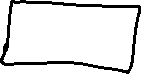
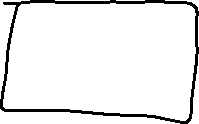
Flow of info:

First, the client listens for inputs. From there, a NodeJS server sends inputs to the ESP. After, the ESP listens on same network, and drives motors/servos depending on the data sent. Once these commands are finished, the ESP pings confirmation back to the NodeJS server which displays to the user through either video or command output.



Weaknesses: Local network can be accessed by anyone with relevant information, as simple as SSID/password and overwhelmed. If the IP of the PI or ESP is found, bad data can be sent to disrupt/overwhelm the connection. Also in general, an unstable connection can cause inaccurate info to be sent. If someone has access to the network, the data transferred can be sniffed/scanned, and if any of it is compromising, can be extracted.

How can a malicious user attack this system? 1. Denial of service (DOS) - send thousands of requests to ESP connected to the car, which the buffer can’t handle, so it loses the Wi-Fi connection. 2. Conversely, DOS the server so none of the commands can even send. 3. Physical tampering of motors or the Raspberry PI server, so the connection can’t be made or the instructions cannot be executed. 4. A user could spoof their IP or send bad data to trick the system. 5. A disruption of the local network, or a DOS attack to the router itself would prevent any part of the system from property connecting and sending data.

Many of these problems can be remedied by starting off with a mostly closed system. A password protected/encrypted local network increases the time and effort that an intruder must put in to breach a system. Having all sides of the system (e.g. PI/laptop, ESP) only be listening for certain IPs or instructions also reduces the change that incorrect information or spam will be transmitted amongst the involved systems. Data validation, as in only functioning with a certain instruction set, will even further reduce errors. Periodic monitoring of the network and data buffer for unauthorized/ unexpected users and kicking them out of the bubble adds an extra layer of security. By keeping a system as barebones as possible, the probability of a data breach is minimal.