General notes:
- Start meeting with the hard parts (things that need the most attention from Prof. Schafer
- Webpage for group is not yet up but will be shared later.
- Arduino IDE is a fine environment for coding but won't be able to make microcontroller do everything it theoretically could.
    - If more control is necessary, consider using SDK of the chip manufacturer.

Suggestions:
- Make sure there is a location (human descriptive) and address (computer descriptive) so that user can change settings or send information to the correct device and the base station can identify that device correctly.
- Think about solar power as a possible future implementation that could mitigate some of the frustrations of remote devices (notably battery changing) by increasing uptime without battery changes, and ease strict regulations on battery usage that are currently necessary to make the device work for a long time without a battery charge.
- Identify some ways to increase battery life.
    - One way might be to use an extra microcontroller with a MOSFET that would turn on and off the entire system when not in use.
- Create a webpage that is accessible from the ESP-32 to input settings for the device. This allows the user to update settings in the field, and not bring the device in and plug back into the computer to reprogram.
- Battery management system (BMS) design should include voltage regulation, over/undercharge protection, battery level test, and possible solar hook-up with charging circuitry
- Very good weatherproofing required for all seasons outside.

Action items:
- Identify and order the microcontroller(s) that will be used in the project.
- Written documentation how we will be coding and programming the chip.
- Identify controls/actuators/sensors that can demonstrate the device's usefulness.
    - Example: relays (might have to use optoisolators)