks and maximize efficiency throughout the project.

4.5 Slack

The team used Slack to communicate with sponsors and advisors. Slack is a messaging app for business that keeps teams easily communicated, and allows groups to organize via "workspaces" and

“channels”. Slack helped the group communicate and ask questions day-to-day and in between meetings. Slack also helped cross-communicate between the different groups, and was more secure, faster, and better organized than email.

5. Software Requirements

5.1 Software Requirement Gathering Strategy

Our overall project requirements from Worcester IS Fab Lab were determined early on through conversations with the Fab Lab, our project description and the concept paper for the lab. However, our specific machine learning and blockchain strategies changed many times. These requirements were determined through conversations with Cyvl, Embue, and Boston Blockchain Association. Some of these meetings were individual meetings, while some included as many sponsors together as possible to make sure everyone was on the same page with the changing requirements.

5.2 Functional and Non-Functional Project Requirements

5.2.1 Worcester IS Fab Lab

Our sponsor, the Worcester IS Fab Lab, had multiple project requirements. First, our project had to involve some implementation of blockchain to help Embue and Cyvl. Second, our project had to involve some machine learning or artificial intelligence application with both Embue and Cyvl. These three project components had to help Embue and Cyvl increase social good in some way.

5.2.2 Embue

Embue's project requirements consisted of creating a model that can predict when a window is open within an apartment unit in real time. Therefore, the model can only be trained using information that would have already been collected. In short, data collected after when the window is predicted to be open cannot be considered because at the time the model is making a prediction when deployed it will not have access to that data.

5.2.3 Cyvl

Our team investigated many machine-learning-based approaches to help Cyvl promote social good. Also, although Cyvl can determine pavement conditions, they cannot predict pavement deterioration. Therefore, they can tell their customers if a road currently needs maintenance, but not when roads will need maintenance in the future. To help them, we decided to create a regression model that will predict PCI scores of roads in the future.

5.2.4 Blockchain

Our team investigated ways that blockchain could help Embue and Cyvl promote social good. Due to our seven week time constraint, we determined our goal to be a minimal viable product. Our minimal variable project designs a basic framework of blockchain that can store data and retrieve data to enhance the security of Embue and Cyvl's data. Blockchain could support data verification, security, and sharing for these two companies. From a social good perspective, this could help more people access the

data securely, and prevent hackers from being able to access all of Embue and Cyvl's data. To decide which platform to use for blockchain, we first looked into Substrate and their documentation for their tutorial. We used our learnings from Substrateto help us decide on a platform among Filecoin, Web3.Storage, and Skynet.

5.3 User Stories and Epics

Each of the following epics correspond with the following story owners:

1. Background research leads - entire team
2. Paper documentation leads - entire team
3. Blockchain leads - Qingbei Shang and Cameron Morreale
4. Embue lead - Joshua DeBare
5. Cyvl lead - Kelsey Moody
6. Business lead - Mabel Konadu

Sprint	User Story/Issue	Points
Epic: Background Research		
2	As a programmer, it would benefit me to research clustering, so that we can implement clustering later on in the project.	2
3, 4	As a developer, I want to research about IPFS in relation to Blockchain	2
3, 4	As a developer, I want to research shrinkage estimation.	2
3	As a developer, I want to provide a high level diagram on blockchain systems	2
3	As a developer, I want to research blockchain applications for security, so that we can prepare to implement it into our project.	2
Epic: Document progress through paper		
2	Write section on blockchain applications for security	1
2, 3, 4, 5	Continue methodology section	1
2, 3, 4, 5	Start business section of paper	2
2, 3, 4, 5	As a project manager, it would benefit me to send the retrospective to the professors, so that I can receive feedback on it.	1
2, 3	Write visual studio code, python, and jupyter notebook sections	1
2	Write tensorflow, pandas sections	1
2	Make user story paper templates	1

2	Write week 1(sprint 0) overview and retrospectives sections, and introductory paragraphs for ch7	2
3, 4	Write section on blockchain applications for security	1
3	Write math research section	2
3,4,5	Write about business risk section	2
3, 4, 5	Write IPFS section	1
4	Write executive summary	2
4	Write sprint sections	3
4	Write functional and non-functional project requirements and use cases for Cyvl and Embue	2
4	Write findings and discussions and design sections for Cyvl and Embue	2
4	Write section about Blockchain future	1
5	Write WSL and ubuntu sections	1
5	Write blockchain portion in executive summary	1
5	Write Filecoin, Web3.Storage sections	3
5	Write blockchain section in introduction	1
5	Write Skynet section	1
5	Write software requirements (Faker.js, NPM, Node.js)	2
5	Write Polkadot section	1
5	As writers, we want to respond to professors' feedback to improve the paper.	1
5	Create business flow diagrams	1
5	Write future work sections for Cyvl and Embue	1
5	Write Agile methodology section	2
5	Write learning assessment sections	1
5	Write conclusion	1
5	Rewrite introduction	1
5	Write nonlinear regression section	1
5	Cite references	2

5	Do formatting, lists of tables and figures	1
5	Write sprint 4 and 5 documentation sections	2
5	Write blockchain discussion and findings section	3
Epic: Integration of Blockchain for Security		
3, 4	As a programmer. I want to figure out the exact scope of the blockchain portion of the project so we can begin the implementation process by Monday 11/14th	2
3	As developers, Suki and Cam want to meet to discuss blockchain.	1
3, 4	As a programmer, I want to set up a meeting with the expert from BBA	1
3, 4	As a developer, I want to present my blockchain findings to BBA	1
4	As a programmer, I want to start watching tutorials on how to set up a blockchain environment.	2
4	As a programmer, I want to download IPFS Software for the Blockchain environment.	2
4	As a developer, I want to create a fake data json file for future tests in Blockchain.	2
4	As a developer, I want to explore Web3. storage and Skynet for Blockchain implementation and data storage.	2
5	Send an email to a blockchain professor	1
5	Send meeting invitations to a WPI blockchain professor	1
5	As a developer, I want to watch the Polkadot tutorial to learn how to use it.	3
5	As developers, we want to read papers from Sarnie to learn more about blockchain.	2
5	As developers, we want to do the platforms tutorial (Filecoin, Web3.Storage, Skynet, Storj) to learn about the different platforms and how they work.	3
5	As developers, we want to create high-level architecture diagrams for Embue’s business application to illustrate the entire process from IoT devices to application layer.	2
5	As developers, we went to create process flow diagrams for Embue’s business application to explain the data transmission.	2
5	As developers, we want to create a Data Request Smart Contract diagram to explore feasible use cases for Embue.	2
5	Write blockchain design sections	3
Epic: Create Machine Learning Model for Embue		
2	As a programmer, it would benefit me to set up a set-temp vs actual-temp graph, to visualize some of the data from Embue.	3

3	As a data scientist, I want to match the sensor and temp device to get more accurate temp readings	3
3	As a programmer, I want to set up time series graphs with temperature and set temperature from the censors, to visualize some of the data from Embue.	3
3	As a programmer, I want to merge tables to match sensors.	2
3, 4	As a data scientist, I want to compile temperature changes following a change in set point over all units to determine if it is a bimodal distribution	2
4	As a data scientist, I want to implement a decision tree branch based on temperature	1
4	As a data scientist, I want to implement a decision tree branch based on if the thermostat is on for long enough	1
4	As a data scientist, I want to implement a decision tree branch based on sequential readings in the dataframe	2
4	As a data scientist, I want to look for a good tutorial on how to implement K-means clustering for python packages	1
4	As a data scientist, I want to explore how many clusters are the optimal amount using the elbow method	2
4	As a data scientist, I want to explore what are the best parameters to cluster by.	3
5	As a data scientist, I want to attempt clustering on Embue data to begin a machine learning model.	2
Epic: Create Machine Learning Model for Cyvl		
2	As a learner, it would benefit me to plan our meeting with the civil professor, so that we can maximize our time with her to learn as much as possible.	1
2	As a programmer, it would benefit me to research derivatives/data for pavement deterioration, so that we can use them in our model.	2
2	As a programmer, it would benefit me to research factors affecting regression (use paper from Cyvl), so that we can implement these factors into our machine learning model.	2
2	As a programmer, it would benefit me to search for pavement data from the list of towns for Cyvl so that we have baseline data.	2
3	As a developer, I want to clean the PSI and MassDOT data.	3
3	As a data scientist, I want to convert the PSI scores into PCI scores.	2
3	As a programmer, I want to combine MassDOT, PSI, and Cyvl dataframes into one dataframe.	3
3, 4	As a data scientist, I want to clean the main dataframe and organize it.	3
3, 4	As a developer, I want to build a linear regression for Cyvl.	2
4	As a data scientist, I want to do a correlation test on the variables and analyze the results.	2
4	As a data scientist, I want to print a summary of the OLS linear regression.	1

4	As a data scientist, I want to analyze the results of the regression summary.	2

Table 1: User stories and epics

5.4 Use Cases

5.4.1 Embue

Embue can use clustering and decision trees to predict window openings in apartment units. Using this algorithm, they can automatically turn off heating and cooling systems when they detect open windows to limit energy loss. This lowers the energy bills for tenants and reduces the building's carbon footprint.

5.4.2 Cyvl

Cyvl can use the regression model to help their customers better allocate money for future road repairs. This proactivity prevents roads from falling into complete disrepair, which keeps drivers of these roads safe. Additionally, road repairs are much cheaper than road reconstructions, which happen when a road is neglected for too long.

This can drastically help cities and towns plan for future road repairs within their budgeting processes. Furthermore, this model empowers states to give money to cities and the national government to give money to states in a systematic and efficient way. Finally, the model could potentially discover underappreciated areas where certain underprivileged communities' roads are being neglected.

5.4.3 Blockchain

Blockchain can be used to enhance security, accelerate data sharing, and validate data integrity. Unlike traditional databases (e.g., MongoDB), blockchain prevents users from changing data unless they break the chain. Our proof of concept for blockchain can serve as a building block for future blockchain projects, and can serve as a blueprint for Embue and Cyvl to enhance their security through blockchain.

6. Business and Project Risk Management

6.1 Risk vs Reward

This is a minimal risk project with a high-level reward. The goal is to implement machine learning techniques and docker storage based blockchain for Embue and Cyvl to build and add to their IoT networks. The reward will be a blockchain platform where both companies can successfully connect with third parties to exchange data, as well as a better understanding of regulations in the exchange of technical and financial data. The associated risk with this project would be an uncompleted project due to our 7-week time limit. Another risk to consider is security, as in any event of a hack, data would be exposed which could put Embue and Cyvl at risk.

6.2 Risk Culture

Startup culture tends to be one of the most overlooked aspects when companies focus on positioning themselves, growing, overcoming competitive obstacles, and surviving in the labor ecosystem. This has led Embue and Cyvl to conduct all necessary checks on all of their employees and take steps to minimize data breaches. Both companies did not provide us with access to their company database; however, we were provided with a sharp file from Cyvl, and Cyvl's parent company Beta, provided the team with PCI scores. Embue, on the other hand, provided us with a CSV file. In regards to this project, there were no background checks on all team members.

6.3 Additional Risk

6.3.1 Operational Risk

Operational risks are caused by ineffective processes, people or the system that can disrupt the flow of business. If the project team. Another risk is being hacked or data breached. Embue provided the team with a csv file, which contained information about each device such as the device ID, unit number and whatever the device is a thermostat sensor. In addition, Cyvl provided the team with a sharp file including PCI scores from their parent company Beta. Due to Cyvl being a startup company, there was a lot of information the company did not have access to provide to the team and sensitive files the team could not obtain during this project. Another operational risk that might have led to us having slow response time to our questions is that both representatives from the companies had to ask from their higher ups or other departments for the information.

6.3.2 Financial Risks

Financial risk refers to the possibility of a business losing money. With Embue and Cyvl's current models, data breach may lead to poor decisions and financial losses. Current mitigation for this is file-encryption, which can protect their customers' data and harder for hackers to access. In connection with this project, the project team was not paid and used some crypto currency from Coinbase to exchange for Filecoin during our blockchain research.

6.3.3 Reputational Risks

Reputational risk is a threat to a good name or the standing of a business. This can greatly affect the status of a company either in a positive or negative way. Both Embue and Cyvl are new to the WPI Fintech MQP project, or WallStreet for short. If the project team were to have really bad experience with either company, it would tarnish the reputation of WPI and the company's future projects. Also, if the team is not successful in completing our deliverables, this may affect future projects as well.

6.3.4 Innovation and Change Management Risks

Innovation risk is the risk you take on when you make improvements to processes (Harvard Business Review, 2013). Change management risks are factors that might prevent a solution from being adopted (LaMarsh Global, 2020). Both Embue and Cyvl are risk averse and tend to be cautious when innovating because both are still new in the business environment. In order to stay relevant and within their markets, Embue and Cyvl must continue to provide a higher level of profitability and value for their customers. They must also find new ways to integrate smart contracts and the business flow of blockchain to increase a wide variety of their business processes. Innovation is more reliable when backed by tangible research and value. Providing evidence of why you think the innovation will work, steps taken to implement the innovation, and the value-added gives reassurance. Understanding the benefits of innovation may help both companies attract investors.

7. Software Development

7.1 Agile Scrum Schedule and Set Up

For project management, we utilized the Agile Scrum methodology, due to its adaptability and flexibility. To implement this methodology, we used Jira. Our schedule consisted of weekly sprints, starting on Monday and ending on Friday. At the beginning of each sprint, the team created user stories, and assigned them to group members based on roles and preferences. These stories dictated the work done each week. Every day, the scrum master led a daily standup, where each member described what they had been working on, any blockers they encountered, and what they plan to work on next. At the end of each week, the team held weekly retrospective meetings to detail what was accomplished that week, and the challenges that the group faced that week.

We completed five main sprints over the course of our project. During our first sprint, the team set up Jira and learned about the Agile Scrum methodology, as the methodology and technology were new to the entire group. At the beginning of our second sprint, we determined user stories, epics, and a point system. For our point system, 1 equated to 0-1 hours of work, 2 equated to 1-3 hours of work, and 3 equated to 3-6 hours of work. At the end of each sprint, the status of user stories was either complete or incomplete; incomplete user stories rolled over to the following sprint. We also included risk analysis with potential mitigation strategies at the end of each sprint.

7.2 Sprint 1

7.2.1 Overview

For our first sprint, we set up Jira and learned about the Agile Scrum methodology. This occurred throughout our entire first week, so we did not have any user stories set up until the beginning of our second sprint.

We also set up our GitHub repository, and made sure everyone had PyCharm, Jupyter Notebook, and the necessary Python packages installed.

We researched blockchain and machine learning during our first sprint. We also set up and familiarized ourselves with the necessary tools to complete this project. Throughout this sprint, we also established weekly meeting times with our various sponsors and advisors.

7.2.2 Retrospective

During our first sprint, we focused on the set up of tools and schedules, and preliminary research. The Agile Scrum methodology was new to the entire group, so Jira was especially difficult to set up and utilize properly. While there were some challenges to this part of the sprint, it was a fairly seamless process. At the end of the week, we detailed a few of the main challenges from the week. This mainly included narrowing down the scope of the project and determining a basic overview of the project.

We also determined that we had areas we could improve upon. Specifically, we determined that our research had to be more project-focused going forward, and we had to finish more sections of the report, due to our time constraint.

7.3 Sprint 2

7.3.1 Overview

For our second sprint, we began to research machine learning techniques, and possible blockchain implementations. We also searched for additional data sources, such as historical pavement scores, pavement indices, and weather data. We also continued to document our progress in the paper, and edit eachother's work in the paper. We met with our sponsors and advisors to narrow down the scope of the project, and to determine the feasibility of our proposed ideas.

7.3.2 Retrospective

During our second sprint, we met with professors and our advisors to discuss the details of the project. We also continued to research machine learning techniques and search for data. One team member found important historical data, which was a high point of this sprint.

However, we continued to have scope issues with the blockchain aspect of the project during this sprint. Due to our limited knowledge of blockchain and our time constraint, we began to have scope creep as the implementation ideas became unfeasible. This was a low point of this sprint. We decided to stop trying to solve this problem by ourselves, and instead we invited our advisors to our meetings to help advocate for us.

During this week the Embue epic shifted from focusing on the central heating system energy usage calculations to predicting when windows are open. As a result, the data needed to be cleaned all over again. Although a shifting project is part of the agile methodology, much of the previous work has been made obsolete. The team engaged in a specific conversation with the Embue CTO where he explained how Embue is planning on installing sensors in the central heating system, which would make our calculations obsolete. The second half of the week after this point was creating a dataframe to combine the different types of sensors.

At this point, the team needed more clarity on the project. The blockchain portion had been workshopped multiple times and still very much in flux. Additionally, the Embue project changed. The team recognized that this challenge came from a struggle of communication between the group and the sponsors. The team decided to reach out more regularly to their advisors on slack and set up an additional meeting with an advisor to mitigate this problem.

7.3.3 Documentation

Story Owner	User Story/Issue	Points	Completed?
Background research leads	As a programmer, it would benefit me to research clustering, so that we can implement clustering later on in the project.	2	No
Embue lead	Write section on clustering in paper	1	No
Paper documentation leads	Write section on blockchain applications for security	1	Yes

Paper documentation leads	As a project manager, it would benefit me to add to the Jira weekly Excel sheets to document our sprints.	1	No
Paper documentation leads	Continue Methodology section	2	No
Paper documentation leads	Start Business section of paper	2	Yes
Paper documentation leads	As a project manager, it would benefit me to send the professors the retrospective by Friday so I can receive feedback.	1	Yes
Paper documentation leads	As a project manager, I have to set up weekly meetings with our sponsors and Professors.	2	Yes
Paper documentation leads	Write Visual studio code, python version, and jupyter notebook sections	2	No
Paper documentation leads	Write tensorflow, pandas sections	2	Yes
Paper documentation leads	Write week 1 (sprint 0) overview and retrospective sections and introductory paragraphs for ch7.	2	Yes
Background research leads	As a developer, it would benefit me to research blockchain applications for security, so that we can prepare to implement it into our project.	2	Yes
Embue lead	As a programmer, it would benefit me to set up set-temp vs actual-temp graph, to visualize some of the data from Embue.	3	Yes
Cyvl lead	As a learner, it would benefit me to plan our meeting with the civil professor, so that we can maximize our time with her to learn as much as possible.	1	Yes
Cyvl lead	As a programmer, it would benefit me to research data for pavement deterioration, so that we can use them in our model.	2	Yes
Cyvl lead	As a programmer, it would benefit me to research factors affecting regression (use paper from Cyvl), so that we can implement these factors into our machine learning model.	2	Yes
Cyvl lead	As a programmer, it would benefit me to search for pavement data from the list of towns for Cyvl so that we have baseline data.	2	Yes
Cyvl lead	As a developer, I want to clean the PSI and MassDOT data.	3	Yes
Cyvl lead	As a data scientist, I want to convert the PSI scores into PCI scores.	2	Yes
			Total Points: 27

Table 2: Sprint 2 User Stories

7.4 Sprint 3

7.4.1 Overview

For our third sprint, the team continued research on both machine learning and blockchain.Our blockchain lead also narrowed the scope of the blockchain implementation, through research and meetings. Furthermore, Embue and Cyvl lead cleaned and organized data, combined dataframes, and began to investigate questions and trends of the data.

We also continued to document our progress through the paper.

7.4.2 Retrospective

Sprint 3 went better than Sprint 2. We were able to narrow the scope of the blockchain aspect of the project, which relieved a lot of stress from the team.

The Embue and Cyvl leads took time to learn about their data and its trends, which helped us identify the most appropriate machine learning techniques to implement for Cyvl and Embue.

There were many small issues with the data frames for both Embue and Cyvl, so the Cyvl and Embue leads spent a lot of time cleaning the data.

High points of this sprint were when we defined the blockchain aspect of the project, and found out that we would get additional historical data to help with our Cyvl regression implementation.

One team member had an important commitment outside of this project, so we were down to 4 members this week. This will continue next week, so we plan to communicate better to pick up the "slack" of being down one member.

7.4.3 Documentation

Story Owner	User Story/Issue	Points	Completed?
Embue lead	As a programmer, I want to research clustering, so that we can implement clustering later on in the project.	2	No
Background research leads	As a developer, I want to research IPFS in relation to Blockchain.	2	Yes
Background research leads	As a developer, I want to research the shrinkage estimator.	2	Yes
Background research leads	As a developer, I want to provide a high level diagram on blockchain systems.	2	Yes
Blockchain leads	As a developer, I want to research blockchain applications for security, so that we can prepare to implement it into our project.	2	Yes
Paper documentation leads	Write section on clustering in paper	1	No
Paper documentation leads	Write section on blockchain applications for security	1	Yes
Paper documentation leads	As a project manager, I want to add to the Jira weekly Excel documents to document our Agile sprints.	1	No

Paper documentation leads	As a project manager, I want to send Professors the Retrospective by Tuesday so that I can receive feedback on it.	1	No
Paper documentation leads	Start business section of paper	1	No
Paper documentation leads	Write Visual studio code, python, and Jupyter Notebook sections	1	Yes
Paper documentation leads	Write Math Research section	1	Yes
Paper documentation leads	As a project manager, I want to write about the business risk culture.	1	No
Paper documentation leads	As a project manager, I want to send retrospective to Professors on Tuesday 11/15 and add it to the paper.	1	No
Blockchain leads	As a programmer. I want to figure out the exact scope of the blockchain portion of the project so we can begin the implementation process by Monday 11/14th	2	Yes
Blockchain leads	As developers, Suki and Cam want to meet to discuss blockchain.	2	Yes
Blockchain leads	As a programmer, I want to set up a meeting with the BBA blockchain expert.	1	Yes
Blockchain leads	As a developer, I want to present my blockchain findings to BBA	1	Yes
Embue lead	As a data scientist, I want to match the sensor and temp device to get more accurate temp readings	2	Yes
Embue lead	As a programmer, I want to set up time series graphs with temperature and set temperature from the censors, to visualize some of the data from Embue.	2	No
Embue lead	As a programmer, I want to merge tables to match sensors.	3	Yes
Cyvl lead	As a programmer, I want to combine MassDOT, PSI, and Cyvl dataframes into one dataframe.	3	Yes
Cyvl lead	As a data scientist, I want to clean the main dataframe and organize it.	3	Yes
Paper documentation leads	Write IPFS section	1	No
			Total Points: 28

Table 3: Sprint 3 User Stories

7.5 Sprint 4

7.5.1 Overview

Due to Thanksgiving break, this sprint was extended. Our blockchain leads continued exploring and researching and watching tutorials to potentially use for blockchain. Josh continued working on

clustering Embue's data while Kelsey continued working on the machine learning techniques. We also continued working and documenting our progress through the paper.

