Nala: Smart Companion for Seniors

TEAM: Randy Garcia (Project Manager), Hongyu Shi, Parmanand Shiwmangal, Thomas Wilk, Mohammad Zilon
MENTOR: Professor Xiao

Introduction

- We want to provide support for older adults and family members
- Nala will support its user by providing reminders, life alert, and functions for fun.
- A pleasant appearance, a comfortable size, and a natural way of interacting with our the user are all necessary to help relieve isolation.

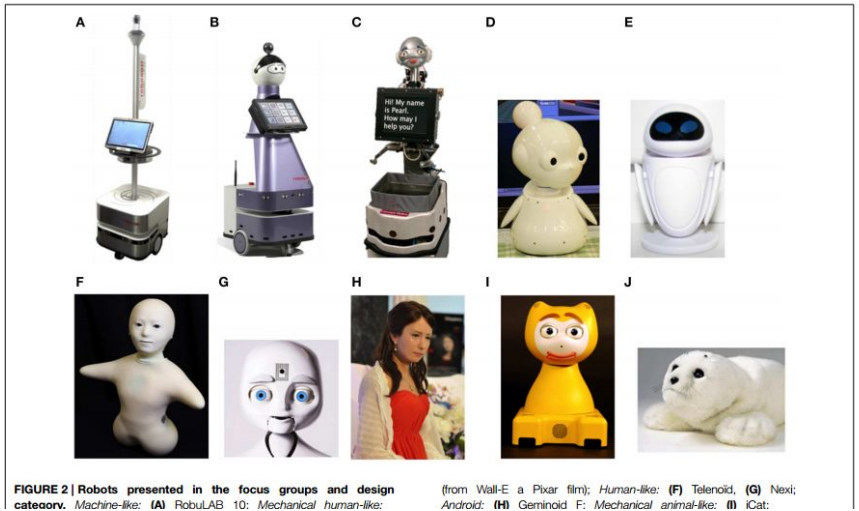
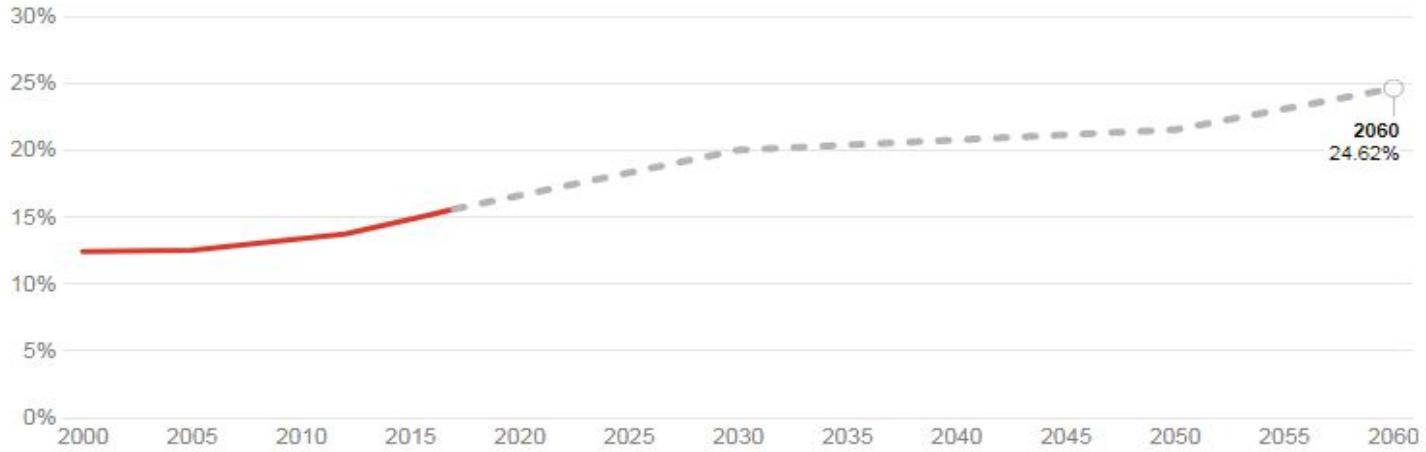


FIGURE 2 | Robots presented in the focus groups and design category. *Machine-like:* **(A)** RobuLAB 10: *Mechanical human-like:* (from Wall-E a Pixar film); *Human-like:* **(F)** Telenoid, **(G)** Nexi; *Android:* **(H)** Geminoid F: *Mechanical animal-like:* **(I)** iCat:

Problem

- Seniors have become a fast-growing share of the U.S. population. According to projections from the Population Reference Bureau, nearly one in four Americans will be over 65 by 2060. (Source: U.S. Census Bureau)
- Most adults are out working, while their elderly parents are often left alone, or are at a senior care center where they might not be given adequate attention.

