

## Introduction

Aging is a fact of life. However, as we age, we become more vulnerable to injury and go through a vastly more extended healing cycle when we become injured. For example, according to the CDC, one out of five falls for individuals over 65 causes serious head injury or broken bones. This information is staggering since over 3 million patients over 65 are treated in emergency rooms every year for injuries from falls.

The nearly 800,000 patients who end up hospitalized due to a fall tend to have head and hip injuries so severe they end up in long-term hospice care. These falls are responsible for more than 50 billion in medical costs within the US alone. With Medicare and Medicaid covering around 37.5 billion of that total cost in 2020.

In some cases, it's not the fall itself but what happens after the fall that causes long-term damage to the patient. For example, being trapped on the floor for hours unable to get up can add to the injury leading to long-term disability. In addition, dehydration, blood loss, and other medical conditions can occur while trapped on the floor leading to life-threatening situations.

## Solution

The fall detection created by our team is a wearable device that can detect a fall and notify loved ones, medical staff, and first responders of the incident. The device is cheap and easy to use but has all the benefits of keeping caregivers in the loop if the elderly or disabled person they care for is in a fall situation. The device is worn on the wrist or upper arm to maximize comfort with heart attack, stroke, and palpitation detection and warning, coming with version 1.1.2.

To build our product, we utilized the sensorTile running the ALLMEMS1 firmware. The development was possible via the Cradle expansion board connected via USB. The information from the tile is sent to a local raspberry Pi4 via Bluetooth for monitoring the wearer's status. Once the algorithm detects the fall, the program uses the Twilio API via python to alert the predesignated parties.

Utilizing the Python SDK blue\_st\_sdk.manager module combined with Bluepy and Blue-st-SDK, we connected the sensorTile and the raspberry pi.

```
alkene — pi@raspberrypi: ~/Downloads — ssh pi@192.168.0.145 — 80x24
pi@raspberrypi:~/Downloads$ sudo python3 accelerometerlogger.py

#####
# SensorTile data logger #
#####

Scanning Bluetooth devices...

Discovery started.
New device discovered: AM1V310.
Discovery stopped.

Connecting to AM1V310...
Device AM1V310 connected.
Connection done.

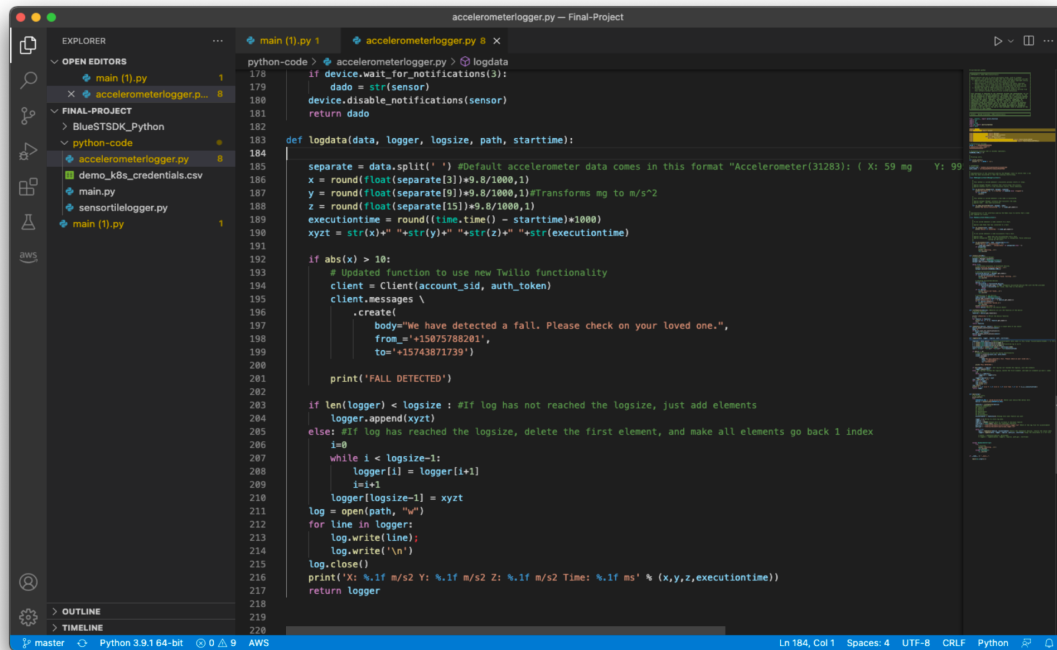
Features:
0) Activity Recognition
1) Temperature
2) Pressure
3) Magnetometer
4) Gyroscope
5) Accelerometer
6) ADPCM Audio
```

