The Rest Egg

An Audio Sensor Alert System

The Rest Egg team

Khalid Abdulla
Zarah Khan
T. Blake Jennings
David Yoon

May 2, 2016

Table of Contents

List of Illustrations

Abstract

The founder of modern nursing, Florence Nightingale, once proposed that "unnecessary noise... is the most cruel absence of care which can be inflicted either on sick or well". [1] Our group project, the "Rest Egg," was designed to tackle the ongoing and rarely addressed issue of excessive noise in the hospital. This device monitors a patient's resting environment for noisy disturbances, and alerts a nurse or staff member if one occurs. The alert is triggered by exceeding a programmed threshold. Poor quality of rest has been shown to worsen health and compromise growth and development. Hospitals do not provide an appropriately restful environment, which is most harmful to the sick, injured, and those with special needs, all of whom typically cannot express their discomfort. The Rest Egg would provide a cost-effective modern approach to this century old issue.

This report will summarize the progress made on the project since November 30, 2015 to the present, April 27, 2016. Listed below is the final progress made in preparation for the demonstration. Final notable updates include: a completed thesis defense and establishing that there is a demand for our project; selecting sound as our main focus and building out the full system.

The Device

The Rest Egg is a Raspberry Pi based project which sends smartphone push notifications when it senses sound levels above a certain threshold. This alerts the user - the caretaker, for example - that the surrounding noise is high and action is required. There is a need for this because a recent study showed that intensive care units (ICUs) exceeded noise recommendations for hospitals set by the World Health Organization, reaching around 50-70 decibels [2]. Patients in these hospitals experience high noise levels which prevent them from sleeping. Having a decibel range exceeding 30-40 decibels will disturb patients and prevent them from resting and the recovering process will take longer. Hospitals without quiet environments find themselves holding patients longer, increasing medical bills [3]. The rest egg solves these issues by alerting the hospital staff of potential noises that can wake up patients.

Introduction

The main system parts included a Raspberry Pi 2, its charging cord, microphone attachment, and a 16GB micro SD memory card. After consulting the team mentor, the ordered microphone attachment was taken apart, in order to extract the microphone. This part was then soldered to a headphone jack wire for easy input access - attaching this to a dongle gave the pi unfiltered noise sensing capability. The prepackaged MicroSD card was replaced with a 64 GB so it can hold different applications that were tested for the project- this size is excessive for testing purposes only, and will not impact the price point for our end users.

Pushover is the mobile application of choice- as opposed to other free apps on the market, a nominal one-time fee of $5 indicates that the creators will keep the app updated and working for the foreseeable future. The monetary model will hopefully prevent this application- which works on almost all devices- from becoming outdated in around a year, which often happens to other free notification apps.

Additional parts of the project include the light sensor and LED. The LED and parts related were received early January. The LED has a ‘glow’ function derived from emulating PWM at the GPIO pins of the Raspberry Pi. The light sensor (Photoresistor) arrived mid-April, as our most recent research then indicated light may be a larger problem than we had initially anticipated. However, this is still a side concept, with sound remaining the most critical aspect of the project.

Research

The research done for this project came established a few conclusions. Zarah has defended an honors thesis regarding the research for the the project. First, she established that hospitals are on average too noisy for proper sleep. Then she found that excessive noise deteriorated sleep quality, and exposed patients to a variety of adverse health effects. For those with sensory sensitivities derived from conditions such as autism spectrum disorder (ASD), sleeping was found to be an even greater challenge. Noises that would affect a healthy adult are much more painful to those with such sensitivities. We looked at some attempted solutions that hospitals have used, and found them to either be ineffectual or limited. Based on these findings, we believe there is a market and a demand for our product. Some of the papers found showed an interest in creating a device that monitors patients for signs of discomfort.

