



**UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA DE INGENIERÍA EN TELEINFORMÁTICA**

**TRABAJO DE TITULACIÓN
PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN TELEINFORMÁTICA**

**ÁREA
TECNOLOGÍA DE LOS ORDENADORES**

**TEMA
“DESARROLLO DE UN PROTOTIPO DE APLICACIÓN
MÓVIL CON ALGORITMOS DE MACHINE LEARNING Y
NLP PARA APOYO DE PERSONAS CON DISLALIA
EVOLUTIVA”**

**AUTOR
VERA VERA JUAN ANDRÉS**

**DIRECTOR DEL TRABAJO
ING. ACOSTA GUZMÁN IVÁN LEONEL, MSIG**

GUAYAQUIL, SEPTIEMBRE 2022



**ANEXO XI.- FICHA DE REGISTRO DE TRABAJO
DE TITULACIÓN
FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA			
FICHA DE REGISTRO DE TRABAJO DE TITULACIÓN			
TÍTULO Y SUBTÍTULO:		Desarrollo de un prototipo de Aplicación Móvil con algoritmos de Machine Learning y Nlp para apoyo de personas con Dislalia Evolutiva	
AUTOR(ES) (apellidos/nombres):		Vera Vera Juan Andrés	
TUTOR /REVISOR (apellidos/nombres):		Ing. Comp. Acosta Guzmán Iván Leonel, MSIG. / Ing. Telecom. Veintimilla Andrade Miguel Ángel, MG.	
INSTITUCIÓN:		Universidad de Guayaquil	
UNIDAD/FACULTAD:		Facultad Ingeniería Industrial	
MAESTRÍA/ESPECIALIDAD:			
GRADO OBTENIDO:		Ingeniería en Teleinformática	
FECHA DE PUBLICACIÓN:		28 de septiembre 2022	No. DE PÁGINAS: 167
ÁREAS TEMÁTICAS:		Tecnología de los Ordenadores	
PALABRAS CLAVES/ KEYWORDS:		Aplicación móvil, Dislalia, fonoarticulación, red neuronal, OpenGL. Mobile application, Dyslalia, phonoarticulation, neural network	
<p>RESUMEN: El siguiente trabajo aborda sobre el desarrollo del lenguaje comunicativo en infantes, si bien la dislalia es tratada por terapeutas de lenguaje, en muchas ocasiones, los tratamientos son costosos, no se toman las indicaciones por profesionales a falta de uno en los hogares y la falta de importancia sobre los representantes hacia el infante que la padezca. Por estos motivos, se propone realizar como apoyo tecnológico el desarrollar un software aplicativo móvil que contengan ejercicios de práctica de articulación de fonemas. Para que sirva como complemento de práctica de los ejercicios de reforzamiento de lenguaje en casa.</p> <p>Para la metodología de este trabajo se realizó las metodologías bibliográfica, cualitativa y cuantitativa con el fin de poder recolectar la información, y poder llevar a cabo el presente trabajo investigativo. La presente metodología de desarrollo del software es bajo el método prototipado el cual destaca en realizar versiones betas tanto en diseño como en funcionalidad, debido a que existen variedades de los tratamientos sobre la dislalia.</p> <p>Como hallazgo durante la investigación se tiene que los tratamientos de dislalia se realizan mediante ejercicios de intervenciones directas e indirectas para así estimular la fonoarticulación y articulaciones de fonemas. Dentro del campo de desarrollo de software, Flutter es la tecnología más usada para aplicaciones nativas.</p> <p>Como conclusiones las métricas de calidad fueron resultados de acc, 89,29%, precisión, 98,94%, auc, 99,97%, recall, 98,89%. Diseñando un aplicativo móvil con algoritmos NLP, para reforzamiento de lenguaje para dislalia evolutiva.</p>			

ABSTRACT: The following work deals with the development of communicative language in infants, although dyslalia is treated by speech therapists, in many cases, the treatments are expensive, the indications are not taken by professionals due to lack of one at home and the lack of importance of the representatives towards the infant who suffers from it.

