Free market blockchain P2P energy trading

PROJECT PLAN

Team #41

Arun Sondhi: Hardware/software API

Alec Dorenkamp: Smart contract / blockchain design

Noah Eigenfeld: Smart contract / blockchain design

Brendon Geils: Blockchain and API development

Jack Myers: Smart meter circuit design

Joe Staudacher: Power sensing/hardware

Client: Sodima Solutions

Adviser: Goce Trajcevski

sdmay18-41@iastate.edu

sdmay18-41.sd.ece.iastate.edu

Revised: 10/30/17

Table of Contents

List of Figures

1 Introductory Material

1.1 Acknowledgement

To preface our project plan, we would like to thank those who have made contributions to the project, outside of the members of our team. First and foremost, we would like to thank our adviser, Dr. Goce Trajcevski, for his technical assistance and comprehensive guidance throughout the project. We would also like to thank Sodima Solutions for providing the funding for the required hardware for the project, as well as providing the overall idea of the project. Lastly, we would like to express our gratitude towards the faculty of Iowa State University for their support in giving us the technical background and knowledge for us to handle a project of this scale. Without the support of these individuals and organizations, our project's success would not have been possible.

1.2 Problem Statement

The main aim of our project is to incentivize renewable energy generation from individuals and small businesses by facilitating peer to peer trading of surplus energy. By creating this free market environment for energy trading, individuals will think more about how they produce and consume energy, and will be inclined to generate energy of their own. With this new understanding and market accessibility, energy prices will fluctuate to be at parity with their true value, not just what the utility company dictates. A more detailed description of our implementation of this free market solution is found later in this document.

The secondary goal that our project aims to reach is aiding in the decentralization of the power generation market. The interconnectedness of grids has already contributed to the reduction of blackouts, as one individual power plant or utility company is not solely responsible for all energy generation. As things currently stand, a Chicago grid can pull from a Toronto plant if they approach their capacity curve. Our system would aid this interconnectedness a degree further, as the power loss and cost to move energy a mile up the road would be less than the power loss and cost to move that energy from Toronto to Chicago. A more decentralized grid would help the overall grid to be robust to fluctuations, as the sources of energy would be widespread and independent.

The connection between decentralizing energy generation and our solution is clear. We hope that an open market will allow individuals to feasibly operate their own renewable energy sources like solar panels or wind turbines. If this goal is realized, energy will not be produced only at large plants. Because of this, consumers will be able to rely on a more diverse array of energy sources than those that are currently available.

A positive side effect of incentivizing renewable energy generation is the effect that it will have in slowing down climate change. Renewable energy usage is seen as one of the key ways to target this problem, but a large portion of worldwide energy does not come from this "clean" energy generation. To change this, the generation of renewable energy needs to be more accessible to individuals and businesses, rather than only those who have the resources and capabilities of a large energy company.

Our project has two major components: the development of an inexpensive and user friendly smart power meter and software to facilitate peer to peer trading of surplus energy. When these two stages of the project are complete, individuals using our hardware and software will be able to buy or sell surplus energy at significantly better rates than could be obtained going through a utility company.

A stretch goal for the project would be to develop a marketplace where these peer to peer interactions could be facilitated. We would have to find a way to automate this process so users would simply have to say how much power they need at what times and the system would arrange for that transaction to take place.

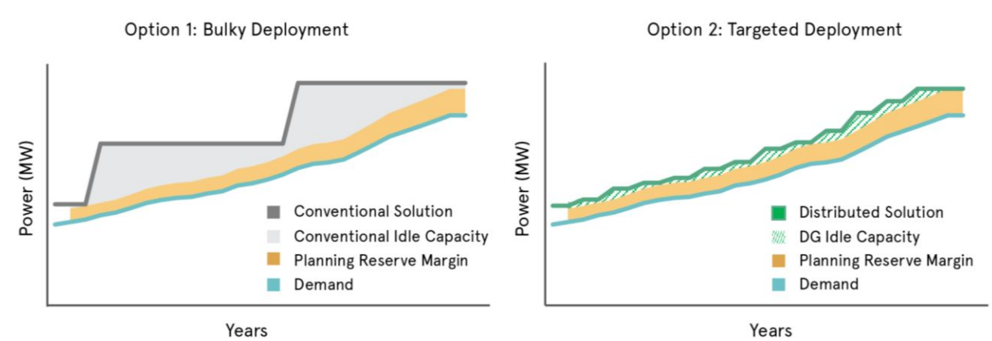

Figure 1: Bulky vs Targeted Deployment of Energy Resources

1.3 Operating Environment

The operating environment for our solution will certainly be a relevant factor in the final implementation. The meaning of “operating environment” is quite different for the hardware and software aspects of our project. For the hardware component the operating environment is taken in a literal sense. Essentially, our smart power meter has to be able to withstand all of the conditions that existing power meters currently withstand. It will have to be able to survive in various weather conditions that naturally come with being a product that lives outside of the home.

