Raspberry Pi Peer-to-Peer Network with IoT Security

**Project Description:** Create a Raspberry Pi-based peer-to-peer network with a focus on networking concepts, basic networking security, Internet of Things (IoT), and troubleshooting.

**Skills/Objectives:**

1. **Networking Concepts:** 
   * Gain hands-on experience in setting up a peer-to-peer network using Raspberry Pi devices.
   * Understand the basics of IP addressing, routing, and network configuration.
2. **Basic Networking Security:** 
   * Implement and configure firewall settings on Raspberry Pi devices.
   * Set up secure remote access using SSH key authentication.
   * Explore encryption methods for securing communication between devices.
3. **Internet of Things:** 
   * Use Wi-Fi connectivity on the Raspberry Pi to send collected sensor data to a cloud platform.
4. **Troubleshooting:** 
   * Implement error handling mechanisms and troubleshoot issues that may arise during the setup or data transmission.

**Project Phases:**

1. **Planning:**
   * Define the scope and objectives of the project.
   * List the hardware and software requirements.
   * Plan the network architecture and assign roles to each Raspberry Pi.
   * Create a project timeline with milestones.
   * Identify potential risks and mitigation strategies.
2. **Design:**
   * Specify the hardware components required for each Raspberry Pi.
   * Design the network topology and connectivity.
   * Develop a security plan, including firewall configurations and encryption methods.
   * Design the structure of Python scripts for IoT device communication.
   * Create a plan for error handling and logging.
   * Design the layout and features of the dashboard.
3. **Testing and Feedback:**
   * Set up Raspberry Pi devices according to the design.
   * Test the peer-to-peer network connectivity.
   * Implement basic networking security measures and perform security testing.
   * Integrate and test IoT devices, ensuring proper data transmission.
   * Implement troubleshooting mechanisms and conduct testing.
   * Gather feedback from testing and make necessary adjustments to the design.
4. **Documentation:**
   * Document the hardware setup process for each Raspberry Pi.
   * Record step-by-step instructions for configuring the network.
   * Document the implementation of security measures and troubleshooting mechanisms.
   * Create detailed guides for integrating and programming IoT devices.
   * Develop documentation for the dashboard.
   * Compile all documentation into a comprehensive project report.

**Python Libraries:**

You can use Python libraries like tkinter, socket, netifaces, Adafruit CircuitPython DHT and paramiko for creating a graphical user interface (GUI) and generating displays, security and connecting via networks.

**Project Structure:**

* server.py: The main Python script containing the server logic.
* client.py: The main Python script containing the client logic.
* app.py: The script for creating the GUI and handling user interactions.
* README.md: Documentation for using the calculator app.
* requirements.txt: List of required libraries and dependencies.