# Project Configurability

* Code modularity, with each subsystem being separated into a separate file.
* NRF52840, M5Core and Thingy:52 can all be reprogrammed in C/C++. The GUI is written in Python. Thus, everything is written in a popular programming language, making it easy for further development.
* All Python libraries used can be installed using pip: pyserial, Tensorflow, matplotlib, keyboard, numpy, customtkinter, PIL, mqtt, pandas

# Field Deploying Plan

To deploy the system:

* Flash Thingy:52 with code under mycode/apps/ibeacon
* Flash NRF52840 with code under mycode/apps/central\_hr
* Flash M5Core2 with code in the home folder name m5core2.ino
* To open the GUI, run main.py

The plan/next steps:

* Make the ML model more generalized by collecting more data
* Thinking of turning this into a Fitbit with a cloud service
* Thingy:52 and M5Core combine into the Fitbit. The PC GUI access the info via the cloud server.

To help non-project members understand the code:

* Code documentations: block diagram, message protocol diagrams. It is under the documentations folder on GitHub.
* The code has useful comments.