On the software side, “operational environment” has a less literal and physical meaning. Instead, the operating environment of our software will be economic and political climate in which our solution is being used. There are many legal factors that could come into play with this kind of trading, like use of the utility company's infrastructure or trading between different cities, states, countries. These are all factors that would need to be explored in detail if this project were to be expanded beyond a simple test environment with two nearby homes or small businesses.

As the project continues (potentially with future senior design groups), more focus can be put into refining the robustness of the hardware and making sure our software implementation integrates effectively with the economy and politics of the location where our solution is being used.

1.4 Intended Users and Intended Uses

The users can be split into two groups, which we'll call producers and consumers. The producers are the users who will be supplying excess energy that they produce into the system. Producers are looking to maximize the profit that they can create from producing energy, and our tradable energy market will enable them to do just that. In order to best serve these users, we are minimizing the transaction costs and maximizing the ease with which they can find buyers for their energy. The consumers are the users who will be consuming the excess energy that producers create. We can best serve them by minimizing the transaction costs and making it as easy as possible for them to find producers whose energy they can consume.

The intended use can therefore be described as incentivising personal generation of energy by creating an accessible market for both the producers and consumers. Ideally, the intended users would eventually include anyone with an electrical service connected to their home or business but in the current utility climate, the intent is simply to attract as many users that already have direct generation installations as possible. As we get more producers in our user base, that will allow us to get more consumers by making the market more competitive and attractive. In due course, the flow of consumers to the marketplace will encourage more people to become producers and install their own direct generation setups. This circular growth will eventually lead to us reaching our intent of widespread incentivisation of direct generation.

1.5 Assumptions and Limitations

<u>Assumptions:</u>

- Enacting the distribution of power after our transaction is completed is outside of the scope of our project, including the new power equipment that could be required for this distribution
- For a full implementation of our project, an agreement will have to be completed with the utility company owning the power infrastructure so they will allow these transactions to take place
- The level of testing that we will complete will be within an individual municipality, so interstate/international trading laws will not be applicable

<u>Limitations:</u>

- The cost of the IoT smart power meter must not exceed that of the average power meter used in Ames, IA (the area of testing)

1.6 Expected End Product and other Deliverables

Blockchain Implementation

Power transactions will be made and recorded using a blockchain implemented on Ethereum's platform. This blockchain will be the backbone of our project, as it will keep a ledger of all power transactions that occur in a network of users. The decentralized-ledger approach of a blockchain will ensure all transactions are trustworthy, even though users may remain completely anonymous. The blockchain will consist of a set of smart contracts (deployed on Ethereum) defining our own cryptocurrency for power transactions and logic to initiate and accept transactions.

IOT Smart Meter

An "internet of things" smart meter will be installed at a user's property to read the flow of energy into their home/building. This smart meter will be connected to the internet, and will interact with the blockchain to physically enact transactions. The smart meter will be able to verify transactions of power over set periods of time, as determined by the agreement between the buyer and seller in the transaction. The goal for the end of this semester is to have functioning basic communication between the smart meter and the software side of our project. By the end of the year, we will develop a working prototype that is able to verify the completion of a transaction, connect to the blockchain network and web application, and display vital information to the user via a user interface directly on the physical meter.

Web Application

Users will manage their power transactions through a simple web application, which will interact with our blockchain. Since the focus of our project is on the blockchain and smart meter implementations, this application will provide only the basic functionality needed for users to track and trade their power. Such functions include login/account creation capability, viewing transaction history, and initiating/accepting transactions. In the future, this web application may be expanded to better meet the needs of the customers.

2 Proposed Approach and Statement of Work

2.1 Functional requirements

Functional requirements:

1. A blockchain software implementation
2. An IoT Smart Meter device
3. Web app for management of transactions
4. API for communication between the smart meter and the blockchain

The requirements will grow and change as progress is made but these requirements are the core of the project and will dictate our goals over the coming months.