For these reasons, it is proposed to develop as a technological support the development of a mobile application software containing phoneme articulation practice exercises. To serve as a complement to the practice of language reinforcement exercises at home.

For the methodology of this work, the bibliographic, qualitative and quantitative methodologies were used in order to collect the information, and to carry out this research work. The present methodology of software development is under the prototyping method which stands out in making beta versions both in design and functionality, because there are varieties of treatments on dyslalia.

As a finding during the research, we have found that dyslalia treatments are performed through direct and indirect intervention exercises to stimulate phonoarticulation and phoneme articulations. Within the field of software development, Flutter is the most used technology for native applications.

As conclusions, the quality metrics were acc, 89.29%, accuracy, 98.94%, auc, 99.97%, recall, 98.89%. Designing a mobile application with NLP algorithms for language reinforcement for developmental dyslalia.

ADJUNTO PDF:	SI <input checked="" type="checkbox"/> X	NO
CONTACTO CON AUTOR/ES:	Teléfono: 593-963551722	E-mail: jandres.verav@ug.edu.ec
CONTACTO CON LA INSTITUCIÓN:	Nombre: Ing. Ramón Maquilón Nicola, Mg.	
	Teléfono: 593- 2658128	
	E-mail: direcciónTi @ug.edu.ec	



**ANEXO XII DECLARACIÓN DE AUTORÍA Y DE
AUTORIZACIÓN DE LICENCIA GRATUITA
INTRANSFERIBLE Y NO EXCLUSIVA PARA EL USO NO
COMERCIAL DE LA OBRA CON FINES NO ACADÉMICOS**



**FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA INGENIERÍA EN TELEINFORMÁTICA**

LICENCIA GRATUITA INTRANSFERIBLE Y NO COMERCIAL DE LA OBRA CON
FINES NO ACADÉMICOS

Yo, **VERA VERA JUAN ANDRÉS**, con C.C. No. **0931423693**, certifico que los contenidos desarrollados en este trabajo de titulación, cuyo título es **“DESARROLLO DE UN PROTOTIPO DE APLICACIÓN MÓVIL CON ALGORITMOS DE MACHINE LEARNING Y NLP PARA APOYO DE PERSONAS CON DISLALIA EVOLUTIVA”** son de mi absoluta propiedad y responsabilidad, en conformidad al Artículo 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN*, autorizo la utilización de una licencia gratuita intransferible, para el uso no comercial de la presente obra a favor de la Universidad de Guayaquil.

VERA VERA JUAN ANDRÉS
C.C.No. 0931423693



**ANEXO VII.- CERTIFICADO PORCENTAJE DE SIMILITUD
FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



Habiendo sido nombrado **ING. COMP. ACOSTA GUZMÁN IVÁN LEONEL, MSIG** tutor del trabajo de titulación certifico que el presente trabajo de titulación ha sido elaborado por **VERA VERA JUAN ANDRÉS C.C. 0931423693**, con mi respectiva supervisión como requerimiento parcial para la obtención del título de Ingeniera en Teleinformática.

Se informa que el trabajo de titulación: **DESARROLLO DE UN PROTOTIPO DE APLICACIÓN MÓVIL CON ALGORITMOS DE MACHINE LEARNING Y NLP PARA APOYO DE PERSONAS CON DISLALIA EVOLUTIVA**, ha sido orientado durante todo el periodo de ejecución en el programa Antiplagio Turnitin quedando el 8% de coincidencia.



https://ev.turnitin.com/app/carta/en_us/?o=1898547573&s=1&lang=en_us&u=1133714199



Firmado electrónicamente por:

**IVAN LEONEL
ACOSTA GUZMAN**

**ING. COMP. ACOSTA GUZMÁN IVÁN LEONEL, MSIG
DOCENTE TUTOR
C.C. 0914940812
FECHA: 12 DE SEPTIEMBRE DEL 2022**



**ANEXO VI. - CERTIFICADO DEL DOCENTE-TUTOR
DEL TRABAJO DE TITULACIÓN
FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



Guayaquil, 12 de septiembre del 2022

Sr (a).

