SMART STRIDE

Pennsylvania Governor's STEM Competition

Tanush Shrivastava

Index

Ms. Vetter: a Dedicated South Fayette Citizen

Ms. Vetter is part of the administrative personnel at the South Fayette School District campus. She has a condition, Multiple Sclerosis (MS), which inhibits her flexible movements and periodically restrains her from walking. This is due to foot drop, or dropped foot, which is when her foot becomes completely numb and does not lift during her stride, causing her to fall. On one instance of her foot drop occurring, she fell on her wrist, fracturing it. Furthermore, she was once an avid hiker and traveler, a passion that she can no longer embrace due to the waking restrictions from MS and foot drop. After hearing Ms. Vetter's stories about the injuries and restrictions of her condition, we were compelled to assist her and create a device to prevent foot drops from occurring.

What is Multiple Sclerosis?

Multiple Sclerosis (MS) is a condition that arises when nerve damage impairs the brain-body connection. Vision loss, discomfort, exhaustion, and poor coordination are just a few of the numerous symptoms of this disease. Each person will experience the symptoms, intensity, and duration differently, so while some people may live symptom-free for most of their lives, others may experience severe symptoms that never disappear. Furthermore, when speaking with Ms. Vetter, we learned about a condition some individuals may experience: sidedness. This is when one side of the body has especially weakened nerves, making the limbs on that particular side more prone to the symptoms of MS. With the potential to be fatal, Multiple Sclerosis creates a variety of challenges for those who have it, making even the smallest of tasks a challenge.

How Multiple Sclerosis has Impacted Ms. Vetter

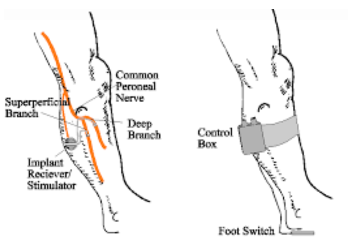
When speaking to Ms. Vetter about the impact of MS on her daily life, she shared a variety of stories, from her home life to her occupation. At school, she works in the Student Services building, often completing tasks in spreadsheets on her laptop. However, due to constant numbness in her hands from Multiple Sclerosis, she struggles to type, often hitting an incorrect key close to the one she intended. Also stemming from this condition, Ms. Vetter struggles with a variety of fine motor skills, from dicing vegetables to wearing earrings. Along with having issues with her hands, she also told us about her struggles with uncomfortable weather. Especially

in high-humidity or high-heat scenarios, she is sensitive to the environment around her, overheating and experiencing fatigue more often than individuals without MS. This especially impacted her passion for traveling, as she could not enjoy the locations she went to because of constant exhaustion.

When we asked Ms. Vetter what her greatest struggle was, she responded by telling us about foot drop and the variety of hindrances it has caused her. From feeling embarrassed in public to having another condition that restricted her in traveling, foot drop has taken away some of the peace of mind she once had. Furthermore, she explains that once foot drop happens, it often has a “snowball effect,” occurring repeatedly for a period of time after the first instance and increasing her risk of falling and potential injury. Inducing a variety of symptoms, Multiple Sclerosis greatly affects Ms. Vetter, both physically and emotionally.

Creation of Solution

From our feedback and further interaction with Ms. Vetter, our team understood that resolving her foot drop should be our primary concern as it impacts her daily life the most. From there, we started brainstorming ideas to solve this issue, looking at solutions that were already on the market and evaluating their benefits and drawbacks with Ms. Vetter.

Device Name	Pros	Cons
Electrical Stimulus 	-Activates nerve to ease pain and stimulate movement -Minimal physical presence	-Has to be activated by human -Not automatic
STIMuSTEP  	-Ensures walking capability through sending an electrical impulse directly to the peroneal nerve (Source 6).	-Wires needed -Surgery needed -Bulky -Shock felt when unnecessary
SaeboSTEP 	-Lifts top of the foot to correct walking by pulling with elastic threads (Source 4).	-Bulky brace -Constant resistant felt -Does not work efficiently with no-lace shoes

From our research, the most prominent solutions either require a bulky attachment, external activation, or constant stimulation (Source 2). If external activation is needed, Ms. Vetter would not be able to react in time to trigger the device. On the other hand, if the device was always on, she would have to experience electric impulses even when her foot drop is not present, unnecessarily bothering her (Source 3).