Requirements/standards:

Project development will follow a structured model for Git version control. We will use pull requests that'll require at least one other member's approval. Each task will have it's own branch and those branches will be appropriately named based on their task. All commits will have will have sufficiently descriptive comments. All code will include descriptive objects and class names. Each method and class will be sufficiently commented except obvious getters and setters.

2.2 Constraints considerations

The project must meet certain standards that we as a group agree on. We want the platform for the blockchain to be robust and have the ability to expand on it if desired. This robustness will lead to trust in our energy trading system, and the scalability will enable growth. Code conciseness goes hand in hand with robustness, and makes testing and expanding our project simpler and quicker. Portability is an important requirement for the smart meter, allowing it to be installed easily and effectively, and portability for the web app is important so that users can access it from multiple different devices.(such as laptops, tablets, and mobile devices) Our goal is to end up with a product that we can be proud of and have it be something that we are eager to present to the client. Each group member will strive to put their best work forward and each contribute their own unique values to the project.

These values will include efforts by all team members to adhere to the engineering ethics that we have been studying while at Iowa State University. Every member of the team has a bevy of university-mandated ethics training in addition to their own personal values informing their decisions as engineers. The ethics program at Iowa State emphasizes the importance of integrity, honesty, and responsibility in the work of engineers. Keeping these in mind and following the IEEE Code of Ethics is of the utmost importance for our group as we work on this project.

Project development will follow a structured model for Git version control. We will use pull requests that'll require at least one other member's approval. Each task will have it's own branch and those branches will be appropriately named based on their task. All commits will have will have sufficiently descriptive comments. All code will include descriptive objects and class names. Each method and class will be sufficiently commented except obvious getters and setters.

2.3 Technology considerations

Arduino

Advantages:

- Entire team has experience and can develop MVPs
- Multiple libraries and shields for wifi and other extraneous processes
- I/O system is the easiest

Disadvantages:

- Little learning in terms of team's intellectual growth
- Can't house the ethereum web3.js code base
- Doesn't provide a strong IP story

Arduino would be best used if we encounter high I/O in our system while not requiring heavy web based protocols. The shields and libraries are cohesive for simple wifi connectivity, but lack libraries for connecting to blockchain networks. Every member of our team has worked with arduino.

Raspberry Pi

Advantages:

- Easiest connectivity process (wifi)
- Experienced members on our team
- Libraries and shields for extraneous processes
- Allows web server code/processing
- Rolling new updates is simpler

Disadvantages:

- Doesn't provide a strong IP story
- Single point of failure on the system

A Raspberry Pi system will allow us the greatest extensibility and code reusability. Many of the arduino advantages are present with the Raspberry Pi while also allowing us to use js libraries that connect directly to the blockchain. By connecting direct, we can remove a node that would be required for computational logic (cloud or local server) that both the arduino and PCB require. Our team similarly has experience working with these.

PCB/Embedded

Advantages:

- We can improve our knowledge or fabrication and low level hardware/software
- Capability to be fastest processing
- Least amount of resources used - power/computation

Disadvantages:

- Learning curve is highest
- Development cycle is the longest
- Documentation for our application is minimal

In going the PCB/Embedded route we have the opportunity to become more adept in a technology our team isn't familiar with. This would also allow us the best intellectual property scenario as the technology isn't easily reproducible in code. The drawbacks are large in terms of development cycles and resources that we can tap to work through any issues. Many seamless processes such as wifi will be more difficult on this platform. The finalized product has the potential to be faster while consuming less resources.

2.4 Safety considerations

The one safety concern which needs to be addressed when approaching our project is the linking of the smart meter to the power lines going into each of the homes, businesses, or other entities that are going to be included in the network. As long as the meter clamps are safely attached to the power lines (by the entity or a hired professional), there will be no safety issues.

2.5 Previous Work and Literature

TransActive Grid + LO3 Energy

TransActive Grid is the IP holding company of LO3 Energy. The Consensys team previously teamed up with LO3 Energy to test the viability of energy blockchain. In 2016 there was a blockchain energy implementation in New York City, specifically the Brooklyn borough. The pair wired up two Brooklyn residences and traded energy on the blockchain. The details of this transaction were left out due to IP implications from the organization [7].