7.5.2 Retrospective

For this week's sprint, our team was limited with time due to Thanksgiving and also having a team member in a different time zone. This did not allow our team to achieve all the intended stories, however, our group was still very productive and worked during Thanksgiving break in order to achieve set milestones.

In addition, the team continued to search and test for a reasonable platform to implement the blockchain so we communicated with the advisors and experts and narrowed down applications to be used for Filecoin, Skynet, and later introduced Substrate for further experimentation. The Embue and Cyvl leads continued working on machine learning models such as decision trees and regression.

Lastly, our team wanted to start planning out additional stories we believed could be completed as we continued to work on analyzing data and blockchain.

7.5.3 Documentation

Story Owner	User Story	Points	Completed?
Embue lead	As a data scientist, I want to implement a decision tree branch based on temperature	1	Yes
Embue lead	As a data scientist, I want to implement a decision tree branch based on if the thermostat is on for long enough	1	Yes
Embue lead	As a data scientist, I want to implement a decision tree branch based on sequential readings in the dataframe	2	Yes
Embue lead	As a data scientist, I want to look for a good tutorial on how to implement K-means clustering for python packages	1	Yes
Embue lead	As a data scientist, I want to explore how many clusters are the optimal amount using the elbow method	2	Yes
Embue lead	As a data scientist, I want to explore what are the best parameters to cluster by.	3	Yes
Cyvl lead	As a data scientist, I want to do a correlation test on the variables and analyze the results.	2	Yes
Cyvl lead	As a data scientist, I want to print a summary of the OLS linear regression.	1	Yes
Cyvl lead	As a data scientist, I want to analyze the results of the regression summary.	2	Yes
Paper documentation leads	Write executive summary	2	Yes
Paper documentation leads	Write sprint sections	3	Yes
Paper documentation leads	Write functional and non-functional project requirements and use cases for Cyvl and	2	Yes

	Embue		
Paper documentation leads	Write findings and discussions and design sections for Cyvl and Embue	2	Yes
Blockchain leads	As a programmer, I want to download IPFS software for Blockchain	2	Yes
Blockchain leads	As a programmer, I want to research about IPFS in relation to Blockchain	2	Yes
Blockchain leads	As a programmer, I want to start watching tutorials on how to set up a blockchain environment	2	Yes
Paper documentation leads	Write IPFS section under blockchain	1	No
Blockchain leads	As a developer, I want to create fake data json file for future tests in Blockchain.	2	Yes
Blockchain leads	As a developer, I want to explore Web3. storage and Skynet for Blockchain implementation and data storage.	2	No
Paper documentation leads	Write a section about Blockchain future	1	No
			Total Points: 32

Table 4: Sprint 4 User Stories

7.6 Sprint 5

7.6.1 Overview

This sprint focused on the application of blockchain, the integration of machine learning for Embue, and the documentation of the project in the paper. The Embue lead implemented clustering, then documented their work in the paper. The Cyvl lead summarized the results of their regression model, and wrote sections of the paper.

7.6.2 Retrospective

For this week's sprint, we completed all the stories that were attainable from last week's sprint and from our backlog. These stories were mostly on further blockchain investigation and implementation and finishing our paper. To start our week, we communicated with our BBA expert and Blockchain leads to watching tutorials on Substrate, Web3.Storage and Filecoin tutorials. The blockchain leads continued working on the user stories and process flow. Sprint 5 suffered, because our blockchain leads and BBA expert were working diligently to produce a high-level diagram of process flow for Embue. We worked on the final deliverables within the last sprint and we were able to complete the carried over points as well.

7.6.3 Documentation

Story Owner	User Story	Points	Completed?
Paper documentation leads	Write WSL and ubuntu sections	1	Yes
Paper documentation leads	Write blockchain portion in executive summary	1	Yes
Blockchain leads	Send an email to a blockchain professor	1	Yes
Paper documentation leads	Write Filecoin, Web3.Storage sections	1	Yes
Blockchain leads	Send meeting invitations to a WPI blockchain professor	1	Yes
Blockchain leads	As a developer, I want to watch the Polkadot tutorial to learn how to use it.	3	Yes
Paper documentation leads	Write blockchain section in introduction	1	Yes
Paper documentation leads	Write IPFS section	2	Yes
Paper documentation leads	Write Skynet section	1	Yes
Paper documentation leads	Write software requirements (Faker.js, NPM, Node.js)	2	Yes
Paper documentation leads	Write Polkadot section	1	Yes
Blockchain leads	As developers, we want to read papers from Sarnie to learn more about blockchain.	2	Yes
Blockchain leads	As developers, we want to do the platforms tutorial (Filecoin, Web3.Storage, Skynet, Storj) to learn about the different platforms and how they work.	3	Yes
Paper documentation leads	As writers, we want to respond to professors' feedback to improve the paper.	2	Yes
Embue lead	As a data scientist, I want to attempt clustering on Embue data to begin a machine learning model.	2	Yes
Paper documentation leads	Create business flow diagrams	1	Yes
Paper documentation leads	Write future work sections for Cyvl and Embue	1	Yes
Paper documentation leads	Write Agile methodology section	2	Yes
Paper documentation leads	Write learning assessment sections	1	Yes
Paper documentation leads	Write conclusion	1	Yes
Paper documentation leads	Rewrite introduction	1	Yes
Paper documentation leads	Write nonlinear regression section	1	Yes
Paper documentation leads	Cite references	2	Yes
Paper documentation leads	Do formatting, lists of tables and figures	1	Yes
Paper documentation leads	Write sprint 4 and 5 documentation sections	2	Yes

Paper documentation leads	Write blockchain discussion and findings section	2	Yes
Blockchain leads	As developer, we want to create high-level architecture diagrams for Embue's business application to illustrate the entire process from IoT devices to application layer.	2	Yes
Blockchain leads	As developer, we went to create process flow diagrams for Embue's business application to explain the data transmission.	2	Yes
Blockchain leads	As developer, we want to create Data Request Smart Contract diagram to explore feasible use case for Embue.	2	Yes
Paper documentation leads	Write blockchain design sections	3	Yes
			Total Points: 48

Table 5: Sprint 5 User Stories

7.7 Product Burndown

To track our progress throughout the project timeline, we created a burndown chart. The burndown chart shows the amount of work completed or left in the sprint in our planned stories and actual stories. We tracked our work using story points. Tracking our project through weekly sprints gives us an idea of how much work has been done in each sprint and how much work is left to be done at the end of the project timeline.

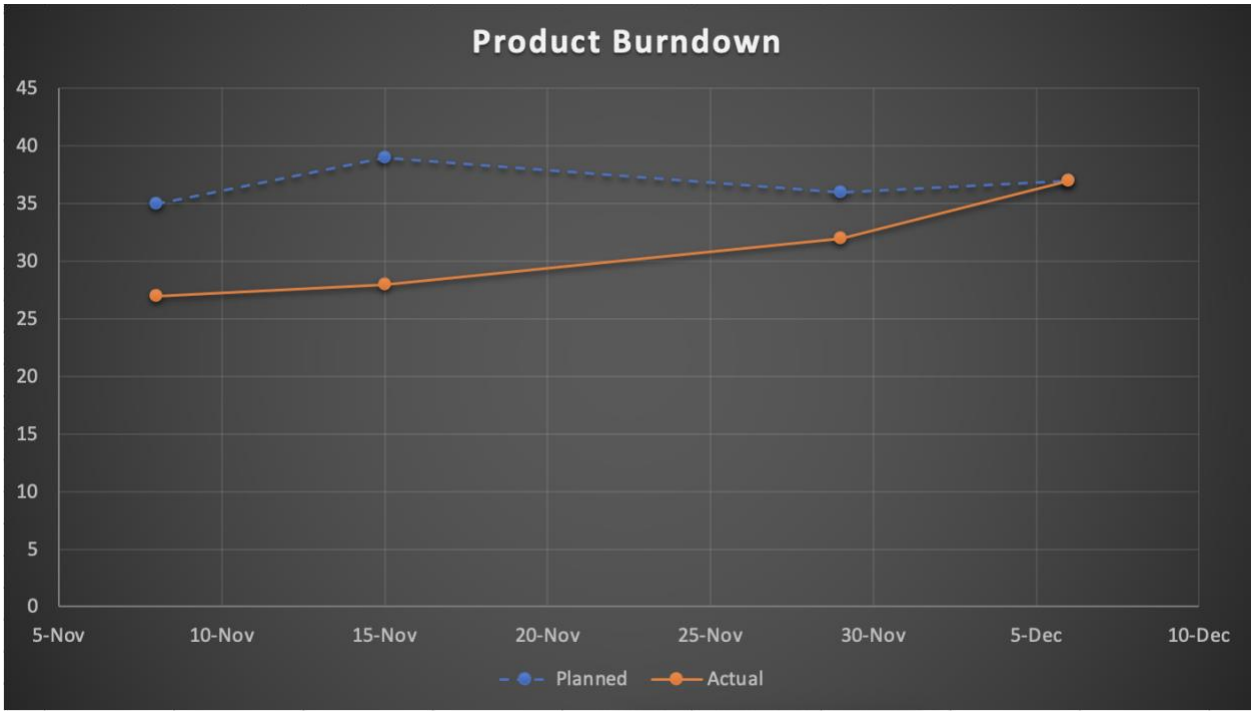


Figure 1: Product Burndown of Points Intended vs Actual Points Completed Each Sprint

8. Machine Learning

8.1 Research

8.1.1 Company Backgrounds

8.1.1.1 Embue

All over the country, cities and towns are mandating reduced carbon emissions from buildings. However, many older buildings have outdated infrastructure that causes heating and cooling inefficiencies, which inflates costs.

Embue, a start-up located in Worcester MA, describes themselves as entrepreneurs and engineers who love buildings. Embue is the first building intelligence, automation and control platform designed specifically for multifamily apartment building portfolios – 40+ unit buildings. They modernize "building infrastructure by monitoring building conditions (temperature, humidity, leaks, noise), central and per-apartment equipment (heating, air conditioning and water) and building elements (window and door open/close)" ("Worcester IS FAB Lab for Social Good," n.d.). They collect data through IoT devices and build dashboards to provide insights for property owners to "reduce energy use, operational waste, and infrastructure risk."("Worcester IS FAB Lab for Social Good," n.d.)

Their mission is to "make every building in your multifamily portfolio smarter and more energy and carbon efficient" ("Embue - Home Page" n.d.). Their expertise includes IoT, SaaS (Software as a Service), building automation, intelligent building controls, and energy efficiency. Embue's customers are owners of large apartment complexes for families, students, and senior housing. Embue's product allows customers to combine per-apartment equipment control with alerts and analytics, to help their customers save up to 25% on utility costs while maintaining a comfortable and healthy indoor environment. These initial customers have also been from mostly underprivileged and vulnerable populations. In addition, Embue's unique predictive maintenance solution allows customers to know the problems in advance to lower risks and improve labor usage. Since residents pay the energy costs, these savings directly help the residents.

For this project, Embue's machine learning engineer was our contact. Due to the sensitivity and identity protection concerns of the data being gathered by Embue's IoT devices, we researched the implementation of a Blockchain platform with Embue's IoT data. We also identified a way to use Embue's data to improve energy efficiency and reduce energy costs in their customers' apartment buildings.

8.1.1.2 Cyvl

Cyvl describes themselves as "next-generation infrastructure management with 3D scanning and artificial intelligence" ("About" n.d.). The team's Cyvl contact contributed to the founding of Cyvl in 2020. The company headquarters are in Somerville, Massachusetts. Cyvl works with nearby cities, civil engineering firms, and governments to manage infrastructure assets such as buildings, roads, sidewalks, trees, and more using 3D map sensors and artificial intelligence. The company hopes to expand its reinvention and innovation throughout the New England area.

For their product, Cyvl first installs low-cost mobile LiDar sensors on sedans and trucks. These sedans and trucks are then driven around to capture high-quality imagery data to obtain all necessary data at once. Then, Cyvl uses 3D artificial intelligence algorithms to process the data and determine a pavement condition index (PCI) for geographically located road data. Cyvl then makes all results available on GIS, CAD and point cloud platforms to provide their customers simple, affordable and effective information about their respective roads ("About" n.d.).

To help Cyvl promote social good, the team created a regression model to predict PCI scores. The regression will help properly allocate money towards maintenance to reduce the number of pavement-related accidents on the roads.

8.1.2 Algorithms Research

The following sections describe the many machine learning algorithms the team researched to consider implementing for Cyvl and Embue. Some of these machine learning algorithms were used, some were suggested for future use, and some were not used.

8.1.2.1 K Nearest Neighbors

The K Nearest Neighbors algorithm (KNN) is a "non-parametric, supervised learning classifier," that uses proximity of labels to make predictions or classifications about individual data points ("What Is the K-Nearest Neighbors Algorithm?" n.d.). KNN algorithms can be used for either regression or classification problems. Figure 1 below shows an example of a KNN algorithm for classification, where k = 5.

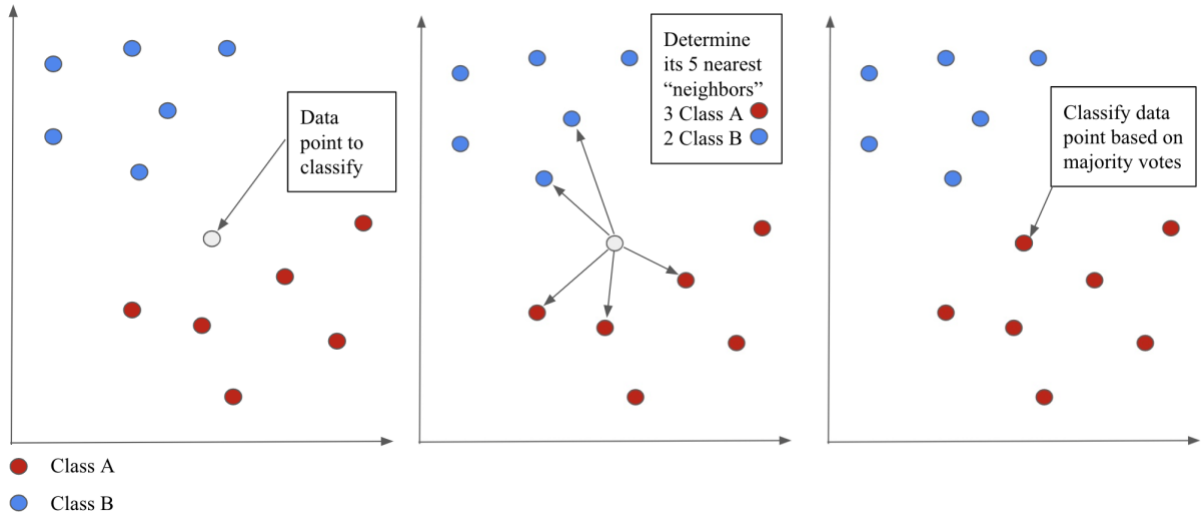


Figure 2: KNN Algorithm for Classification example, where k = 5

For classification, KNN algorithms classify a given data point by identifying the classification of the k nearest data points to the given point, taking the "majority vote" of the classification of those nearby points and giving that classification to the given data point. KNN algorithms can also be used for regression problems ("What Is the K-Nearest Neighbors Algorithm?" n.d.). In these cases, the average of the k nearest neighbors is taken to make a prediction about a classification – the main distinction is "that

classification is used for discrete values, whereas regression is used with continuous ones." ("What Is the K-Nearest Neighbors Algorithm?" n.d.)

In both cases, the distance between the given point and the other data points will need to be calculated. The most common measures used to calculate this distance are: Euclidean distance, Manhattan distance, Minkowski distance, and Hamming distance. Additionally, the variable "k" determines how many nearest neighbors the algorithm uses to determine the classification.

KNN algorithms are easily implemented and adapted, because they only have two hyperparameters. However, KNN algorithms do not scale well or perform well with high-dimensional data inputs, and it is prone to overfitting. KNN algorithms are commonly used in "data preprocessing, recommendation engines, finance, healthcare, and pattern recognition" ("What Is the K-Nearest Neighbors Algorithm?" n.d.).

8.1.2.2 Decision Trees

Decision trees split the data in a tree-like pattern into smaller and smaller subsets. Then, "when predicting the output value of a set of features, it will predict the output based on the subset that the set of features falls into" ("What Is a Decision Tree" n.d.).

Decision trees start with a question that demands a series of additional questions (called decision nodes) to split the data. Each question helps to arrive at a final decision, denoted by a leaf node, that determines the class labels ("What Is a Decision Tree" n.d.). Figure 2 below shows an example of a decision tree for the question "Should I surf?"

Figure 3: Decision tree classification example, where the class labels are "surf" and "don't surf."("What Is a Decision Tree" n.d.)

The decision tree algorithm can be used for either classification or regression.

While decision trees themselves are prone to bias and overfitting, multiple uncorrelated decision trees can predict more accurate results.

8.1.2.3 Random Forests

The random forest algorithm combines the output of a collection of uncorrelated decision trees to solve regression or classification problems. Each tree in the ensemble is made from a random "data sample drawn from a training set with replacement" ("What Is Random Forest?" n.d.). Feature bagging is then used to decrease the correlation among the decision trees. For a regression task, the results of the decision trees will be averaged; for classification, a majority vote will be taken. Figure 3 below shows how the algorithm works.

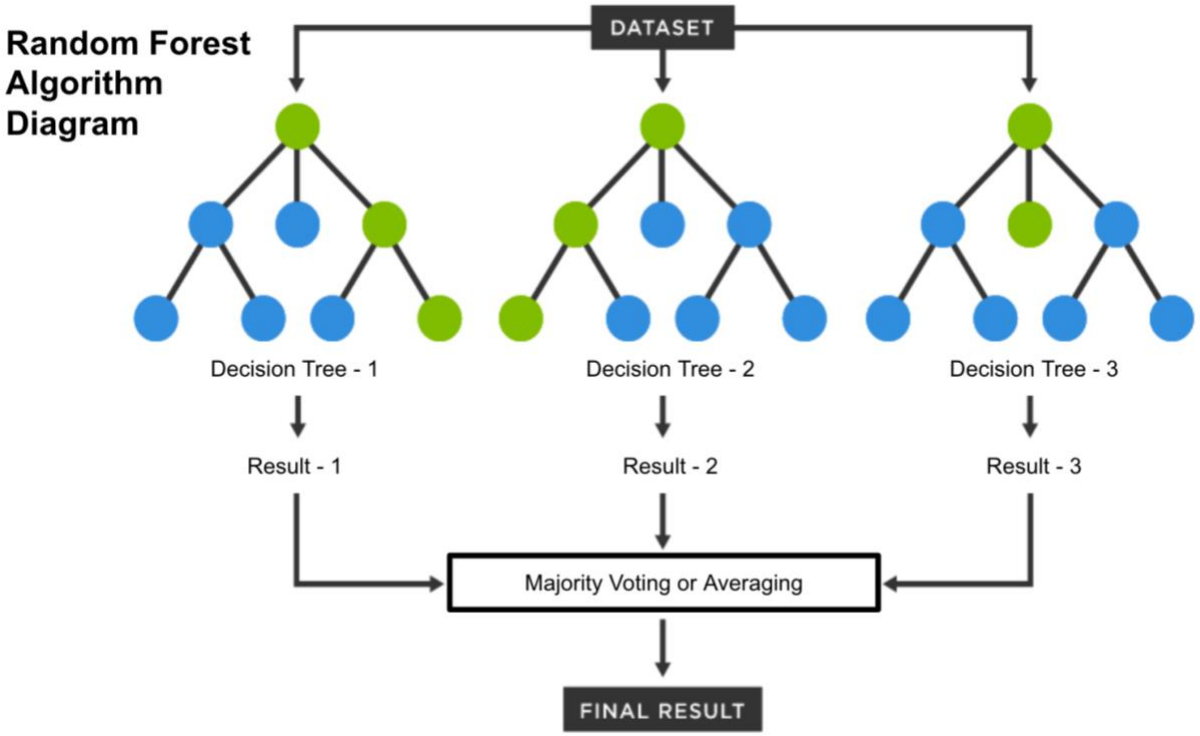


Figure 4: Random Forest Algorithm Diagram for Classification or Regression

Random forest algorithms have three hyperparameters that must be set: node size, number of trees, and number of features sampled.

Random forest algorithms reduce the risk of overfitting with decision trees, provide flexibility, and make it easy to evaluate feature importance to the model ("What Is Random Forest?" n.d.). However, they are time consuming and complex, and require resources to store the data. Random forest algorithms are commonly used in finance, healthcare, and e-commerce ("What Is Random Forest?" n.d.).

8.1.2.4 Linear Regression

Linear regressions predict the value of a dependent variable based on the values of independent predictor variables by estimating the coefficients of the predictor variables. The regression then fits a straight line, plane, or hyperplane.

Linear regressions take the following form:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots \beta_p X_p + \varepsilon_i$$

In this equation, Y is the dependent variable being predicted, β_0 is the intercept of the equation, and ε_i is the random error of the regression. X_i are the independent variables used to predict Y. Lastly, β_1, β_2, … β_p are the coefficients of the independent variables.

We can derive estimates of the coefficients and the intercept that minimize the error between the regression function and sample data (also known as the residual sum of squares(SSR)) through Ordinary Least Squared (OLS) estimation. (Maurizio 2020) OLS estimation is very common, because we commonly want to minimize SSR (Maurizio 2020).