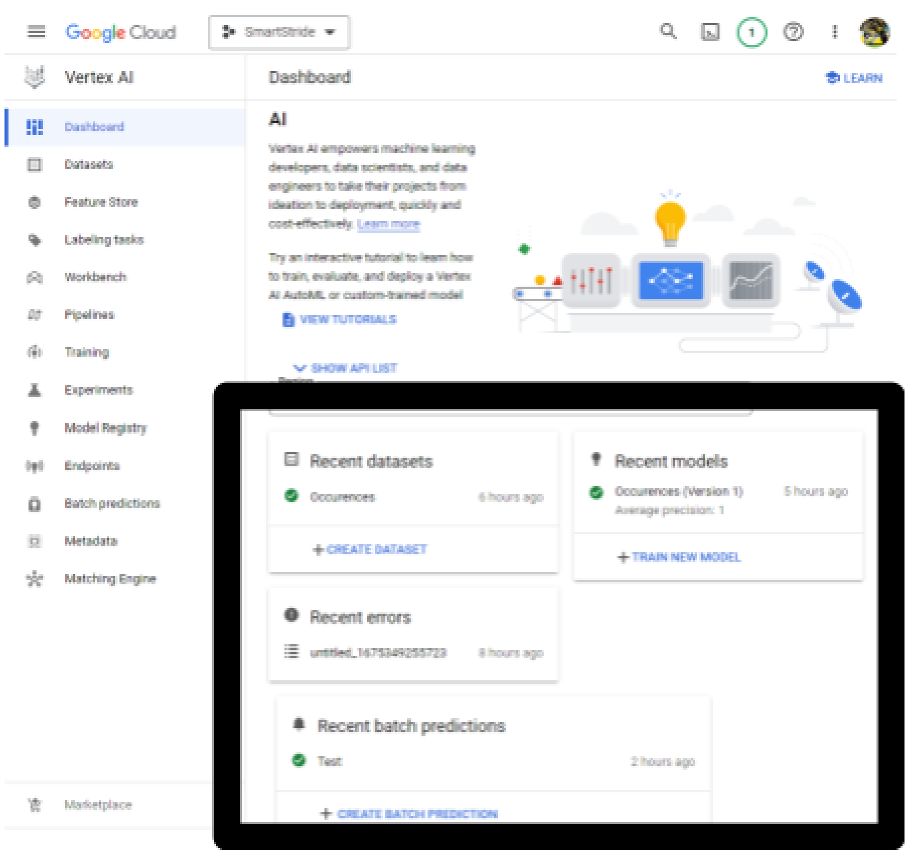
Our Solution

To solve this issue, we decided to automate the activation process using artificial intelligence to predict when foot drop may occur and activate the device only when required. The device we created would communicate to electrical stimulus patches and as a result of the activation, would move the foot to the proper position. The communication would occur only when assistance is necessary, which is determined by an AI system utilizing Google Cloud to analyze data factors to accurately predict when someone like Ms. Vetter would need assistance. The accuracy of the AI system would be ensured with an "R" value, comparing calculated results to this value to determine whether or not foot drop is likely at that time.

This data used for analysis is gathered from various sensors such as a humidity sensor, temperature sensor, heart-rate sensor, and pedometer. Based on the trained AI, the program is able to determine when the foot drop may occur and activate the electrical pads. The benefit of this solution is that the user will feel a slight shock to activate their nerves but only at times of risk rather than at all times of movement.

Explanation of Google Cloud System

By collaborating with Ms. Vetter, we were able to analyze initial data of factors she experienced when foot drop occurred. For example, on one instance of her foot drop, the temperature was 71 degrees, her heart rate was 73 bpm, and the humidity was 40%. We were able to feed many of these datasets to Google Cloud and trained a program to be able to predict if foot drop will occur based on inputted data values when compared to the stored dataset. The process of data, training, and output was used for AI predictions. If a value of one is returned, foot drop will occur and if 0, it will not occur.



