EXPO Presentation Script

1. Intro: **(Ryan)** Hello everyone, welcome to our EXPO Presentation for Electric Blocks II. My name is <name here>. **I'm Greyson. And I'm Samuel**. And we have been working to further develop the existing project Electric Blocks I. So, what is the goal of this Minecraft mod?

2. Background – **Sam:** the goal of Electric blocks 1 was to create a tool useful in conducting power flow study: which is a numerical analysis of electric flow of power in a system. Without a tool like this, power flow engineers make complex circuits and diagrams to represent the flow of power in a system such as the picture to the right. Commercial power systems are usually too complex to allow for hand solutions such as this, hence special tools like this project are created to provide physical models of power systems.

3. Background- The original Electric Blocks team worked hard to provide us with implementation of a working modeled powers system within Minecraft with accurate information and values .To describe how the mod works best, I will talk about the tech stack it uses. Starting at the bottom left for the Minecraft side of things. Minecraft is a Java-written game and therefore most developers use Java to implement most mod functions. Minecraft Forge is a modding framework that lets us create Minecraft mods with relative ease. The Electric Blocks mod uses Forge to implement the GUI, the new blocks implemented, and sends simulation information to another program – Electric Blocks PandaPower – for processing.

At the bottom-right of this tech stack, we have the Python programming language. Above that is PandaPower - an open-source tool for power system modeling where all our calculations are performed. Above *that* is Flask which allows for information to be received over the internet and then parsed for later use. At the top right is Electric Blocks PandaPower is a server which takes the data from Minecraft and creates the simulated power system. It then transforms the system into an easily readable format to be sent back to the Electric Blocks mod in Minecraft. All communication is done using JSON formatting. So, all we must do is parse the received information.

The picture on the right is the original Electric Blocks team's representation of an external grid connected by wires to a lamp. In PandaPower this connection is defined as an external grid connected by lines to a load.

4. Value Proposition: **(Ryan)** The previous iteration of Electric Blocks brought accurate power flow simulation using the medium of Minecraft. We extended onto this by adding all-around improvements to user experience, and existing documentation for the mod to become a viable product for educational and engineering purposes. We want to be able to provide anyone, no matter what experience level of power flow, ways to test, learn, and interact with power systems in a safe, and easily changeable environment.

5. **Greyson:** When discussing Electric Blocks with our client, Professor Daniel Conte de Leon, we were effectively given carte blanche to take the project in any direction we saw fit, within reason, so long as we were proud of our work by the end of development. With that said, Daniel gave us a general idea of where he imagined the project going and making Electric Blocks a viable product to learn power flow for primary and secondary education was part of his vision. So with that in mind, we decided to focus on

improving the user experience so Electric Blocks is viable for educational settings, as well as laying groundwork to on-ramp future Capstone teams working on Electric Blocks.

6. **Ryan:** Development process: For the development process, all the software packages currently used are open source, and the licenses allow us to code without needing to pay. Our schedule for the project was divided into 'sprints' of 2 weeks that were focused solely on smaller, more palatable objectives. Outside of this was meeting our capstone deadlines as a team.

Because this project exists purely in software, we were allowed more flexibility during development than other Capstone teams who have tight manufacturing and design deadlines. Additionally, each of us can work on separate features in tandem with minimal conflicts in version control.

7. New Implementation roadmap: *(Read the slide).*

8. New electrical modeling elements: Here are some of the new elements we created that help in modeling a functioning power system.

9. **Sam:** The first new electrical element we added is the battery block which simulates a storage element, this allows users to have a more extensive look into an active power system. Pictured below are the three different levels of user interfaces associated with the element that reflects data to a storage element. In Panda Power, the state of charge will not change over time, it is instead manually changed by the user to reflect a storage system with specific charge percentages and active charge or discharge values. The active power can be changed by assigning positive values for a charging storage unit and a negative value for a discharging unit.

10. Electric Furnace: One of the new blocks added is the electric furnace, an extension of the generic load which is a block that takes in power and use that power to produce some sort of function. This element allows the user to smelt and cook items with the power provided by the model they construct. This block implements the storage system that Minecraft uses, as well as recipe handling.