Noisy Hospitals

Ideally, hospitals would adhere to the following noise level conditions: [4]

- WHO recommends 30 dB in general hospital area and 35 dB in patient room
- EPA suggests 45 dB daytime and 35 dB nighttime
- Long term exposure to 70+ dB levels creates risks of hearing loss

Sound Level (dBA)	In the ICU	In the Community
0		Threshold for normal human hearing
20	Sound of breathing at 1 meter	Grand Canyon at night, no birds, no wind
30	According to the WHO: L_{Aeq} should not exceed 30 dBA in general hospital areas and 35 dBA in rooms where patients are treated or observed	Quiet room or whisper Quiet bedroom at night
40	According to the WHO: L_{Amax} indoors should not exceed 40 dBA during the night	Noise of normal living, talking, or radio in the background
50	Endotracheal aspiration unit	Library
60	Ventilator Oximeter alarm Conversation at the nurses' station	Conversational speech at 1 meter Television Noisy lawn mower at 10 meters
70	IV infusion alarm Monitor alarm Ventilator alarm	Vacuum cleaner at 1 meter
80	Supply cart Nebulizer Pager Connection of gas supply Rattling side rails	Heavy traffic at 10 meters Door closure
90	Loud crying Items falling onto floor Portable X-ray machine	Circular saw at 1 meter Motorcycle
100		Jackhammer at 10 meters
110		Siren at 10 meters A club or disco

dbA = A-weighted decibels; WHO = World Health Organization.

Figure I. Noise in the ICU and the Community

Hospitals on average well exceed these noise levels (around 50 - 70 db during the daytime, with 80 dB peaks). Medical devices, as well as conversations, make the environment less than ideal as seen in Fig. 1.

We also looked at the state of hospital lighting. It was found that hospitals were consistently too dim for proper sleep [5]:

- Indoor light ranges from 150 - 250 lux
- Average hospital emits 104.8 lux nearly 24 hours a day
- Constant low light compromises circadian rhythm -- poor sleep quality
- Patients need doses of bright light during day then darkness when resting

During rest periods, flashing alarms can be a problem as well. This research was to go towards understanding the need for a light sensor. While it would be up to the medical professionals to ensure patients were exposed to ample sunlight, our product would ensure that resting time had the appropriate low levels. In addition, if a light was on in a patient's room during rest hours, then either the patient, a visitor, or a staff member has turned it on, and a nurse ought to be notified to investigate.

Effect of Poor Sleep Quality

Evidence was found to support that the average hospital often exceed these noise levels, and it is affecting patient quality of sleep and well being: [4]

- On a psychological level, poor sleep can lead to pain, fatigue, stress, anxiety
- On physiological level, biological functions such as the immune system are compromised

The following figure shows just a few of the adverse health effects that are caused by lack of sleep or deteriorated sleep [6].

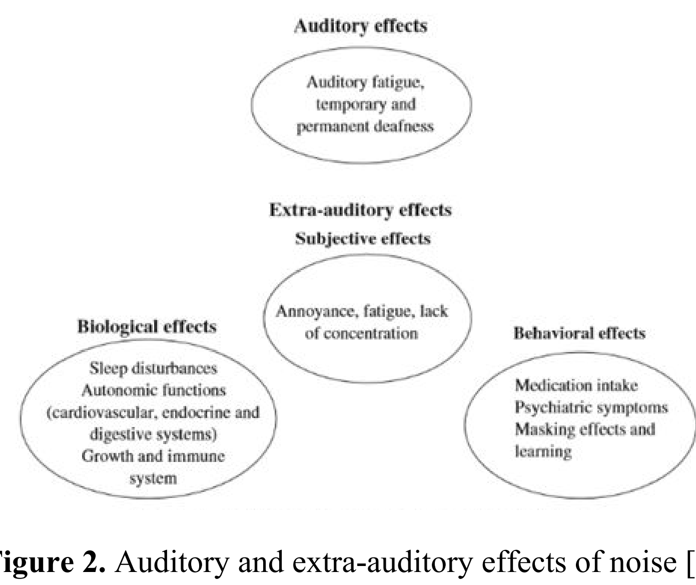

Figure 2. Auditory and extra-auditory effects of noise [6].

Hospitals are where the most vulnerable, the sick and injured, go to rest and recover. If hospitals are placing those patients at risk of conditions such as the conditions mentioned above, then ultimately, modern hospitals are failing to give patients proper care.