To display the data output by the Blue ST SDK\_Python in g unit.

```
alkene — pi@raspberrypi: ~/Downloads — ssh pi@192.168.0.145 — 80x24
Accelerometer(2374): ( X: -25 mg Y: 75 mg Z: -967 mg )
Accelerometer(2388): ( X: -24 mg Y: 74 mg Z: -966 mg )
Accelerometer(2402): ( X: -26 mg Y: 73 mg Z: -967 mg )
Accelerometer(2416): ( X: -23 mg Y: 75 mg Z: -965 mg )
Accelerometer(2430): ( X: -26 mg Y: 72 mg Z: -966 mg )
Accelerometer(2444): ( X: -22 mg Y: 73 mg Z: -965 mg )
Accelerometer(2458): ( X: -27 mg Y: 76 mg Z: -967 mg )
Accelerometer(2472): ( X: -24 mg Y: 76 mg Z: -965 mg )
Accelerometer(2486): ( X: -24 mg Y: 72 mg Z: -966 mg )
Accelerometer(2500): ( X: -25 mg Y: 75 mg Z: -965 mg )
Accelerometer(2514): ( X: -23 mg Y: 75 mg Z: -964 mg )
Accelerometer(2528): ( X: -27 mg Y: 75 mg Z: -967 mg )
Accelerometer(2542): ( X: -25 mg Y: 72 mg Z: -967 mg )
Accelerometer(2559): ( X: -27 mg Y: 74 mg Z: -965 mg )
Accelerometer(2573): ( X: -27 mg Y: 75 mg Z: -966 mg )
Accelerometer(2587): ( X: -25 mg Y: 75 mg Z: -966 mg )
Accelerometer(2601): ( X: -27 mg Y: 76 mg Z: -966 mg )
Accelerometer(2615): ( X: -22 mg Y: 72 mg Z: -965 mg )
Accelerometer(2629): ( X: -24 mg Y: 72 mg Z: -965 mg )
Accelerometer(2646): ( X: -22 mg Y: 71 mg Z: -967 mg )
Accelerometer(2660): ( X: -22 mg Y: 73 mg Z: -965 mg )
Accelerometer(2674): ( X: -22 mg Y: 72 mg Z: -965 mg )
Accelerometer(2688): ( X: -26 mg Y: 71 mg Z: -965 mg )
```

```
alkene — pi@raspberrypi: ~/Downloads — ssh pi@192.168.0.145 — 80x24
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 6710.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 6822.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 6935.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 7047.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 7178.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 7291.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 7403.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 7516.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 7647.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 7760.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 7872.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 7985.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 8097.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 8228.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 8341.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 8453.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 8566.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 8678.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 8791.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 8903.0 ms
X: -0.3 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 9016.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 9128.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.4 m/s2 Time: 9260.0 ms
X: -0.2 m/s2 Y: 0.7 m/s2 Z: -9.5 m/s2 Time: 9372.0 ms
```