Training Data:

	A	B	C	D	E	F
1	Humidity	Stress	WalkTime	Temperature	RValue	Occurence
2	26	0	49	15	22.1	0
3	47	0	48	23	30.6	1
4	42	1	35	42	32.4	1
5	45	0	58	38	36.5	1
6	54	0	2	50	31.6	1
7	26	1	27	5	14.9	0
8	37	1	25	52	31.9	1
9	24	0	38	19	20.5	0
10	58	1	13	45	33.7	1
11	44	0	44	54	38.2	1
12	56	0	2	55	33.7	1
13	54	0	11	44	31.6	1
14	51	0	20	28	27.7	1
15	39	1	58	40	35.5	1
16	43	0	0	52	28.5	1
17	31	1	24	21	20.6	0
18	30	0	23	20	19.6	0
19	58	1	36	31	34.1	1
20	43	0	35	54	36.1	1
21	47	1	12	46	30.5	1
22	20	1	60	6	20	0

Example Input:

Humidity	Stress	WalkTime	Temperat	RValue	Occurrence
42	1	41	45	34.5	
33	0	39	30	26.7	
58	1	54	39	40.1	
28	1	19	8	14.8	
53	1	14	52	34.5	
51	0	36	41	34.8	
32	0	17	54	29.2	
42	0	16	50	30.8	

Example Output For One Row:

```
Humidity,Stress,WalkTime,Temperature,RValue,Occurrence,Occurence_1_scores,Occurence_0_scores
28,1,19,8,14.8,,9.573798820383672e-08,0.9999998807907104
```

In this case, the last output value (0.99999) is very close to one, which means foot drop would occur.

Explanation of R-Value Calculation

The program used to check the AI system's prediction is a predictive algorithm classified as a "Least Absolute Shrinkage and Selection Operator" (LASSO). This program would parse through the data and determine the factors that have the greatest impact on whether or not someone would experience

```r
  ## get training set
  tempTr <- train[, (names(train) %in% variables)]
  tempTrain <- cbind.data.frame(train$Y, tempTr)
  #tempTrain<-train
  ## get testing set
  tempTe <- test[, (names(test) %in% variables)]
  tempTest <- cbind.data.frame(test$Y, tempTe)
  #tempTest<-test

  colnames(tempTrain)[1] <- "Y"
  colnames(tempTest)[1] <- "Y"

  Y <- append(Y, tempTest$Y)
  #Y<-tempTest$Y

  # svm
  svmfit = svm(Y~ ., data = tempTrain , kernel="linear", cost=10, scale=FALSE)
  yhat.SVM = predict(svmfit, newdata = tempTest[, -1])
  append.SVM <- append(append.SVM, yhat.SVM)

  # RF
  RFfit <- randomForest(Y~ ., data = tempTrain, importance=TRUE, ntree = 500)
  yhat.RF = predict(RFfit, newdata = tempTest[, -1])
  append.RF <- append(append.RF, yhat.RF)
}

# tempdF <- cbind.data.frame(Y, append.SVM)
ACC[replicateCV] <- length(which(Y == append.SVM))/nrow(data)
aucRF[replicateCV] <- auc(Y, append.RF)
aucSVM[replicateCV] <- auc(Y, append.SVM)

accRF[replicateCV] <- coords(roc(Y, append.RF), x = "best", input = "threshold",
                             as.list=FALSE, drop=TRUE, best.method=c("closest.to
                             transpose = FALSE, as.matrix=T)
```

foot drop at that particular time. For instance, some individuals' conditions may rely more on the temperature they are exposed to, while others may be more sensitive to the amount of time they have spent walking. This analysis would be conducted on a sample set of data from the individual, tracking times when they do and do not experience foot drop a. After this portion of the program, the parsed data would be fed into a random forest prediction algorithm, using machine learning to compute a prediction on how the factors contribute to foot drop, similar to how variables act in an equation to produce a final result. After running this portion of the algorithm, an accurate method of making predictions can be applied to data collected from sensors in real time. Based on correct and incorrect predictions from updated data, the data can be fed through the program again, producing an updated algorithm that factors in recent data more than past occurrences. Using this program, the device can predict when to activate the electrical stimulus, preventing instances of foot drop.