Linear regression models are simple, efficient, and easy to interpret. Therefore, they are commonly tried first when trying to fit a regression to a dataset. However, linear regressions sometimes oversimplify real world complexity; they also assume linearity and are greatly affected by outliers. Linear regressions are commonly used to evaluate trends and sales estimates, analyze pricing elasticity, assess risk, and analyze sports data ("About Linear Regression" n.d.).

8.1.2.5 Nonlinear Regression

Nonlinear regressions "portray a nonlinear relationship between dependent and independent variables," and take the following form:

$$Y = f(X, \beta) + \varepsilon$$

where X is a vector of p predictors, β is a vector of k parameters, f is some known regression function, and ε is an error term (Vaidya n.d.).

These models are typically more accurate and flexible than a linear model, because they can accommodate diverse curves that derive complex relationships.(Vaidya n.d.) Nonlinear regressions are commonly used in insurance, agricultural research, and forestry research, as many relationships in nature are nonlinear.

8.1.3 Math Research

8.1.3.1 Shrinkage Estimation

Shrinkage estimation can be used to improve the determination of betas/coefficients (β_1, β_2, … β_p) in linear regressions when sample observations are "relatively few but regard multiple assets." (Fusai and Rocoroni, 2000) For example, shrinkage would be helpful if we have multiple variables, but only have historic observations from 2-3 points in time.

When we construct a beta vector with few observations, by grouping the betas estimated individually, we produce a beta vector that is "too sensitive to sampling errors" (Fusai and Rocoroni, 2000). To avoid this problem, shrinkage estimation adjusts the individual beta estimates to constrain them near a mean target value.

The simplest shrinkage technique is given by the James-Stein estimator:

$$\beta_{i,\mathrm{JS}} = \bar{\beta} + \alpha_i(\hat{\beta}_{i,OLS} - \bar{\beta}).$$

Equation 1: Coefficient estimate using shrinkage estimator (Fusai and Roncoroni 2000)

In this equation, the beta with subscript 'OLS' is the original beta/coefficient estimate for the variable. The beta with the line above it is the mean of all betas (Fusai and Roncoroni 2000). The alpha is the shrinkage parameter – the shrinkage parameter produces a biased estimator, basically "what we pay for introducing the bias, we get back reducing the variance" (Fusai and Rocoroni, 2000). Therefore, we must pay particular attention to the determination of the shrinkage parameter.

A common practice is to define the shrinkage parameter as:

$$\alpha_i = 1 - \frac{(k-3)v}{(\hat{\beta}_{i,OLS} - \bar{\boldsymbol{\beta}})'(\hat{\beta}_{i,OLS} - \bar{\boldsymbol{\beta}})},$$

Equation 2: Shrinkage parameter (Fusai and Rocoroni, 2000)

In this equation, *k* is the number of assets/variables (k > 3). *v* is the pooled variance of betas (obtained through k individual regressions), with

$$v = \frac{1}{k}(\hat{\beta}_{i,OLS} - \bar{\boldsymbol{\beta}})'(\hat{\beta}_{i,OLS} - \bar{\boldsymbol{\beta}}).$$

Equation 3: pooled variance of betas (Fusai and Rocoroni, 2000)

8.1.3.2 Pavement Indices

The pavement conditions index (PCI) and the pavement serviceability index (PSI) quantify pavement conditions based on multiple observations. The PCI scale goes from 100 (pavement is in perfect condition, newly paved) to 0 (pavement is in complete disrepair). The PSI scale goes from 5 (perfect) to 0 (complete disrepair).

The relationship between these two variables is defined as:

$$\left\{ \begin{array}{lll} \text{PCI} = 100 & \text{for} & \text{PSI} \geq 4.2 \\ \text{PCI} = 27.510 \bullet \text{PSI} - 16.691 & \text{for} & 1.5 < \text{PSI} < 4.2 \\ \text{PCI} = 25 & \text{for} & \text{PSI} \leq 1.5 \end{array} \right\}$$

Equation 4: Equation Relating PCI and PSI (Sidess, Ravina, and Oged 2020)

If given the PSI scores for roads, we can use the above equations to determine the PCI score of those given roads. The PCI score of a road is also a function of time. As the road gets older, its PCI score decreases nonlinearly. We can see this relationship in the pavement deterioration curve below.

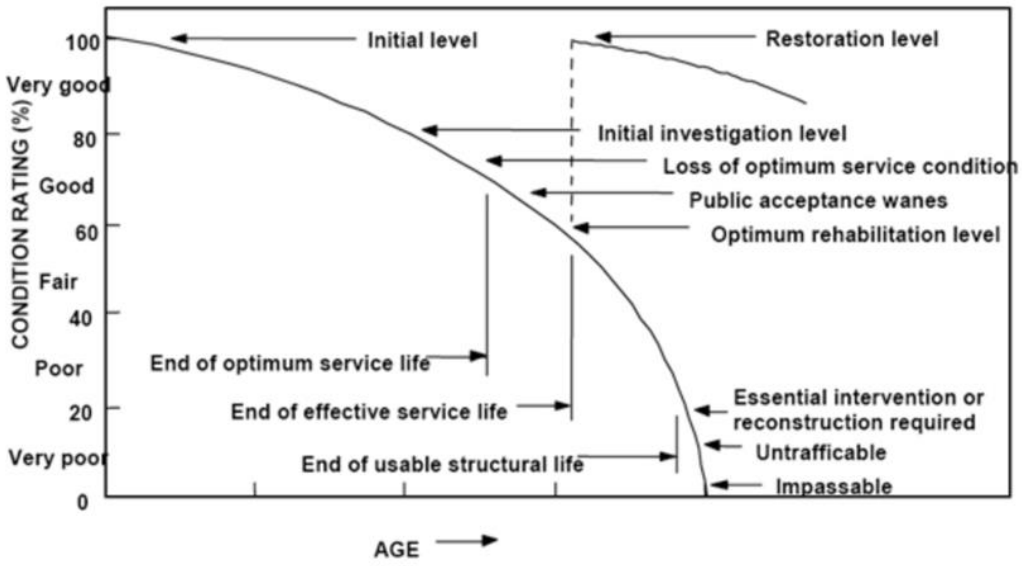


Figure 5: Pavement Deterioration Curve (Al-Mamany, Hussein Hameed, and Zehawi 2021)

8.2 Software Development Environment

8.2.1 PyCharm (v2022.2.3)

One team member used PyCharm (an IDE), as different developers on our team preferred different IDEs based on what they were already familiar with. Not only does PyCharm support Jupyter notebooks, but it also has the capability to write longer form code and a suite of test cases. Additionally, PyCharm has error and warning detection, Git integration, and a debugger, which makes debugging and version control much easier. PyCharm offers functionality to let the developer have multiple scripts open at once, within the same window (Di Russo 2019).

8.2.2 Visual Studio Code (v1.73)

Visual Studio Code (VSC), a free streamlined code editor, has support for debugging, task running, and version control. VSC aims to "provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs" ("Visual Studio Code Frequently Asked Questions" 2022). For this project, some members of our team utilized VSC to code and debug.

8.2.3 Python (v3.7.3)

Python is an "interpreted, object-oriented, high-level programming language" ("What Is Python? Executive Summary" n.d.). Python has high level data structures, dynamic typing, and easy to learn syntax ("What Is Python? Executive Summary" n.d.). Python supports modules and various packages,

some of which are described below. For this project, Python was the programming language we used to clean and manipulate our data, and build machine learning algorithms.

8.2.4 Pandas (v1.5.1)

The Pandas package is built on top of Numpy, as it uses many of the array related functionality provided by Numpy. Common uses of Pandas include data cleaning, data visualization, statistical analysis and loading data. Pandas can convert the files to dataframes, which are easy and fast to process. For example, a developer can easily add extra columns or select data only from certain censors with Pandas ("What Is Pandas in Python?", 2022).

8.2.5 Numpy (v1.23.4)

Numpy, a python library that encompasses arrays, processes data up to 50 times faster than a typical python list. Numpy was built within the C programming language which has more dynamic memory allocation. In short, Numpy is used to assist data scientists in linear algebra and array manipulation, which may come in handy for certain machine learning mechanisms. Numpy offers many pre-built functions ready to modify arrays with any number of dimensions ("Introduction to NumPy", n.d.).

8.2.6 SciKit Learn (v1.1.3)

Scikit Learn (SKLearn) is a python library dedicated to machine learning, and based off of the MatPlotLib, SciPy and Numpy libraries. SKLearn offers functionality surrounding Clustering, Supervised and Unsupervised Machine Learning, Dimensionality Reduction, Ensemble Methods and Feature Extraction. These features may be useful as we implement machine learning algorithms on our data ("Scikit Learn - Introduction", n.d.) .

8.2.7 Tensorflow (v2.11.0)

Tensorflow is an open source library that simplifies implementation and preprocessing of machine learning algorithms. Tensorflow also possesses an extensive collection of pretrained models that data scientists can easily implement. Developers can model mathematical operations by using dataflow graphs that represent data moving between nodes. Although developers wrote Tensorflow in C++, Tensorflow is designed to seamlessly integrate with Python. (Yegulalp 2022).

8.2.8 Jupyter Notebooks (v3.5.0)

Jupyter notebook, a free web application to create and share computational documents, offers a "simple, streamlined, document-centric experience" ("Home" n.d.). The notebooks are in an open document format, and they run interactive code in particular languages and return the output to the user. Jupyter supports over 40 programming languages ("Home" n.d.). Jupyter Notebook allowed us to work

with big data, and with Python and the various packages described above. We chose Jupyter Notebooks because various members of our team already had experience with the application.

8.3 Databases Selected & Data Sources

The following sections describe the sources of the data used to create the machine learning models for Embue and Cyvl.

8.3.1 Embue

The final dataframe for Embue had sensor readings from Embue and outside weather data. This dataframe was a combination of a public dataset and Embue's data. To get the sensor data in the final dataframe, the team determined the specific temperature readings, humidity readings, setpoints and timestamps for the corresponding thermostat and indoor sensor. The hourly outside temperature and humidity data came from a subscription to Visual Crossing, a weather aggregation website.

8.3.2 Cyvl

The machine learning model for Cyvl required a large dataframe. This dataframe included PCI scores of roads from 2018, 2019, 2020 and 2022, road criteria (traffic counts, material, etc.) and the geographical location of the road. This data came from various sources:

1. shapefiles from Cyvl containing geospatial data about the 2022 PCI scores of roads in MA
2. 2020 Road Inventory geospatial dataset from Massachusetts Department of Transportation (MassDOT)
3. 2019 and 2020 geospatial datasets containing yearly PSI scores from MassDOT
4. 2018 PCI scores for roads in Haverhill MA and Milford MA from BETA Group, Inc.

8.4 Data Cleaning, Denormalization, Organization

The following sections describe how the team cleaned and organized the data used in the machine learning models for Embue and Cyvl.

8.4.1 Embue

8.4.1.1 Cleaning Data for Embue: Data Representation

The team received a .csv file from Embue with five columns and seven million rows of readings from thermostat and indoor temperature sensors. Each reading contained the following data:

- a device ID to determine which device the reading originated from,
- a timestamp for when the reading was recorded,
- a type which determines the reading type
- a value which contains the value for that reading.

For example, a reading with the fan type may have either a value of "Off" or "On" to represent the two possible states of the fan. A reading with type "SetPoint" or "Temperature" will have a decimal

number value to represent the degrees celsius. The following reading represents a censor with device ID 100 recording the temperature to be 25 degrees at 5:00 pm on January 1, 2022.

Index	DeviceId	Timestamp	Type	Value
0	100	1/1/2022 17:00	Temperature	25

Table 6: Embue Sensor Reading Example I

8.4.1.2 Cleaning Data for Embue Part 2

Embue gave our team the following data for Council Tower, where DeviceID = 100. The table below is a representation of the data.

Index	DeviceId	Timestamp	Type	Value
0	100	1/1/2022 02:15	Humidity	38
1	100	1/1/2022 00:43	Temperature	23.2
2	100	1/1/2022 01:21	Temperature	23.7
3	100	1/1/2022 01:03	SetPoint	24
4	100	1/1/2022 04:45	FanState	Off
5	100	1/1/2022 00:56	Humidity	40
6	100	1/1/2022 3:24	Humidity	41
7	100	1/1/2022 00:04	SetPoint	21.5
8	123	1/1/2022 00:37	Temperature	24.5

Table 7: Embue Sensor Reading Example II

Each row represents an individual reading. Although each reading has a different timestamp, the humidity, temperature and set point values must be compared to each other at the same time. For example, even though the time for the humidity reading with index 5 is (00:56) is 13 minutes after the temperature reading with index 1 (00:43), they are close enough to be compared.

To compare these data points, the team kept all the timestamps for humidity and matched all the other rows to these timestamps. The group chose humidity because it contained the most readings. As a result, assumptions were made about all other readings to fit the humidity timestamps. For example, our collaborators informed the team that temperature readings do not change by more than 0.5 degrees between each reading, regardless of how much time has elapsed between readings. For SetPoint, a reading is only added when a user physically changes the setpoint temperature through the thermostat. The most recent reading of each of these values was added to the humidity data to compare the reading types at the

same time. Finally, additional auxiliary information, such as if the fan is running was ignored because it was not in the scope of the project. Table 8 below represents the above data from Table X after the data has been combined by timestamp.

Index	Timestamp	Humidity	Temperature	SetPoint
0	1/1/2022 00:56	40	23.2	21.5
1	1/1/2022 02:15	38	23.7	24
2	1/1/2022 3:24	41	23.7	24

Table 8: Embue Data Example 3 - Other Data

Embue also provided the team with device data, which contained information about each device, such as the device ID, unit number and whether the device is a thermostat or an indoor temperature sensor. From this, we paired each unit's thermostat and temperature sensor together to compare data collected from the thermostat and data collected from the indoor sensor within each unit.

8.4.2 Cyvl

First, we loaded in the four datasets used for Cyvl as geodataframes. Afterwards, we subsetted the geodataframes to select only their useful columns, and retitled the columns to match the column names of the Cyvl dataset. Due to RAM limitations, we then had to change the datatypes of the columns to save memory (int 64 and float 64 columns were changed to int 32, int 8, and float 32, depending on the data). We also changed the coordinate systems of the geodataframes so that they were all EPSG 3857. We then dropped any rows in the geodataframes with null/0 values.

After this initial cleaning, we used the PSI to PCI conversion equations, and created 2019 PCI and 2020 PCI columns in the PSI datasets. We then spatially combined these two geo dataframes, so that we had 2019 and 2020 PCI scores for different geographically subsetted road sections.

Next, we combined the MassDOT Road Inventory data with the data from Cyvl. This gave us a geodataframe containing 2022 PCI scores of many roads sections in MA, as well as data about the variables that might impact the deterioration of the pavement, such as traffic counts and surface type. Then, we spatially joined the MassDOT/Cyvl geodataframe with the 2019/2020 PSI data, to create a large database with 2019, 2020, and 2022 PCI scores. Next, we spatially joined the MassDOT/Cyvl/PSI data with the data from BETA to create a master geodataframe with 2018, 2019, 2020, and 2022 PCI scores from Haverhill and Milford MA.

Last, we subsetted the master geodataframe for rows where 2018 PCI $\geq$ 2019 PCI $\geq$ 2020 PCI $\geq$ 2022 PCI. This took out any road that had maintenance done on it during that time, which helped create a better-fitting regression model.

8.5 Design

8.5.1 Existing Software Frameworks and Architectures

Our final deliverables for the machine learning aspect were:

1. Jupyter notebook with data organization, regression model and tests for Cyvl using Python
2. Jupyter notebook with data organization, clustering model, and tests for Embue using Python
3. Google Drive folder with all data files for Cyvl
4. CSV data file for Embue
5. Python packages used: pandas and numpy for data cleaning and organization, Sci-Kit Learn for correlation and machine learning
6. Management software: GitHub to share code between team

8.5.2 Embue Main Dataframe Data Dictionary

Column Name	**Description**	**Datatype**
Timestamp	Time of sensor reading in number of milliseconds after the 1970 epoch	float64
ThermostatHumidity	Number of grams of water vapor per meter cubed as read by the thermostat in the unit	int64
IndoorSensorHumidity	Number of grams of water vapor per meter cubed as read by the indoor sensor	int 64
IndoorTempSensor	Degrees Celcius as read by the indoor sensor	float64
thermostatTemps	Degrees Celcius as read by the thermostat	float64
setPointReadings	The most recent set point reading in degrees celsius	float64
timeSinceLastSetPoint	The time since the set point was most recently changed in minutes	float64
TimeThermostatOn	The time since the thermostat has been on in minutes or 0 if the thermostat is off	float64
EllapsedTime	The time since the last reading	float64
outsideHumidity	Humidity outside in degrees n by number of grams of water vapor per meter cubed	float64
outsideTemp	Temperature outside in degrees celsius	float64

Table 9: Embue Main Dataframe Data Dictionary

8.5.3 Cyvl Main Dataframe Data Dictionary

Column Name	Description	Data Type
address_st	street name	string
City	City number (1 = Abington, 2 = Acton, etc.)	int32
town	town name	string
predict	2022 PCI score from Cyvl	int8
PCI_2019	2019 PCI score	int8
PCI_2020	2020 PCI score	int8
AADT	Average annual daily traffic, measured by counting the total number of vehicles in a year and diving by 365	int32
Mun_Type	Municipal type (0 = empty, 1 = City, 2 = Town)	int8
Terrain	Terrain type (1 = level, 2 = rolling, 3 = mountainous)	int8
Speed_Lim	Speed limit	int8
Curb	Curb Type (0 = none, 1 = left side only, 2 = right side only, 3 = both sides, 4 = along median only, 5 = all curbs (divided highway))	int8
Surface_Tp	Material used to build road (1 = unimproved, graded earth, or soil surface road, 2 = gravel or stone, 3 = brick, 4 = block, 5 = surface-treated, 6 = bituminous concrete, 7 = Portland cement concrete, 8 = composite road; flexible over rigid, 9 = composite road; rigid over flexible or rigid over rigid, 10 = stone dust)	int8
Surface_Wd	Surface width in ft, excluding shoulders	int8
Urban_Type	Population density (1 = large urbanized area (200k people or more), 2 = small urbanized area (between 50k-200k people), 3 = large urban cluster (between 5k-50k people), 4 = small urban cluster (between 2.5k-5k people), 5 = rural)	int8
geometry	coordinates	geometry
RSR	2018 PCI score from Beta	float64

Table 10: Cyvl Main Dataframe Data Dictionary

8.6 Findings and Discussion

8.6.1 Embue

The team had difficulty finding an accurate window predicting model. First, since Embue has not installed window sensors, the team could not confirm the accuracy of the predictions from the model. Due to this issue, we could not use supervised machine learning models, which left us with very limited options for the types of models.

Through conversations with a data analyst and data scientist from Embue, the team generated a Python script that could sort through the records for the desired readings using an approximation of a decision tree. Although we cannot confirm its accuracy, the team used three criteria to approximate window opening based on common temperature behaviors when windows are open. The model filtered readings off of these major criteria:

1. The temperature is stagnant or moving away from a setpoint towards the outside temperature
2. The thermostat being on for more than 6 minutes
3. Persistent behavior of the two readings above

The team chose the first criteria because that is the problem that is being addressed. If the temperature is moving away from the set point and towards the outside temperature, then the unit may have an open window. Next, a thermostat being on would indicate the system is actively heating or cooling. The team chose six minutes as a threshold because the readings were getting confused with a change in set point which could lead to false positives in the model. Finally, the third model criteria filters for multiple consecutive records that fit the other two criteria. Embue is most interested in finding units with the window open for extended periods of time, which the third criteria would indicate.

From this, the team implemented a k-means cluster model that could separate out readings with open windows. In order to know how many clusters to use, the team used the “elbow method” to calculate the SSE (Sum of the Squared Error) for a normalized value using Min-Max normalization. In Min-Max normalization, all values are divided by the largest possible value so that each value is between zero and one. The elbow method analyzes the SSE of many different numbers of clusters and chooses the point where the cluster's SSE changes less and starts to level out.

The two values that the team clustered for were the amount of time the thermostat has remained on for and the difference between the set point and recorded temperatures. We expect the window to be open when the thermostat has been on for a long time and the difference between the setpoint and actual temperature to be large. Therefore, we are looking for a cluster that has both of these characteristics. The SSE values for the different cluster values are in the table below.

Number of Clusters	SSE Value
3	37
4	25
5	16

6	13
7	10
8	8
9	7

Table 11: SSE Values for Embue Clustering

As shown by the table above, the SSE scores drop by 9 points between four and five clusters, while only dropping by three or less afterwards. Therefore, the best number of clusters to use is five. With five clusters the centers of the clusters are at the following coordinates: [0.00464048, 0.39621742], [0.00286022, 0.02377616], [0.00270105, 0.2473421], [0.00939878, 0.54971344], and [0.61499474, 0.57790522]. Based on these values, the team expects the fifth cluster listed to include values when the window is open. Therefore each of the readings in this cluster has the potential to have been taken when a window was open.

8.6.2 Cyvl

The goal of this aspect of the project was to help Cyvl promote social good through machine learning techniques.