Besides the 2016 article LO3 energy has been relatively quiet about their progress. The gist from their end is that energy blockchain is viable from a technical perspective. The Brooklyn Microgrid is now being brought to other countries, most recently Germany.

Grid+

Grid plus is a Austin, TX based energy retailer. The parent company is New York City based ConsenSys. Grid+ provides a smart argent and blockchain implementation. The smart agent is used to buy and sell the GRID token. They also have another coin, BOLT. This coin is known as a stable coin. The advantage to stable coins is while a cryptocurrency will readily fluctuate in price a stable coin reflects a more stable currency, in this case the USD. A single BOLT is equivalent to a single USD. The end user will have the advantage of near real time service and security of the blockchain while not having a wildly fluctuating cost of energy.

Grid+ however doesn't have a peer to peer energy model. The business model they use is to cut administrative costs from the distributor and retailer in the energy supply chain. From a technical perspective the Grid+ energy blockchain implementation is using the ethereum blockchain [5]. The ERC20 token standard is their building block for the smart contract. The focus of their contracts is on the token and ICO, while they have long term development outlines for their energy business.

SolarCity

SolarCity wrote a report [11] to which they explain the distributed energy environment and its subsequent advantages. They specify that the distributed energy model is a net benefit for society through benefits related to voltage and power quality, conservation voltage reduction, grid reliability and resiliency, equipment life extension, and reduced energy prices. The hurdle they present is that the current utility incentive model doesn't coincide with a distributed energy focus. To convert to their model they propose legislative changes.

Power Ledger

From the Power Ledger whitepaper [12] the Australian based energy blockchain startup looks to provide peer to peer energy trading. The will facilitate these energy transactions with the ethereum blockchain. Power Ledger provides very little technical documentation of their work but boast a trading matching algorithm, meter reading device and token sale.

2.6 Possible Risks and risk management

As we proceed with development, there are a few concerns that we are aware of that have a chance to hinder our progress. Though we are fairly knowledgeable about most areas of our project, because we are inexperienced in interacting with power lines and implementing smart contracts, there may be unexpected roadblocks that we run into that will need to be addressed. We are in the process of eliminating any timing issues associated with sending energy over a multiple-hour period of time while working with blockchain transactions, which are typically one-time events. This will help us mitigate the risk of users exploiting other users. Finally, we are working diligently to ensure that the software we write is error-free and secure, through the use of smart contracts within the blockchain.

2.7 Project Proposed Milestones and evaluation criteria

Please refer to section 3.4 for milestones and 2.11 for evaluation criteria.

2.8 Project tracking procedures

We are using a Gantt chart to outline the variety of tasks that are necessary to complete our project. This chart serves as a visual tool with clear milestones that need to be completed, including dates, task owners, duration, and more. This schedule can and will be adjusted as progress is made and issues arise, but it will be in our best interest to keep up with the deadlines in order to stay on track to complete the project in a timely manner.

2.9 Objective of the task

We anticipate that this project, in the long run, has the potential to be implemented in a large scale. Taking into account the time and resource constraints associated with this project, our goal is to have a working prototype that can be tested first on a preliminary level and potentially on a network of 400 homes.

A stretch goal for the project would be to develop a marketplace where these peer to peer interactions could be facilitated. We would have to find a way to automate this process so users would simply have to say how much power they need at what times and the system would arrange for that transaction to take place.

An inherent goal of our peer to peer marketplace is that of anonymity/privacy. The idea of using blockchain for an energy market is not only to provide users with an economically viable and accessible application, it is also to provide them with a secure environment for them in which to trade. Our blockchain market will not require users to put their trust in any one other user, therefore allowing them to remain completely anonymous and help them keep their privacy intact. By keeping their transactions disassociated with any personal information about them, users will be able to confidently trade on the market without worrying about their transaction habits or other personal information being tracked.

In addition to our goals related to the application of our solution, we also have some goals throughout the process of developing our solution. We hope that this design process will be a beneficial experience for us, aiding us in learning how to interact with contacts in industry and respond to their feedback in order to tailor our work to most effectively solve the problem at hand.

2.10 Task approach

We have researched various ways to go about solving our problem, the most significant of which are detailed below.

Software

1. Ethereum Approach

Using the open-source ethereum (blockchain) platform to develop our smart contract for buying and selling assets (energy). Using ethereum we can utilize their stable Solidity language to implement our smart contract. The advantage here is there are many projects built with this stack allowing for more resources and support.