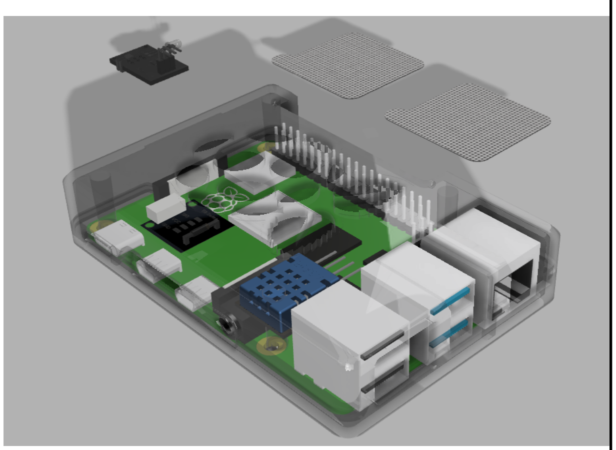
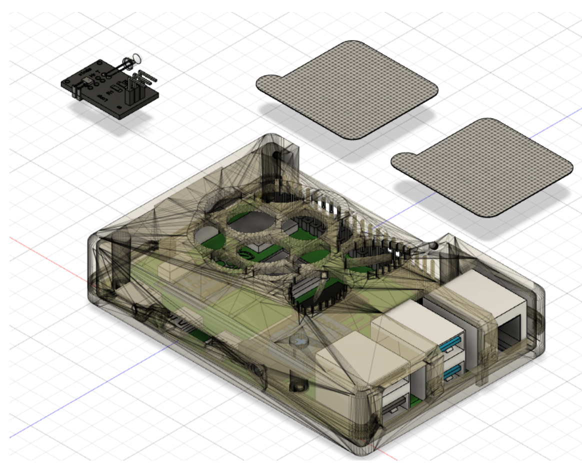
Python Code Controlling Operation

```python
#import statements
import RPi.GPIO as GPIO
import time
import board
import Adafruit_ADS1x15
import Adafruit_DHT
import sys
import adafruit_dht

#initialize sensor values
heartBeat = 0
temp = 0
humid = 0
steps = 0
tilt = 0
move = False

#GPIO Setup
GPIO.setmode(GPIO.BOARD)

#heart
GPIO.setup(3, GPIO.IN)
GPIO.setup(5, GPIO.IN)

#step
GPIO.setup(32,GPIO.IN)

#humid
GPIO.setup(18, GPIO.IN)

#electic pads
GPIO.setup(8, GPIO.OUT)
GPIO.setup(10, GPIO.OUT)


def callAzure():
  pass
def rCalculation():
  pass

def activate():
    GPIO.output(8, GPIO.HIGH)
    GPIO.output(10, GPIO.HIGH)

#get heart rate
def setHeart():
  if __name__ == '__main__':


    adc = Adafruit_ADS1x15.ADS
    # initialization
    GAIN = 2/3
    curState = 0
    thresh = 525
    P = 512
    T = 512
    stateChanged = 0
    sampleCounter = 0
    lastBeatTime = 0
    firstBeat = True
    secondBeat = False
    Pulse = False
    IBI = 600
    rate = [0]*10
    amp = 100

    lastTime = int(time.time()*1000)

    # Main loop.
    while True:
        # read from the ADC
        Signal = adc.read_adc(0, gain=GAIN)
        curTime = int(time.time()*1000)

        sampleCounter += curTime - lastTime;
        lastTime = curTime
        N = sampleCounter - lastBeatTime;
        if Signal < thresh and N > (IBI/5.0)*3.0 :
            if Signal < T :
              T = Signal;

        if Signal > thresh and  Signal > P:
            P = Signal;

        if N > 250 :
            if  (Signal > thresh) and  (Pulse == False)
              Pulse = True;
              IBI = sampleCounter - lastBeatTime;
              lastBeatTime = sampleCounter;

              if secondBeat :
                secondBeat = False;
                for i in range(0,10):
                  rate[i] = IBI;

              if firstBeat :
                firstBeat = False;                        # 
                secondBeat = True;                        # 
                continue

              runningTotal = 0;

              for i in range(0,9):
                rate[i] = rate[i+1];
                runningTotal += rate[i];

              rate[9] = IBI;
              runningTotal += rate[9];
              runningTotal /= 10;
              BPM = 60000/runningTotal;
              heart = BPM

          if Signal < thresh and Pulse == T
              Pulse = False;
              amp = P - T;
              thresh = amp/2 + T;
              P = thresh;
              T = thresh;

          if N > 2500 :
              thresh = 512;
              P = 512;
              T = 512;
              lastBeatTime = sampleCounter;
              firstBeat = True;
              secondBeat = False;

          time.sleep(0.005)
def setTemp():
  dhtDevice = adafruit_dht.DHT22(board.D1
  while True:
    try:
        # Print the values to the serial 
        temp = temperature_c * (9 / 5) + 
        humid = dhtDevice.humidity

    except RuntimeError as error:
        # Errors happen fairly often, DHT
        print(error.args[0])
        time.sleep(2.0)
        continue
    except Exception as error:
        dhtDevice.exit()
        raise error

    time.sleep(2.0)

#change step if sensor moved
def setStep():
  GPIO.setmode(GPIO.BCM)
  if GPIO.input(18) != move:
    steps+=1
    move = not move

#function to check if activation needed o
while True:
```

