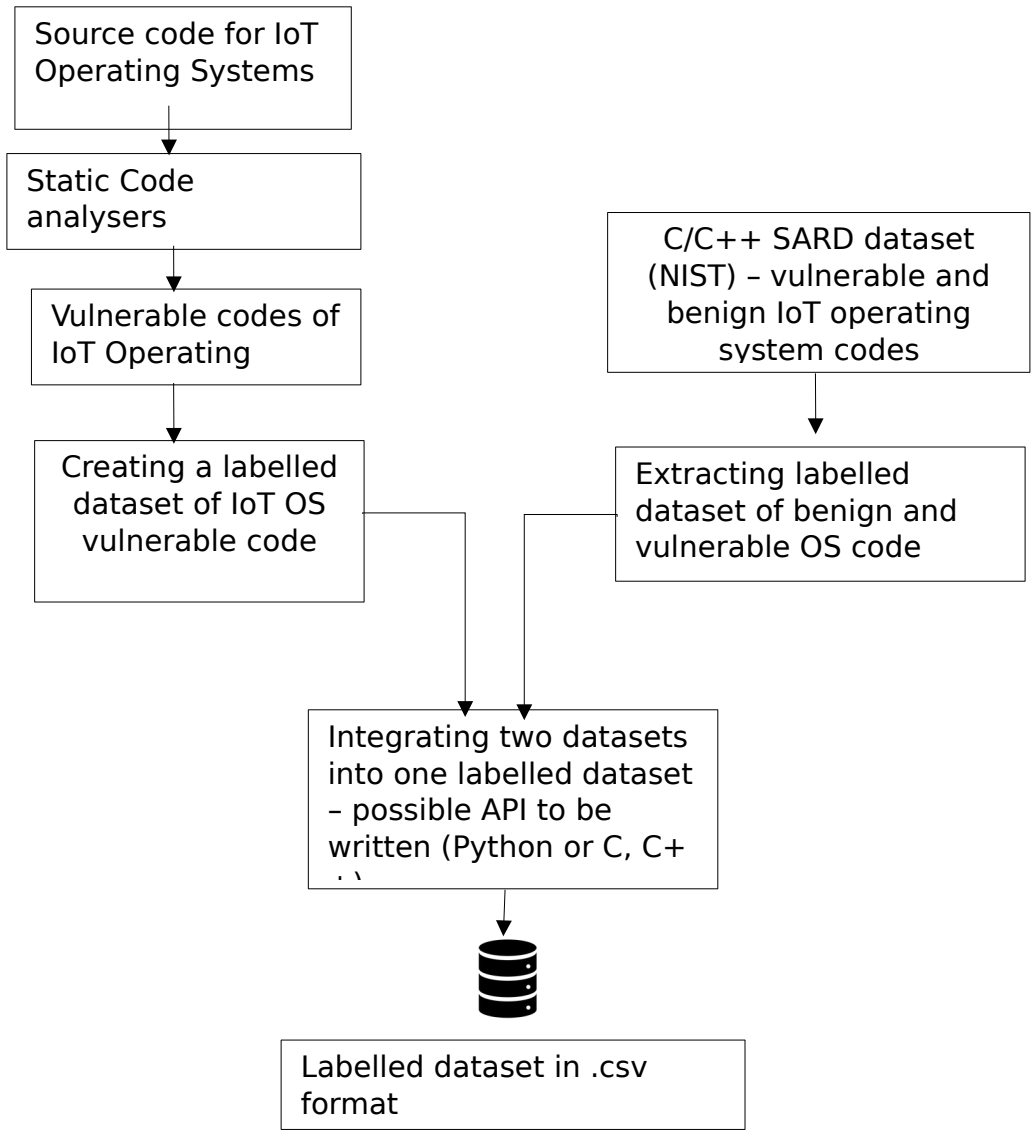
1) Workflow / Requirements for creation of the labelled dataset tool



2) IoT Operating systems
   - A review on IoT Operating Systems: link
   - Top 7 IoT Operating systems in 2020 : link
   - Also consider Amazon's free RTOS

S. No.	IoT OS title	Language
1	Contiki / Contiki NG	C, C++
2	Riot OS	C, C++
3	TinyOS	nesC
4	Raspbian	C,C++, Python
5	Ubuntu Core	C, C#
6	Zephyr	C, C++
7	Windows 10 for IoT	C, C++
8	Amazon RTOS	C

Note: Most of the embedded IoT devices are written in C programming language, whereas C++ is the preferred for more complex Linux implementations

Points/ Tasks to be considered:

- Can we extract the OS source code for all of the above OS?
- What about different versions of the OS?
- Possibly consider, plugging all of them from Github to project repo

3) Static Code analysers
   a. Static analysis tool -I C++
   b. Flawfinder
   c. RATS
   d. More SAST scanners can be found at https://owasp.org/www-community/Source_Code_Analysis_Tools
      Note: Need to check whether they are open source and work with C, C#, nesC and /or Python

4) Creation of labelled datasets from both the workflows
   a. Collate and compile a dataset from the SAST utilities
   b. Extracting data from NIST's SARD dataset into a compatible list consisting of benign and vulnerable code

5) Write an API to merge both the datasets towards a format compatible as input to Dev's vulnerability detection engine.