Ing. Annabelle Lizarzaburu Mora, MG.

Director (a) de Carrera Ingeniería en Teleinformática / Telemática

**FACULTAD DE INGENIERÍA INDUSTRIAL DE LA UNIVERSIDAD DE
GUAYAQUIL**

Ciudad. -

De mis consideraciones:

Envío a Ud. el Informe correspondiente a la tutoría realizada al Trabajo de Titulación **DESARROLLO DE UN PROTOTIPO DE APLICACIÓN MÓVIL CON ALGORITMOS DE MACHINE LEARNING Y NLP PARA APOYO DE PERSONAS CON DISLALIA EVOLUTIVA** del estudiante VERA VERA JUAN ANDRÉS, indicando que ha cumplido con todos los parámetros establecidos en la normativa vigente:

- El trabajo es el resultado de una investigación.
- El estudiante demuestra conocimiento profesional integral.
- El trabajo presenta una propuesta en el área de conocimiento.
- El nivel de argumentación es coherente con el campo de conocimiento.

Adicionalmente, se adjunta el certificado de porcentaje de similitud y la valoración del trabajo de titulación con la respectiva calificación.

Dando por concluida esta tutoría de trabajo de titulación, **CERTIFICO**, para los fines pertinentes, que la estudiante está apta para continuar con el proceso de revisión final.

Atentamente,



Firmado electrónicamente por:

**IVAN
LEONEL
ACOSTA
GUZMAN**

**ING. COMP. ACOSTA GUZMÁN IVÁN LEONEL, MSIG
DOCENTE TUTOR
C.C. 0914940812
FECHA: 12 DE SEPTIEMBRE DEL 2022**



**ANEXO VIII.- INFORME DEL DOCENTE
REVISOR FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



Guayaquil, 23 de septiembre del 2022.

Sr (a).

Ing. Annabelle Lizarzaburu Mora, MG.

Director (a) de Carrera Ingeniería en Telemática / Telemática

FACULTAD DE INGENIERÍA INDUSTRIAL DE LA UNIVERSIDAD DE GUAYAQUIL
Ciudad. -

De mis consideraciones:

Envío a Ud. el informe correspondiente a la REVISIÓN FINAL del Trabajo de Titulación **“DESARROLLO DE UN PROTOTIPO DE APLICACIÓN MÓVIL CON ALGORITMOS DE MACHINE LEARNING Y NLP PARA APOYO DE PERSONAS CON DISLALIA EVOLUTIVA”** del estudiante **VERA VERA JUAN ANDRÉS**. Las gestiones realizadas me permiten indicar que el trabajo fue revisado considerando todos los parámetros establecidos en las normativas vigentes, en el cumplimiento de los siguientes aspectos:

Cumplimiento de requisitos de forma:

El título tiene un máximo de 21 palabras.

La memoria escrita se ajusta a la estructura establecida.

El documento se ajusta a las normas de escritura científica seleccionadas por la Facultad. La investigación es pertinente con la línea y sublíneas de investigación de la carrera.

Los soportes teóricos son de máximo 30 años. La propuesta presentada es pertinente.

Cumplimiento con el Reglamento de Régimen Académico:

El trabajo es el resultado de una investigación.

El estudiante demuestra conocimiento profesional integral.

El trabajo presenta una propuesta en el área de conocimiento.

El nivel de argumentación es coherente con el campo de conocimiento.

Adicionalmente, se indica que fue revisado, el certificado de porcentaje de similitud, la valoración del tutor, así como de las páginas preliminares solicitadas, lo cual indica que el trabajo de investigación cumple con los requisitos exigidos.

Una vez concluida esta revisión, considero que el estudiante está apto para continuar el proceso de titulación. Particular que comunicamos a usted para los fines pertinentes.

Atentamente,



Firmado electrónicamente por:
**MIGUEL ANGEL
VEINTIMILLA
ANDRADE**

ING. TELC. VEINTIMILLA ANDRADE MIGUEL ÁNGEL, MG.