11. New Graphical User Interface enhancements: Next we'll talk about changes to the user experience while using the mod.

12. **Greyson:** One of the most obvious issues with Electric Blocks I was the development textures, seen earlier in this presentation. While they were high contrast and easy to identify from a distance, they were mostly just text, or very basic graphical designs. Our goal was to replace all of these with designs that were still easy to identify from a distance, but also resemble their real-life counterparts, so new users can intuit what they're used for. Also, as you might've noticed, we've also snuck in some extra information into their designs – Yellow borders indicate that the block supplies power, and blue corners mean they can be interacted with via a multimeter. As a bonus, these are also colorblind-friendly.

13. Further aiding in usability is a new plug-in for the mod "Here's What You're Looking At," or HWYLA for short. For otherwise-unmodified Minecraft, HWYLA displays useful information about blocks and entities in a tooltip just by hovering over them. With the goal of improving the workflow of Electric Blocks in mind, we wrote plug-in code for HWYLA that lets you see service status, active power, and other fun numbers for Electric Blocks at-a-glance, without needing to pull out a multimeter every single time.

14. **Sam:** One of the things we noticed about this project was the overwhelming UI that displays all the information to the user upon request. While this information is all a part of power flow, for those inexperienced in power flow, many of these variables are not understood. For Electric Blocks II we wanted to lower the barrier of entry to be more welcoming to users of any understanding level. Here we can see the previous UI for a bus block in the Electric Blocks Mod. For users new to power flow, they have extremely limited or even no understanding of what variables such as Voltage Magnitude and Voltage Angle are.

15. Here is the result of the UI reconstruction. Users can choose between an intro, intermediate, and advanced level multimeter item that displays the responding level of UI. This does not change the calculation of the power flow at all, it merely changes the UI to simplify the input and output information displayed. Here we can see an example of each different level with the bus block. The intro multimeter simplifies the UI and only displays Voltage and the active power flow in the bus block. The intermediate level introduces concepts such as reactive power. The advanced multimeter displays *all* variables associated with this block, this is meant for those who are completely familiar with power flow. These UIs are currently applied to every interactive block in Electric Blocks. What you see here does not include all the possible variables for a block - if we are to look at other blocks such as transformers, we will see many other variables specific to that block. As we mentioned before, there are specific multimeters that you can use to change this reflected data.

16. Here are our designs for the three levels of multimeters which are usable for interacting with the electrical elements. For all three multimeters, left clicking will result in the current highlighted block to be toggled off/on depending upon the status of the block at the time. Right clicking with each multimeter will present the level of UI depending on the type of multimeter you are using. The yellow multimeter will display intro level UI. The green multimeter will display the intermediate level of UI and the blue multimeter will display the advanced level of UI.

17. Multiplayer enhancements: Next we will talk about how we constructed a multiplayer session.

18. **Sam:** Since this mod is collaborative, that means that there is multiplayer functionality. What we did was create a running Minecraft Forge server using our mod on a local machine. We set up a LogMeIn Hamachi VPN to have users connect to the servers IP which we set as the server hosts IP address. This allows users to connect to the server, assuming they have installed the same version of the Electric Blocks mod as the server does. Since Electric Blocks PandaPower does not have any built-in security features, isolating the server by only allowing authorized users to communicate is important in maintaining security, which is why we decided to use Hamachi. To maintain security for our Electric Blocks PandaPower backend, we changed the address from where it receives information to the Minecraft server IP to ensure that only the server can communicate with it. A larger vulnerability called Log4j was discovered at the end of 2021 which allowed for remote code execution, meaning a malicious person could take over someone's computer by simply using the in-game chat feature. This vulnerability is present in multiplayer, so we solved this by making changes to the server startup files by disabling message lookups. In the picture to the right, we can see the Minecraft servers file directory which creates a multiplayer server. The mods folder will load up all related mods to Electric Blocks, the Log4j XML file which is used to get rid of the vulnerability, and the run.bat file which starts the server with the Log4j XML file in effect.

19. Documentation

20. **Greyson:** First off, we’ve updated documentation on the project website for the new features we’ve added, including a handoff document that will help future teams get started quickly – giving them a rundown of all the code we’ve touched and tips to avoid roadblocks we encountered.

21. Next, we've created an orientation map to help new users get started with the mod. Let’s take a walk through this right now.

22. <Video – 5:26>

23. Since Electric Blocks is an ongoing project for Senior Capstone students, we've provided recommendations for future teams on where this project can improve. A wider variety of interactive loads, implementing three-phase power, and a larger scale tutorial map would make great contributions to the project. There is also potential to implement realistic limitations on electrical equipment, time-based appliances, or an in-game menu to describe interactions between elements in a power system.

24. That’s all from us – any questions?

25. Thanks for coming!