Grading Criteria

- Ability to successfully achieve the objectives (40%)
- Repeatability and robustness of unlock sequence (via video demo) (20%)
- Ease of use (10%)
- Creativity (10%)
- Well written code (10%)
- Complexity (10%)

Github

https://github.com/VeronicaGupta/Sentry

Functionality

1. Data collection (hand movement data)
   a. Accelerometer
   b. Gyro
2. Events
   a. Sequence/Key/Password storage
      i. Start - Record Key
      ii. Stop - Enter Key .
         1. Successful- LED GREEN
         2. Failed- LED RED
      iii Remove/Add Sequence: Long press "Record Key" to reset blink Red LED twice.
      iv Recording: Blink Green LED during recording

3. Event handling
   a. At start event
   b. At stop event

4. Re-record Gesture Functionality
   a. Reset button or long press on the Record Key to overwrite the saved gesture

5. Re-lock: Auto-lock after 10 seconds or button press

Gestures and Data Collection

a. Define how gestures will be captured

Resources Enforced

1. Board: STM32F429ZI
2. Coding framework: PlatformIO
3. Functions: Mbed drivers/HAL
4. Sensors: STM32 integrated sensors
5. Sensor must be held close to the fist while performing the gesture sequence
6. “Enter key” for unlocking and “Record Key” for recording the sequence should be used

Decide-

1. Sequence format ?
2. Remove and add sequence ??
3. How to show the unlock event ?
4. Once unlocked, how to lock again?
5. Axis and rotations to use for unlocking?

Class notes about project from prof:

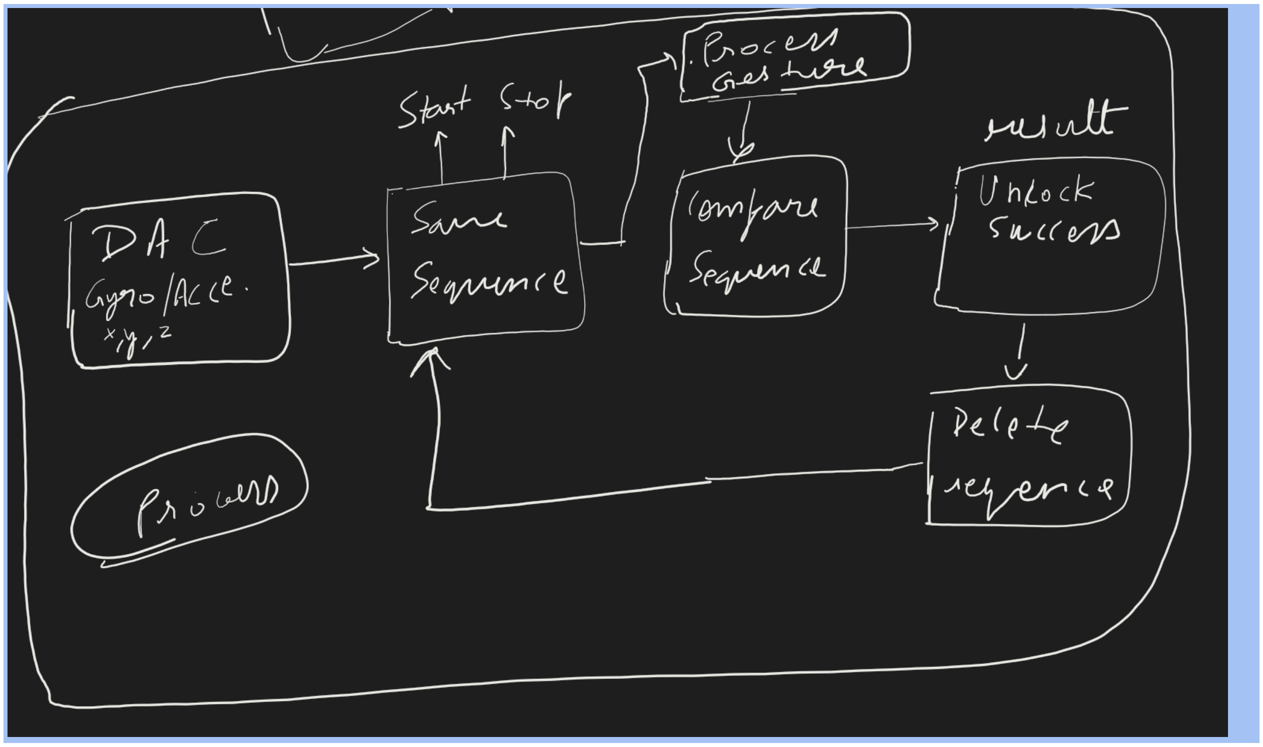
1. Preventing access and granting access
2. Get a complex 3sec gesture and board in hand required
3. Train against the gesture and then lock and unlock
4. Put board on learn mode
5. Get a test function
6. One team person trains and other will demonstrate the lock and unlock
7. Get information from gyro through SPI
8. Deliverable is folder and demo
9. One submission per group - video and folder
10. One person from group uploads the submission