Sensory Sensitivity

Many factors can lead to sensory sensitives, be it genetics, diseases, etc. This product would be most beneficial to people who have such sensitivities. It can be difficult to express or alleviate discomfort when one is half-asleep, is vulnerable to stress and anxiety, or has certain disorders. Zarah's defended honors thesis focused on how excessive noise affects those with Autism Spectrum Disorder (ASD), due to this group's notorious sensitivity to stimuli, poor sleep record, and severe vulnerability to stress and anxiety: [7]

- Autistic individuals have a notoriously poor sleep quality, leads to higher stress
- Autism Spectrum Disorder affects sensitivity and pain tolerance
- The concern here is that unnecessary stress could affect growth and development

For those ASD, especially the young, proper growth and development is key to establish higher functions, social skills, and independence. Unnecessary stress can compromise not only growth and development, but also prevent treatments and operations that such people were undergoing in the hospital (such as melatonin therapy) from being fully effective.

Other Approaches

The team found some evidence of hospitals addressing the solution, but these solutions have mixed results far from reaching ideal sound levels. Fundamentally, hospitals would rather implement technology or policy changes to address the issue rather than redesign the hospital layout due to cost effectiveness [4]:

- Hospitals have attempted measures such as quiet time, ear plugs, Yacker Tracker
- No existing solution is perfect, some more effective than others
- Yacker Tracker reduced noise by 2 - 3 dB; only cut down peak noise
- Quiet time reduced sound levels by 10 dB, but noise returned 30 mins later

Noise levels in the hospitals measured never seemed to drop below 50 dB, varying in noise between 50 - 70 dB, which would be like sleeping with the TV on [4]. Devices like the Yacker Tracker were found to only reduce noise by 2 to 3 dB, which is barely detectable to the human ear. However, it reduced peak noise. Fighting noise with noise has been shown to be a problem however. Quiet time was most effective, reducing noise by 10 dB (roughly 60 dB to 50 dB); however, this is louder than WHO or EPA recommended levels. And noise levels returned 30 mins later [4]. What we argue is that in conjunction with quiet hours, our device would ensure that quiet hours were better enforced. By force of accountability, this system would let staff members be fully aware of who specifically is disturbed by an aggravating noise, and thus it would be a nurse's responsibility to correct the issue.

Another paper called for a hypothetical patient-monitor device, listing the benefits of efficiency and better communication between patients and doctors during their stay. Some hospitals already use devices like the Yacker Tracker. One hospital has gone so far as to use literal video monitoring [8]. While effective, this could lead to potential privacy issues.

Budget

Below is a list of materials used in the project and their appropriate costs:

- Raspberry Pi 2 CanaKit Starter kit ($69.99)
- Microphone: Atomic Market ($6.99)
- 3D printed Egg Case (less than $5 for materials)
    - Alternative Cheaper solution for the egg shell is to buy an egg shaped plastic/foam and modify it: ($10 on Amazon)
- Expected Total: +- $100

The total build cost is less than $90, cheaper than the competition's Yacker Tracker. After announcing the Raspberry Pi Zero - late November of 2015 - which is more than 80% cheaper than the Raspberry Pi 2, here is a list of materials to be used with the RP Zero with their costs:

- Raspberry Pi Zero: ($5)
- Power Supply: ($4.99)
- 4 Port USB Hub with power adapter: ($10.95)
- USB WiFi: ($7.98)
- Microphone: Atomic Market ($6.99)
- Pushover subscription ($5, optional)
- Total: $40.41

Based on these values, the concept design of the Rest Egg is more than 70% cheaper than the Yacker Tracker. This will ease the burden on potential users, as our end goal has shifted towards open-sourcing the project for anyone to build and modify. Due to the project's affordability and open source content, future ASU engineering students are encouraged to build upon the project in future capstone courses, perhaps adding new sensors or features. However, our main goals were achieved: affordability, utility, and ease of use.

Work Hours

At their peak, work hours constituted 7-8 hours per week per student; during breaks and vacation time, this time dwindled to a 1-2 hour per week average. Average overall was slightly over 6 hours per week per student.