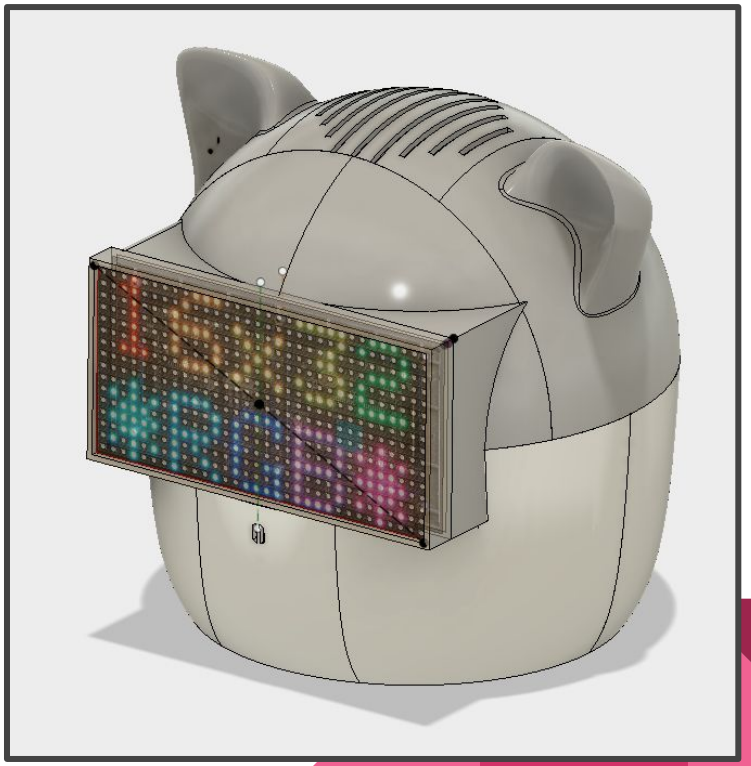


Problem cont.

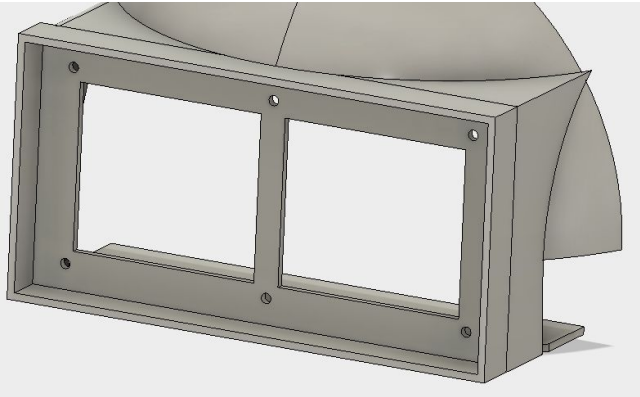
- Study conducted Selwyn Height retirement home in New Zealand.
- 40 seniors participated in the from ages ranging from 55-100 years.

- Where a group of 20 interacted Paro for 12 weeks
- Compared to the control group the group with Paro were noted to feel less lonely