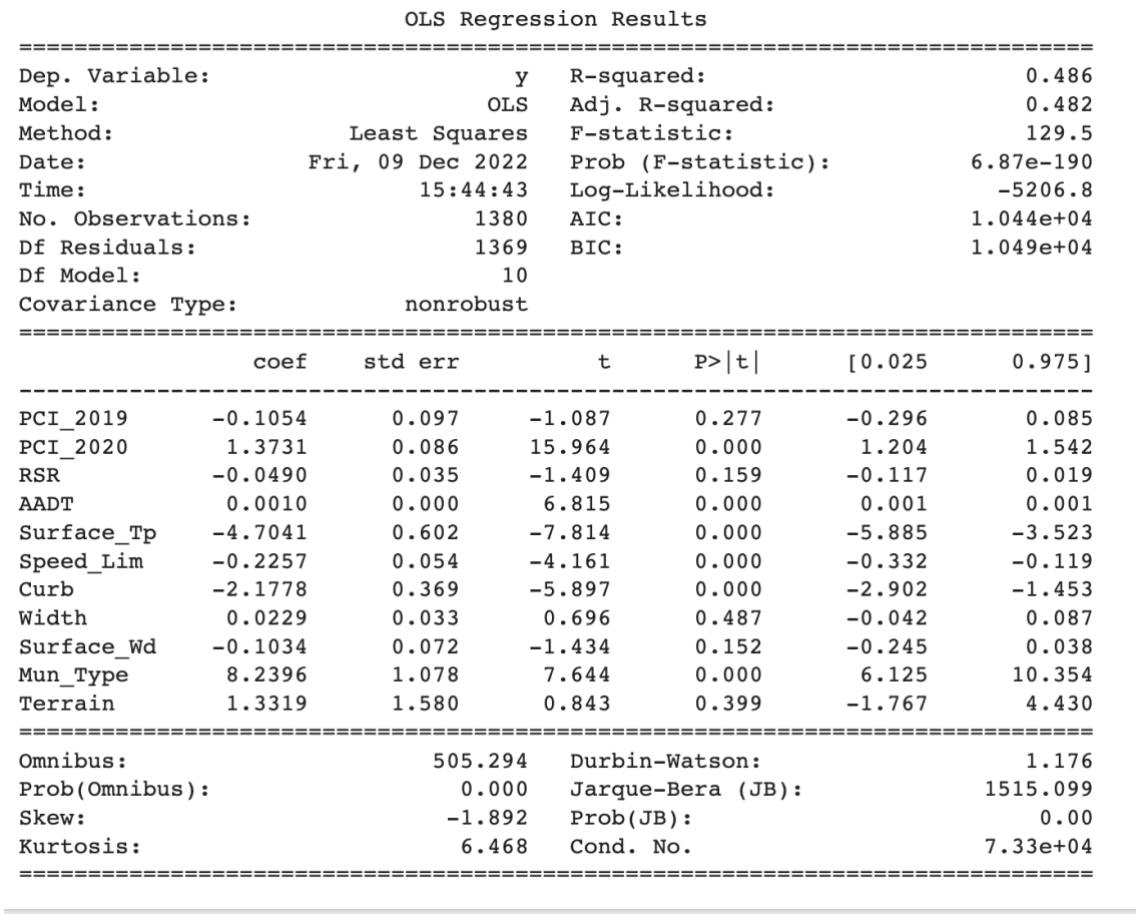
First, the team ran a correlation test on the variables of the cleaned data, to determine if the variables were correlated with the 2022 PCI score – 'predict'. This helped us determine what variables would be most helpful in a regression. The results are in Figure 6 below:

x	City	predict	PCI_2019	PCI_2020	AADT	Mun_Type	Terrain	Speed_Lim	Curb	Surface_Tp	Surface_Wd	Urban_Type	index_right	Width	RSR
predict	-0.2248211	1	0.5979315	0.66651611	-0.0335289	-0.2248211	-0.0299678	-0.0759032	0.07341042	NaN	0.20075305	NaN	-0.1976617	0.18703347	0.34292268
index_right	0.78856243	-0.1976617	-0.5299914	-0.5193289	0.08138166	0.78856243	0.11947828	0.10251282	-0.0529204	NaN	0.04307162	NaN	1	-0.070059	-0.2692882
Width	-0.2556317	0.18703347	0.22726801	0.30229089	-0.2682737	-0.2556317	-0.0383844	-0.1413322	-0.1806453	NaN	-0.114869	NaN	-0.070059	1	0.25306775
Urban_Type	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Terrain	0.11042592	-0.0299678	-0.0186567	-0.0474464	0.07271854	0.11042592	1	0.42137859	-0.0463981	NaN	0.04948439	NaN	0.11947828	-0.0383844	-0.0126298
Surface_Wd	0.12011814	0.20075305	0.29289862	0.23944707	0.54181249	0.12011814	0.04948439	-0.0676993	0.6443151	NaN	1	NaN	0.04307162	-0.114869	0.58281676
Surface_Tp	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Speed_Lim	0.1939216	-0.0759032	-0.1385147	-0.1502123	0.49622944	0.1939216	0.42137859	1	0.10532427	NaN	-0.0676993	NaN	0.10251282	-0.1413322	-0.1467491
RSR	-0.3481056	0.34292268	0.63685837	0.56755104	0.29567871	-0.3481056	-0.0126298	-0.1467491	0.51598949	NaN	0.58281676	NaN	-0.2692882	0.25306775	1
PCI_2020	-0.499907	0.66651611	0.93271592	1	-0.1299401	-0.499907	-0.0474464	-0.1502123	0.14172434	NaN	0.23944707	NaN	-0.5193289	0.30229089	0.56755104
PCI_2019	-0.4634297	0.5979315	1	0.93271592	-0.1031328	-0.4634297	-0.0186567	-0.1385147	0.19173984	NaN	0.29289862	NaN	-0.5299914	0.22726801	0.63685837
Mun_Type	1	-0.2248211	-0.4634297	-0.499907	0.05624426	1	0.11042592	0.1939216	0.04377104	NaN	0.12011814	NaN	0.78856243	-0.2556317	-0.3481056
Curb	0.04377104	0.07341042	0.19173984	0.14172434	0.69121075	0.04377104	-0.0463981	0.10532427	1	NaN	0.6443151	NaN	-0.0529204	-0.1806453	0.51598949
City	1	-0.2248211	-0.4634297	-0.499907	0.05624426	1	0.11042592	0.1939216	0.04377104	NaN	0.12011814	NaN	0.78856243	-0.2556317	-0.3481056
AADT	0.05624426	-0.0335289	-0.1031328	-0.1299401	1	0.05624426	0.07271854	0.49622944	0.69121075	NaN	0.54181249	NaN	0.08138166	-0.2682737	0.29567871

Figure 6: Correlation test of Cyvl data

We must note that our data contained only two towns. Therefore, surface type and urban type only had two different values. We can see the impact of this on the correlation test above, where we have null values for those two variables. Other than that, we can see that most of the variables are correlated with 'predict'. However, Speed_Lim, Curb, Surface_Wd, and Width are not correlated with 'predict'.

Next, the team ran a linear regression on the data, to see if a linear model would fit the data and accurately predict PCI scores for Cyvl. We used OLS estimation to determine the coefficients of the model. The following table summarizes the regression:



OLS Regression Results

Dep. Variable:	y	R-squared:	0.486
Model:	OLS	Adj. R-squared:	0.482
Method:	Least Squares	F-statistic:	129.5
Date:	Fri, 09 Dec 2022	Prob (F-statistic):	6.87e-190
Time:	15:44:43	Log-Likelihood:	-5206.8
No. Observations:	1380	AIC:	1.044e+04
Df Residuals:	1369	BIC:	1.049e+04
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
PCI_2019	-0.1054	0.097	-1.087	0.277	-0.296	0.085
PCI_2020	1.3731	0.086	15.964	0.000	1.204	1.542
RSR	-0.0490	0.035	-1.409	0.159	-0.117	0.019
AADT	0.0010	0.000	6.815	0.000	0.001	0.001
Surface_Tp	-4.7041	0.602	-7.814	0.000	-5.885	-3.523
Speed_Lim	-0.2257	0.054	-4.161	0.000	-0.332	-0.119
Curb	-2.1778	0.369	-5.897	0.000	-2.902	-1.453
Width	0.0229	0.033	0.696	0.487	-0.042	0.087
Surface_Wd	-0.1034	0.072	-1.434	0.152	-0.245	0.038
Mun_Type	8.2396	1.078	7.644	0.000	6.125	10.354
Terrain	1.3319	1.580	0.843	0.399	-1.767	4.430

Omnibus:	505.294	Durbin-Watson:	1.176
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1515.099
Skew:	-1.892	Prob(JB):	0.00
Kurtosis:	6.468	Cond. No.	7.33e+04

Figure 7: OLS Regression Summary

This summary also tells us the coefficients, standard errors, and p values of the independent variables. A lower p value (<0.05) indicates high statistical significance. Therefore, all of the variables except for RSR, PCI_2019, Width, Surface_Wd, and Terrain are statistically significant to the regression. While there are only two values for urban type and surface type, they were still statistically significant to the model.

The standard error (std err) of a variable measures how precisely the model estimated their coefficient. Therefore, we typically look for smaller standard errors, as they indicate more precise estimates of the coefficients. Surface_Tp, Mun_Type, Curb and Terrain have significantly higher standard errors than the rest of the variables, indicating high levels of uncertainty around the estimation of their coefficients.

Additionally, the r-squared value of the regression is 0.486; this means that the regression model explains 48.6% of the variation in the data. While we would prefer a higher r-squared value, this value does not necessarily mean this regression is a bad fit for the data. In the real world, we often have some degree of unexplainable variance in the data, due to high variability in the data, which produces a lower r-squared.

From this analysis, Cyvl can decide which variables to include in the regression, based on their correlation to 'predict', their p values, and their standard errors. They may also consider whether this regression is accurate enough, or whether they should consider other types of regressions.

8.7 Future Work

8.7.1 Embue

First, we recommend that Embue install window sensors in some of the Council Tower units to enhance the machine learning models the team implemented. This would allow the implementation of supervised machine learning methods, which could be more accurate than unsupervised. Also, the current models could be tested for accuracy using the information gathered with window sensors.

We also recommend that the clustering model be expanded upon. Embue data scientists have the potential to improve the accuracy of the models through further exploration of different values in the clustering model. For example, Embue could explore using more than two values or trying different values, such as change in temperature since the last temperature reading or time between readings.

We also recommend that Embue rerun the model on all of their accurate data, to create a generalized model. The current model used only Council Tower data, so we do not know its accuracy for other buildings.

After a model is complete that can predict window openings in the past, it can be adapted to operate in real time, and double checked through window sensors to ensure accuracy, which will be the final step before Embue can use this data to save energy when a window is open.

Embue can also measure the impact of this machine learning model. They can determine how much money landlords save, and how much carbon emissions are saved using this model.

8.7.2 Cyvl

We recommend that Cyvl consider many other independent variables to implement in the machine learning model, such as: weather, whether trucks drive on the roads, and whether the road has been maintained recently. We also suggest that Cyvl look for more varying data, in terms of towns, urban types, and surface types. If they can find this data, we recommend that Cyvl rerun the regression model because it will likely improve the fit of the model.

In a few years Cyvl will also likely have historical data from some of their towns, as they re-survey the roads. We recommend that Cyvl rerun the regression model once they have more historical data to create a better performing model. More historical data would also allow Cyvl to use more towns in the model; more towns would create more variation in some of the variables, and would lead to a better-fitting model.

To create a better fitting model, we also suggest Cyvl investigate the use of shrinkage estimation to determine the coefficients of the linear regression. This type of estimation helps the accuracy of the regression in cases where there are few points in time. The team did not have enough time to investigate this, but it may improve the accuracy of the model.

We also recommend that Cyvl investigate nonlinear regression models. The pavement deterioration model (that graphs PCI over time) is nonlinear. We hypothesize that a nonlinear regression would lead to a better fitting model, but our time constraint did not allow us to investigate this.

Last, we recommend that Cyvl investigate how specific towns differ in terms of historical road statuses. Below are some questions they may consider:

- Does the socioeconomic status of the town impact the general PCI scores?
- What are the characteristics of the towns with the best roads? How do they differ from the towns with the worst roads?
- What areas in specific towns have better/worse PCI scores? How do these areas differ from the town as a whole?

9. Current Business Process Flows

9.1 Embue

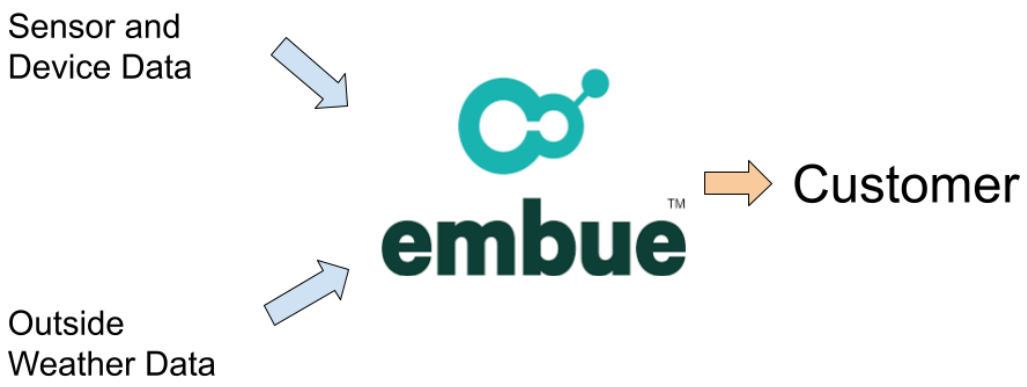


Figure 8: Current Business Flow for Embue

Once Embue's data is in their central database, only Embue and their customers have access to that data. Embue collects data from indoor sensors and thermostats, puts the data into Embue's central database, and gives access to the data to their customers through dashboards. In this business flow, no outside third parties have access to the data.

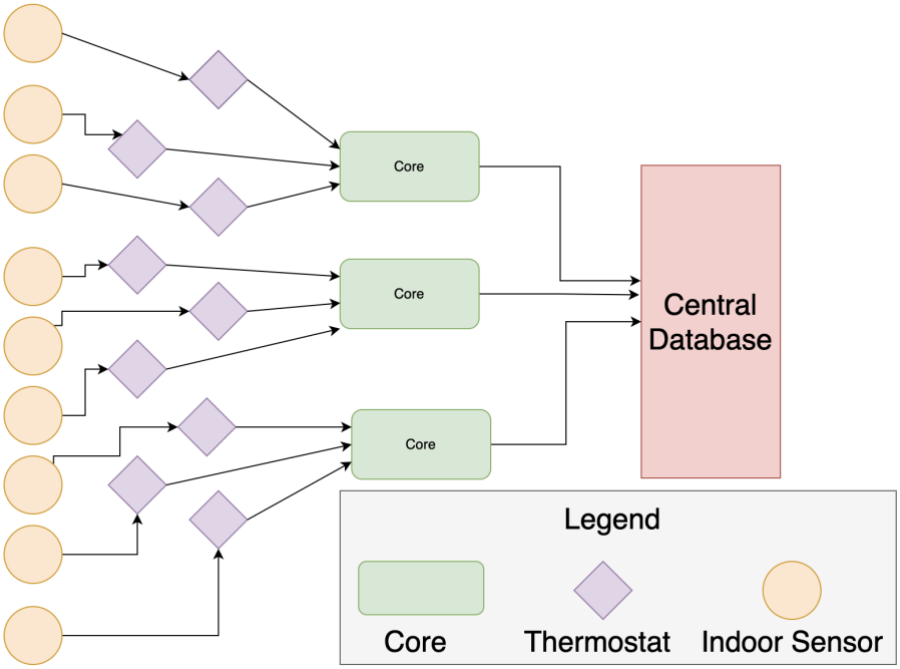


Figure 9: Relationship of Central Server, Thermostats, Indoors Sensors, and Core for Data Collection at Embue

The complicated part of Embue's data regards getting the data from the sensor to the central database. There are three types of IOT devices in this network. Firstly, indoor sensors record humidity and temperature, sending their data to a thermostat within the same unit. For example each individual unit within an apartment complex will have one thermostat and one indoor sensor. The thermostat not only records a plethora of data, but also acts as a relay between a core and an indoor sensor. Each apartment complex has a core where all the data for that apartment complex aggregates before the core sends it to Embue's central server. Sometimes a core will go offline due to an unstable wifi connection, in which case the data remains in the core until wifi connection restores, when the core transfers all data collected during the wifi blackout to Embue's central server.

9.2 Cyvl

Figure 10: Current Business Flow for Cyvl

For Cyvl's business flow, a truck drives around municipalities, and takes pictures to gather geographical data. Once Cyvl collects the data, they add it to their database and send it to their parent company Beta Group Inc., who distributes it to the customers. The customers currently only receive data that is within their jurisdiction, and can edit the data to account for when roads are repaved. In the future, Cyvl may make their data publicly available, while still giving municipalities edit access within their jurisdiction.

10. Blockchain

10.1 Research

10.1.1 Boston Blockchain Association (BBA)

Boston Blockchain Association (BBA) is a “community of innovators, collaborators and entrepreneurs excited about the promise of blockchain technology” (“Home” n.d.).

On their site, the mission of BBA is to establish the greater Boston area as an international hub for blockchain innovation. BBA seeks to support, educate, promote, and advance blockchain technology. BBA plans to accomplish this by supporting and connecting entrepreneurs with useful resources, to develop local partnerships, networks and resources to accelerate the adoption and development of blockchain technology. An expert from the BBA provided guidance and knowledge regarding blockchain and its practical applications.

10.1.2 Background Research

10.1.2.1 Blockchain

Blockchain is a public access, anonymous digital immutable ledger that can verify transaction data between two individuals. The data is encrypted and stored in a block (a data structure that contains a set of transaction data). Blockchain forms a chain of data with multiple blocks (Hayes 2022).

Currently, blockchain is most commonly used to ensure the transparency and security of transaction data. Blockchain utilizes public key cryptography – a system that uses pairs of keys, known as public keys and private keys – for security. The generation of key pairs depends on cryptographic algorithms which are based on complex mathematical problems (“Blockchain - Public Key Cryptography -” 3033). The public key encrypts the data or “locks the door”, and the private key decrypts the data or “unlocks the door”. Public key cryptography makes it so only the desired party can “unlock the door”. In blockchain, When you initiate a transaction the data from both parties is sent to a block, or a locked container holding the metadata around the transaction that was initiated. To verify the transaction, you need the majority of computers in the network to decrypt all the data, or “unlock the container”, and determine if the keys match. These computers are referred to as nodes (Abrol 2022). If the majority of nodes are able to “unlock the container” and verify that the transaction did occur, the block is added to the network. Once a block is added to the blockchain network, it stays on the blockchain network permanently. Therefore, valid blocks are only added to the network and cannot be removed. Although you cannot track who did the transaction, you can see that a transaction occurred between computers, since blocks in the network contain verified transaction data. It is important to note that blockchain itself does not provide privacy, although it allows users to control their data through private and public keys. If someone gets access to your private key they have full access to your data.

10.1.2.2 Blockchain in energy sector

With the technology's development, more advanced equipment has been installed for energy systems, such as smart meters, which leads to the challenges of centralized management and operations. In 2016, 24.6% of the UK gross electricity consumption was generated by renewable energy resources (RES) through onshore or offshore wind farms and PV solar plants, which accounts for 44.9% and 12.5% of the total 35.7 GW ("Digest of the United Kingdom Energy Report 2017" 2017). However, the RES is facing the problem of variability, prediction, and weather, which requires more flexibility measures, including the integration of fast-acting supply, demand response, and energy storage services, to enhance security and safety (Eid et al. 2016, Luo et al. 2014). Among all the countries, the UK alone planned to install 53 million electricity and gas smart meters for homes and small businesses (Wang et al., 2021). Centralized management and operation face more challenges due to the need for advanced communication and data exchanges between different IoT devices of the power network.

Blockchain can improve wholesale energy trading and supply. For the current wholesale energy markets, trading energy consists of a complex procedure that requires different third-party intermediaries, such as brokers, trading agents, exchanges, price reporters, logistics providers, and banks and regulators as shown in the below figure (Andoni et al. n.d.). The current procedures (Figure 11) involve a lot of manual post-processing, which are time-consuming to consolidate information and transactions. With distributed ledger technologies and smart contracts through autonomous trading agents, two companies no longer need to wait for a response from the intermediaries. Instead, the agents can search for the best deal in the marketplace that satisfies the need of a consumer; meanwhile, the smart contract, or the agreement, can be stored in the blockchain safely, the delivery procedure executed automatically, and the payment sent out spontaneously (Andoni et al. n.d.). The entire process is available to all parties involved in the transaction.

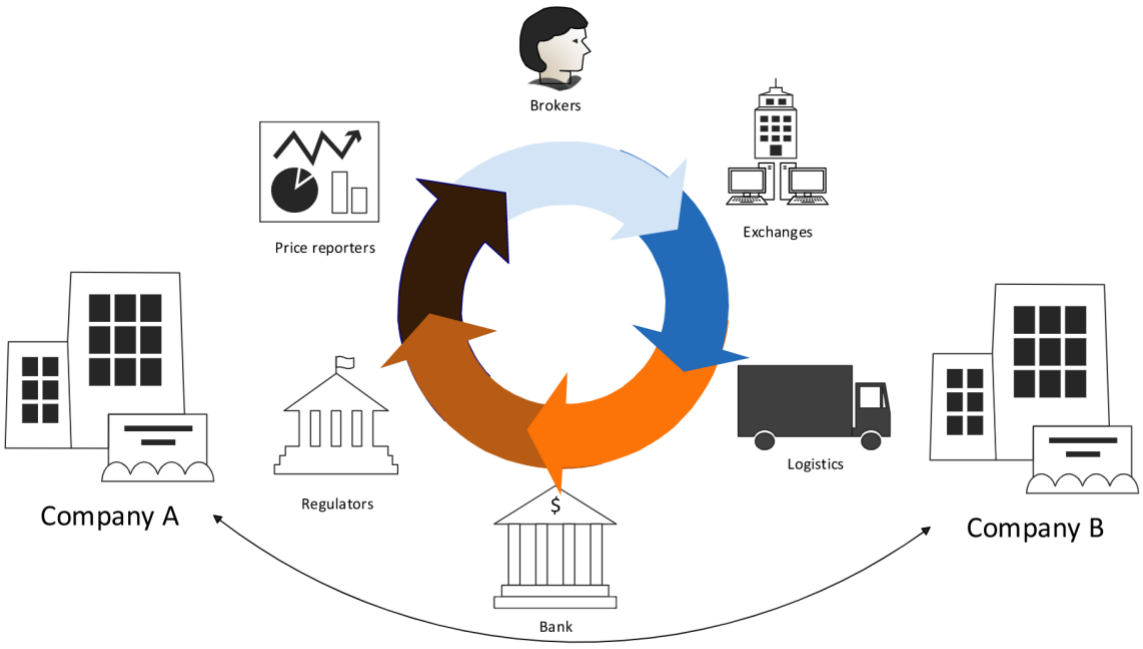


Figure 11: What can blockchain do? (Sahil 2022)

10.1.2.3 Blockchain in infrastructure

The decentralized nature of blockchain as well as the immutability of transactions on the network, security, and transparency makes blockchain a useful tool to improve the infrastructure in our society. For example, blockchain offers additional solutions to improve road safety. The current approach to improving road safety is known as SS (safe system). SS is an approach adopted by the WHO (World Health Organization). The goal of the SS approach in traffic management policies and operations is to achieve the outcome of roads with no accidents or fatalities. To do this the SS approach focuses on preventing collisions, particularly high-speed collisions that result in an increased chance of serious injury or death. The key elements of this system are safe road networks, safe speeds, safe users, safe vehicles, and safe roads. Blockchain can be used to expand the scope of the current SS approach being adopted by the WHO to improve road safety by addressing the interconnected nature of the several elements of safe driving to include safe computation, safe data, and safe communication (Abu Ali, Taha, and Barka 2020).