Cumulative Person Hours	760 - 800 hours	Health research - sensor research - microcontroller research - Noise levels - Coding - Course work (Reports and meetings)
Khalid Abdulla	190 - 200 hours	Hardware research/fabrication - budget control - egg shell fabrication - coding - contribution to course work (presentation, reports, mentor meetings, group meetings)
T. Blake Jennings	190 - 200 hours	Application development - software compilation - Egg to App connections - coding - contribution to course work (presentation, reports, mentor meetings, group meetings)
Zarah Khan	190 - 200 hours	Autism research - Hospitals atmosphere research - noise levels and threshold - coding - Algorithm - contribution to course work (presentation, reports, mentor meetings, group meetings)
David Yoon	190 - 200 hours	Software research/resources - Eggshell design - hospital feedback/research - coding - contribution to course work (presentation, reports, mentor meetings, group meetings)

Table I. Person hours described and justified.

Design

The team has narrowed down the components required to those needed for sound detection and analysis. The overall structure of the design is based on three sections: audio recording, audio process, and the web server. The recording is done through the microphone by Audacity. The software provides a graphical representation of the incoming audio and stores it in a usable format for processing such as a wav file. ALSA, a software framework for graphical analysis, contains commands for manipulating the stored audio. The command *arecord* allows the Raspberry Pi to access the Python script and sample the .wav file for easier analysis.

Audio processing is done with the command *sox* which processes the audio and outputs the desired result in a simple format. In the case of the rest egg, this command retrieves the file from *arecord* and determines a mean value for the sound level using the additional command *stat*. Once the threshold has been breached, the Pi will send a signal to the web server which sends a notification through the smartphone application. Users are able to adjust the threshold value from the application. The feedback returns to the Pi which changes the compared threshold value.

To summarize: the sound is recorded and sampled, averaged, then compared to the threshold derived from research: 35 db, which is adjusted to preference by the end user. If it is greater, a signal is sent via Wifi to the application. The code for noise detection is in Appendix A.

Pushover is a multi-platform notification service by Superblock, LLC. It caters to all major smartphone environments, and many additional devices including smart watches. After a seven day trial, a $5 one time licence purchase allows users 7,500 messages per month, which far exceeds the alert needs of the Rest Egg. With a few lines of code, a program within the Raspberry Pi sends a notification to the specified user. In the code in Appendix B, taken from their FAQ, the 'token' is the name of the Raspberry Pi device profile created on the Pushover website, and 'user_key' are given as soon as the user makes an account. Since the messaging code itself is minimal, this can be inserted into other functions on the system without straining the CPU. If a future group adds another functionality or component, this small bit of code can be added as an alert for that function- for example, a push notification message for a light sensor when the lights are turned on during a resting period.