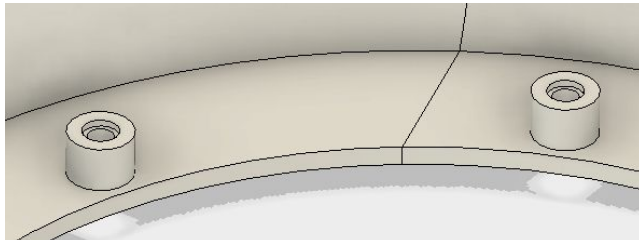
3D Design: pleasant, cute, fun



3D Design: Durable



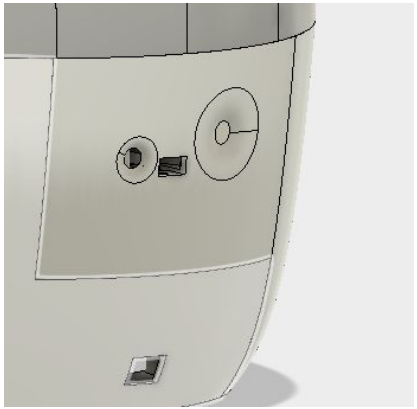
Sturdy LED matrix holder



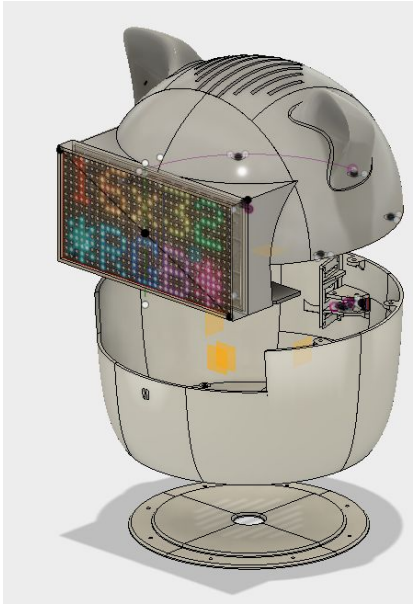
Brass M3 bolt hardware with cylindrical holders

Ribbed bolt holders

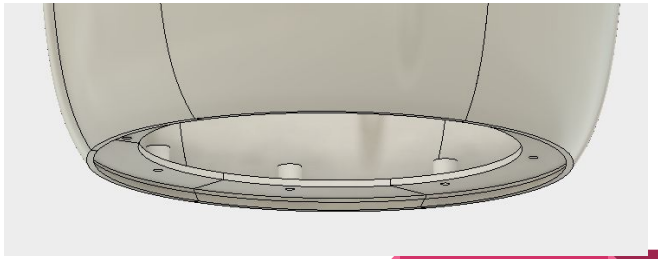
3D Design: ease of use



Accessible knobs and cable outlets



Accessible, replaceable parts



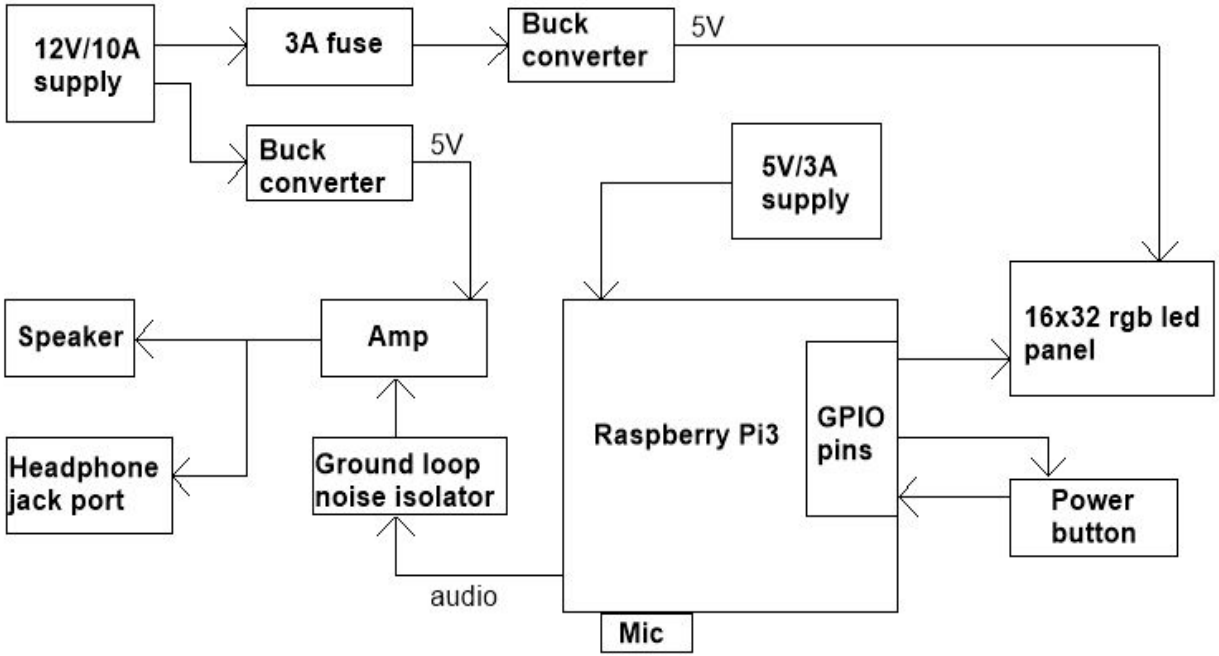
Clear visual response; wide, stable base