2. Hyperledger Approach

Using the open-source hyperledger (blockchain) platform which is newer compared to the tried and true ethereum approach. Hyperledger is supported by larger organizations such as IBM. We believe this will allow the technology to stabilize long term with the backing of a large company compared to the burn-out many open-source projects that lack an organization have seen.

Hardware

1. Arduino + Wifi Shield Approach

Using GPIO to continuously calculate available energy and control the power management system. The data will then be used for buying and selling. A wifi enabled shield will provide IoT capabilities to support transactions.

The arduino will be linked to a two-way meter and work as a smart meter. Current and power usage will be tracked and data will be transmitted to the marketplace.

2. Microcontroller/custom PCB Approach

Functionally identical to the arduino approach, only the built from a different set of components. Both will record and send data for buying and selling.

3. Raspberry Pi Approach

Again, functionally identical to the arduino approach, but will require a different set of sensors to track power usage. This approach will also allow us to make use of existing open source libraries.

Approach Taken

After considering the above approaches, we determined to implement our blockchain through Ethereum, and to use a Raspberry Pi approach for our smart meter. Each user/property will use the smart power meter to monitor the flow of power from the property (and any power-generating devices, such as solar panels or wind turbines) to their power company. All power is transmitted through the power company for any given transaction. Each smart meter will be linked to the Ethereum blockchain via an internet connection, and will have a copy of the transaction ledger. A user for any given property may access the ledger and their transactions through a web application, which we will design.

