Project Review: Raspberry Pi Automation Tool

This project aimed to develop a tool using a Raspberry Pi that could control home appliances like coffee machines, lights, or heaters based on a user-defined schedule. In this review, we reflect on our group dynamics, technical challenges, achievements, and the key lessons we learned throughout the process.

What went well

1. Group Coordination:
   Our team managed to divide tasks effectively. Each group member knew their responsibilities, and this clarity reduced overlap and confusion, allowing us to maintain steady progress.

2. Ideating the Concept:
   We successfully brainstormed various ideas for the project, and each idea was critically evaluated using multiple ideation techniques like the 5 Whys and the Disney method. Defending the concept to others helped us solidify our approach.

3. GitHub Management and Group Board:
   We used GitHub to track our progress and manage version control effectively. Each member could easily access the latest code and updates, preventing any version conflicts. We also maintained a group board for task allocation and tracking, which helped us stay on the same page.

What went wrong

1. Time Management Issues:
   Our team took too long to set up the prototype, which delayed the later stages of development. This was partly due to underestimating the time it would take to get the basic framework up and running.

2. Raspberry Pi Setup Problems:
   We encountered issues with the Raspberry Pi hardware itself, as it seemed to be faulty. Unfortunately, we spent too much time trying to troubleshoot and resolve this issue, which further delayed our progress.

Top 3 Lessons Learned

1. Avoid last-minute rushes:
   Procrastinating until the final stages of the project led to a stressful time crunch. We realized that it's crucial to allocate time more effectively and avoid leaving major tasks until the final week to minimize risks.

2. Effective GitHub Use:
   We learned that maintaining clear and consistent updates on GitHub, without spamming unnecessary commits, improves the clarity and accessibility of the codebase. This streamlined the workflow and allowed

us to collaborate more efficiently.

3. Don't over-focus on one problem:
　　We spent too much time trying to fix the Raspberry Pi issue instead of considering alternative solutions. The takeaway is to shift focus when a problem cannot be solved within a reasonable timeframe and explore backup plans more quickly.

Reflection of Learning Goals

Throughout this project, we used various ideation methods to define our product, including the 5 Whys, AEIOU, Empathy Map, and the Disney Method. These methods helped us explore different facets of the problem and come up with a well-rounded solution.

Although we initially planned to incorporate sensor kits into our project, time constraints forced us to adapt by using indicator LEDs instead. This pivot was a valuable learning experience as it allowed us to focus more on the programming aspects of the Raspberry Pi, enhancing our technical skills in that area.