Electrical System

Block diagram showing major components and general flow within Nala

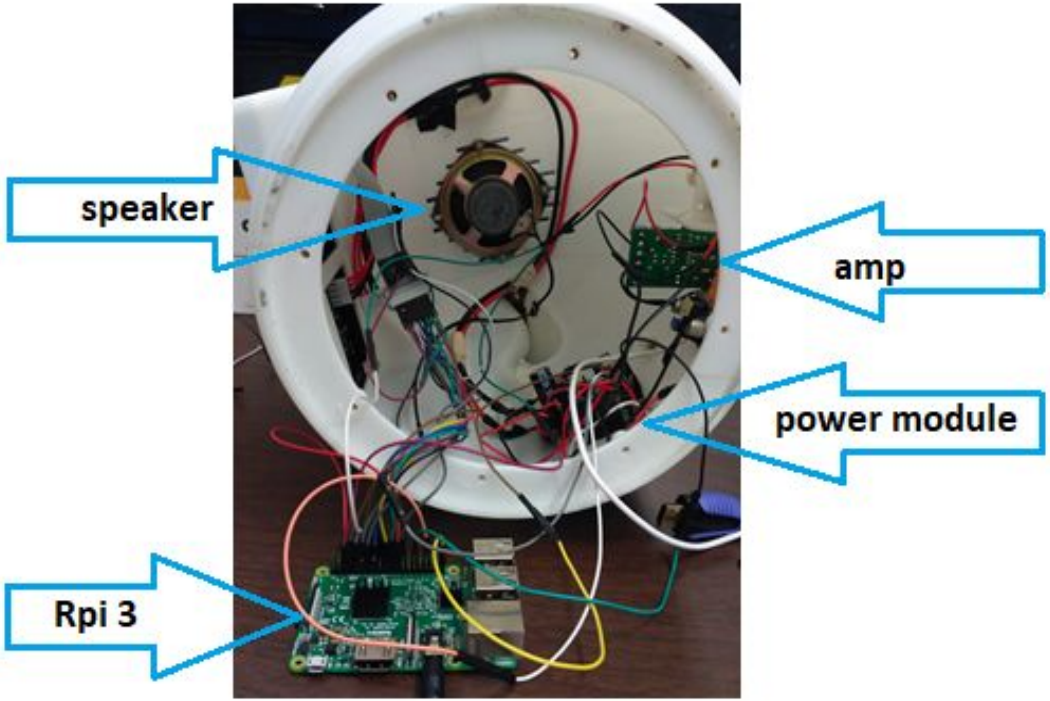
Components:

- Researched
- Purchased
- Tested
- Wired accordingly



Electrical System cont.

Nala’s Interior Components:



Electrical System cont.

Major Problem: Loud hum/buzz from speaker

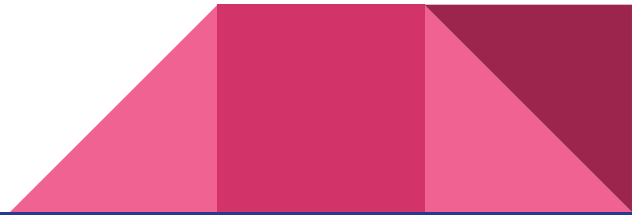
Cause:

- Using audio in conjunction with LED panel.
- Both features require the use of pulse width modulation module in RPi.
- Also power supply noise

Solution:

- Grounding unused GPIO GND pins of RPi.
- Use of ground loop noise isolator on RPi audio output.
- Decoupling capacitor across power supply line (+ve to -ve).