Tasks

4 Dec 2024

1. Individual -
    1. a. Download- Recitation 3 to 7
    2. b. Implement on board
    3. c. Theory
        - i. Timers,
        - ii. teleplot,
        - iii. Communication protocol (SPI),
        - iv. State machines
        - v. Filters
        - vi. LCD screen
2. Gyro and accelerometer (see recitation)
    - a. Process
        - i. Gyro
            1. X
            2. Y
            3. Z
        - ii. Accelerometer
            1. X
            2. Y
            3. Z
    - b. Show in teleplot
3. Recording - Process Collected Data (Sample from file, independent of MC)
    - a. Send record/data in file (txt, excel, json)
    - b. Receive record/data that recording
4. Filtering into a struct- gesture
    - a. Research, attend office hours for same, mail TA, averaging, low pass, high pass-> decision
    - b. Threshold setting
5. Add Events to Trigger functions
    - a. Saving action
        - i. Start button
        - ii. Stop button
    - b. Unlocking action
        - i. Successful- LED GREEN
        - ii. Failed- LED RED
    - c. Locking action
        - i. Successful - LCD Screen
        - ii. Failed- LCD Screen
6. Event handling

a. Saving action
    i. Delete old records
    ii. Save new record
b. Unlocking action (algo):
    i. Check if new record matches to saved record
        1. Research, github, neha’s algo
        2. Basic implementation POC
        3. Add POC algo to our code



10 Dec 2024

1. State Diagram- Deterministic solution
    a. Each critical code in different functions
2. Psuedo Code
    a. Recording and Retrieving data (independent of hardware) (Gx, Gy, Gz, Ax, Ay, Az)
    b. Collection, Filtering and add new parameters (based on Gx, Gy, Gz, Ax, Ay, Az) algo
        i. Study how to capture this data<-
        ii. Finalise the filter (combination of filters)
        iii. Try 5 gestures
    c. Recognition algo->(base maths-> if and else/ ml algo)
        i. Study (ml/if else)

ii. Hal library (what add for ml)

12 Dec 2024-Task

1. Showing something christmassy on LCD for output- Neha (3 screens-> unlocked (success), failure reaction, locked screen and idle (something moving))
2. Recognition Algo (projects)- Sharmitha
3. Record parameters to file (json) (Take sample values) 3 seconds, retrieve them back in a structure- Sharmitha
4. Filtering (types of gestures best filtering)-Parnika
5. Interrupt from button to lock again- Parnika

Suggestion-TA
The techniques you used should show promise and might get you your solution!
Memory constraints are tricky, but here are some ideas to think about:

1. Reduce Data: Instead of storing every sensor point, you could extract key points (like peaks/valleys) or downsample the data.
2. Sliding Window: Can use buffers to store only the last few sensor points and compare incrementally.
3. Memory-Efficient DTW: Use a method to limit the DTW matrix or process it incrementally.
4. Use CMSIS-DSP: Functions in this library can reduce computation and memory overhead.
5. Simplify Comparison: Instead of comparing full gestures, use a summary (like an average or key features) and compare those.
Links:

Touch Screen Demo --> https://os.mbed.com/teams/ST/code/DISCO-F429ZI_LCDTS_demo/

LCD Demo --> https://os.mbed.com/teams/ST/code/DISCO-F429ZI_LCD_demo/

Gyro Demo --> https://os.mbed.com/teams/ST/code/DISCO-F429ZI_Gyro_demo/

14 Dec 2024-Task