// Borko

* Intro
  + What smartie connect is recap
  + How it works recap
    - Sensor and activators class to mimic real life objects
    - Nodes to manage values gotten from sensors and use activators
    - Mqtt makes nodes communicate with each other and with the discord
    - The discord is how we communicate with the user
* Setup -
  + Libraries used
    - Made a python environment, so you shouldn’t need to install anything, but if you encounter problems this is the list of libraries used (enter as a diagram so we don’t use words)
      * --os --sys --threading --datetimer --time --discord --random --termcolor --paho.mqtt.client --ssl --tkinter --json --unittest
  + software to be installed
    - You will need a discord account
    - Two terminals
      * One to simulate the main raspberry pi node and one for the discord
  + How to run it
    - Once the files have been downloaded into a folder, enter the \*enter folder name\* directory using the cd command
      * If we have extra words explain how to find directory
    - Do the following commands
      * Terminal 1
        + Enter “python3 code/helper\_functions/tkinter\_gui.py” for mac and linux
        + To run it without a gui “python3 code/nodes/main”
      * Terminal 2
        + Enter “python 3 code/discord/main\_discord.py”
    - How to make tests run
* Services
  + MQTT
    - Explain use
    - Say that we use the uni’s credencials
  + Discord
    - Explain use
* Implementation Review
  + Design and changes
    - We separated the speakers node from the safety one so users that want their doorbell to ring without wanting a smoke detectors can do that, also makes code more organised
    - We decided that storing the user info in a json wasn’t necessary as we were already storing it in a json-like dictionary, it could be saved in a json when the data is sent to the company if wanted
    - The user gathering design has changed from the one described on page seven pseudocode 1, explain how, the why is basically just because it made more sense, but write it better
  + Details of any features in addition to the core required features that you have implemented or those that you have not implemented.
    - List and small description of the following (not how it works, just what they are)
      * Discord
      * Unit tests
      * GUI
      * API
  + Details of any features / work that has been undertaken in addition to implementing the core elements required, to justify the achievement of an A grade
    - One paragraph per every feature mentioned above
  + Things we weren’t able to implement
    - Spontaneous discord message for events
    - Check what is on/off/open/closed
    - More unit tests
    - If this was a real thing we would have made a custom app with custom messaging instead of discord
    - Wanted private channel but made public, would be ideal if he had our own app
    - Multiple users mqtt messages
    - Being able to delete, create and modify devices
    - (if we don’t have time) we were planning on making the user agree to our terms during the registration process, but we didn’t have time (make up better excuse)
* Conclusion
  + Even if we didn’t implement everything we wanted, smartie connect covers all the required core elements and more
  + We implemented extra features that are extracurricular
    - Discord
    - Unit test
    - GUI

and proved the capacity to research

* New language(?)
* Overall smartie connect is good

Add repository link on cover page and appendix