Result: Very low hum/buzz from speaker

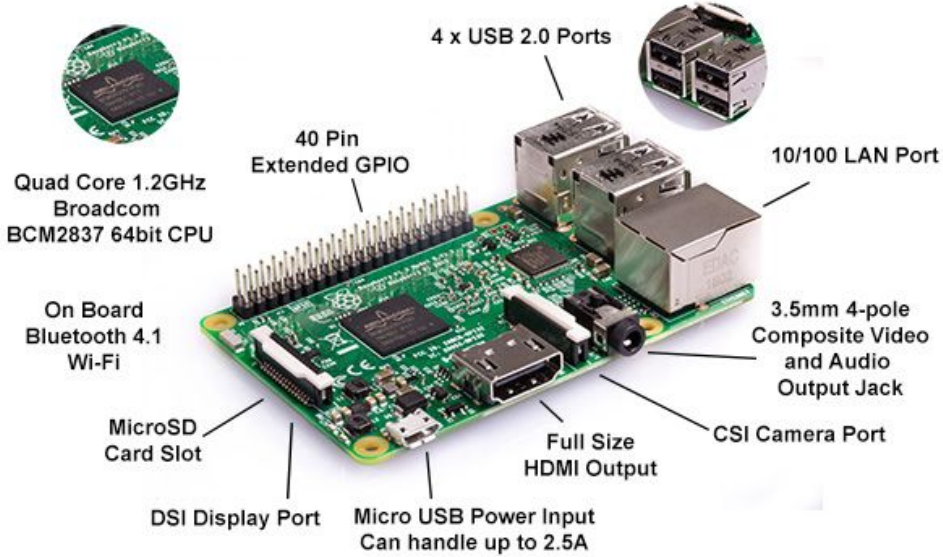


Onboard Computations on the Raspberry Pi

Raspberry Pi is a credit-card sized computer.

The Raspberry Pi handles the onboard computations:

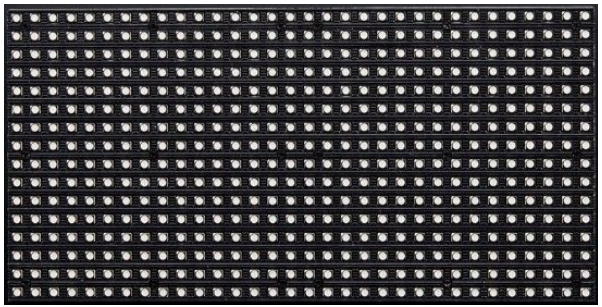
- Alexa Voice Service
- Drives the the speaker.
- Controls the 16x32 RGB LED Matrix.
- Runs a boot script to automate various processes such as activivating Alexa, initiating the speaker, power button and display.



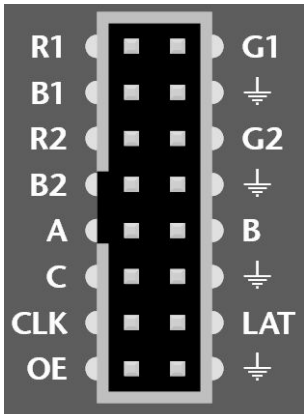
Pi Controls The LED Matrix via its GPIOs

For Visual interaction with Nala, we are using The LED Matrix.

Through the Pi’s GPIOs (general-purpose input/output pins), we have full control of the 16x32 LEDs.