Schedule

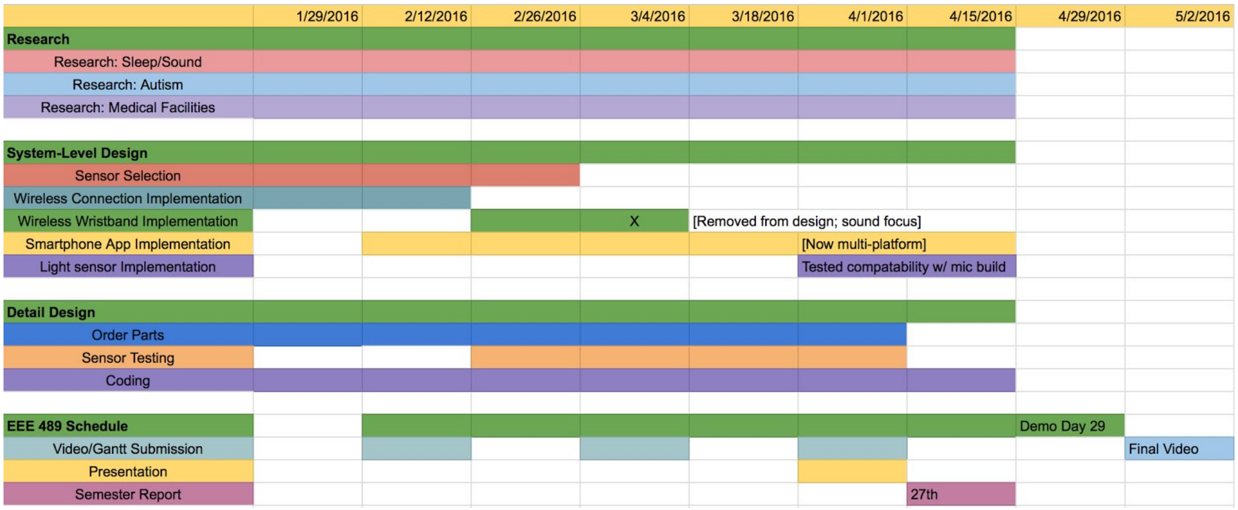

Figure 3. Updated gantt chart for Spring 2016, towards project build and demo.

Date	Task	Member
Sept – Nov	Begin research about autism, epilepsy, anxiety, etc. Study stress and stress factors. Study sleep patterns.	Zarah, Tyler
Sept 18	Turn in Prospectus. Begin meetings with committee members.	Zarah, Tyler
Sept 25	Progress Report 1 Due – EEE 488	Team
Sept 28	Post 1st Video Progress Report	Team
Oct – Nov	List Device Parameters. Interview Medical Professionals	Khalid, David
Oct 5	Draft, Design and Budget	Team
Oct 12	Post 2nd Video Progress Report	Team
Oct 19	Continue Drafting Our Design	Team
Oct 26	Post 3rd Video Progress Report	Team
Nov 6	Review Our Project with Peers	Team
Nov 9	Post 4th Video Progress Report	Team
Nov 11	Finalize Goals; Final Video	Team
Nov 13	Progress Report 2 Due – EEE 488	Team

Table II. Timeline of work completed during EEE 488.

Date	Task	Team Member
Dec 1	Finalized third-party WristBand selection	Khalid
Dec 4	Finalized shopping list. Ordered parts	Team
Dec 7	Individual Semester Summary Reports, Intra-team Evaluations	Team
Dec - Jan	Write thesis paper over the break	Zarah
Jan	Continue research regarding stress and diseases	Zarah
Jan	Publish update videos	Khalid; Team
Feb 17	Received all parts for the Rest Egg	Zarah; Team
Feb 23	Created microphone	Team
Feb 24	Published update video	Khalid
Mar 27	Tested audio capture	Team
Mar 29	Designed/tested code	Team
Mar 30	Publish update video	Zarah
Mar 31	Complete Thesis Draft	Zarah
Apr 1	Publish update video	Khalid; Team
Apr 7	Thesis defense	Zarah
Apr 15	Thesis defense; Submit Honors Thesis	Blake; Zarah
Apr	Publish update videos	Team
Apr 19	Finish main, debugging	Team
Apr 27	Finish 3D egg design and print	David
Apr 28	Finalized Thesis	Blake
Apr 29	Final Demo	Team

Table III. Timeline of work completed during EEE 489.

Limitations

Money could have been saved by buying the Raspberry Pi Zero, but it is a risk that professor Goryll ill-advised, due to the reduced processing power of the Zero. The Raspberry Pi Zero was advertised for $5 in late November of 2015 (80% cheaper than the Raspberry Pi 2). Because it did not have USB ports, it would have been necessary to purchase a powered USB Hub for it. Yet, the Zero had not been tested or reviewed and materials were needed by Christmas time. The team plans to post the project as open source, which will include the cheapest theoretical method to build the Rest Egg.

The “Rest Egg Wristband” proved to be too large of an addition to the project workload. The amount of medical research needed to synthesize a legitimate noise algorithm is daunting; adding heart or skin readings to the workload would not have been sustainable with our budget and timeframe. Coding an additional device and its various inputs, with backup research would have been too difficult for something not directly aligned with the main concept. This leads into the decision to focus on sound irritant, instead of the aftereffects of stress the wristband could record.

