The doorbell sends an email to the home owner, and their phone receives the email and connects via BLE to a smart wristband, which vibrates, makes a noise and lights up. This means the home owner can be alerted that someone is at the door wherever they are – whether they are in the garden or out of the house.

Currently, for this proof of concept, the doorbell is connected to a laptop via USB, however in the final fully integrated version, the doorbell itself will be connected to the internet. Once the phone receives the email, it uses BLE to communicate with the BLE board in the wristband.

Possible Expansion: Video feed from door to phone.
The wristband could be turned in to a smart watch and you could receive the video feed onto your wrist.
You could remotely unlock the door if it is someone you know.
The wristband could be used as an alarm clock.

The ultimate aim would be to have a product which has a camera and remote door unlock, meaning you can let guests you know in from somewhere else in the house or if you are out of the house. The wristband would also be waterproof meaning you could be notified even in the shower.

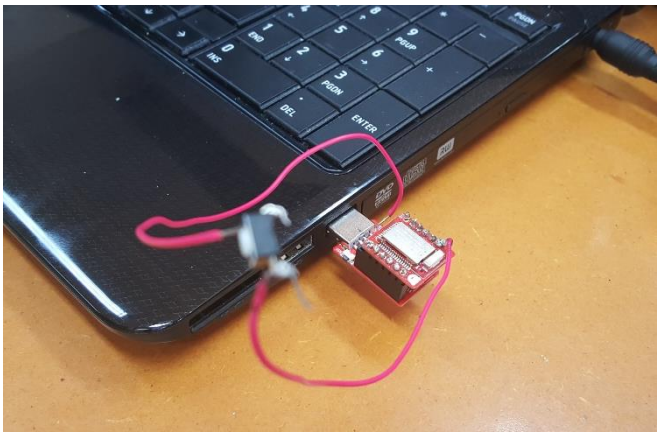
Alternate uses: Use by deaf people to be alerted when their doorbell is rung, or an alternate alarm clock to the usual bed vibrating alarm system.

Development issues: When the BLE board is attached to the mbed, which is connected to the computer, the debugging software for the board overrides some of the inputs to the board. It was not obvious that this was the case and it was difficult to find out which inputs we could use that wouldn't be overridden.

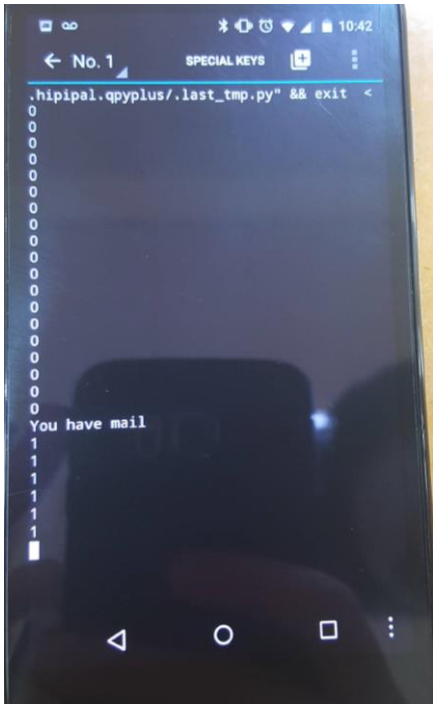
Large traffic on the wifi meant that the websocket and webserver were temperamental, which meant that our initial solution of reading the data on a webpage instead of transmitting button states via email only sometimes worked.

Little documentation for using python on BLE.

The doorbell



Phone program scans for mail



The smart wristband