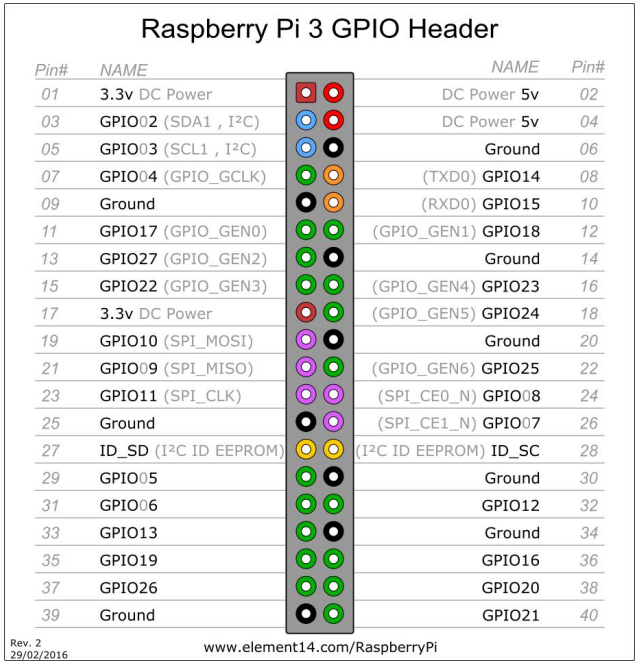
Front of the 16x32 RGB LED Matrix



Display Input Pins to control the LED’s via Pi’s GPIOs

Back of the 16x32 RGB LED Matrix

Pi Controls The LED Matrix via its GPIOs cont.



Raspberry Pi 3 GPIO Header

Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I²C)		DC Power 5v	04
05	GPIO03 (SCL1 , I²C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I²C ID EEPROM)		(I²C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

Rev. 2
29/02/2016

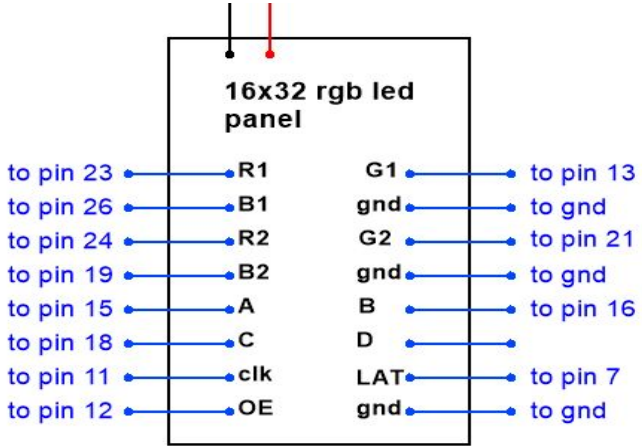
www.element14.com/RaspberryPi

Raspberry Pi 3 GPIO Header

[1] = 😊, [2] = 💥 and [3] = 💧 ; signals that go to all chains have all icons.

Connection	Pin	Pin	Connection
-	1	2	-
💧 [3] G1	3	4	-
💧 [3] B1	5	6	GND 😊 💥 💧
😊 💥 💧 strobe	7	8	[3] R1 💧
-	9	10	E 😊 💥 💧 (for 64 row matrix, 1:32)
😊 💥 💧 clock	11	12	OE- 😊 💥 💧
😊 [1] G1	13	14	-
😊 💥 💧 A	15	16	B 😊 💥 💧
-	17	18	C 😊 💥 💧
😊 [1] B2	19	20	-
😊 [1] G2	21	22	D 😊 💥 💧 (for 32 row matrix, 1:16)
😊 [1] R1	23	24	[1] R2 😊
-	25	26	[1] B1 😊
-	27	28	-
💥 [2] G1	29	30	-
💥 [2] B1	31	32	[2] R1 💥
💥 [2] G2	33	34	-
💥 [2] R2	35	36	[3] G2 💧
💧 [3] R2	37	38	[2] B2 💥
-	39	40	[3] B2 💧

Raspberry Pi's GPIO connection scheme



note: rgb nodes connect to RPI3 pins

Pi Controls The LED Matrix via its GPIOs cont.

Controlling the LED Panel via Python or C/C++ Code:

The Display can be controlled by one simple python code.

C/C++ Code:

To use the C code, this Hzeller's Git [https://github.com/hzeller/rpi-rgb-led-matrix] has to be downloaded and be made on the Pi. The instruction are on the readme.md file.

Running Commands:

After the making the "Demo" file, we can run various demos. For example to run a rotating square script we have to run the following command via the Pi's Terminal app. "sudo ./demo -D0 --led-no-hardware-pulse --led-cols=32 --led-rows=16 --led-chain=1 --led-brightness=50"

There is also an API to create a program to display an image or video. The link to the API is https://github.com/hzeller/rpi-rgb-led-matrix/tree/master/utils.

