Mini Project - "The remote MQTT controlled PixelEater"

1. Implementation Phase

Work to be done in Mini-Project.

- Implementation
- Live Demonstration

Outline of work

- Setup and configure the Mosquitto broker
  - Config files, cert creations, acl & password files are to be documented.
  - Consider lifetimes and experiations
- Design and implement the three clients.
  - use crontab to boot with clients
- Logger
  - The logger may be setup using mosquitto_sub or MQTT.fx
  - The logger is not to be hosted on the RPi
  - Make sure the transaction logs are written to a file
    - Start a new file every 24 hour period
- Player
  - Create a minimal user interface on the SenseHat
  - To be programmed with python and paho-mqtt
  - Input validation
  - Start - publish start messages as appropriate
  - publish joystick events as appropriate
  - Optional:
    - read a logfile and publish
    - play the pixeleater game locally too
- Recorder
  - to be programmed with python and paho-mqtt
  - input validation
  - start - subscribe to start messages
  - when started:
    - initial board color: white
    - pixelEater color: yellow
    - setup board, and place the joystick at position (0,0)
  - subscribe to joystick events

The start topic

- Strings are json encoded
  - "ordered" - the joystick events shall be played out when recived. That is, ignore timestamp information
  - "timed" - the time difference between the is considered (if possible). This means that one way have to introduce delays. This options is used when a recorded file if used by
  - "stop" - The recorder shall the blackout the Sense Hat

Topics

- dat235/groupname/start - retained, qos-1
- dat235/groupname/joystick - qos-0

Payloads

- Start
  - "ordered" - a start order
  - "timed" - a start order
  - "stop" - a stop-and-wait order
  - "EXIT" - an order to exit the program
- joystick json encoded event tuple