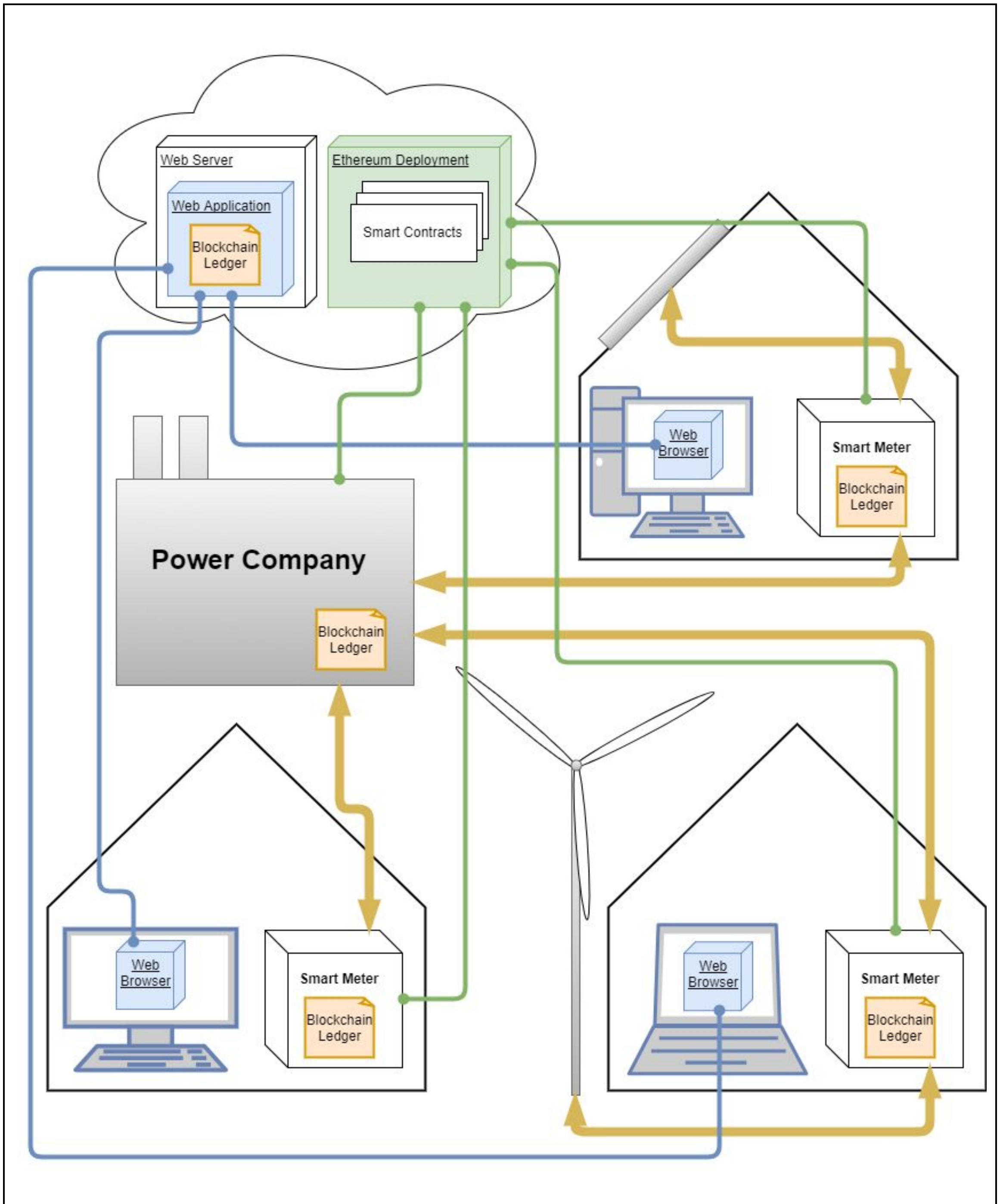

Figure 2: A rough diagram of our proposed system

The web application will interact with the Ethereum smart contracts through a JavaScript interface, implemented on the back end. Any front end task that requires querying or changing the ledger will call a function written in JavaScript on the backend. These secondary functions will call functions within our Ethereum smart contracts, which will perform the queries or updates.

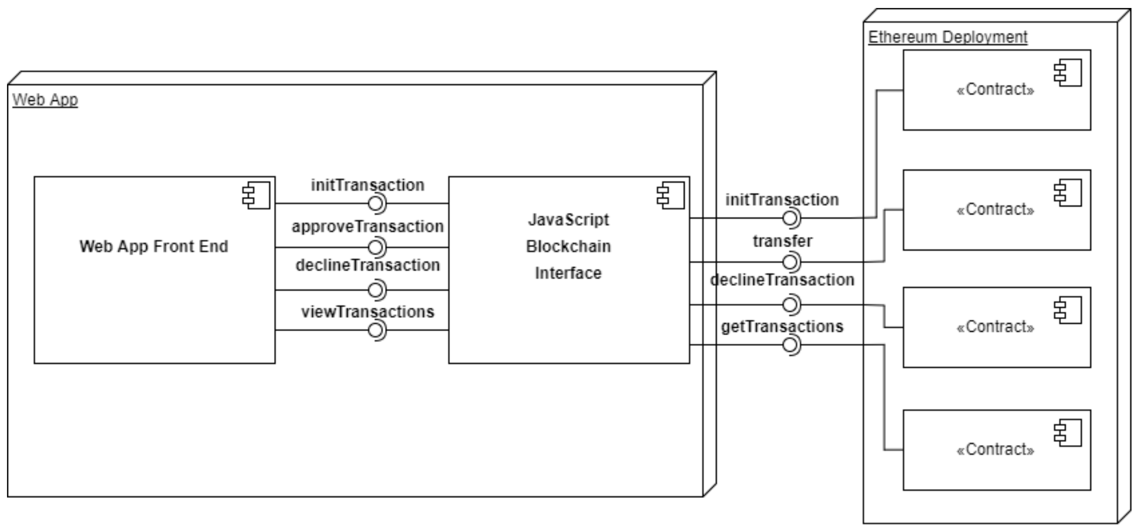

Figure 3: A macro-component diagram of how our web application interacts with the smart contracts in our blockchain

2.11 Expected Results and Validation

The simplest way for us to confirm that our solution works is testing it. Because our project is broken into two parts (smart meter and blockchain software), there will be multiple stages of testing. We will test the functionality of the smart meter to determine that it is able to serve as an accurate and non-intrusive method of reading power input to a house. Concurrently, we will be testing the functionality of the software to make sure we can reliably and securely process transactions. When the unit testing stages are completed, we will move on to integration testing to make sure the entire system works together cohesively. If we have the opportunity, it could also be beneficial to work on testing in a system with a large number of points (houses) to make sure that everything is scalable.

2.12 Challenges

1. Does a user have to put energy into the system at the same rate that it is being consumed? Will there be any timing issues? More research is being done.
2. We will need to figure out the optimal way to connect the IoT device to the Blockchain.