Sound is the main issue in hospitals and homes. The eyes can be closed or covered- the ears, however, are harder to protect. Many people find their own internal noises too audible when wearing earplugs or other such devices, and the device may be too uncomfortable for proper rest, or may not be wearable due to medical situations. This focus on sound allows us to zero in on the fastest route to that goal, and will make the project ‘simpler’ for open sourcing. A simple list of steps for a product is a good marketing strategy for open source designs.

For light sensing, the Phantom YoYo Mini Luminance Light Sensor was our initial selection. The implementation would be simple, as we cannot afford to divert CPU usage to other functions- sound is the main focus. The final implementation concept contains a simple ON/OFF signal through the device, only sensing if the lights are on once light levels breach 150 lumens. However, this sensor did not make it to the final demo- additional code and functions could be a threat to the functionality of the noise sensor. Again, this sensor would only be able to see if the lights are on, something caretakers can easily achieve on their own; noise, however, is a more elusive sense that needs empirical monitoring.

Future Plans

The group is ready to demo in front of the public with a live experience of the Rest Egg and an attractive poster. The Rest Egg group is also going open source with this project, creating potential for edits or improvements, perhaps by future 488/489 students. While the project has gone through many conceptual changes, each of those ideas that didn't make it were still useful- they simply didn't match the time constraints or our end goals. To allow for improvements or added 'modules', instructions will be provided on adding additional components to the Rest Egg system on an instructable.com article, including a github.com.

ABET Criteria

An acceptable capstone project must meet most, if not all, of the set ABET considerations, including: health and safety; social; political; ethical; manufacturability; economic; sustainability; environmental. This project meets many of these requirements. Economic - this project is based in the Raspberry Pi environment, an affordable product by definition. The introduction of the Raspberry Pi Zero- a $5 computer that we were unable to acquire- could push the price down even further. The other components are either on-hand for many people or affordable; the final cost for an end user could be lower than $50. Health - the project's goal was to improve the atmosphere in hospitals so the patients can recover from their illness faster. Safety - the Rest Egg can prevent hearing loss in patients unable to eliminate nearby sources of loud noises. Proper rest & healing reduces the chance infection and complications, preventing hospital readmission. Environmental - the device helps to reduce noise pollution in high traffic areas. Ethical - nurse accountability is increased, and proper healing can be ensured for the sick & injured. Social - children with Autism Spectrum Disorder may have improved social skills later in life by having a dedicated safe zone free of noise pollution [7]. Sustainability - faster recovery leads to shorter hospital stays and increases hospital efficiency by reducing the resources needed per patient.

Conclusion

The Rest Egg has the potential to hasten the recovery for hospital patients, and to ensure rest for special needs persons, especially those with special needs who cannot express discomfort. While it's also cheap and easy to use for caretakers at home to insure a better recovery atmosphere. Being a modular, open source design, the platform we've created can be customized to fit any number of needs. With the device recording data over time, doctors could create a rest log to monitor patient sleep; a smart watch or wristband can measure the effect of stressors on the patient; improved light sensors can be implemented. The Rest Egg will help create a safe zone, alerting doctors and parents that their patient or child's rest is genuine. This is potentially a very appealing product for people who are in real physical need of rest. When the Rest Eggs was developing, we have identified all the negative in the technology of our competitors, and from there, we made sure that our product is built using the most reliable components and technologies- while still remaining affordable. Since this product can be used at home, we understand that the consumer will not pay for an overpriced product that will not create a safe zone. So putting this in mind, we chose to built a dependable, relevant product within a budget that both an individual and hospitals can afford.

Final Remarks

Conceptual changes have been large since our project's beginning. However, our main focus - sound - is still at the forefront of our design. Our goal was to identify excessive noise and alert caretakers where they will actually listen- their phones. The other devices in the market use lights or sound as an warning system, but these often go unnoticed or are counter productive. Notifying people in a modern way is what sets our product apart and makes us relevant. As this course reaches its conclusion, we hope that this devices, or its concept, make a positive impact on people's lives.