IoT/ITS entities rely on data coming from different sensors located in different vehicles and other road entities. Centrally storing and managing this data impedes the ability of the network elements to react and make real-time decisions. Safe computation is a commitment to real time computing (to ensure safety we need to be able to accurately assess the user/vehicles' immediate state). The decentralized nature of blockchain in theory makes blockchain an effective tool to facilitate safe computation (Abu Ali, Taha, and Barka 2020). However, based on what we learned working with Filecoin, the process of storing and retrieving data on chain often takes over 24 hours. This limitation makes it so blockchain is currently not an effective tool to facilitate safe computation in practice, though in the future if a blockchain based storage system comes up with a way to speed up the storage and retrieval process significantly while maintaining security, this could be a viable use case for blockchain in the future.

With the rise of IoT/ITS technologies that allow for a vehicle's location and speed to be tracked with respect to the infrastructure on a central network, there is a dependency on data collection and processing. This is where the concept of safe data comes in, as data security and privacy are essential to ensure the safety of the road network. The combination of blockchains' immutability and security makes it a useful tool to ensure that the data is not only accurate, but it is also safe. This ties into the next element of an expanded SS framework, safe communication. Blockchain's combination of anonymity and security makes it a useful tool to facilitate safe communication between the central road network and individual drivers (Abu Ali, Taha, and Barka 2020).

A possible blockchain use case to try and improve road safety would be the usage of smart contracts to control the transfer of digital assets (data in this case) between traffic management control systems (TMCs) and drivers. The TMCs would receive all relevant traffic information about the road conditions, such as how many drivers passed X location in a certain timeframe, speeds ect. TMCs need this data to determine where and when traffic occurs. The decentralized nature of blockchain would allow for this data to be relayed from the drivers to TMCs and back to the drivers in real time, which would address the issue of drivers not knowing how much traffic there will be on the road until it is too late. This would allow drivers to make an informed decision about their commute rather than driving recklessly, in a rush since their GPS did not anticipate the traffic (Abu Ali, Taha, and Barka 2020).

In this use case there would need to be something to bridge the gap between the data from the drivers that is stored on chain to the off chain TMCs. This issue is known as the blockchain oracle problem. An oracle is needed to bridge the gap between blockchain and external systems. This has applications to improving road safety, as something is needed to connect the on-chain data (vehicle location, speed, timestamps, road conditions etc) to the external traffic management control systems in a

secure way. Since a central oracle would defeat the purpose of using blockchain for this use case, Chainlinks' decentralized oracle network would be used to bridge the gap between on-chain and off-chain data. A Decentralized Oracle Network, or DON for short, combines multiple independent oracle node operators and multiple reliable data sources to establish end-to-end decentralization ("What Is an Oracle in Blockchain?" 2021).

Another possible blockchain use case to improve the infrastructure would be the usage of blockchain to track the temperature recorded inside buildings at any given time. A potential application of this data is that it can be used to hold landlords accountable. Since blockchain is immutable, it is impossible for landlords to alter the reported temperature data and timestamps and then claim they are providing suitable living conditions for their tenants, when they are not. On the other hand, this also allows for landlords who are doing their jobs properly to verify that they are providing comfortable living conditions for their tenants, allowing tenants to know which landlords are and which landlords are not providing comfortable living conditions before they sign a lease.

10.1.2.4 Blockchain and IPFS

Blockchain and IPFS (InterPlanetary File System) are two technologies that are often discussed in the context of distributed systems and decentralized networks. While blockchain is a technology for storing and managing data in a distributed and secure manner; IPFS is a protocol for sharing and distributing files in a peer-to-peer network.

Blockchain is a distributed ledger technology that allows for secure and transparent record-keeping of transactions. It eliminates the need for a third-party verifier and operates in a decentralized manner. The transactions which result in changes to the blockchain ledger are digitally signed, verified, and validated by miner nodes, which always keep a duplicate of the ledger. This results in decentralized, secure, and tamper-proof ledes that are shared among all participants in the network (Nakamoto, 2008).

IPFS, on the other hand, is a decentralized content distribution protocol that allows users to store and share large files in a distributed manner. It allows users to host and access files through a peer-to-peer network without the need for a centralized server ("IPFS Documentation | IPFS Docs" n.d.). IPFS uses *content addressing* (Figure 12), which identifies content based on its contents rather than its location. IPFS assigns each piece of content a unique *content identifier* called CID, which is a *hash* of the content. This hash is unique to the content, even though it may appear shorter than the original content ("IPFS Documentation | IPFS Docs" n.d.). In addition, IPFS uses a data structure called *Merkle directed acyclic graphs*, or Maerkle DAGs, which is optimized for representing directories and files. IPFS uses *distributed hash tables* (DHTs) that allow users to access and share data. A hash table is a database that maps keys to values; a distributed hash table is one where the table is split across all the peers in a distributed network ("How IPFS Works" n.d.). To sum up, IPFS uses content addressing to create unique identification, Merkle DAGs to link contents, and DHTs to discover contents.

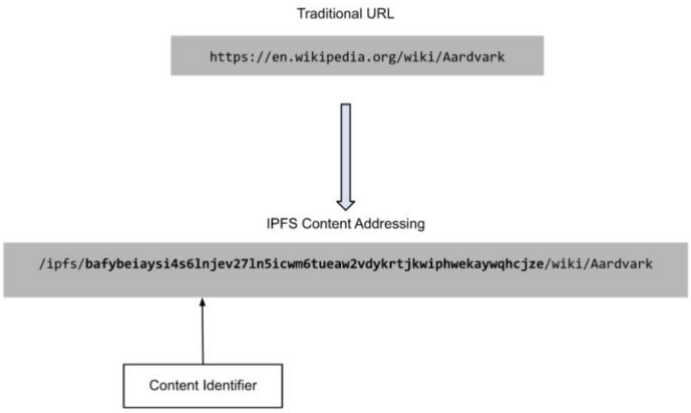


Figure 12. Difference between Traditional URL and IPFS Content Addressing ("What Is IPFS?" n.d.)

The limitation of blockchain is storing large volumes of data. For example, storing large documents can be costly as the 1MB block size limit in Bitcoin's blockchain limits the file that can be uploaded (Benet, 2014), which leads to the need for IFPS. As previously mentioned, IPFS assigns each piece of content a unique content identifier with a hash value for each piece of content. Thus, storing the hashes of the documents in the chain is more effective than storing the document itself (Ernst 2022). Whenever a document is uploaded to IPFS, a hash is generated and stored in the corresponding smart contract. This hash is used to access the document and changes each time when the document is modified (Nizamuddin et al.).

In conclusion, while both blockchain and IPFS are decentralized technologies that have the potential to revolutionize industries, they have different features and use cases. Blockchain is a distributed ledger technology primarily used for applications requiring a secure and transparent ledger of transactions. At the same time, IPFS is a decentralized file-sharing system primarily used for applications requiring the storage and sharing of large amounts of data.

10.2 Software Development Environment

10.2.1 Node.js (v18.12.1)

Node.js is an open-source and cross-platform JavaScript runtime environment, which is feasible for building fast and scalable network applications for developing server-side and networking applications ("Node.Js - Introduction" n.d.). The language Node.js uses is JavaScript, which can be run on OS X, Microsoft Windows, and Linux. With its rich library of various JavaScript modules, developers can simplify the development of web applications to a great extent.

Node.js = Runtime Environment and JavaScript Library

Figure 13: What is Node.js?

10.2.2 NPM (v8.19.2)

NPM is Node Package Manager, which is considered as the world's largest Software Registry with over 800,000 code packages ("Introduction to NPM Scripts" 2022). NPM allows open-source developers to share software and manage private development. Npm scripts are the entries in the scripts field of the *package.json* file, which are used to automate tasks, such as removing unnecessary characters CSS and JavaScript code, and building projects ("Introduction to NPM Scripts" 2022). In this project, NPM helped us to install the dependencies from various open-source software, such as Faker.js and Web3.Storage.

10.2.3 Faker.js(v7.6.0)

Faker is one of the popular libraries that generate fake and reasonable data that can be used for unit testing, performance testing, building demos, and working without a complete backend ("Getting Started | Faker" n.d.). We use Faker.js to generate simple test data based on Embue's temperature devices, which includes the following fields: randomID, randomUnit, and randomType (see below).

```
{
    "deviceID": "f2191403-244e-4368-972e-2888ee41bd39",
    "UnitNumber": "376",
    "DeviceType": "Thermostat"
},
{
    "deviceID": "68cd843f-5f63-4f0c-9e36-c73c3d4c5cc4",
    "UnitNumber": "534",
    "DeviceType": "IndoorSensor"
},
```

Figure 14: Fake data example with Faker.js

10.2.4 Ubuntu

Ubuntu is a free open source Linux based operating system that we used in our project to install the packages needed to create a Filecoin node to interact with the blockchain. For this project, we used the latest version at this time which is 22.4.1 LTS ("Install Ubuntu on WSL2 on Windows 11 with GUI Support" n.d.).

10.2.5 WSL

WSL is software that allows windows 10 and above to install and use linux applications and command line tools on a windows system. For this project, WSL was used to help us install Ubuntu ("Install Ubuntu on WSL2 on Windows 11 with GUI Support" n.d.).

10.2.6 Storj

Storj is a blockchain based storage system where ordinary people share their unused storage space. It configures a trust based cloud storage system between clients and hosts, and all client data must be encrypted before transmitting on the network. The encrypted portion of files stored on the network are known as shards. One of the main limitations of blockchain is that as more and more data is stored on a blockchain, nodes are required to process and validate the information which is a complex process that slows processing speeds. While this validation process is a crucial part of blockchain's security, it makes it very inefficient to store massive amounts of data on a blockchain network. This limitation is known as blockchain bloating. Storj addresses this issue by storing only the metadata of each block on the blockchain, which holds essential information such as the encrypted file location and hash (Wilkinson, Brandoff, and Buterin 2014).

The integrity of shards stored on the network is verified by a protocol known as proof of storage. Proofs of storage (PoS) are cryptographic protocols that allow a client to efficiently verify the integrity of remotely stored data. To use a PoS, the client sends an encoded version of its data to the server while keeping a small amount of state locally. At any point in time, the client can then verify the integrity of its data by executing a highly-efficient challenge-response protocol with the server. In Storj, there are 2 ways this is accomplished, proof of storage via Merkle audits (Zahed Benisi, Aminian, and Javadi 2020) and proof of storage via pre-generated audits (Kamara n.d.).

Once the Shards are verified, Storj uses ethereum blockchain and uses a process called proof of redundancy. It stores the metadata in Satoshi style, enabling users to retrieve their information in full whenever needed. Satoshi style is a similar approach to privacy that the Bitcoin network uses, it is named after the founder of Bitcoin. The way this approach to privacy works on the Bitcoin network is that in a blockchain, each person assigns a public key that is visible to all nodes in the network. On the other hand, each person has a private key, which is their Bitcoin wallet. As long as users keep their private key securely, it will not be possible to reveal their true identity. In other words, unlike traditional systems, a central party can retrieve a user's identity neither by knowing their public key nor private. The traded cryptocurrency in this blockchain is also called bitcoin and goes by the ticker "BTC" (Zahed Benisi, Aminian, and Javadi 2020). Satoshi style implies that this process is used for non-Bitcoin cryptocurrencies. In this use case, the cryptocurrency used is called "STORJ", which is a token that is used to pay for storage space. The Storj token works on the Ethereum blockchain rather than the Bitcoin blockchain (Rosenberg, Chavarria, and Velasquez 2022).

An application called Metadisk runs on the Storj platform and checks the network periodically to make sure the files stored are available and untampered. Each file is stored in at least three locations with the option of increasing this amount to meet the client's demands. As a result, even if the original file is destroyed, the data can still be retrieved through one of the copies. If a node or an audit is not available, another copy of the file would be saved on a new node (Wilkinson, Brandoff, and Buterin 2014).

The downside of Storj is that currently it lacks mobile support, so if you want to access data on a Storj network, you would have to use a computer. For people who are willing to lug around their computers everywhere they go, this is not a big issue. However, if you do not want to do this, you would not be able to access the data anywhere and anytime you want.

10.2.7 Docker

Docker is an open platform that allows developers to easily create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts, such as libraries and other dependencies, and deliver those as one package. Docker allows easier ways to create and manage applications, because they can be run in various environments without the need to install additional dependencies (Docker Documentation n.d.).

10.3 Different Platforms

10.3.1 Filecoin

Filecoin is a blockchain based decentralized storage system that runs on top of IPFS. The purpose of Filecoin is to help clients store data on space offered by storage providers who are not their trusted parties in a safe way, and to ensure that said data remains safe over time. To do this, Filecoin uses 2 methods. The first method is Proof-of-Spacetime. Through this method, clients will be assured that during a specific period of time their data is being stored. The next method is Proof-of-Replication. Proof-of-Replication shows that data is safe in its dedicated physical storage locations and not a single node in the network has duplicated files on their own hardware. Storing data on a decentralized storage platform costs a fraction of the price of a centralized storage provider ("How Filecoin Works" n.d.). While it can cost upwards of $75 to store 1 GB of data on AWS or Google for one year, it's nearly free on Filecoin (Levin 2022). The downside to Filecoin is a lack of profitability and lack of interest among large investors (Schoeman 2021). As useful as Filecoin is for ensuring that stored data is secure whether or not the storage providers are trusted parties, there is the risk of the coin itself being scrapped. If the coin is scrapped, people will be unable to retrieve their data, as data is stored and retrieved through smart contracts that would no longer exist if the coin were scrapped. Considering that Filecoin is in the very early stages of its development, the risk of this is significantly higher than the risk of a more established cryptocurrency such as ethereum or bitcoin being scrapped.

10.3.2 Web3.Storage

Web3.Storage is an API that manages the data flow process involving the data file you upload on the site, the IPFS network, the process to find Filecoin storage nodes to store your data, and the retrieval of your data behind the scenes. The Web3.Storage API can be used by setting up the Javascript client library or the Go client library. There is also the option to use the web interface instead, to do this you need an API token which you get by creating an account. The Web3.storage API does not encrypt data for you, so it is recommended that you encrypt your files before uploading them onto Web3.storage. Once you upload your file to Web3.Storage, it is converted to CID format and stored on the IPFS network while being aggregated for storage on the Filecoin network. This process takes up to 48 hours before a Filecoin storage provider is found. Once this process is complete, data can be retrieved by using an IPFS HTTP gateway that lets you view CID files. This can be done on any IPFS gateway, but it is faster on the IPFS gateway that Web3.storage runs. Although we found this to be the easiest way to retrieve data, data can also be retrieved by using web3.storage client libraries, in the terminal using IPFS command line tools, or in terminal using curl or powershell ("How to Retrieve Data from Web3.Storage" n.d.).

10.3.3 Skynet

Skynet is an open protocol for hosting data and web applications on the decentralized web using the Sia blockchain network that is guided by Sia Foundation (“Skynet Overview - Skynet Guide” 2021). Skynet's decentralized database allows users and applications to alternate storing data options from a single central authority to a decentralized manner. The decentralized database gives users the maximum ability to access data across the globe on any device and any application.

Skynet is an open-source software, which means any developers can use Skynet to develop their own blockchain application for free. For developers, Skynet does not require any payment for the application’s storage. And developers do not need to take corporations pulling access to their resources in consideration. For users, Skynet prevents corporate oversight and ensures users’ privacy.

10.3.4 Polkadot

Polkadot is a platform for Web3; they describe their mission as “envision a Web where our identity and our data is our own - safely secured from any central authority” (“About Polkadot” n.d.). Polkadot will establish a decentralized web environment where users are in control of their own data. Polkadot Relay Chain is the central core of Polkadot, which is responsible for the network’s shared security, consensus, and cross-chain interoperability. Parachains, a type of parallel transaction execution model, are used for sovereign blockchains that can have their own tokens and optimize their functionality for specific use cases. Parathreads are similar to parachains but with a pay-as-you-go model, which is more economical for blockchains that do not need continuous connectivity to the network. Bridges allow parachains and parathreads to connect and communicate with external networks (“About Polkadot” n.d.). Overall, the combination of parachains, parathreads, and bridges allows Pokadot to support a wide range of decentralized applications and provide a scalable, interoperable blockchain platform.

10.3.5 Substrate

Substrate is a blockchain technology platform that offers a modular and customizable framework for buildings centralized applications. It was developed by the team at Parity Technologies and is written in the Rust programming languages. Substrate is designed to be modular and flexible, which allows developers to create and deploy their own blockchain applications and networks easily. In addition, it allows developers to focus on the unique features and functionality of their blockchain, rather than spending time and resources on low-level implementation details. Substrate has a low barrier to entry, allowing even those with limited blockchain experience to get started building decentralized applications quickly. Also, it has strong support for smart contract development, making it a good choice for decentralized applications that require complex, programmable logic (“Substrate Blockchain Technology” n.d.).

Substrate Connect offers a fast, secure, and decentralized way to connect with other blockchain networks, including Polkadot and Kusama. Its light-client-first design allows users to interact with a blockchain in a more efficient way than running a full node. Light clients can quickly sync with a blockchain and run locally on various devices, making them well-suited for powering decentralized applications. In contrast, running a full node requires significant knowledge, effort, and resources, and

connecting to a remote third-party node can be insecure and unreliable. Substrate Connect is one of the most convenient ways to securely and trustfully connect your app to any Substrate-based chain. When run as a browser extension, it allows for multiple light clients to run simultaneously and continuously sync as long as the browser remains open. It is compatible with all major browsers and does not require a TLS certificate for node connections (“Substrate Connect” n.d.).

10.4 Blockchain Design for Embue

10.4.1 Architectural Design

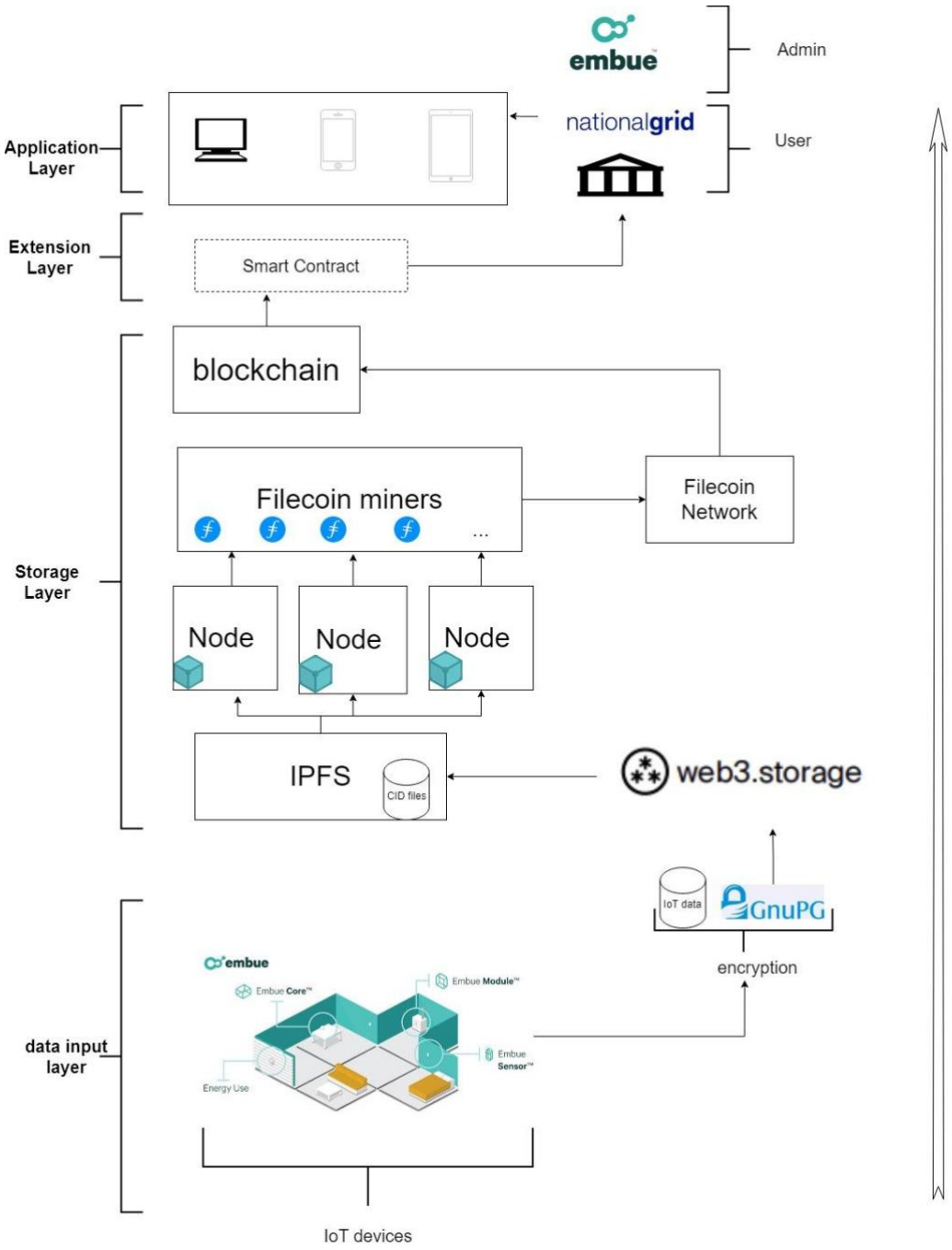


Figure 15: Overview of Proof of Concept

To give a high level overview of how all the moving parts for our proof of concept interact with each other, we created Figure 15 and structured our design based off of a blockchain protocol smart contract architecture for clinical trials (Omar et al. 2020). This diagram is based off of the following steps:

