Assessment 2: Python group project

Demonstration and online submission; week 23;

20% - group project

Students will be writing code related to a Python project. The final version of their software will be demonstrated to the instructors at week 23.

Students will have to write Python code for creating an original light animation on the 6 LED bars mounted in the lab. They will be provided with:

- interfacing library `opc.py` which allows wireless access to the LED device
- simulator software which allows to test the code locally before connecting to the LED device
- example scripts showing how to use the library and produce some basic animations

Originality in the design and implementation of animations is strongly encouraged, as is the use of programming tools beyond those explained in class. Additional grades are awarded for employing external interfacing methods (e.g. Arduino based).

Final assessment will be done in class on 27/03 (week 23). You will be allocated a 5 minute slot on that day to demonstrate and talk through your code during the class. We will mark your presentation and code to award you a grade on the Middlesex 20 point scale. If you are not present for your allocated slot we will run the code based on your online submission, but your mark will be significantly reduced - the online submission deadline is at 11PM 26/03/2019.

Summarising, project 2 will be assessed based on quality of final submission, and knowledge demonstrated during the live presentation, according to the rubric below.

Project 1 evaluation rubric

Item	1	2.1	2.2	3	Fail
Code 35 %	High quality code exhibiting deep knowledge and creativity	Code shows structure, understanding of programming tools and personal initiative	Code shows some personal initiative and good understanding of basic concepts	Code works, but is mostly based on given examples with little personal contribution	Code does not work, and shows little understanding of language/algorithms
Documentation 10 %	Advanced comments and documentation	Satisfactory comments and documentation	Basic but correct documentation and comments	Poor documentation and comments	Missing or minimal comments or documentation
Animations 25 %	Highly creative and varied animations using a range of techniques	A range of novel animations with clear personal ideas	Some new ideas incorporated into simple animations	Mostly variations on given examples, with few personal ideas	Same as in examples, with no demonstration of any personal contribution
User interface 15 %	Use of sensors for getting user input and changing animations	Keyboard interface with some options, but no use of external sensors	Basic (e.g. mouse movement, multiple keypresses)	Very basic (e.g. single keypress)	No interface
Presentation 15 %	Very good and organised demonstration	Careful demonstration of code and user guide	Demonstration is clear but not very organised	Basic explanation of code and animations	No clear explanation, lack of understanding

Technical information

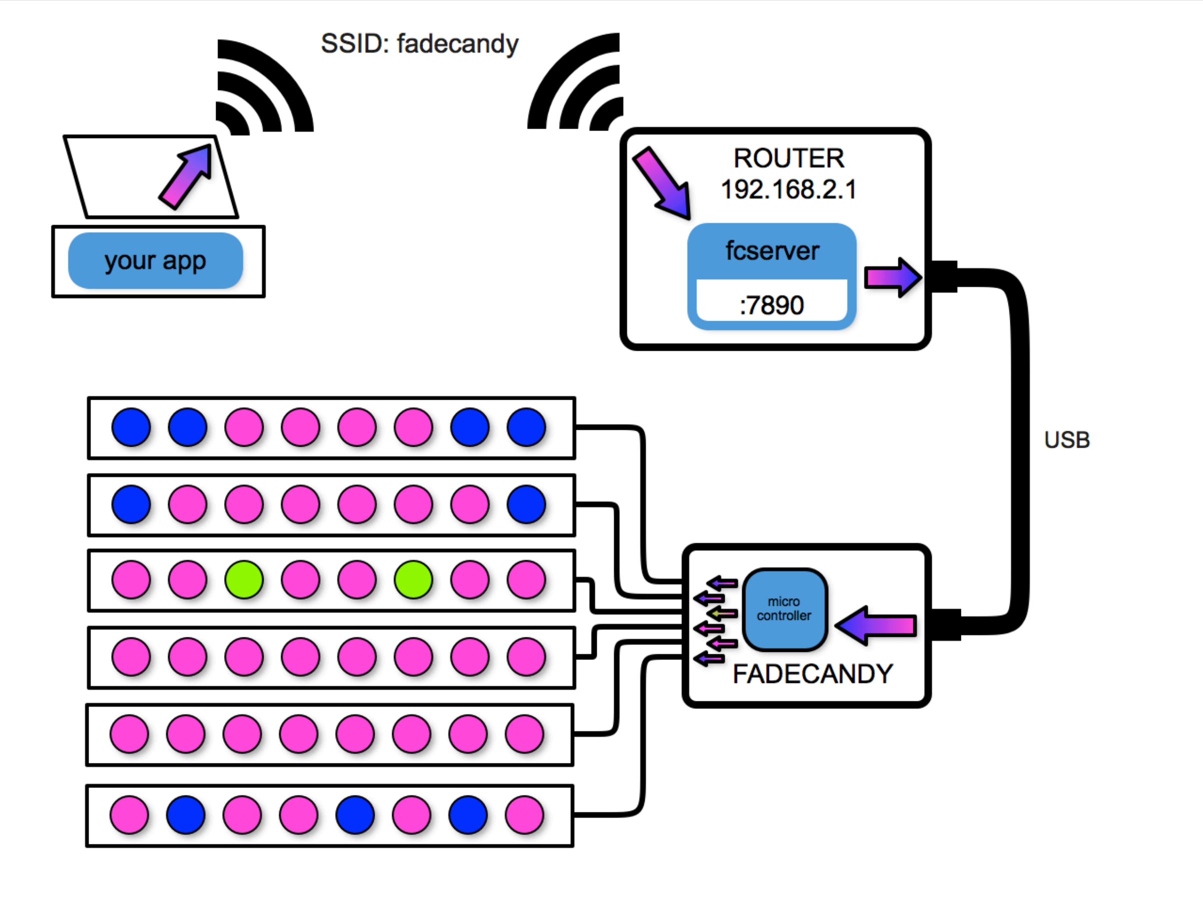
The installation has 6 led strips suspended in the space between the ground and first floor in the labs in the Ritterman builiding.

Each strip has 60 RGB LEDS. The strands are attatched to a fadecandy controller which enables them to be controlled over USB. The fadecandy controller is plugged into a wifi router with a USB socket which is running the fcserver application. This application enables us to send data to it using the OPC format by connecting to the wifi network the router creates. This data is then used to control the LED strips.

When all the strips are installed they will appear to be connected in a long line. The first strip will be addressable as the first 60 LEDs, (0-59), the next strip will be LEDs 60-119, going up all the way to 359. If you send to many values the additional ones will have no effect, but will not generate an error. If you send less values then only the ones that you send data for will be updated.

The fcserver will be running at the address 192.168.2.1:7890 once you connect to the wifi network, you will need to set this address in your application.

When you are running the simulator you need to send your data to the address 127.0.0.1:7890 or localhost:7890 (the local loopback address which will resolve to your machine) as both applications will be running on the same machine.


SSID: fadecandy
your app
ROUTER
192.168.2.1
fcserver
:7890
USB
micro controller
FADECANDY