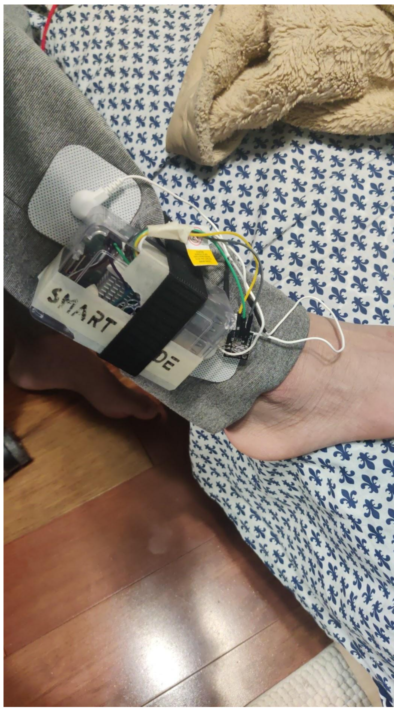
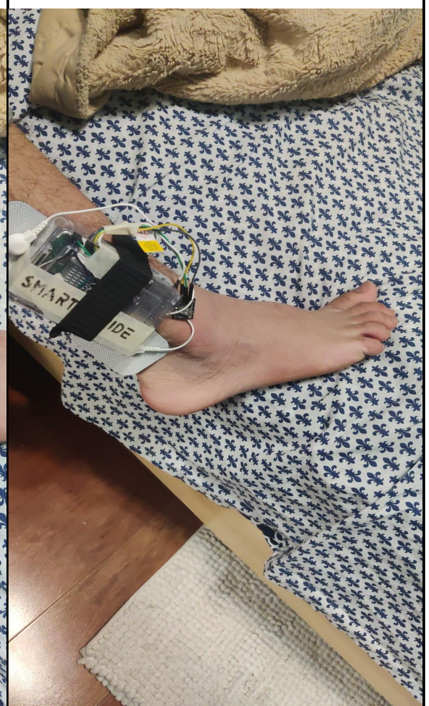
Device Explanation

The Google Cloud platform and R-Value Calculation would be accessed from a Raspberry Pi, a full-sized computer compressed into one board. The Raspberry Pi would be attached to the various sensors such as heart rate, humidity, and temperature sensor via GPIO (general purpose input output) pins. These would be placed inside a comfortable, small 3D-printed case. The electrical stimulus pads would also be wired to the GPIO pins but in an orientation such that the pads contact the skin. The combination of this hardware would be minimal, yet effective.