1. Data input layer
    a. Gather Embues' IoT sensor data containing the ID signature for each sensor
    b. Gnupg converts the .docx file to a .csv that can only be unlocked by entering your private key ("GnuPG — The Universal Crypto Engine" n.d.)
2. Storage layer
    a. Encrypted data file is uploaded to the web3.storage API ("Say Hello to the Data Layer" n.d.)
    b. Data file is converted to CID form to be stored on IPFS nodes
    c. Filecoin miners gather the CID from IPFS nodes and perform data aggregation on the Filecoin blockchain ("Home | Filecoin Spec" n.d.)
3. Extension layer
    a. When the admin (Embue) grants access to the IoT sensor data for the user, the smart contract executes a command that begins the retrieval process from the blockchain to the user
    b. The Filecoin miner sends an idenical copy of the CID file to the IPFS network for retrieval
    c. If Embue denies access, the retrieval process does not occur
4. Application layer
    a. Anyone who wants access to Embues' IoT sensor data such as the national grid, national government, etc. requests access to specific IoT sensor data using a mobile application
    b. Embue grants or rejects access to IoT sensor data
    c. If access is granted, the user will be able to retrieve the the file and decrypt it. If access is denied, they will get a notification on their application that informs them their request was denied

10.4.2 Generalized Process Flow

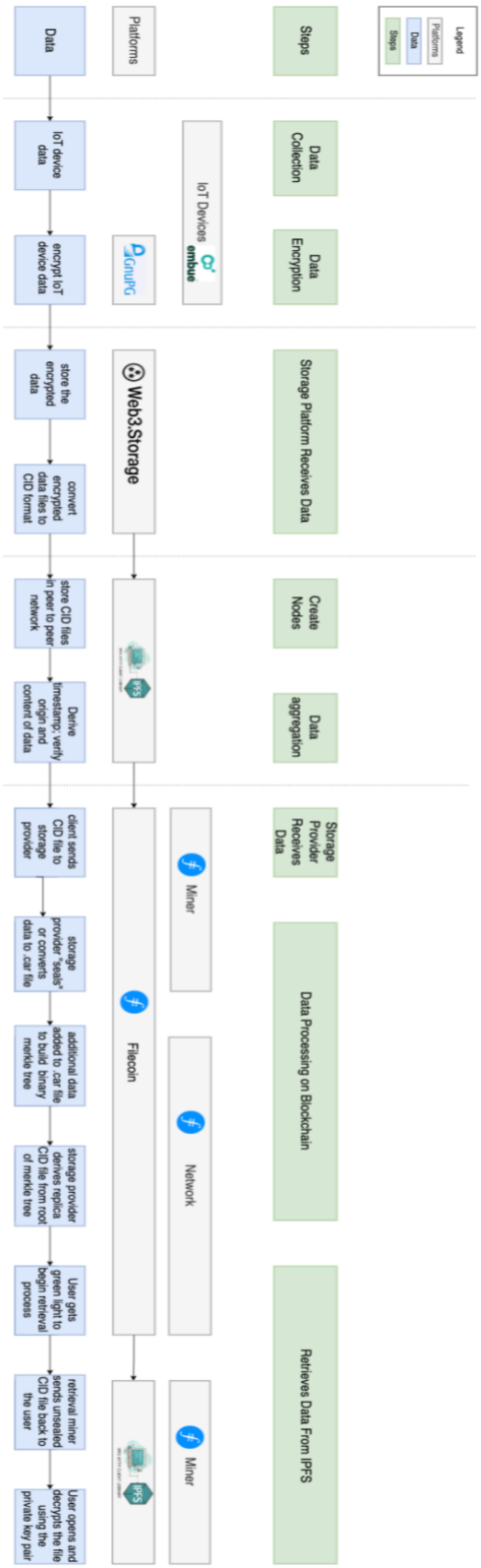


Figure 16: Process Flow Diagram

The process flow diagram shows the data transmission from IoT devices to data retrieval. Detailed steps include:

1. Data collection is run by Embue's IoT devices, including thermostats, indoor sensors, and other IoT devices.
2. Data encryption is performed through *GnuPG Made Easy* (GPGME) to generate private key pairs to decrypt the data.
3. Storage Platform (web3.storage) receives the encrypted data file and converts it into content identifier (CID) format, which can be used to fetch the files over IPFS later.
4. Once the CID files are received from the storage platform, IPFS creates the nodes to store CID files in a peer-to-peer (P2P) network.
5. IPFS runs data aggregation to derive the timestamp, verify the origin, and verify the content of the data.
6. The valid storage provider (Filecoin miners) receives CID files from the client (Filecoin).
7. At the start of the data processing on blockchain, the storage provider "seals" or converts the data to the CAR file.
8. In the Filecoin network, the chain verifier node synchronizes the chain to confirm whether the data is the starter node to reach the current consensus. When additional data joins the network, the chain verifier node constantly fetches the addition to the chain and validates it to build a binary Merkle tree ("Home | Filecoin Spec" n.d.).
9. The storage provider derives a replica CID file from the root of the Merkle tree.
10. When data is ready for retrieval, the users may begin the data retrieval process.
11. Storage Miner node sends back the unsealed CID file upon the user's request through the IPFS network.
12. The user receives the data file from the IPFS network and accesses the file by decrypting the file with a private key pair.

10.4.3 Use Case: Data Request Smart Contract

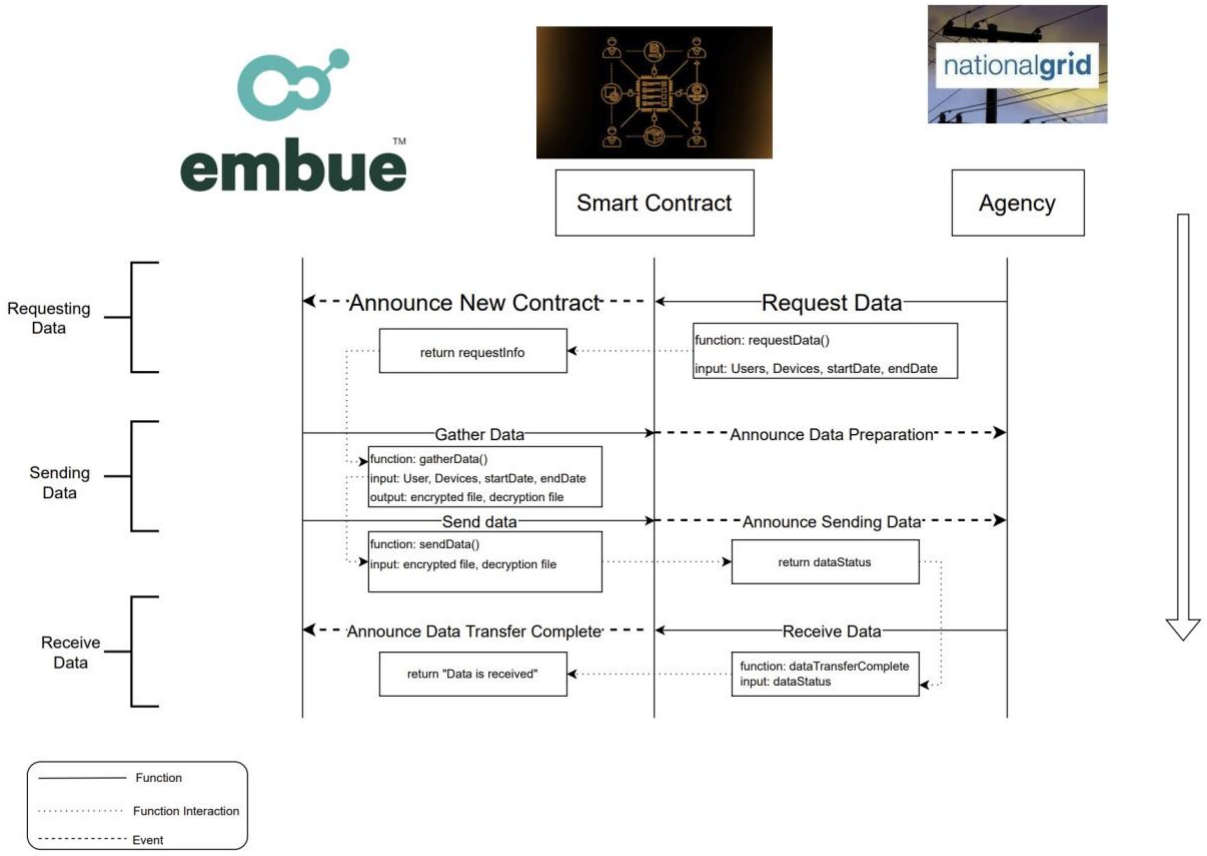


Figure 17: Sequence Interaction Diagram among Embue, Data Request Smart Contract, and Agencies

In the data request smart contract in Figure 17, Embue and different agencies (e.g., National Grid, government agencies, etc.) are the main participants.

The entire process will start with the agency representative, the user, requesting data through Smart Contract. Smart Contract will validate user's information to either create a new smart contract with requested information or reject the user. Once the new smart contract is deployed, Embue will receive the notification and start sending data. First, Embue will gather the data based on the user's request and use GunPG API or other encryption API to encrypt the data file. Smart Contract will announce the update of the fetch data process to the user. Second, Embue will send the encrypted file and decryption file to the user. Smart Contract will announce the update of the sending data process to the user. Once the user receives the data file, Smart Contract will notify Embue that data is received.

From social good aspects, Embue and agencies will benefit from the automated process. For Embue, they will save time on communicating with agencies and organizing the data file. For agencies, all they need to do is logging into the client website and selecting the devices based on their need.

Function 1: Requesting Data – requestData()

Input: User_ID, Agency_ID, Devices. Start_Date, End_Date
User_ID is the ID of the user requesting data
Agency_ID is the ID of which agency user comes from
Devices is the list of devices ID from user's request
Start_Date is the date of the beginning date from user's request
End_Date is the date of the ending data from user's request

if *User is an authorized user* **then**
    Validate user's information
    Create a new smart contract including the details of user's request
    Send user's request information to Embue
    Request status set to pending
else
    Revert request
end

Function 2: Gathering Data – gatherData()

Input: Devices, Start_Date, End_date
if *Devices have valid data* **then**
    Request status to start fetching data
    Find the first timestamp of the Start_Date
    Find the last timestamp of the End_Date
    for date in range **(Start_Date, End_Date):**
        generate the data file from the first timestamp to the last timestamp
        Request status set to fetching data from (MM/DD/YYYY)
    return encrypted file and decryption file
Request status set to fetching complete
end

Function 3: Sending Data – sendData()

Input: encrypted file, decryption file
encrypted file is the data file based on user's request
decryption file is the private key to decrypt the file

Request status set to sending data now
Send status set to incomplete
if *send status is incomplete* **then**
    Request status set to sending data (sent data size/total file size)
else
    Request status set to sending data complete
    Send status set to complete
end

Function 4 Transferring Data Complete – dataTransferComplete()

Input: request_status
if *request status is complete* **then**
    Embue receives the message "Data is received"
end

10.5 Findings and Discussion

10.5.1 Filecoin

Our goal initially for the Filecoin tutorial was to store test data onto the blockchain to see if it works, and then if it does use Embue and Cyvl's data. To set up Filecoin to store test data we followed the tutorial listed on filecoins page ("Setup" n.d.).

This tutorial broke the setup down into 4 sections, the first being to install a lotus node, the second being to create a lite-node, the third being to get an FIL address, and the fourth being to sign up with filecoin plus. A full lotus note is a computer running the "lotus daemon". This full lotus note gives the user complete access to the Filecoin blockchain, but the specs required to run a full node (64 gb ram, 8+core CPU, ect) are only seen in computers well in excess of $3-4k. Therefore to resolve this blocker instead of creating a full lotus node, I installed a lite node instead. A lite node interacts with the Filecoin network by routing all blockchain based requests to a full node managed by protocol labs (the people helping run the site). This allows me to do what I want to do, which is to store test data onto the Filecoin blockchain without having a $4k laptop.

Below are the 4 steps taken to complete the Filecoin tutorial:

1.
    a. To complete the first step for this section, we opened up Ubuntu, and updated our systems by typing "sudo apt update -y && sudo apt upgrade -y" into the command line.
    b. Next, we downloaded the latest AppImage file from the Lotus GitHub releases page. For this step, we ran into a blocker. The site tells you to type in "get + the appimage file found on the github repository"(jennijuju n.d.) into the command line, but the link doesn't work because it is outdated. The latest version if you click on the Lotus GitHub releases page link is v1.18.0, so we needed to replace v1.17.2 with v1.18.0 in the above link, and type that into the command line instead.

c. Before going to the next step in the tutorial “Make the AppImage executable”, you need to install fuse by typing in “sudo app install libfuse2” into the command line. After doing that, we went to step 3 of the Filecoin tutorial, and typed “chmod +x Lotus-v1.18.0-x86_64.AppImage” into the command line.
d. The final step is to move the AppImage file to usr/local/bin and rename it lotus. To do this, we typed “sudo mv Lotus-v1.18.0-x86_64.AppImage /usr/local/bin/lotus” into the command line. After all these steps are completed, the lotus node has successfully been installed on your computer, to confirm this go follow the linux/ubuntu/usr/local/bin directory and you will see a file that says “lotus”. If the file is not there, that means something went wrong and you need to redo the above steps. If the file is there, step 1 of the Filecoin tutorial is complete.

2.

a. For the first step of the second section in the Filecoin tutorial, we needed to create a lite-node. To do this, we typed “FULLNODE_API_INFO=wss://api.chain.love lotus daemon --lite” into the command line and the node ran in the background. To run further commands from here, you need to open a new terminal window so we did that.

3.

a. For the first step of the third section in the tutorial, we needed to create a FIL address. To do this, we typed “lotus wallet new” into the command line as instructed to on the tutorial, and after doing this it showed our public filecoin address below.
b. This address is going to be needed later, so we copied it and saved it elsewhere.
c. For the next step, you need to back up your Filecoin address by typing “lotus wallet export f1... > my_address.key” into the command line, but replace f1…. with your public address. This creates a new file called “my_address.key” in the current directory, which contains your private key.
d. To find the directory of a file in ubuntu, type the command “realpath -s my_address.key” into the terminal.
e. Once you do that, the Filecoin tutorial recommends moving the file to a drive or somewhere that you will be able to access it even if something happens to your computer. This is extremely important, because the file contains your private key which cannot be recovered if lost.

4.

a. For the final section of this tutorial, we went to plus.fil.org. We were instructed to click “Proceed” under “For Clients”.
b. We then requested to “Get DataCap”, or storage space for Filecoin storage.
c. For the next step, you are instructed to click “Automatic Verification”, then to connect to your Github account. In the Request field, enter the public address you got from running lotus wallet list. To complete this step, you need a github account over 180 days old.

We learned a lot from doing these tutorials. One of the things we learned was how cheap it is to store data on the Filecoin blockchain. You can actually store data on the Filecoin network for free if you sign up for filecoin plus and apply for storage via the notary service. The drawback to this is it often takes some time to get approved, often a few days. When compared to other storage networks like AWS, by using Filecoin instead it would save upwards of 75 dollars per gigabyte of storage used per year. Thus, Cyvl and Embue could use Filecoin to save money on data storage costs. Another thing we learned was

how useful IPFS is, as the use of a peer-to-peer network service for storing and retrieving data from the blockchain makes it much easier to integrate Filecoin with Embue or Cyvls multiple stakeholders and an already existing IoT network.

10.5.2 Web3.Storage

Based on our learnings from the Filecoin tutorial, we decided to look for a practical way to simplify the data process flow between Embue or Cyvls IoT sensor data, Filecoin miners, IPFS, and the Filecoin blockchain. While researching ways to simplify this process we came across Web3.storage, which is an API that simplifies the data process flow so much that all you need to do to use it from a user standpoint is make sure you encrypt your files before uploading them to the API. The interactions between Filecoin miners, IPFS, and the Filecoin blockchain are all taken care of by Web3.Storage making it much easier to use for companies who may not be familiar with blockchain technology. Web3.storage offers 5gb of free storage per month for making an account. Web3.storage has 2 monthly paid plans, with the lite plan offering 30gb a month for 3 dollars, and an additional 10 cents for every additional gb used. The other plan offered is the expert plan, which offers 120 gb a month for 10 dollars, and an additional 8 cents for every additional gb used ("Pricing - Storage That Grows With You" n.d.). This is cheaper than AWS, where transferring data out from AWS alone charges 9 cents per GB, and that is on top of storage fees, and web request fees (Frankel n.d.). Therefore, Cyvl and Embue can use Web3.Storage to save money if they switch over from centralized storage services such as AWS.

10.5.3 Substrate

For this project, the team was instructed to use Polkadot to determine which platform should be used for the implementation process. The team used the tutorial from Substrate to learn how the Polkadot network works. Substrate-based chains we used are connect to Polkadot networks, which granting the access to Polkadot's system for parallel transactions, cross-chain transfers, and an expanding support network ("Home | Substrate_" n.d.). From the tutorial, we would gain a basic understanding on building a minimal functional development environment.

The first blocker we encountered was the source code change. The initial tutorial video was out-of-date; thus, the team struggled on how to get started besides watching the video. In the following meeting with the Boston Blockchain Association, we realized that Substrate changed their source code a few weeks before we looked into it. We then received the new tutorial to get started.

The second blocker we encountered was a build error. After the team followed the guideline to install required packages and Rust and compile the node template with command cargo build –release, we kept getting an error and were stuck at the building process. After comparing all the command lines, we realized that the command sudo apt update somehow removed the Make Utility from the Ubuntu operating system. After running command `sudo apt install make -y`, the build process was successfully completed.

The third block we encountered was being unable to simulate the blockchain network. The team tried to start the local blockchain node using provided account; however, we kept receiving the connection error showing `Error while dialing /dns/telemetry.polkadot.io/tcp/443/x-parity-wss/%2Fsubmit%2F: Custom { kind: Other, error: Timeout }`. To address this, we looked at different ways how others fix the error. We eventually found a way to resolve this blocker from a Chinese website, CSDN (China Software Developer Network). CDSN suggested to turn off our firewall on public internet and private internet. Then, we added `–unsafe-rpc-`

`external -unsafe-ws-external` to the original command lines. Since we were developing the custom local node, there may have been an insecure situation. Therefore, we needed to inform that the remote server of the local node's corresponding interface call may be insecure through command lines.

Overall, the Substrate tutorial helps us to better understand how blockchain is running. Even though we were unable to go through the Filecoin and Web3.Storage tutorials, they focus more on how to store and retrieve the data. Through the Substrate tutorial, we learned how to build a local blockchain and simulate a network, where we can see how the node is connected on the blockchain. When we continue adding nodes that are linked to the starter node, we can increase the peers number on every node and not just the starter node. Those interactions grant the team a better comprehension of the decentralized and distributed network.

10.5.4 Blockchain Applications for Cyvl

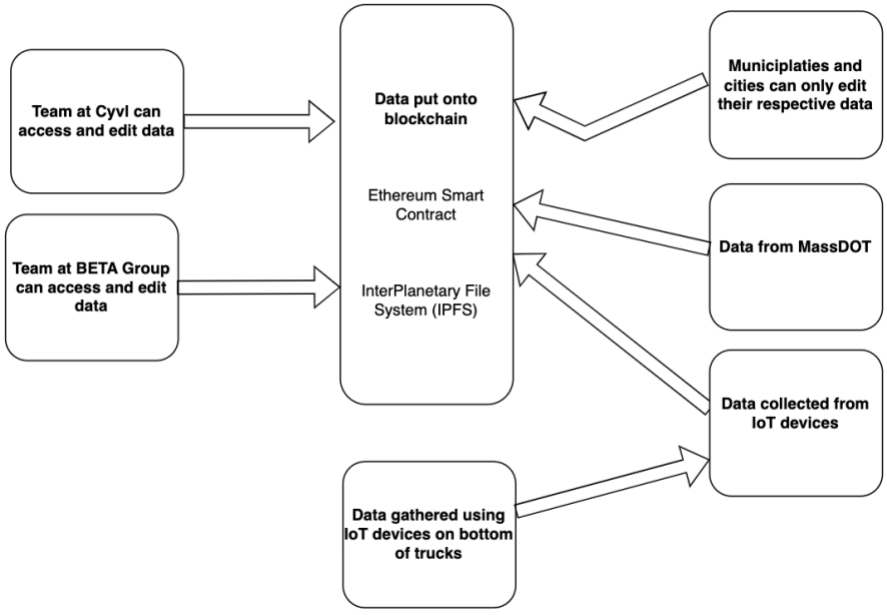


Figure 18: Cyvl Business Flow with Blockchain

The team explored a possible implementation of blockchain for Cyvl's distribution of data and found little business value to current use cases. The advantage to adding a blockchain based decentralized system would be to manage all of the stakeholders' data privileges. As Cyvl has control over the data, there is no need for blockchain use cases surrounding verification and authentication. There are better ways to control access to the data with permissioning. For example, the team recommends creating a portal with all the available data that is widely available, while keeping the current system in place for municipalities to edit data within their jurisdiction.

Another potential blockchain application regards the authentication of captured data from the IoT devices located within the trucks. Local governments hire Cyvl to measure road conditions within a specific area. The location data can be verified to be within the limits of the municipality being

investigated. For example, if data location pointed to California for a Massachusetts based municipality, the system could disallow that location. Even though this is possible, the location verification is solving a problem that Cyvl doesn't have at the moment, as the location data is for the most part very accurate and does not need the additional layer of verification.