Python Code:

```
1.  import RPi.GPIO as GPIO
2.  import time
3.
4.  delay = 0.000001
5.
6.  GPIO.setmode(GPIO.BCM)
7.  red1_pin = 23
8.  green1_pin = 13
9.  blue1_pin = 5
10. red2_pin = 24
11. green2_pin = 21
12. blue2_pin = 19
13. clock_pin = 11
14. a_pin = 15
15. b_pin = 16
16. c_pin = 18
17. latch_pin = 7
18. oe_pin = 12
```

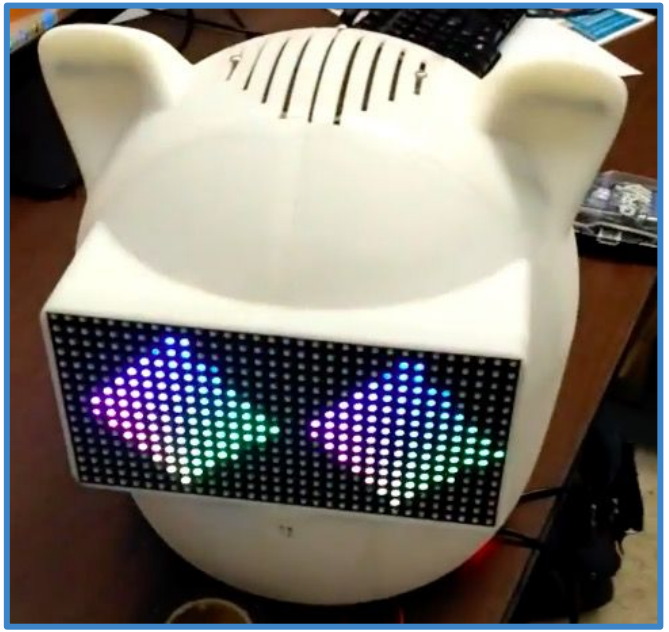
Display’s Capabilities

Basic Functionalities:

- Scrolling and still Text.
- Image
- Video

The LED Matrix can display:

- Time
- Reminders/Notifications
- Monitored health status.
- Alexa’s Feedback text.