CAD Drawings

Render Isometric	Raw Isometric
	

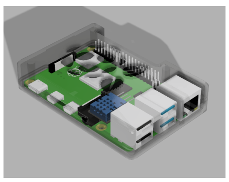
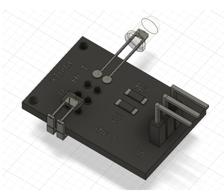
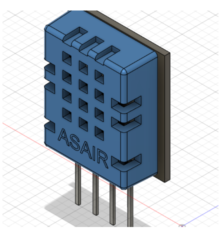
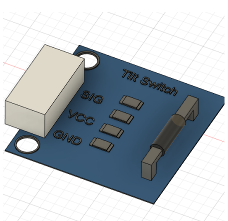
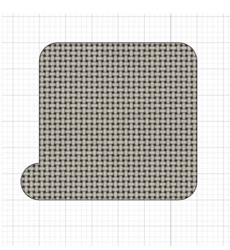
Functionality Images

Over Pant Sleeve	No Pants	Inside Pant Sleeve
		

Explanation for Electrical Stimulus and Corporate Partner

We were advised that electrical stimuli could be used to help solve this by our corporate partner in Mount Lebanon, Pennsylvania: Orthopedic & Sports Physical Therapy Associates (OPTSA) Inc., a physical therapy center that has experience using electrical stimuli. We worked with the center to understand the functionality of how electrical stimuli works, learning how we can make them effective with proper safety. From our discussion with them, we learned that electrical stimulus works in the form of wave patterns that resemble those of neural signals. Once the nerves identify the signal, they activate the muscles they are connected to (Source 5). From this finding, we understood that our activation of the electrical stimulus would have to provide a varied amount similar to a wave to activate the peroneal nerve correctly. Furthermore, we wanted to ensure our device was safe for use and did not cause harm to its users. Our partner told us that because our device was just like e-stim and the only variation is the time of activation, it would have all the benefits that e-stim has, which is supported as no psychometric differentiations have been found through this source (Source 1). This is an advantage for our product, as a major downside of e-stim is that it is not a permanent solution, which can be solved through multiple activation instances.

Materials/Budget

Image	**Device**	**Purpose**	**Cost**	**Cost (per unit if 100 bought in bulk)**
	Raspberry Pi	Computation	$10.00	$2.60
	Heart Rate Sensor	Track user heart rate data	$1.44	$0.27
	Humidity/ Temperature Sensor	Track temperature and humidity data	$3.50	$0.40
	Tilt Switch	Identity motion/step count	$0.40	$0.15
	Electrical Pads	Provide electric shock	$2.00	$0.15
			$17.34	**$3.57**

Future Plans

In the future, we hope to eliminate the need for an external device and simply use the onboard sensors from the user's smartphone/smartwatch. The sensors on the user's device would communicate with an app to analyze the data, and if danger is detected, it will activate the electrical pads in a similar manner to our current solution. We also plan to make the algorithm more accurate by feeding more data for analysis and comparison, advancing the code. With these two improvements, the solution to foot drop would become more practical and effective.

Works Cited

Andreopoulou, G., Mercer, T. H., & van der Linden, M. L. (2018). Walking measures to evaluate assistive technology for foot drop in multiple sclerosis: A systematic review of psychometric properties. Gait &Amp; Posture, 61, 55–66. https://doi.org/10.1016/j.gaitpost.2017.12.021

Kane, H. (2022, August 26). *Treating Foot Drop With A Brace*. Perform Podiatry. https://www.performpodiatry.co.nz/treating-foot-drop-with-a-brace/

Renfrew, L., Lord, A., Warren, J., & Hunter, R. (2019, July). Evaluating the Effect of Functional Electrical Stimulation Used for Foot Drop on Aspects of Health-Related Quality of Life in People with Multiple Sclerosis. National Library of Medicine. Retrieved December 5, 2022, from https://pubmed.ncbi.nlm.nih.gov/27680426/.

SaeboStep Foot Drop Brace - Neurorehabdirectory.com. (2022, September 6). Neurorehabdirectory.com. https://www.neurorehabdirectory.com/rehab-products/saebostep-foot-drop-brace/

Smith, A., Prokopiusova, T., & Rasova, K. (2020, August 24). Functional electrical stimulation for foot drop in people with multiple sclerosis: The relevance and importance of addressing quality of movement. SAGE Journals. Retrieved December 5, 2022, from https://journals.sagepub.com/doi/abs/10.1177/1352458520923958

Taylor, P., Hart, I., Khan, M., & Slade-Sharman, D. (2016, September). Correction of Footdrop Due to Multiple Sclerosis Using the STIMuSTEP Implanted Dropped Foot Stimulator. National Library of Medicine. Retrieved December 5, 2022, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5087579/