One action that the team does recommend to Cyvl regarding blockchain is the use of oracles as a verification tool. For example, when a municipality repairs a road and updates the record in the database, there is currently no reliable way to verify this, as it is not feasible to drive the data gathering vehicle over all of these streets before the entire municipality is reassessed in a following year. As a result, the team recommends Cyvl verify the existence of construction on the road through something similar to a feature on the Waze app where residents can confirm the update on the road.

In short there is really only one major scenario in which a blockchain application would be of use. In this scenario, there is a complex system where Cyvl is entering its data into a third party with their competitors. Even in this situation, the business value would go to the third party that created this platform and not to Embue specifically.

10.5.5 Blockchain Applications for Embue

The team theorized a blockchain project of Embue that has potential business value for the future. The implementation, based on the proof of concept and diagrams above, consists of device authentication with randomly generated key pairs and ledge managed storage that would include timestamp and content verification for data payloads. This system would provide multiple potential benefits for Embue.

Firstly, the plan would reduce storage costs by transferring the data from AWS which charges to Web3 storage, which reduces the storage cost from $6.25 per month per gigabyte to $0.08 per month per gigabyte (Levin 22). Next, blockchain adds a timestamp redundancy, as this system itself can calculate an approximate timestamp for each record, which may be useful in confirming any errored timestamp. Furthermore, this system has a built in authentication system, which prevents malicious actors from spoofing a device to send inauthentic data. Moreover, this blockchain based system can utilize smart contracts to predict against deletion and revision. Finally, it would make sending the raw data to other entities, such as National Grid, easier.

10.6 Future Work

Although blockchain technology currently has limited applications for practical use right now, once the industry catches up, based on our research we feel that this will change in the future. A good analogy from the BBA blockchain expert brought up was the internet in the early 1980s, where it was extremely early in the development stage of the internet and the industry was not yet ready to utilize everything it had to offer. With blockchain, we feel there is a lot of room to expand on the minimal viable product we produced with Filecoin. The big issue Embue is trying to solve is they are concerned about how centralized their IoT network is, because if a malicious actor hacks into one IoT device, they have control over the whole network. A potential future use case to build on the proof of concept we developed would be figuring out a way to integrate a blockchain based storage system like Filecoin with already existing IoT systems. Right now, one of the biggest challenges to the adoption of blockchain based storage systems is that one would have to completely overhaul the existing IoT system Embue is using to adopt a blockchain based storage solution, as there is currently no viable way for on chain data to interact

with off chain data. This is known as the blockchain oracle problem, and it applies to this specific use case as something is needed to facilitate the communication between the blockchain and off chain IoT devices. Another potential project that we recommend exploring in the future is exploring a way to validate the identity of individual IoT devices on chain. Since the IoT ID signature is currently off chain, an oracle is needed to bridge that gap and verify that the signature on chain matches the ID of off-chain IoT devices ("What Is the Blockchain Oracle Problem? Why Can't Blockchains Solve It?" 2020).

Future work that builds off of the proof of concept we developed has significant implications from an economic and social good standpoint. If a way to validate the identity of Embues' IoT devices on chain by connecting on-chain data to an external off-chain system is accomplished, that would be a solution to the oracle problem that can be applied to other blockchain use cases. The significance of this is the same oracle problem that is a barrier to this specific use case for Embue is a barrier for upwards of 90 percent of all blockchain use cases. Thus, if future projects building off of our proof of concept for this MQP discover a way to validate the identity of Embues' individual IoT devices on chain, the same mechanism that would solve this problem could be used to render many other seemingly unrelated blockchain use cases that had no practical real world value to all of a sudden become viable. The economic implication of this is that people would pay Embue money to use their framework to develop their own blockchain based security system, as right now most companies are using centralized cloud computing platforms like AWS which is much more expensive than a blockchain based storage system like Filecoin, and is not as secure as discussed in our executive summary. The developers of this potential future project would also make money themselves by patenting it, as the solution would make a lot of previously unviable blockchain use cases viable. That 90% figure of blockchain use cases being limited by the oracle problem suddenly being viable means that there is a lot of money to be made, and they would get a piece of it.

In addition, the team highly recommends finishing the Substrate Tutorial before the actual implementation on Filecoin and Web3.Storage. The Substrate tutorial is suitable for people who do not have much background information about how blockchain runs. It starts with the basics from building a local blockchain to integrating a light client node with minimal hardware requirements. After the Substrate tutorial, the team can implement the code based on our PoC with Docker. Docker will include all the dependencies and required packages to get started on Filecoin and Web3.Storage. Thus, the team will not encounter our previous blockers to re-solve those problems.

From a social good standpoint, if a practical way to validate Embue's IoT ID signatures on chain is discovered, the mechanism used to do that would be a solution to the oracle problem, and could be used in a variety of other blockchain use cases that are currently not viable. For example, another project a co-author is working on is a blockchain use case that offers an alternative to the current cap and trade carbon credit system. The oracle problem means that there is no way to validate the identity of the carbon credits on-chain, which means a blockchain based carbon credit token system is not viable right now. This shows how wide reaching the impact of future work that builds off the proof of concept developed in this project could be. In regards to identifying a way to validate Embue's IoT ID signatures on chain, the solution that allows this could also be used for seemingly unrelated blockchain use cases that can help with climate change, financial literacy, etc.

11. Assessment

11.1 Accomplishments

In this project, our team was able to create machine learning models for both Embue and Cyvl and a proof of concept with blockchain that can be built off to add security to IoT networks. To achieve this, we met a few major milestones. We implemented the Agile Scrum project management methodology, to help us organize projects with two companies with different backgrounds. Agile also provided concrete guidelines to tackle major technical challenges of this project. Our sponsors and future teams can use this along with the recommendations we have laid out, to continue the work we have done.

The Future Work section of this paper are our recommendations for Embue and Cyvl on what they might do if they would like to continue this project. Additionally, we presented our findings, and we also detailed some next steps for Embue and Cyvl moving forward. Our team believes these recommendations will guide both companies and future project teams to build off the things we have accomplished over the course of the project. The project has potentially huge financial value and social benefits.

11.2 Business Learnings

11.2.1 Leadership Learnings

Working with Embue and Cyvl provided the team with many insights into teamwork and working with multiple stakeholders. There are a number of key takeaways that helped us improve over the course of this project and we believe future groups can benefit from this. The first takeaway we had is strong leadership and a clear vision is important to a successful project. This was something we found challenging in the beginning which caused us to struggle. Over the next few weeks, our project benefited greatly from having well-defined project goals. This was accomplished through our daily stand-up meetings both with the group, our advisors and sponsors. This helped our team with risk mitigation and risk management throughout the course of our project. A lot of our experience throughout the project has been a learning experience as we prepare for industry.

One aspect that helped the team develop a stronger vision was through learning how to identify scope creep. Throughout the beginning of the project, the blockchain related deliverables changed very often. It was unclear what an MVP would look like for a specific part of the project. This made it very hard to start the project when the goal remained unclear. Through the process of forcing the project to take a goal, the team learned about scope management and scope creep. Moreover, the team learned how to prevent scope creep in the future and how to identify it earlier. Overall, scope creep can lead a project to start behind schedule or never start at all by making it impossible to start as the requirements are constantly shifting. The team also learned how to manage the scope after a late start due to scope creep by narrowing the scope to something reasonable. If the goal is to first scan and try before scaling. You can't scan the entire time and never try anything. At some point a team has to pick a project, go with it and learn from it.

Furthemore, the team had to learn how to balance many stakeholders. For example two separate startups, a sponsor, two consultants and three direct advisors all were stakeholders. Each stakeholder had their own separate interests and goals. The team learned that you can not please everyone. Additionally,

the team learned the importance of listening to the stakeholders first before being understood. Each stakeholder needed to have their business needs met or at least an explanation on how their needs were impossible.

Communication was something our team struggled with throughout the project, both internally and with our sponsors. We did not start our project with setting up Jira and as a result we had to find other ways to document our sprints. We ended up using Excel and Google doc to document so the whole team could easily access information. This approach resulted in our scrum meetings being disorganized and focused on tasks rather than user stories. We often did not know how what we had worked on impacted progress on the user stories and a lot of time was spent having discussions when not necessarily appropriate. When the team was able to start working on Jira, it took some adjustment to better organize our scrum meetings. Through our highs and lows, the team continued to work together and manage different leadership roles.

Externally, our team learned how to communicate different things to different people. Different styles of communication worked on different sponsors. For example, some sponsors were very attentive to slack, and checked often, while others did not check as frequently. Some sponsors liked having defined, pre-planned meetings while others preferred doing things completely on the fly. The team had to learn each person's personal meeting style to collaborate effectively with the collaborators, as their input was vital to completing the project and ensuring it met their business needs in a helpful way.

Dividing tasks up in a way where there was still some interconnectivity was challenging. The team's MQP was effectively divided up into three smaller projects, one for each of Embue, Cyvl and blockchain. The team assigned each project at least one developer, which in retrospect was a good idea.What the team could have done better was communication between teammates. For example, the developers on the Cyvl and Embue portions did not receive updates on the blockchain portion and vice versa. This meant that it was hard to support each other and ask each other for help when needed. The team learned that it is important to have overlap and be able to delegate tasks and responsibility.

Effectively communicating with the sponsors and each other was also challenging. Sometimes we would organize a meeting with a sponsor to talk about a specific part of the project, but they would be interested in other parts of the project. For example,one of our meetings was intended to be part of the technical side of blockchain, where blockchain developers would explain the different technologies used. However, the wrong people were at the meeting and the purpose of the meeting was not expressed before the meeting started. It would have been more effective to plan for this and be ready for the business questions anyways. Additionally, sometimes there were meetings when a collaborator would use technical language with the team that we did not understand and that was challenging to get information because it was being stated in a complicated way. In this situation, the team could have slowed down the collaborator in order to make sure they understood the situation instead of staying silent and feeling lost.

Finally, another important takeaway is the willingness to express different ideas with superiors. Our advisors and sponsors were open minded and encouraged the team to speak up and ask questions for clarification.

11.2.2 Culture Learnings

One of the experiences the team gained was utilizing WorcLab. It's a building at Worcester Downtown where both Embue and Cyvl rent a coworking space. The team usually had our weekly Tuesdays meetings with Embue at WorcLab. It was beneficial for the team because we were able to work effectively as we present our weekly updates to Embue and also ask any questions we had. On the other

hand, our meetings with Cyvl were mostly on Zoom, since they are located in the Boston area. Overall, WorcLab provided us with a great space to work and the team had a good experience. The downside to this was transportation, which the team mostly carpooled and difficulties finding free parking in Worcester city center.

11.2.3 Time Management Learnings

Working with Agile Methodology required the team to keep up with the demands of the project. In some of the sprints, the team did not have the best time management. For instance, the team had to work during Thanksgiving break and the last week of the project to finalize our paper and deliverables. Although challenging, it boosted the team morale to continue to deliver our good work. Besides the team's regular weekly meetings, we also did team reflections on Tuesdays where each team member shares their challenges and roadblocks with others for solutions. We also reached out to our advisors, sponsors, and BBA Expert for their guidance throughout the project.

11.2.4 Team Management Learnings

In order to divide the different parts of the project to complete the project, team management was critical. Sometimes, team members have to meet in person at WorcLab to share ideas and work together. We also continued to work remotely anytime we cannot meet face-to-face. Communication was a crucial aspect for the team's success. The team mainly used discord and text messages for immediate responses.

11.3 Technical Learnings

11.3.1 Machine Learning

Working on this project taught the team many technical skills ranging from software, blockchain and machine learning models which we implemented. We used linear regression for Cyvl's data, and learned why other machine learning algorithms wouldn't work in Cyvl's case.

The team also implemented a K-means clustering model that isolates readings with an open window to analyze the Embue data. Through this, the team learned about Python and many of its packages, and when to use specific machine learning algorithms. The Embue lead learned what the elbow method is and how to implement it within K-means clustering, and successfully determined a useful amount of clusters. Furthemore, the team learned how to use sklearn to implement K-means clustering and how to change the number of clusters and maximum iterations. In addition, the team learned when it is appropriate to apply clustering.

Next, the team learned about normalization and how to properly relate machine learning features to each other. This also used python and sklearn. We learned why normalization was important, what were the risks of failing to implement it, what the different types of normalizations were and how to effectively choose a normalization method. We also learned the challenges of expressing what that normalization value means and how to make it useful once the machine learning process is finished. For example, a difference in 15 percent could be really different things depending on the data set and that is a problem the team had not experienced before.

A supervised method the team learned was decision trees. Usually decision trees are used as a supervised model, but in this context the team had to learn how to adapt a model to fit unusual

circumstances. There was also a lot of uncertainty the team had to learn how to deal with. For example, knowing how to present that information is new, as in the past most findings have been more definite. While it is impossible to know if the information we are presenting is correct, the team still needs to present them, and this changes the way we present, particularly in the wording.

For example, the team learned how to iterate through a dataframe and use more complex operations, such as referencing a different dataframe that is dependent on a specific column value. We also learned the cost of that type of action in terms of computational efficiency, Furthermore, through both the Cyvl and Embue sections of the project, developers learned how to integrate data found from the web with company provided data, even when they did not have a primary key.

All of the code we wrote and used was in Python, which the team was familiar with. Overall, we got to learn not only about machine learning models and blockchain, but we also learned about various technologies and engineering skills while working with multiple stakeholders. Data cleaning, clustering and blockchain were learning experiences for the team as this was a real-world project that required us to deliver a minimum viable product at the end.

11.3.2 Blockchain

From this project, the team was introduced to many new softwares and platforms to investigate blockchain implementation. These platforms included Filecoin, Substrate, Web3.Storage, IPFS and many more. We explored how blockchain can be used to improve Embue's IoT network. We also determined that a blockchain-based storage system would be a good solution to this problem and determined Filecoin would be the one of best softwares. We produced a proof of concept with Docker that both Cyvl and Embue can build off to add security to their IoT networks in the future.

Before this project, the team had only a basic understanding of blockchain's benefits in the field of financial technology (FinTech) from the FIN3300 class.

Through the project, the team gained a better understanding of the potential uses of blockchain technology in various industries, including healthcare and supply chain management. They learned that blockchain's decentralized and distributed nature could improve transparency and security in these industries. We also learned more about smart contracts, which are digital protocols that enforce the terms of an agreement automatically. When the smart contract is utilized in blockchain implementation, it can enable secure and transparent transactions. They can be used in a range of applications, including financial transactions, supply chain management, and voting systems. By using a decentralized and immutable ledger, smart contracts provide an efficient and secure way to conduct business.

We learned that in the healthcare industry, blockchain can be used to securely store and share patient data, allowing for better coordination between healthcare providers and enabling more efficient delivery of care. From the publication, *Blockchain-Based Management for Organ Donation and Transplantation* (Hawashin et al. 2022), we learned from their proposed blockchain solution to strengthen our comprehension on smart contract utilization. Because details of donated organ transplantation are confidential, they used the Ethereum blockchain to develop their own private-permissioned blockchain to improve data privacy, security, and confidentiality. Blockchain network serves as the basis for recording transactions and events permanently to ensure accountability and data provenance. The developed smart contracts for organ donation and transplantation must be deployed on the blockchain to ensure they are accessible at all times. For the organ donation phase, the smart contract interacts among different participants in three phases, which include creating a waiting list, accepting donors, and auto-matching. For organ transplantation, the smart contract includes removing organs, delivery process, and

transplanting organ phases. In both scenarios, the smart contracts are responsible for announcing events and recording the main attributes from all parties once the corresponding API functions are called. The publication inspired the team to create Embue's smart contract use case (Figure 17), the potential API functions, and high-level architecture diagram.

In addition, the team gained a deeper understanding of IoT devices and their relationship with blockchain. IoT (Internet of Things) refers to a network of interconnected devices that can collect, share, and exchange data. Examples of IoT devices include smart home appliances and wearable technology that communicate with each other over the internet. The relationship between IoT devices and blockchain is one of integration and collaboration. IoT devices generate large amounts of data, which can be stored and managed using blockchain technology. This allows for secure and transparent record-keeping, as well as real-time tracking and analysis of data. In turn, blockchain technology can also enhance the security and reliability of IoT devices. The decentralized and distributed nature of blockchain makes it difficult for hackers to access sensitive data, protecting the privacy of IoT device users and preventing security breaches.

After completing the tutorials for Filecoin, Web3.Storage, IPFS, and Polkadot, we applied our learnings to build a high level architecture diagram for Embue. The team had already narrowed down the proof of concept as a way to improve Embues security system using blockchain. We understood that Embue was concerned about how if a malicious agent had access to one IoT device, they had access to the entire network. We also understood from our research that the way blockchain solves this problem is via decentralization, so instead of having all the IoT data stored on a central network, they are distributed among various Filecoin storage providers. What we didn't understand was how to go about integrating Filecoin with Embue's existing IoT security system in a way that ensures privacy of the data, avoids requiring them to overhaul their existing IoT security system, and is easy for outside stakeholders to interact with. Thus, we looked for a good way to piece together Embue's existing IoT network, outside stakeholders, and the blockchain. During our research, we determined that a good way to break things down would be to create an architecture diagram for our use case.

When building our architecture diagram we broke it down into 4 layers. First, we had the data input layer. For this layer, we identified what data we need to store (Embue's IoT sensor data), and what Embue needs from us before storing the data on the blockchain. Since blockchain does not inherently provide privacy, we needed to encrypt the data before storing it on a chain. From here, we researched encryption methods compatible with a large scale IoT network, and we came across GnuPG. GnuPG is a free software with an easy-to-use interface that utilizes public key cryptography. In a public-key system, each user has a pair of keys consisting of a *private key* and a *public key*. A user's private key is kept secret, and the public key may be given to anyone with whom the user wants to communicate. Since Embue may want to share different parts of the data with different stakeholders, this is especially useful.

Next, we had the storage layer. For the storage layer, we needed to figure out what software is used to transfer the data onto the Filecoin blockchain, how the Filecoin blockchain interacts with the software we use to help with the storage process, and how to make this process easy enough to be practical for Embue to use. From our research, we learned that Web3.Storage is an API that manages the data flow process involving the data file you upload on the site, the IPFS network, and the process to find Filecoin storage nodes to store your data on to the Filecoin blockchain. The way the Web3.Storage API does this is it converts the encrypted data file you uploaded to the site into CID format. This is important, because CID format is a format that is readable by the IPFS network. Moving the data onto the IPFS network is a necessary step because CID files stored on the IPFS network can be retrieved by anyone

(side note- This is another reason why the initial encryption step with Gnupg was so important!). In order to integrate blockchain with Embues's IoT security network, we need something to facilitate the storage as well as the retrieval process so the stakeholders can access it, which is exactly what IPFS does. Once the CID files are stored onto the IPFS network, they are dispersed into separate IPFS nodes to prepare the files for data aggregation from a variety of Filecoin miners (this is where the decentrailization comes in) to prepare it to be stored on the Filecoin blockchain.

We then have the application and extension layers. These layers are what facilitates the interaction between the Admin (Embue), user (various stakeholders), and the Filecoin blockchain. The application layer is where the user, or Embues different stakeholders, requests data from the admin (Embue), and where the admin grants or denies access. The Extension layer is essentially the step that facilitates the interaction between the storage layer and application layer. For the extension layer, we have a smart contract that automatically executes the transfer of data from the admin to user. Although technically the extension layer is the third layer and the application layer is the last layer, we will be explaining the application layer first because that will make it easier to understand what the extension layer is doing.

In the application layer, the user requests access to Embues IoT data. Once Embue receives this request (automated by the extension layer), they are given the option to accept or deny the request, which is then relayed back to the user (automated by the extension layer). For example, anyone who wants access to Embues' IoT sensor data such as the national grid, national government, etc. requests access to specific IoT sensor data using a mobile application. Embue then grants or rejects access to IoT sensor data. If access is granted, the user will be able to retrieve the the file and decrypt it. If access is denied, they will get a notification on their application that informs them their request was denied.

The extension layer automates the interaction between the Admin, User, and the Filecoin blockchain. The way it does this is by the use of a smart contract. For example, when the admin (Embue) grants access to the IoT sensor data for the user, the smart contract executes a command that begins the retrieval process from the blockchain to the user. Once this command is executed by the smart contract, the Filecoin miner sends an idenical copy of the CID file to the IPFS network for the user (vairous stakeholders) to retrieve. If Embue denies access, the smart contract executes a command to shut the process down, and the retrieval process does not occur.

Overall, building the architecture diagram was a great learning experience for us because it forced to learn about the various platforms we used at a deeper level. For example, knowing what Filecoin is was not enough. To build the architecture diagram we needed to not only know what Filecoin is, we had to know how it interacts with the platforms within the storage layer such as Web3.Storage and IPFS, as well as how Filecoin interacts the Extension, application and data input layers. To know this, we had to have an in depth understanding of all 4 layers and the interactions between the various platforms we used. It took us several tries to make this diagram, as we learned from experience that a lack of understanding of even one platform resulted in a snowball effect, as that platform we didnt understand well enough interacts with various other platforms, so that means there would be a flawed understanding behind the foundation of those interactions as well. This means that a flawed understanding of even one of the platforms we used made the entire diagram a disaster with flawed interactions everywhere. Thus, we had to develop a deep understanding of every platform we used to make an accurate diagram.

Besides the different platforms, the team also gained experience with different programming languages and APIs. We created the fake data file through the Faker.js JavaScprit library. For example, when we generated the fake data file based on Embue's IoT data, we created the universally unique identifier (UUID) for each device. In addition to JavaScript experience, the team gained experience with the Web3.Storage API with Node.js when we uploaded our fake data file to the client. Moreover, the team gained more experience with the Linux system when running the Filecoin network and Substrate blockchain. We encountered different blockers that were not stated in the official documentation. We

spent lots of time solving those blockers by browsing various tech websites, such as Stack Overflow and CSDN. Thus, the team learned how to use Docker to create proof-of-concept work for the future team to build on and prevent future blockers.

