Project 35:
Realistic Modeling of Electric Power Distribution

Product Requirements

Team Electric Blocks
Zachary Sugano
Christian Whitfield

Product Requirements	24 September 2020	Revision 0

Document History

Revision #	Date	Modified By	Reason
0	24 September 2020	Zachary Sugano, Christian Whitfield	Initial Release

Table of Contents

1. Objective

The objective of this document is to document the requirements for the Electric Blocks system.

2. Scope

The scope of this document is to define the requirements for all aspects of the Electric Blocks system. Since this project is entirely virtual, mechanical, electrical, and regulatory requirements are ignored as that is considered outside the scope of this project. It is assumed that anyone using our product already has the necessary hardware and that hardware is properly maintained by the user or system administrator.

3. References

3.1. Cited Documents

No documents cited as of this revision.

3.2. Acronyms

PP	PandaPower - A power flow simulation API
MCF	Minecraft Forge - A modding library for Minecraft
PF	Power flow
EB	Electric Blocks - The mod we are creating and our team name

4. Functional Requirements

4.1. User Interface Requirements

Our system takes advantage of the MCF library which should make installation easy for users. The operator should be able to simply install the EB jar package into their “mods” folder and launch the game in order to start using the system. They will then use the standard minecraft interface to connect to a server and start creating power flow simulations in game. This is further detailed in the *Software Requirements* section of this document.

System administrators who run the servers for people to connect to must perform additional steps to install and setup the PP server. This could be mostly automated through a docker container or it could be set up manually by installing the appropriate python packages and launching the server from python. It may be possible to automate startup of the server from the minecraft mod, but we would like to design the PP server in such a way that multiple different servers can connect to a single PP instance to run PF simulations.

4.2. What it should do

The EB mod will contain both the client and server side code needed to allow users to run PFs.

The client portion of the mod is responsible for interfacing directly with the player by taking their inputs, communicating with the MCF server, and rendering the models and textures to create a complete simulated experience.

The server portion of the mod is responsible for keeping the client updated with PF information. The server portion will also take user inputs and watch for world events related to EB so it can update its internal model of the network. The internal model of the network will then convert the information into a format that can be understood by the PP server and send the network model over. The PP server will then perform a PF study on the network it receives and return the results. The server portion will then use those results to update its internal model of the network and forward those results to any connected clients so they can be aware of the powerflow results.

While this design will make it difficult to use in a single player context, it should make running multiplayer servers far easier for system administrators to run. The players will only need to install the mod jar and will not need to set up the PP server. An arbitrary number of servers can be set up using a single PP server and this will allow for the mod to be used effectively in a teaching/team environment.

5. Software Requirements

5.1. Functionality

The Electric Blocks mod provides a set of blocks useful for designing and testing AC power flow. The mod is to interact via a socket with the PP server. The server is in charge of solving the power flow problem when it is passed the data relevant to the structure of the electrical network created by the user through the minecraft interface. Updates on PF data can be viewed inside the chat window, but a more refined system could be developed if desired.

5.2. User Interface

The user will interface by means of the standard minecraft mechanics. The blocks pertaining to the components of the distributed power system (e.g. the wire blocks, bus blocks, generator blocks), will be found in the user's inventory. The chat box is useful for getting direct feedback regarding the performance and other PF data from the system. If deemed necessary or convenient, we can apply a tiling property to specific blocks, giving the user an additional way to interact with that block, such as setting electrical parameters.

6. Cost Requirements

6.1. Prototype Cost

Since this project is done entirely in software, our prototype should cost $0 to complete the prototype. It is possible that a testing server might aid in testing the multiplayer functionality of this system. Such a server would cost approximately $12 per month to run and would cost up to $72 in total if ran for the entire duration of the project. Such a test server has not been deemed necessary as of this revision.

6.2. Production Requirements

Since this project is done entirely in software, it should cost $0 to distribute the production software to end users. This does not take into account the costs of hosting and running the server itself. The hardware and maintenance costs of the host machine are considered outside the scope of this project, but we hope to minimize this cost by making installation of this software easy to do. Assuming the end user has sufficient knowledge of setting up a minecraft server, then only the cost of the physical host needs to be accounted for. If using a VPS, this could cost as low as $12 per month, just like the testing server. If they plan to purchase hardware, the costs will be much higher and much more variable.

7. Schedule Requirements

Date	**Schedule Item**
10/09/20	Load, generator, bus, and wire blocks/times.
10/09/20	Complete PP socket server interface
11/06/20	Block and item components (Transformer, 3 Phase Loads and Generators)
11/11/20	Complete PP socket integration with mod
11/20/20	Blocks and items Switches, Battery, Electric Furnace
01/29/20	PF data using chat interface
02/19/21	Measurement and GUI
04/16/21	Asset models
05/07/21	Final project due with software