C.C: 0922668017

FECHA: 23 DE SEPTIEMBRE DEL 2022

Dedicatoria

El presente trabajo de investigación, se lo dedico a mi madre María Vera debido a que ella ha estado en mi constante apoyo en toda mi trayectoria estudiantil, de primer, segundo y tercer nivel, y por último lo dedico hacia mi persona como una prueba más de superación esfuerzo y constancia.

Agradecimiento

Agradecerle a Dios primero y ante todo debido a que su bendición y gracia fue este trabajo realizado y posible, a mi Madre María Vera por ser la persona que me formo mi mayor ejemplo de vida en superación, que dedico a dedicado sus años de vida para forjarme y prepararme para ser una buna persona, y en un futuro ser un buen profesional con sus valores, frutos de vida y sus principios.

Agradecerle a Luis Eduardo Vera (El Gordo Lucho, Fideo) que me ha compartido consejos y abrazos fraternales para ser más astuto y tener mejor empatía ante la sociedad.

Agradecerle a mi abuelo, Juan Vera Rojas por contribuir con mi madre y forjarme para ser una buena persona con experiencia y ser fuerte para los retos que siguen en adelante, y por siempre mantenerme con consejos de religión y así no alejarme de Dios.

Agradecerle al Ingeniero Iván Acosta por darme la oportunidad y confiar en mí de trabajar en esta investigación además de guiarme por el presente trabajo y desarrollo de esta tesis y por el apoyo que me dio durante la carrera universitaria con sus conocimientos compartidos y experiencias de vida como docente.

Declaración de Autoría

“La responsabilidad del contenido de este Trabajo de Titulación, me corresponde exclusivamente; y el patrimonio de este a la Facultad de Ingeniería Industrial de la Universidad de Guayaquil”

Índice General

N°	Descripción	Pág.
-	Introducción	1

Capítulo I

El Problema

N°	Descripción	Pág.
1.	El Problema	1
1.1.	Planteamiento del problema	1
1.2.	Causas y Consecuencias del Problema	2
1.3.	Formulación del Problema	2
1.4.	Sistematización del Problema	2
1.5.	Delimitación del Problema	3
1.6.	Objetivos	3
1.6.1.	Objetivo General	3
1.6.2.	Objetivos Específicos	3
1.7.	Premisa de la Investigación	4
1.8.	Operacionalización	4
1.8.1.	Variable Independiente	4
1.8.2.	Variable Dependiente	4
1.9.	Alcance del Problema	4
1.10.	Justificación e Importancia	5

Capítulo II

Marco teórico

N°	Descripción	Pág.
2.	Marco teórico	7
2.1.	Antecedentes de la Investigación	7
2.2.	Fundamentación Teórica	8
2.2.1.	Etiología de la dislalia	8
2.2.1.1.	Dislalia	8
2.2.1.2.	Tipos de Dislalia	8
2.2.2.	En función de numero de errores y su persistencia	9
2.2.2.1.	Dislalia Simple	9
2.2.2.2.	Dislalia múltiple	9

2.2.2.3	Retraso del habla	9
2.2.3.	En función del nivel lingüístico implicado	10
2.2.3.1	Dislalias Fonológicas	10
2.2.4.	Clasificaciones de Tipos de Dislalia por su Etiología	10
2.2.4.1.	Dislalia Evolutiva o Fisiológica	10
2.2.4.2.	Dislalia Funcional	11
2.2.4.3.	Dislalia Orgánica	11
2.2.4.4.	Dislalia Auditiva	11
2.2.5.	Clasificación de fonemas por su articulación	11
2.2.6.	Tratamientos	12
2.2.6.1	Proceso de tratamientos	12
2.2.6.2	Intervenciones	13
2.2.7.	Lenguaje de Programación	14
2.2.7.1.	Python	15
2.2.7.2.	Java	15
2.2.7.3.	JavaScript	16
2.2.7.4.	C#	16
2.2.7.5.	C/C++	17
2.2.8.	Front – End	18
2.2.9.	Frameworks	19
2.2.9.1.	Kivy	19
2.2.9.2.	Arquitectura Kivy	20
2.2.9.3.	Lenguaje Kivy	21
2.2.9.4.	Flutter	23
2.2.9.5.	Arquitectura Flutter	23
2.2.9.6.	Android Studio	25
2.2.9.7.	Arquitectura Android Studio	25
2.2.9.8.	Entorno de desarrollo IDE	28
2.2.10.	IDE en la nube	29
2.2.10.1.	Azure Machine Learning	29
2.2.10.2.	Google Colab	29
2.2.10.3.	SageMaker	30
2.2.11.	IDE de escritorio	31