Through this project, the team gained a better understanding of blockchain's potential uses in various industries, including healthcare and supply chain management. We learned about smart contracts and how they can be used in blockchain implementation for secure and transparent transactions. The team also gained a better understanding of the relationship between IoT devices and blockchain. The integration and collaboration between the two allowed for secure and transparent record-keeping and real-time data tracking. In addition, the team understood that blockchain could also enhance the security and reliability of IoT devices. Based on these learnings, the team created a proof-of-concept work that allows the future team to build more detailed implementations for Cyvl and Embue.

11.4 Joshua DeBare - Embue Lead

During the process, I realized how much time went into data cleaning. Several times, either the project changed, or the team realized that the data was not complete. For example, even when I finished creating a dataframe that consisted of all the data they thought necessary to predict window openings, I realized that a better representation may be grouping rows together. For example, it is easier to see a window is open if we can also see the rows above and below it. As a result, I did not have as much time on the machine learning portion as we expected. Additionally, I learned how to use Pandas to a much better degree than before. This was my first time using the agile methodology and it was harder than I thought, as I felt that often the user stories in the beginning of the sprint would sometimes be very different than at the end, making it difficult to plan. Finally, I learned how to work with different stakeholders as a part of a team, as many different people wanted different things from the project and it was difficult managing all of their expectations.

I also learned how to do project management and balancing multiple positions. For example I was the Embue Lead, but I ended up also offering a lot of support to Cam and Suki on the blockchain portion of the project. I helped them come up with a timeline and prioritize tasks and took busy work off their plates. I did most of their citations for them. I ran the scrum meetings for the team and attended a couple of blockchain meetings. I learned how to balance these responsibilities with the work I was doing as Embue Team lead. For example I prioritized pressing work with deadlines and made sure to follow up with Cam and Suki on how I could support them and do what I could. This was a great learning experience on how to work with teammates on a large project as well as capitalize on people's skills.

11.5 Kelsey Moody - Cyvl Lead

I had never utilized the Agile Scrum methodology, or Jira (our project management software), and this project exposed me to both. During this project, I learned firsthand the pros and cons of the Agile Scrum methodology, and what user stories, issues, and epics are. I learned how much Agile can help organize a project, and .

I also hadn't used GitHub before, and I learned a lot about how GitHub works throughout this project. While I had used Slack before, I gained experience using it in a professional environment, as opposed to a casual environment. I gained a lot more experience with Jupyter Notebook as well, because I used that to work with the data.

While I had some Python experience coming into this project, I am now very comfortable with Pandas, Geopandas, and scikit- learn after these last seven weeks.

Through Geopandas, I gained experience with loading and cleaning and working with geographical data and shapefiles. Through Pandas, I learned about cleaning data, subsetting data, and merging dataframes.

I also learned the best practices for searching for open data sources on the web. Our project required many data sources, and I spent a lot of time searching online for reputable data sources. I struggled to find data until I asked our Cyvl contact and a civil engineering professor where to find pavement data.

I learned how to run a linear regression with scikit-learn, and how to view a summary of the regression. While I knew how to interpret some values of the summary (p-values, r-squared values), I learned how to interpret more aspects of the summary. I also learned how to run a correlation test on the data, and what I can learn from a correlation test. While the linear regression didn't do as well as we hoped, we still learned more than I thought we would about the data through the summary and correlation test.

I also learned a lot about how to overcome project challenges, and how to deal with changing requirements. Our project changed multiple times, and I had to be flexible enough to overcome that. We also encountered scope creep throughout the project, and I learned the best practices to manage that issue. I also learned a lot about communicating my concerns and issues throughout the project. I had a couple of times where I got stuck on one line of code, and spent hours trying to figure it out. Then, I would ask my teammates for help, and we would figure it out within minutes. Had I asked for help sooner, I could've saved time.

11.6 Cameron Morreale - Blockchain Lead

Throughout this process I learned a lot about blockchain, initially went into the project with my blockchain knowledge being limited to just Bitcoin. Throughout the process I attended a lot of blockchain conferences, some with the Boston Blockchain Association and some with Mass Fintech Hub. I also attended 2 blockchain hackathons where I learned a lot about smart contracts and a variety of possible use cases for blockchain. I see all of these things as part of my journey towards learning the skills I needed to contribute to the MQP project, so I will be discussing the skills I learned from the conferences and hackathons I attended as well as the meetings and research I did directly on the MQP. As I worked on my MQP, I also learned about some of the limitations of blockchain as I ran into several blockers throughout the project, and needed to pivot multiple times.

Throughout this process, both directly working on the project as well as my outside learnings I gained a lot of technical experience as well. I am now very comfortable using Ubuntu and use it for all my projects. Before working on this project, I had never used Github, and am now comfortable enough with Github that I am using it not only for this project, but am also using it with my side projects. I got exposure to many different programming languages, some that I was familiar with before like JavaScript and Python, and others I had never heard of such as Rust, Solidity, Reach, and Go. I worked with a variety of blockchain platforms such as Filecoin, Algorand, Tron, Polkadot, IPFS, and Storj.

I also developed soft skills such as how to divide up the work in a group with people having a variety of differing skills and backgrounds in a way that sets up everyone to succeed. It is especially important in a project like this to be able to objectively evaluate what your strengths and weaknesses are,

so you don’t end up banging your head against a wall trying to do something for 10 hours that another group member can do in one hour. This is something I struggled with at the beginning of the project, but realized that I was hurting the team more by doing this than asking for help on the things I'm struggling with so I can be more efficient and pull my weight.

11.7 Qingbei Shang - Blockchain Lead

During the project, I learned a lot about blockchain and teamwork.

Before this project, my understanding of blockchain was limited to cryptocurrency transactions and Non-Fungible Tokens (NFTs). I had only heard of one blockchain application, Bitcoin. However, throughout this project, I read many research papers on the various applications of blockchain and how it can benefit different industries. These studies showed me that blockchain technology has much more potential than I previously realized.

Through meetings with the Boston Blockchain Association, I learned that implementing a blockchain application is not as straightforward as I had thought. Running a full node on the blockchain requires a high level of computer hardware. I also gained experience with Linux systems while installing dependencies and packages for platforms like Filecoin and Substrate.

In terms of teamwork, this project gave me the opportunity to work with people from different backgrounds and courses within the different majors. This was a new experience for me, as I had only previously worked on team projects with peers who were taking the same classes and had the same level of knowledge and skills. At the beginning of the project, it was challenging to get everyone on the same page, especially when it came to setting up the GitHub repository, installing Python packages, and using different languages. However, through collaboration and communication, we were able to overcome these challenges and complete the project successfully.

Overall, this project greatly expanded my knowledge of blockchain and improved my teamwork skills. I now have a deeper understanding of the potential applications of blockchain technology and the challenges of implementing a blockchain application. I also learned how to work effectively with people from different backgrounds and skill levels in a team setting.

11.8 Mabel Konadu - Business Lead

This project has been a great learning experience. Throughout this project, I have learned a lot working with multiple stakeholders with different backgrounds. This is my first experience leading a real life project where we had to produce a functionable deliverable. I had never used Agile Scrum methodology or Jira before. I had to download softwares like GitHub, Pycharm, to be able to see the progress of my team with software developments. I also had never used discord nor slack for group projects before. I see all these as part of my learning journey experience and preparing myself for industry. I have learned how to effectively work with people from different backgrounds and skills of improving my project management abilities.

I have also learned a lot about project management skills and how to overcome project challenges. This project was my first exposure to multiple stakeholders and thus required me to work extra hard to meet every expectation. This process was quite demanding. Our project scope changed multiple times, which created challenges for execution. With best practices and Professor’s guidance, I learnt through this

project of my shortcomings. There were times I would ask my team members for help in delegation to deliver quality work.

Overall, this project greatly expanded my knowledge both on business and technical. Although the technical aspects of this project was not in my expertise, I have learned alot about machine learning models and blockchain applications.

12. Conclusion

The FinTech industry uses machine learning, blockchain, and many more technologies. The Worcester IS FAB Lab of Worcester MA utilizes blockchain to allow various Internet of Things (IoT)-based enterprise startups to provide auditability of data for social good. This data is then used for machine learning and AI purposes to promote social good.

Under the sponsorship and guidance of the Worcester IS Fab Lab, our team worked with Embue and Cyvl to implement blockchain and machine learning for social good.

12.1 Embue

Large apartment buildings can be energy inefficient, leading to expensive bills. Units tend to lose energy anytime a tenant opens a window while the central heating system simultaneously runs. Embue tracks temperature, humidity and more to increase energy efficiency for large apartment units – but they do not track when windows open. If Embue could turn off the heating system when windows are open, they could save energy and money for their customers. This decrease in energy bills could make housing more affordable. Energy savings can also lead to less energy emissions which, if applied on a large scale, can make renewable energy more feasible.

The group used Python, Pandas, and Sci-Kit Learn to implement an unsupervised k-nearest neighbors model to determine when a window is open. We determined that five clusters is the optimal number of clusters for the model. We also suggested multiple ways for Embue to improve the accuracy of this model, such as adding sensors to windows, so that they could do a supervised machine learning method. This model and these suggestions will help tenants save money and energy in the buildings where Embue's Iot devices reside.

12.2 Cyvl

Municipalities must keep their pavement in good condition. However, limited funds restrict towns' ability to repair roads, and they must prioritize which roads to fix. Therefore, municipalities want to predict future road conditions, so they can determine where they should allocate resources. Cyvl uses Iot devices to determine the pavement condition index (PCI) for municipalities. These PCI scores are displayed so that municipalities can track pavement conditions. Cyvl does not have the data to predict these PCI scores, so the municipalities only know the current conditions of the roads.

Our team used Python and many of its packages to create a linear regression model to predict PCI scores. Our model had a r-squared value of 0.434, and we also determined correlation between variables. We suggested multiple ways for Cyvl to improve this score in the future, including other potential regressions to investigate. This regression will help municipalities predict when roads will need to be repaired. This proactivity could lead to a decrease in accidents associated with poor pavement conditions, which improves the social good of the municipalities.

12.3 Blockchain

Our team explored how blockchain could improve the auditability of Embue's IoT network. We explored the pros and cons of a wide variety of blockchain based storage systems and determined Filecoin

would be best. Filecoin differs from a traditional non-blockchain storage system because they have multiple different storage providers storing the data with low energy consumption and low fees. Additionally, Filecoin uses a proof of storage to ensure that the data stays intact over time. Using Filecoin, if someone accessed one sensor, they wouldn't be able to access anything else.

We successfully gathered some of Embue's fake data and stored it on the Filecoin blockchain. We produced a proof-of-concept work that both Cyvl and Embue can build on to increase auditability of their IoT networks. In the future, Cyvl and Embue could also leverage the Filecoin network for machine learning. They would be able to see the timestamps of when the sensor is on and off – this will determine what data to filter out to improve their algorithms.

References

Abrol, Ayushi. “What Are Blockchain Nodes? Detailed Guide -.” Blockchain Council, September 9, 2022. https://www.blockchain-council.org/blockchain/blockchain-nodes/.

Abu Ali, Najah, Abd-Elhamid Taha, and Ezedin Barka. “Integrating Blockchain and IoT/ITS for Safer Roads.” *IEEE* 34, no. 1 (January 31, 2020): 32–37. https://doi.org/10.1109/MNET.001.1900146.

Active State. “What Is Pandas in Python? Everything You Need to Know,” August 9, 2022. https://www.activestate.com/resources/quick-reads/what-is-pandas-in-python-everything-you-need-to-know/.

Al-Mamany, Dier, Ali Hussein Hameed, and Raquim Zehawi. “Development of Pavement Management System on Road Network Using Geographic Information System Method- Kirkuk to Erbil Highway.” *Design Engineering (Toronto)* 6 (January 2021): 36–47. https://www.researchgate.net/publication/352694425_Development_of_Pavement_Management_System_on_Road_Network_Using_Geographic_Information_System_Method-_Kirkuk_to_Erbil_Highway.

Andoni, Merlinda, Valentin Robu, David Flynn, Simone Abram, Dale Geach, David Jenkins, Peter McCallum, and Andrew Peacock. “Blockchain Technology in the Energy Sector: A Systematic Review of Challenges and Opportunities.” *Renewable and Sustainable Energy Reviews* 100 (n.d.): 143–74. https://doi.org/https://doi.org/10.1016/j.rser.2018.10.014.

Bizzaco, Michael. “Storj Decentralized Cloud Storage Review.” Tech Radar, October 5, 2022. https://www.techradar.com/reviews/storj-decentralized-cloud-storage.

Boston Blockchain Association. “Home.” Accessed December 2, 2022. https://bostonblockchainassociation.com/.

Chainlink. “What Is the Blockchain Oracle Problem? Why Can’t Blockchains Solve It?,” August 27, 2020. https://blog.chain.link/what-is-the-blockchain-oracle-problem/.

CPrime. “What Is AGILE? - What Is SCRUM?” Accessed December 7, 2022. https://www.cprime.com/resources/what-is-agile-what-is-scrum/.

Cyvl.ai. “About.” Accessed November 30, 2022. https://www.cyvl.ai/about.

Geeks for Geeks. “Blockchain - Public Key Cryptography -,” July 5, 3033. https://www.geeksforgeeks.org/blockchain-public-key-cryptography/.

Di Russo, Julia. “4 Reasons to Use PyCharm for Your Next Python Project.” Medium, October 2, 2019. https://towardsdatascience.com/4-tips-to-get-the-best-out-of-pycharm-99dd5d01932d.

“Digest of the United Kingdom Energy Report 2017.” Duke Energy Corporation, July 2017. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/643414/DUKES_2017.pdf.

Embue. “Embue - Home Page.” Accessed December 2, 2022. https://www.embue.com/.

Ernst, Michael. “Version Control Concepts and Best Practices.” University of Washington, March 7, 2022. https://homes.cs.washington.edu/~mernst/advice/version-control.html.

Frankel, Ryan. “AWS S3” Pricing (Calculator & Sample Plan).” HostingAdvice.com. Accessed December 9, 2022. https://www.hostingadvice.com/how-to/aws-s3-pricing/.

Fusai, Gianluca, and Andrea Roncoroni. *Implementing Models in Quantitative Finance: Methods and Cases*. New York: Springer, 2000.

Faker. “Getting Started | Faker.” Accessed December 2, 2022. https://fakerjs.dev/guide/.

Filecoin. “Home | Filecoin Spec.” Accessed December 9, 2022. https://spec.filecoin.io/#section-intro.arch.

Filecoin. “Pricing - Storage That Grows With You.” Accessed December 9, 2022. https://web3.storage/pricing/.
Filecoin. “Setup.” Accessed December 9, 2022. https://docs.filecoin.io/get-started/store-and-retrieve/set-up/.
Geeks for Geeks. “Introduction to NPM Scripts,” July 5, 2022. https://www.geeksforgeeks.org/introduction-to-npm-scripts/.
gnupg.org. “GnuPG — The Universal Crypto Engine.” Accessed December 9, 2022. https://gnupg.org/software/index.html.
Hawashin, Diana, Raja Jayaraman, Khaled Salah, Ibrar Yaqoob, Mecit Can Emre Simsekler, and Samer Ellahham. “Blockchain-Based Management for Organ Donation and Transplantation.” *IEEE Access* 10 (2022): 59013–25. https://doi.org/10.1109/ACCESS.2022.3180008.
Hayes, Adam. “Blockchain Facts: What Is It, How It Works, and How It Can Be Used.” Inves, September 27, 2022. https://www.investopedia.com/terms/b/blockchain.asp.
“How Filecoin Works.” Accessed December 1, 2022. https://docs.filecoin.io/about-filecoin/how-filecoin-works/.
IBM. “About Linear Regression.” Accessed November 30, 2022. https://www.ibm.com/topics/linear-regression.
IBM. “What Is a Decision Tree.” Accessed November 30, 2022. https://www.ibm.com/topics/decision-trees.
IBM. “What Is Random Forest?” Accessed November 30, 2022. https://www.ibm.com/cloud/learn/random-forest.
IBM. “What Is the K-Nearest Neighbors Algorithm?” Accessed November 30, 2022. https://www.ibm.com/topics/knn.
“Install Ubuntu on WSL2 on Windows 11 with GUI Support.” Accessed December 1, 2022. https://ubuntu.com/tutorials/install-ubuntu-on-wsl2-on-windows-11-with-gui-support#5-install-and-use-a-gui-package.
IPFS.tech. “How IPFS Works.” Accessed December 9, 2022. https://docs.ipfs.tech/concepts/how-ipfs-works/#content-addressing.
IPFS.tech. “What Is IPFS?” Accessed December 9, 2022. https://docs.ipfs.tech/concepts/what-is-ipfs/#decentralization.
jennijuju. “Release Release v1.18.1 · Filecoin-Project/Lotus · GitHub.” Github. Accessed December 9, 2022. https://github.com/filecoin-project/lotus/releases/tag/v1.18.1.
Kamara, Seny. “Proofs of Storage: Theory, Constructions and Applications.” In *International Conference on Algebraic Informatics*, Vol. 8080, n.d. https://doi.org/https://doi.org/10.1007/978-3-642-40663-8_4.
Levin, Jason. “Filecoin Storage Up 128% This Quarter.” The Defiant, August 4, 2022. https://thedefiant.io/decentralized-storage-filecoin.
Maurizio, S. “(Simple) Linear Regression and OLS: Introduction to the Theory.” Medium, May 25, 2020. https://towardsdatascience.com/simple-linear-regression-and-ols-introduction-to-the-theory-1b48f7c69867.
Omar, Ilhaam, Raja Jayaraman, Khaled Salah, Mecit Can Emre Simsekler, Ibrar Yaqoob, and Samer Ellahham. “Ensuring Protocol Compliance and Data Transparency in Clinical Trials Using Blockchain Smart Contracts.” *BMC Medical Research Methodology* 20, no. 224 (April 19, 2020). https://doi.org/https://doi.org/10.1186/s12874-020-01109-5.
Polkadot. “About Polkadot.” Accessed December 2, 2022. https://polkadot.network/about/.
web3.storage. “Better Storage. Better Transfers. Better Internet.” Accessed December 9,2022.https://web3.storage/docs/.
Project Jupyter. “Home.” Accessed November 30, 2022. https://jupyter.org/.
Python.org. “What Is Python? Executive Summary.” Accessed November 30, 2022. https://www.python.org/doc/essays/blurb/.

Rosenberg, Eric, Amilcar Chavarria, and Vikki Velasquez. “Storj (STORJ) Cryptocurrency Definition.” Investopedia, October 2, 2022. https://www.investopedia.com/storj-cryptocurrency-definition-5224160.

Sahil. “Top 20 Blockchain Project Ideas for 2022.” parangat.com, January 6, 2022. https://www.parangat.com/blog/blockchain-project-ideas/.

Schoeman, Louis. “Filecoin (FIL) Reviewed.” SA Shares, November 11, 2021. https://sashares.co.za/filecoin-review/.

Sidess, Arieh, Amnon Ravina, and Eyal Oged. “A Model for Predicting the Deterioration of the Pavement Condition Index.” *International Journal of Pavement Engineering* 22, no. 13 (January 16, 2020): 1625–36. https://doi.org/https://doi.org/10.1080/10298436.2020.1714044.

Skynet Labs. “Skynet Overview - Skynet Guide,” 2021. https://support.skynetlabs.com/?pk_vid=4ee3e7363ef37a5b16699958814ad92f.

Substrate.io. “Substrate Blockchain Technology.” Accessed December 15, 2022. https://substrate.io/technology/.

Substrate.io. “Substrate Connect.” Accessed December 15, 2022. https://substrate.io/developers/substrate-connect/.

Tutorialspoint. “Node.js - Introduction.” Accessed December 2, 2022. https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm.

Tutorialspoint. “Scikit Learn - Introduction.” Accessed November 30, 2022. https://www.tutorialspoint.com/scikit_learn/scikit_learn_introduction.html.

Univeristy of Phoenix. “The Basics of Scrum: Methodology and Framework.” University of Phoenix. Accessed December 7, 2022. https://www.phoenix.edu/professional-development/blog/what-is-scrum/.

Vaidya, Dheeraj. “Nonlinear Regression - Meaning, Analysis, Model, Examples.” Wall Street Mojo. Accessed November 30, 2022. https://www.wallstreetmojo.com/nonlinear-regression/#h-application.

Visual Studio Code. “Visual Studio Code Frequently Asked Questions,” November 2, 2022. https://code.visualstudio.com/docs/supporting/faq.

W3 Schools. “Introduction to NumPy.” Accessed November 30, 2022. https://www.w3schools.com/python/numpy/numpy_intro.asp.

Wang, Qiang, Rongrong Li, and Lina Zhan. “Blockchain Technology in the Energy Sector: From Basic Research to Real World Applications.” *Computer Science Review* 39 (February 2021). https://doi.org/https://doi.org/10.1016/j.cosrev.2021.100362.

web3.storage. “How to Retrieve Data from Web3.Storage.” Accessed December 9, 2022. https://web3.storage/docs/how-tos/retrieve/.

web3.storage. “Say Hello to the Data Layer.” Accessed December 9, 2022. https://web3.storage/.

“What Is an Oracle in Blockchain?,” September 14, 2021. https://chain.link/education/blockchain-oracles.

Wilkinson, Shawn, Josh Brandoff, and Buterin. “Storj A Peer-to-Peer Cloud Storage Network.” Storj.io, December 15, 2014. https://www.storj.io/storj2014.pdf.

“Worcester IS FAB Lab for Social Good.” Worcester IS Fab lab, n.d.

Worclab. “WorcLab.” Accessed November 30, 2022. https://worclab.org/.

Yegulalp, Serdar. “What Is TensorFlow? The Machine Learning Library Explained.” Info World, June 3, 2022. https://www.infoworld.com/article/3278008/what-is-tensorflow-the-machine-learning-library-explained.html.

Zahed Benisi, Nazanin, Mehdi Aminian, and Bahman Javadi. “Blockchain-Based Decentralized Storage Networks: A Survey.” *Journal of Network and Computer Applications* 162 (July 15, 2020). https://doi.org/https://doi.org/10.1016/j.jnca.2020.102656.