References

[1] F. Nightingale. (1946). *Notes on Nursing: What It Is, and What It Is Not.* (2nd ed.) [Online]. Available: http://www.worldcat.org/title/notes-on-nursing-what-it-is-and-what-it-is-not/oclc/6922465

[2] B. Berglund, et al. (1999). Guidelines for community noise. *World Health Organization.* [Online]. Available: http://www.who.int/docstore/peh/noise/guidelines2.html

[3] E. Ryherd and C. Zimring. (2010, Oct). Too Noisy to Heal. *Healthcare Design Magazine.* [Online]. Available: http://www.healthcaredesignmagazine.com/article/too-noisy-heal

[4] A. Stafford, et al. (2014, May). Noise in the ICU. *AJN, American Journal of Nursing.* [Online]. 114(5). Available: http://journals.lww.com/ajnonline/Abstract/2014/05000/Noise_in_the_ICU.28.aspx

[5] E. Bernhofer, et al. (2013). Hospital lighting and its association with sleep, mood and pain in medical inpatients. *Journal of Advanced Nursing. 70*(5). Available: http://www.ncbi.nlm.nih.gov/pubmed/24164506

[6] A. Muzet. (2007). Environmental Noise, Sleep and Health. *Sleep Medicine Reviews.* [Online]. 11(2). Available: http://www.ncbi.nlm.nih.gov/pubmed/17317241

[7] J. Adams, et al. (2012). Advice for Parents of Young Autistic Children. *Autism Research Institute.* [Online]. Available: https://www.autism.com/understanding_advice

[8] M. McKinney. (2010). Keeping a Digital Eye on Patients. *Inland Northwest Health Services.* [Online]. Available: https://www.inhs.info/sub.aspx?id=1816