Once the acceleration threshold is hit, the print('Fall Detected') is run:



```
python-code > accelerometerlogger.py > logdata
178 if device.wait_for_notifications(3):
179     dato = str(sensor)
180     device.disable_notifications(sensor)
181     return dato
182
183 def logdata(data, logger, logsize, path, starttime):
184
185     separate = data.split(' ') #Default accelerometer data comes in this format "Accelerometer(31283): ( X: 59 mg Y: 99
186     x = round(float(separate[3]))*9.8/1000,1)
187     y = round(float(separate[9]))*9.8/1000,1)#Transforms mg to m/s^2
188     z = round(float(separate[15]))*9.8/1000,1)
189     executiontime = round((time.time() - starttime)*1000)
190     xyzt = str(x)+" "+str(y)+" "+str(z)+" "+str(executiontime)
191
192     if abs(x) > 10:
193         # updated function to use new Twilio functionality
194         client = Client(account_sid, auth_token)
195         client.messages \
196             .create(
197                 body="We have detected a fall. Please check on your loved one.",
198                 from_="+15073788281",
199                 to="+15743871739")
200
201         print('FALL DETECTED')
202
203     if len(logger) < logsize : #If log has not reached the logsize, just add elements
204         logger.append(xyzt)
205     else: #If log has reached the logsize, delete the first element, and make all elements go back 1 index
206         i=0
207         while i < logsize-1:
208             logger[i] = logger[i+1]
209             i=i+1
210         logger[logsize-1] = xyzt
211     log = open(path, "w")
212     for line in logger:
213         log.write(line)
214         log.write('\n')
215     log.close()
216     print('X: %.1f m/s2 Y: %.1f m/s2 Z: %.1f m/s2 Time: %.1f ms' % (x,y,z,executiontime))
217     return logger
218
219
220
```

## Future

The program's next iteration will be sampling the incoming information through a Machine learning classification using Edge Impulse. As the data gets more precise and unaddressed edge cases to come to light, the chance of false alerts lessens, providing a better overall user experience. In tandem with this fine-tuning, we'll be isolating the heart rate to alert a caregiver of any potential heart attack, stroke, or abnormal vibration. These machine learning algorithms will form the backbone of third-party technologies via Apple and Android-enabled devices and sensors as well as custom-built devices 3d printed by our team.

## Bibliography

Ausubel, Jacob. "Older People Are More Likely to Live Alone in the U.S. than Elsewhere in the World." Pew Research Center. Pew Research Center, September 24, 2020.

<https://www.pewresearch.org/fact-tank/2020/03/10/older-people-are-more-likely-to-live-alone-in-the-u-s-than-elsewhere-in-the-world/>.

Gomes, Felipe. "SensorTile Sensor Data Monitoring Using a Raspberry Pi." Hackster.io, June 18, 2020.

<https://www.hackster.io/felipsz/sensortile-sensor-data-monitoring-using-a-raspberry-pi-7a3663#toc-lets-connect-our-rpi-to-the-sensortile-4>.

"Important Facts about Falls." Centers for Disease Control and Prevention. Centers for Disease Control and Prevention, February 10, 2017.

<https://www.cdc.gov/homeandrecreationalsafety/falls/adultfalls.html>.

"Keep on Your Feet-Preventing Older Adult Falls." Centers for Disease Control and Prevention. Centers for Disease Control and Prevention, December 16, 2020.

<https://www.cdc.gov/injury/features/older-adult-falls/index.html#:~:text=About%2036%20million%20falls%20are,bones%20or%20a%20head%20injury>.

Monk, Simon. "Adafruit's Raspberry Pi Lesson 6. Using SSH." Adafruit Learning System. Accessed December 10, 2021.

<https://learn.adafruit.com/adafruits-raspberry-pi-lesson-6-using-ssh/using-ssh-on-a-mac-or-linux>.

STmicroelectronics. "Blue\_st\_sdk Package¶." blue\_st\_sdk package - BlueSTSDK 1.4.0 documentation. Accessed December 10, 2021.

[https://stmicroelectronics.github.io/BlueSTSDK\\_Python/blue\\_st\\_sdk.html#module-blue\\_st\\_sdk.manager](https://stmicroelectronics.github.io/BlueSTSDK_Python/blue_st_sdk.html#module-blue_st_sdk.manager).

STMicroelectronics. "STMicroelectronics/bluestsdk\_python: Bluetooth Low Energy Sensors Technology Software Development Kit (Python Version for Linux Gateways)." GitHub. Accessed December 10, 2021.

[https://github.com/STMicroelectronics/BlueSTSDK\\_Python](https://github.com/STMicroelectronics/BlueSTSDK_Python).

Twilio. "No Module Named Twilio · Issue #412 · Twilio/Twilio-Python." GitHub.

Accessed December 10, 2021. <https://github.com/twilio/twilio-python/issues/412>.

Wu, Falin, Hengyang Zhao, Yan Zhao, and Haibo Zhong. "Development of a Wearable-Sensor-Based Fall Detection System." International journal of

telemedicine and applications. Hindawi Publishing Corporation, 2015.  
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4346101/>.