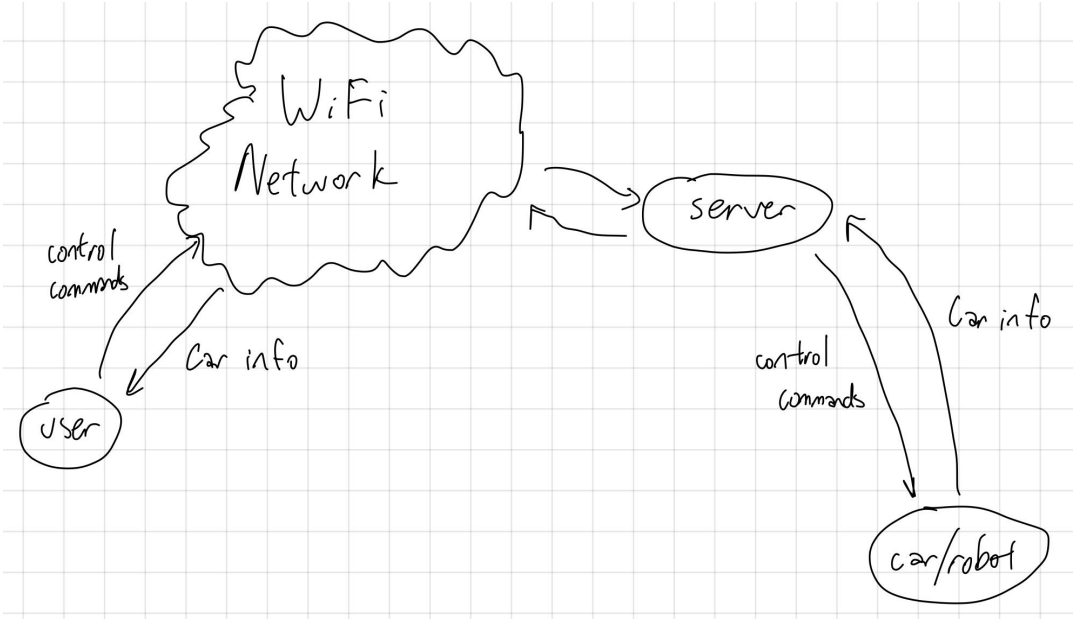
1. Sketch the overall flow of information to drive a car with remote control over the Internet.
    a. The user uses a device capable of connecting to the internet to control the car/robot. From a big-picture standpoint, the communication protocols can be established through wireless communication using WiFi technology. HTTP can be used for message delivery from user to car/robot. Taking a deeper dive, the user information is first communicated to the server, and then the server communicates with the car/robot. The server acts as a middleman between the user and the car/robot. The car/robot is then powered through microcontrollers responsible for handling the various operations of the car/robot, such as moving (other features could include a timer, speedometer, and sensor data). Information gathered from the sensors is then sent back to the user so that the user is aware of what the car is currently doing. The user, server, and the car/robot must have access to the internet at all times to ensure continuous communication between all aspects. Additionally, the overall system must be reactive enough to quickly transmit data.
    b. 

2. Identify weaknesses in your overall system (client, local network, internet, server, node.js, ESP32) with respect to security compromise.
    a. Client
        i. User devices can be vulnerable to malware attacks, which can lead to the hijacking of the car/robot control
        ii. Weak security on the client devices can cause security weakness as attackers could gain access to the car/robot control
    b. Local Network
        i. Weak WiFI network security could allow unauthorized users to connect with the potential of sensitive data to be captured over the network. Additionally, connections could be disrupted which means communication to the car/robot is broken.
    c. Internet

        i. Information transferred over the internet could be intercepted. Especially if the data is weakly encrypted, information could be changed.
        ii. DoS (Denial of Service) Attack could ensue and overload the server with unnecessary information, paralyzing the server from communication.
    d. Server
        i. Weak security could allow unauthorized access to the server meaning important communication information could be intercepted.
    e. Node.js
        i. External libraries included in the code may contain security vulnerabilities which could lead to a compromise of the entire server.
        ii. Code injection could allow attackers to access the server code.
    f. ESP32
        i. Firmware weakness in the microcontroller could lead to unauthorized access.
        ii. Attackers could manipulate the microcontroller by altering the hardware and potentially gaining unauthorized access.
3. List at least five ways can a bad guy attack your specific system. Be very specific
    a. Phishing Attack: Using phishing attacks, a user can be tricked into giving out sensitive information to an attacker. This is a common attack that has been done by tricking the user into thinking they're gaining something by giving out their information. The attacker can then use their own remote application or create their own look-a-like application that uses the user's sensitive information to gain access to the car/robot controls. The attacker would target the user's device (computer or phone) to collect the sensitive information.
    b. Network Attack: By gaining access to the WiFi, an attacker can intercept information transmitted over the WiFi network. If the information transferred over the network is not secure, then information that contains the control information can be altered before reaching the car/robot.
    c. Server Attack: If there is a vulnerability in the server, an attacker can inject code on the server side to gain unauthorized access. Once the Node.js server is accessed, then information can be manipulated for both the user and the car/robot side. For example, the user could see no signs of the car/robot acting abnormally, but the user could potentially gain full access to the car/robot.
    d. Hardware Attack: Knowing that the microcontroller is an ESP32, the attacker could replace the ESP32 microcontroller with their own. Direct command over the car/robot could ensue if the attacker can bypass all software security measures by accessing the hardware directly.
    e. Firmware Attack: If there is a security weakness in the ESP32 microcontroller firmware or framework, the attacker could learn to gain unauthorized access to the car/robot control. A weakness in the firmware includes how the firmware processes the data and control of the vehicle, so the attacker could send a payload that exploits this vulnerability, which allows the attacker to command the car/robot.
4. Describe a way to mitigate each attack

a. Phishing Attack: Implement a Two-Factor Authentication to add another layer of protection. This would make it difficult for attackers to gain access to the remote application.
b. Network Attack: Ensure that sensitive information transmitted over the network is encrypted and secure. Additionally, WiFi security protocols should be up to date.
c. Server Attack: Conduct frequent tests on the server side so that potential flaws in the server can be continuously fixed, including detection systems to alert about the attacks as soon as they happen.
d. Hardware Attack: Physically make the car/robot more secure. Seals can be done on the car to detect unauthorized access. Additionally, authorizing the hardware so that the car/robot is aware of the microcontroller being original or false.
e. Firmware Attack: Verification of the firmware with regular updates continuously to ensure that the firmware is authentic and not altered. Include alert systems or continuous monitoring of the system to detect unusual behaviors of the firmware.

5. Write up your answers and report -- please include graphics as necessary to support your response