Appendix A

Sound sensor code [Ruby]

```
        # Based on Thomer M. Gil First [http://thomer.com/] version template
#
# This program is free to use. It may be distributed under the terms of
# the GNU General Public License as published by the Free Software
# Foundation, version 3.
# This program is distributed in the hope that it will be useful, but
# without any warranty. See the GNU General Public License for more
# details.
#
# This program detects excessive noise ands send out alerts, via web and push notifications.
#
require 'getoptlong'
require 'optparse'
require 'net/smtp'
require 'logger'
require 'date'
HW_DETECTION_CMD = "cat /proc/asound/cards" # name of your microphone from /proc/asound/cards
SAMPLE_DURATION = 5 # sec
FORMAT = 'S16_LE'   # mic generated format
THRESHOLD = 0.05
RECORD_FILENAME='/tmp/noise.wav'
LOG_FILE='/var/log/noise_detector.log'
PID_FILE='/etc/noised/noised.pid'
logger = Logger.new(LOG_FILE)
logger.level = Logger::DEBUG
logger.info("Noise detector started @ #{DateTime.now.strftime('%d/%m/%Y %H:%M:%S')}")
def self.check_required()
  if !File.exists?('/usr/bin/arecord')
    warn "/usr/bin/arecord not found; install package alsa-utils"
    exit 1
  end
  if !File.exists?('/usr/bin/sox')
    warn "/usr/bin/sox not found; install package sox"
    exit 1
  end
  if !File.exists?('/proc/asound/cards')
    warn "/proc/asound/cards not found"
    exit 1
  end
end
options = {} # Parameter settings
optparse = OptionParser.new do |opts|
  opts.banner = "Usage: noise_detection.rb -m ID [options]"
  opts.on("-m", "--microphone SOUND_CARD_ID", "REQUIRED: Set microphone id") do |m|
    options[:microphone] = m
  end
  opts.on("-s", "--sample SECONDS", "Sample duration") do |s|
    options[:sample] = s
  end
  opts.on("-n", "--threshold NOISE_THRESHOLD", "Set Activation noise Threshold. EX. 0.1") do |n|
```

```
    options[:threshold] = n
  end
  opts.on("-e", "--email DEST_EMAIL", "Alert destination email") do |e|
    options[:email] = e
  end
  opts.on("-v", "--[no-]verbose", "Run verbosely") do |v|
    options[:verbose] = v
  end
  opts.on("-d", "--detect", "Detect your sound cards") do |d|
    options[:detection] = d
  end
  opts.on("-t", "--test SOUND_CARD_ID", "Test soundcard with the given id") do |t|
    options[:test] = t
  end
  opts.on("-k", "--kill", "Terminating background script") do |k|
    options[:kill] = k
  end
end.parse!
if options[:kill]
  logger.info("Terminating script");
  logger.debug("Looking for pid file in #{PID_FILE}")
  begin
    pidfile = File.open(PID_FILE, "r")
    storedpid = pidfile.read
    Process.kill("TERM", Integer(storedpid))
  rescue Exception => e
    logger.error("Cannot read pid file: " + e.message)
    exit 1
  end
  exit 0
end
if options[:detection]
    puts “Detecting soundcard"
    puts `#{HW_DETECTION_CMD}`
    exit 0
end
check_required() #Verify all required booleans
if options[:sample]
    SAMPLE_DURATION = options[:sample]
end
if options[:threshold]
    THRESHOLD = options[:threshold].to_f
end
if options[:test]
    puts "Testing soundcard"
    puts `/usr/bin/arecord -D plughw:#{options[:test]},0 -d #{SAMPLE_DURATION} -f #{FORMAT} 2>/dev/null
| /usr/bin/sox -t .wav - -n stat 2>&1`
    exit 0
end
optparse.parse!
raise OptionParser::MissingArgument if options[:microphone].nil? # If no host found
raise OptionParser::MissingArgument if options[:email].nil?
if options[:verbose]
   logger.debug("Script parameters configurations:")
   logger.debug("SoundCard ID: #{options[:microphone]}")
   logger.debug("Sample Duration: #{SAMPLE_DURATION}")
   logger.debug("Output Format: #{FORMAT}")
```

```
   logger.debug("Noise Threshold: #{THRESHOLD}")
   logger.debug("Record filename (overwritten): #{RECORD_FILENAME}")
   logger.debug("Destination email: #{options[:email]}")
end
pid = fork do # Beginning script
  stop_process = false
  Signal.trap("USR1") do
    logger.debug("Running...")
  end
  Signal.trap("TERM") do
    logger.info("Terminating...")
    File.delete(PID_FILE)
    stop_process = true
  end
  loop do
    if (stop_process)
        logger.info("Noise detector stopped @ #{DateTime.now.strftime('%d/%m/%Y %H:%M:%S')}")
        break
    end
    rec_out = `/usr/bin/arecord -D plughw:#{options[:microphone]},0 -d #{SAMPLE_DURATION} -f #{FORMAT}
-t wav #{RECORD_FILENAME} 2>/dev/null`
    out = `/usr/bin/sox -t .wav #{RECORD_FILENAME} -n stat 2>&1`
    out.match(/Maximum amplitude:\s+(.*)/m)
    amplitude = $1.to_f
    logger.debug("Detected amplitude: #{amplitude}") if options[:verbose]
    if amplitude > THRESHOLD
      logger.info("Sound detected!!!")
        filecontent = File.open(RECORD_FILENAME ,"rb") {|io| io.read}   # Read a file
        encoded = [filecontent].pack("m")    # econding in 64
puts  value = %x[/usr/sbin/sendmail #{options[:email]} << EOF
subject: WARNING: Noise Detected
from: tbjennin@asu.edu
Content-Description: "Noise Wav File"
Content-Type: audio/x-wav; name="noise.wav"
Content-Transfer-Encoding:base64
Content-Disposition: attachment; filename="noise.wav"
#{encoded}
EOF]
    else
      logger.debug("No sound detected...")
    end
end
end

Process.detach(pid)
logger.debug("Started... (#{pid})")
File.open(PID_FILE, "w") { |file| file.write(pid) }
```

Appendix B

Pushover Messaging Code [python]

```
import httplib, urllib
conn = httplib.HTTPSConnection("api.pushover.net:443")
conn.request("POST", "/1/messages.json",
  urllib.urlencode({
    "token": "APP_TOKEN",
    "user": "USER_KEY",
    "message": "hello world",
  }), { "Content-type": "application/x-www-form-urlencoded" })
conn.getresponse()
```