3 Estimated Resources and Project Timeline

3.1 Personnel effort requirements

Task	Team	Effort Required	Reference/Explanation
Program User Interface	Hardware	Low	The libraries available for Raspberry Pi devices should make it fairly simple to take in input and display output to the user on the meter.
Coordinate Interfacing with Blockchain	Hardware	High	Because we are fairly inexperienced with blockchain, there will be a learning curve associated with connecting our embedded software with the rest of our system. This step will also require effective communication with the software team on what data they need to receive and what form it will come in.
Test Initial Proof of Concept Prototype	Hardware	Medium	The focus of this step will be to ensure that this version of the system has all of the basic desired capabilities that are needed for an early implementation of the project. Essentially we will need to make sure that we are able to reliably acquire and transmit data.
Design PCB/Embedded Board	Hardware	High	To implement the hardware required for this project from scratch, we would need to acquire various individual components and connect them on a PCB. In addition, we would have to write our own drivers for this board.
Order Board	Hardware	Low	Once we have the design completed, ordering the board will

			just be a matter of determining the best option to have it fabricated
Test and Compare with Raspberry Pi Version	Hardware	Medium	This step should just be a matter of determining which aspects of performance we will give the most weight and comparing the two implementations, including the effort and cost that went into creating them.
Blockchain: Coin Definition	Software	Low	Standard cryptocurrency coin definitions already exist, so we should only have to add small details to tool them to our needs
Blockchain: Init Transaction	Software	Low	Transactions are already a standard Ethereum function, and should be easy to program
Blockchain: Accept Transaction	Software	Medium	We will need to do some extra programming to deal with the timing of when accounts receive payment, such as putting coin into a holding pot until the transaction is complete
Blockchain/Smart Meter API: Power Transaction Signal	Software	High	Requires blockchain/smart meter interaction, which will require collaboration between teams and probably more testing, since this is essential to our project
Web App: Create Account/Register Smart Meter	Software	Medium	We will tie each account to an Ethereum account, so account creation will be handled elsewhere; however, we will need to handle linking an account to a smart meter
Web App: Login	Software	Low	We will just need to pass login through Ethereum, so there should be little programming required for this step
Web App: View Transaction History/Ledger	Software	Medium	Since the web server will have access to the ledger, retrieving data should be easy; we will need to

			program its display and any alternate views we wish to provide to a user
Web App: Init Transaction	Software	Low	Most of the programming for this will be done on the Ethereum smart contracts, and this task will just require programming the JS interface to call contract functions
Web App: Accept Transaction	Software	Low	We will need to display a message for users that indicates they have received a transaction request that they can accept, but most of the programming will be done in the Ethereum smart contracts called through the JS interface
Perform End-to-End Acceptance Tests	Software	Medium	We will want to be thorough with our tests, and will probably need to do quite a bit of debugging and refining, but if our designs are well thought out, and each previous task goes well, we should not have any major changes to make

3.2 Other resource requirements

The main parts that will be required for this project will be on the hardware (smart meter) side, as this is the physically tangible part of the project. For the early prototype, we will need two Raspberry Pi modules (one to act as the buyer and another to act as the seller) and current sensors to detect the power input and output at each of our test "homes." In addition, if we are able to get to testing with fabricating a PCB, there would be quite a few components necessary. While we cannot be sure exactly what would be required for the design at this point, we would certainly need a microcontroller with WiFi capabilities along with the basic components that go into making a PCB (circuit design software, solder, etc.). The software team will require the Ethereum, node, truffle, and testrpc libraries to run smart contract and blockchain tests and implementations.

3.3 Financial requirements

MASTECH MS3302 AC Current 0.1A-400A Clamp Meter Transducer True RMS
at $17 x 2 = **$34**
Justification: To measure the current entering and leaving the building.

Raspberry Pi 3

at $35 x 2 = **$70**
Justification: To connect the current sensing device to the blockchain. Import our logic libraries and subsequent code base.

PCB Board + device parts (Pi 3 substitute)
at $65 x 1 = **$65**
Justification: Time permitting we may substitute the Pi 3 for a custom PCB board that we can run our systems on in the same fashion as we would with the Pi.

Total: **$169**

3.4 Project Timeline

A large portion of the success of our project will be centered around adhering to our timeline. Although at this point, many of our tasks are in the preliminary stage, it will still be vital for us to use this tool to maintain steady progress. An overview of our major tasks and milestones is shown below, while the more detailed version is shown in the attached Gantt charts for both first (EE/CprE/SE 491) and second semester (EE/CprE/SE 492).