Matrix displaying Rotating Squares

LED Matrix displaying a scrolling message

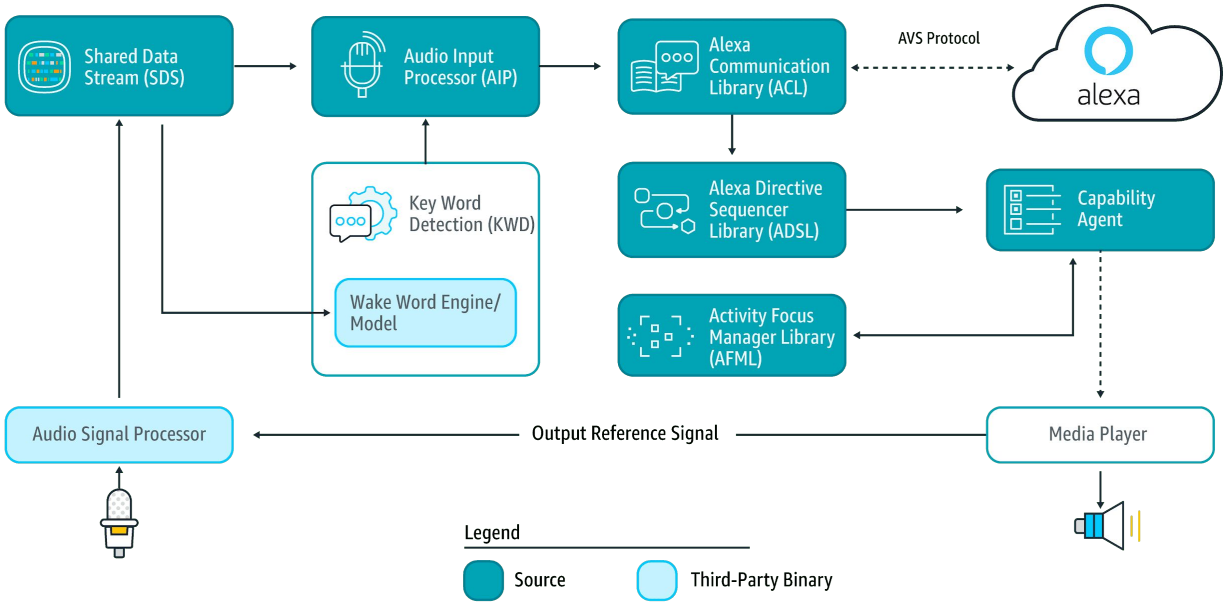
Matrix displaying a still text

Making Nala “Smart”: Alexa Voice Service

Alexa Voice Service (AVS) is Amazon's suite of services built around its voice-controlled AI assistant for the home and other environments. Amazon provides access to its machine learning algorithms, which are freely available as application programming interfaces (APIs). We used these APIs in Nala’s routines to allow Nala to learn to respond in a more natural way. In this way, the more time a user spends with Nala, the more Nala can respond in an engaging way with the user, and provide therapeutic benefits.

Figure #: Block diagram describing Alexa’s use with Nala

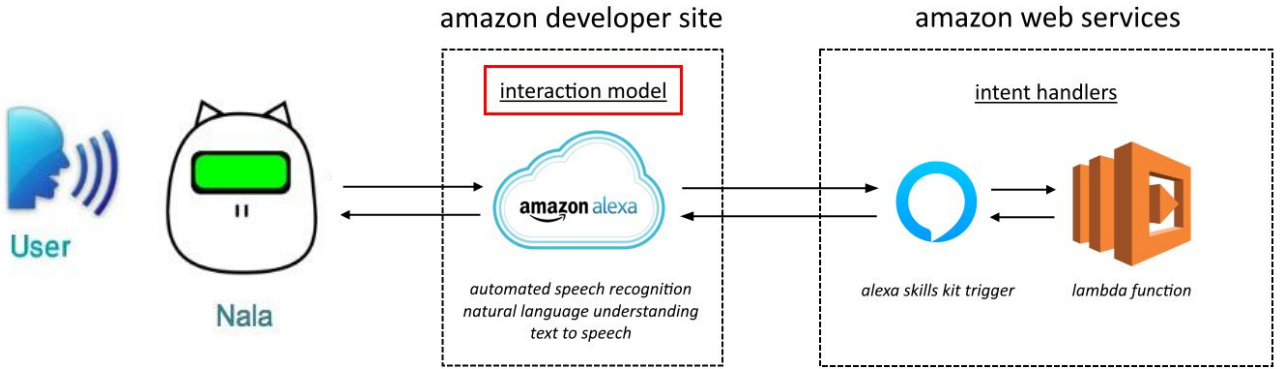
Alexa Voice Service cont.



In depth diagram describing how Nala interacts with the Alexa Cloud

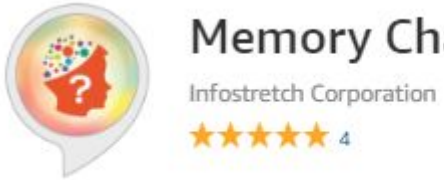
Alexa Web Service (AWS)

- Amazon Web Service is a service that Amazon provides on demand cloud computing platforms for user to use on a paid subscription basis.
- The Alexa Skill Kit is a collection of self-service APIs and tools that makes it easy for you to create new voice-driven capabilities for Alexa.
- The code for the Skills are uploaded to AWS lambda which then executes the code.



Skills

- The interactivity of the Amazon Skills Kit improves the quality of life for its users. Nala can be loaded with different skills depending on the interests of the user, e.g., news, games, and record reminders.



Results

Demonstration Video using Nala

Conclusion

The goals of the project were achieved. We created a cute, user-friendly, therapeutic, and interactive companion robot for the elderly. In a future phase of the project, the following improvements could be made:

- Incorporating a pulse band with Nala to monitor the user's vital signals, and sending necessary alerts in case of emergencies.
- Allowing Nala to control smart home devices such as lights, security systems, air conditioning units, and thermostats.
- Adding motors/drivers to enable physical motion.
- Modifying the current body shape to meet user needs.