School of Electrical and Information Engineering
University of the Witwatersrand, Johannesburg
ELEN4002/4012: Project Specification Outline

To be completed by supervisor
Assessment:
☐ Unacceptable ☐ Poor
☐ Acceptable ☐ Good ☐ Excellent

Project Title: Blockchain based tokenised decentralised grid network (meterBlock)

Group Number:	18G35	Supervisor Names:	Viv Crone and Mitch Cox
Member 1 Name:	Brandon Verkerk	Student number 1:	875393
Member 2 Name:	Christopher Maree	Student number 2:	1101946

Project Specification:

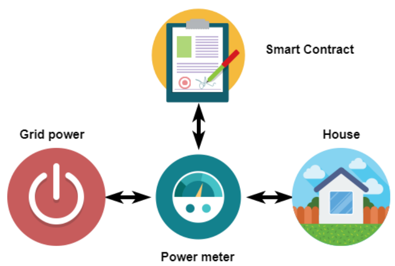
meterBlock provides a decentralised platform to facilitate the governance of electrical energy consumption and production, measured and controlled by pre-paid IoT power meters. meterBlock consists of three distinct pillars:

1. Utility tokens (a blockchain store of value, representing a real world asset), each representing a kWh, are used to account for the energy produced and consumed on the grid. These "KraG" tokens are the economic mechanism used to control energy through IoT based pre-paid power meters. The meters will measure the energy consumed/produced, and control its supply to the load/grid.
    - An IoT device will be used to control a relay or contactor to facilitate and interrupt power supply to a load.
    - An Ethereum Smart contract will be used to manage the utilisation of KraG tokens according to production and consumption of energy
        - This token is based on the ERC20 token standard and is allocated on the production of energy and is deallocated on consumption of energy
        - A website dashboard will be created to facilitate user interaction with the smart contract
    - The IoT device must interact with the smart contract to:
        - Enable the deallocation of tokens, representing the consumption of energy. When these tokens run out, the power turns off
        - Enable the allocation of tokens, representing the production of energy. These tokens are then transferable.
    - *The deliverables related to this pillar are in milestone 1 to 5, with extended implementation up to milestone 7*
2. Using game theory, non-linear pricing curves and other economic incentives will be used to encourage mutually beneficial utilisation of the grid between produces and consumers.
    - By monitoring the load on the grid, the cost per kWh can be adjusted dynamically to encourage advantageous behaviour.
    - *The deliverable related to this pillar is outlined in milestone 8*

Point 1 and 2 above will be implemented without considering how tokens are converted to fiat. The limitation is that no new members can join this system if they don't produce energy. Likewise, members that produce more than they consume are not compensated accordingly.

3. A fiat-to-token exchange is created to facilitate the buying and selling of KraG tokens between consumers and producers.
    - *The deliverable related to this pillar is outlined in milestone 9*
4. Microgrid logic is included to facilitate dynamic load balancing within the grid as well as cost savings for the end user.
    - Each household appliance will be given a priority, enabling intelligent load shedding (turning off of a geyser, before your lights).
    - Through the economic platform from (3), the smart meter can vary its cost model based on user specifications, by turning off high draw, low priority loads when energy is expensive.
    - *The deliverable related to this pillar are outlined in milestone 10*

Points 1 and 2 above is defined as the minimum viable deliverable for the project and are considered the base success criteria. Implementation of point 3 and 4 are seen as stretched goals. Point 3 has inherent challenges associated with it such as normal market forces and human sentiment. Implementation of point 3 requires assumptions to be made that will be addressed in the report. Basic overview of proposed system:



Milestones:

1. Ethereum smart contract capable of creation, allocation and deallocation of KraG tokens (self-standing entity, not linked to anything at this point)
2. IoT microcontroller (Rasberry Pi or Arduino) capable of measuring power consumption (meter).
3. Meter capable of controlling (switching on and off) the load.
4. Interaction between meter and smart contract (connection of meter to the blockchain)
    - This interaction must enable point 1 and 2 (consumption of power results in reduction of tokens. When tokens run out, power turns off).
5. Steps 1 to 4 are then implemented for the production of power
6. User interface implemented on the meter to:
    - Display the real time current token balance of the meter
    - Display of key power metrics(total energy produced/consumed, contract information, tokens consumed, etc.)
    - QR code displayed directly meter enabling tokens to be sent to the meter from a mobile device (loading more tokens)
7. Creation of a web-dashboard to view the current status of meters on the blockchain and manage the smart contracts

These milestones are for specification part 1 above, where the economic forces, such as the price per token and how they are traded, will not be initially taken into account.

8. Non-linear pricing curves implemented through the use of a price oracle.
    - This means that the price of energy changes according the load on the grid

At this point, a fully implemented MVP will have been been created, that could conceivably work in the real world.

9. Simple token market, enabling the buying and selling of tokens for fiat (or other crypto assets)
10. User able to specify individual appliance load priority within her household
    - Smart load sheading enable to selectively turn off low priority devices under high grid load
    - Optimised cost mode to schedule low priority devices to only run when the price of power is low

Budget/resources:

1. Arduino or Raspberry Pi to act as the power meter (R400)
2. Relay or contactor to enable load control (R200)
3. 4 Clip on Current transformers to measure power produced/consumed (R200)
4. LCD user interface for meter (R300)
5. Breadboard ,pcb, wires and other miscellaneous electronics (R100)

Total: R1200

Risks / Mitigation:

Scope creep is a major problem with our idea. As such, we have been careful to define the minimum deliverables (pillar 1 and 2 above), and will only work on the future extensions of the project once these are completed and documented in their entirety. Each component within the project is complex; be it the IoT power meter, the smart contract or token exchange. As a result, design decisions have been kept as simple as possible with the intension of keeping the project as a proof of concept.

Risk	**Mitigation**
Implementation of ERC20 token contract proves too complex for implementation	Use a modified version of existing token standard
Inability to implement power meter user interface via LCD screen	Communicate with PC over USB
Inability to link meter to smart contract via API	Simulate connection via MATLAB
Complexity in implemented non-linear pricing curve oracle to inform meters of current prices	Hard code pricing curves into meters, based on voltage
Fiat to token exchange proves too time complex to implement	Simulate an exchange in MATLAB
Economic game theory relating to the transfer of tokens between a consumer and producer outweighs the intended use case value of the project	Treat economics as a black box, making fundamental assumptions to simplify the model
Inter-meter dynamic load balancing microgrid logic communication mechanism proves complex	Simulate a communication network between multiple meters in MATLAB