First Semester

Class Deliverables

1. Project Plan
2. Design Document
3. Team Website

Hardware Team (Jack, Joe, Arun)

1. Research pars and determine platform to be used
2. Create tentative parts list with pricing
3. Obtain current sensor and meter platform (Raspberry Pi)
4. Research available software libraries for Raspberry Pi module we are using
5. Determine preliminary version of desired capabilities of meter (sensors required, data stored, etc.)
6. Initiate programming of data acquisition and transmission capabilities

Software Team (Brendon, Noah, Alec, Arun)

1. Experiment with Ethereum and creating smart contracts
2. Web app user interface mockups
3. Component diagram for web app
4. Component diagram for blockchain implementation
5. Component diagram for smart meter API

Second Semester

Class Deliverables

1. Working prototype smart meter
2. Basic functional web application
3. Working blockchain cryptocoin implementation
4. All components successfully interact

Hardware Team (Jack, Joe, Arun)

1. Program user interface
2. Coordinate interfacing with blockchain
3. Test initial proof of concept prototype
4. Design PCB/embedded board
5. Order board
6. Test and compare with Raspberry Pi version
7. Program data acquisition and transmission capabilities

Software Team (Brendon, Noah, Alec, Arun)

1. Smart contract defining cryptocoin on Ethereum
2. Smart contract for initiating a transaction
3. Smart contract for accepting a transaction
4. API components for communicating between blockchain and smart meter
5. Create account functionality through web app
6. Login functionality through web app
7. View transaction history functionality through web app
8. Initiate and accept transaction functionality through web app

Full Project Timeline:
https://docs.google.com/spreadsheets/d/1gSY1sHtt_i6-RaNFEG1_S9uw0KnE5iEDKWcexmxvTxs/edit?usp=sharing

4 Closure Materials

4.1 Conclusion

If we complete our goals, our project has the potential to have important implications on the future of the energy market. We believe that with our team, we can make this project reach its full potential. By the end of the second semester of the project, we hope to have a functional prototype that we can use as a proof of concept in order to gain the traction to apply our solution at a large scale. Between the smart meter and blockchain free market trading software, all of the vital components will be present to show the benefits that this approach to distributing surplus energy has over the existing methodology.

While there are groups like Grid+, LO3, and ConsenSys that have already made strides towards a similar solution, we feel that our team is starting our project at just the right time. We are able to learn from the mistakes from our predecessors by taking what they would have done differently and actually doing it differently. While we are not the first to work on this type of project, we are early enough that we are not fighting against any other groups that are dominating or monopolizing the market. With the structural background that we have established in this document, we hope to be able to develop a solution that meets the needs of the users at hand and takes strides towards increasing the worldwide consumption and generation of renewable energy.

4.2 References

[1] "Bitfinex." *BitFinex*. N.p., n.d. Web.

[2] "ERC20 Token Standard." *ERC20 Token Standard - The Ethereum Wiki*. N.p., n.d. Web.

[3] Geils, Brendon. "Bgeils/pwr-blockchain." *GitHub*. N.p., n.d. Web.

[4] Geils, Brendon. "Pwr.company." *Pwr.company*. Blockchain News, n.d. Web.

[5] "Grid+." *GridPlus.io*. N.p., n.d. Web.

[6] "Gridsingularity.com." *Gridsingularity.com*. N.p., n.d. Web.

[7] Lumb, David. "This New York Project Fuses Energy Microgrids With Blockchain Technology." *Fast Company*. Fast Company, 06 May 2016. Web.

[8] Posted 1 Oct 2017 | 15:00 GMT By Morgen E. Peck and David Wagman. "Blockchains Will Allow Rooftop Solar Energy Trading for Fun And Profit." *IEEE Spectrum: Technology, Engineering, and Science News*. N.p., 01 Oct. 2017. Web.

[9] "Project Sunroof - Solar Calculator." *Project Sunroof*. Google, n.d. Web.

[10] Sebnem. "Token Model for Energy - Part 1: Review of the Power Ledger Token Model." *Medium*. Medium, 12 July 2017. Web.

[11] Solar City, comp. "A Pathway to the Distributed Grid." (2016): n. pag. Print.

[12] “PowerLedger Token Generation Event.” PowerLedger Token Generation Event, powerledger.io/.