2.2.11.1.	Jupyter Notebook	31
2.2.11.2.	Spyder	32
2.2.11.3.	Pycharm	32
2.2.11.4.	Visual Studio Code	33
2.2.12.	Inteligencia Artificial	33
2.2.12.1.	Orígenes de Fundamentación de la IA	35
2.2.12.2.	Aprendizaje Automático (Machine Learning)	37
2.2.12.3.	Ciclo de vida del Machine Learning	37
2.2.13.	Librerías que se usan en Machine Learning	38
2.2.13.1.	Pandas	38
2.2.13.2.	Numpy	38
2.2.13.3.	Scipy	38
2.2.13.4.	Matplotlib	38
2.2.13.5.	Tensorflow	38
2.2.13.6.	Aprendizaje Profundo	38
2.2.14.	Tipos de Aprendizaje	39
2.2.14.1.	Aprendizaje Supervisado	39
2.2.14.2.	Aprendizaje No Supervisado	39
2.2.14.3.	Aprendizaje por Refuerzo	39
2.2.15.	Algoritmos de Aprendizaje Supervisado	39
2.2.15.1.	Regresión Lineal	39
2.2.15.2.	Regresión Polinomial	40
2.2.15.3.	Regresión de Soporte Vectorial	40
2.2.16.	Pasos para la creación de un modelo de Machine Learning	41
2.2.16.1	Datasets	41
2.2.17.	Natural Language Processing (NLP)	41
2.2.17.1.	Librerías utilizadas en NLP	42
2.2.17.2.	Nltk	42
2.2.17.3.	Sklearn	42
2.2.17.4.	Pre-Procesamiento	43
2.2.17.5.	Lista de pasos para el Preprocesamiento	43
2.2.17.6.	Tokenización	43
2.2.17.7.	Stop Words	44

2.2.17.8.	Lematización	44
2.2.17.9.	Normalización de palabras	44
2.2.17.10.	Eliminación de signos de puntuación	44
2.2.17.11.	Transformación de símbolos especiales	44
2.2.17.12.	Bolsa de palabras	44
2.2.18.	Modelos NLP	45
2.2.18.1.	Words Embeddings	45
2.2.18.2.	Meta Embeddings	45
2.2.18.3.	Word2vec (W2V)	45
2.2.18.4.	Global for Word Representation (GloVe)	46
2.2.18.5.	FasText	46
2.2.18.6.	Transformer	47
2.2.18.7.	Blazingtext	48
2.2.18.8.	Embeddings from Language Models (ELMo)	48
2.2.18.9.	Bert	48
2.2.19.	Técnicas NLP	49
2.2.19.1.	Redes Neuronales Recurrentes (RNN)	49
2.2.19.2.	Multilabel o multi etiqueta	49
2.2.19.3.	Support Virtual Machine para multilabel o multi etiqueta	49
2.3.	Marco Legal	50
2.3.1	Constitución de la República del Ecuador	50

Capítulo III

Propuesta

N°	Descripción	Pág.
3.	Metodología	51
3.1.	Propuesta tecnológica	51
3.12.	Diseño de la propuesta tecnológica	51
3.2.	Metodologías de Investigación	52
3.2.1.	Metodología Mixta	52
3.2.2.	Metodología para la revisión Bibliográfica	52
3.2.3.	Metodología Cualitativa	52
3.2.4.	Metodología Cuantitativa	52
3.2.5.	Técnicas de Investigación	53

3.2.5.1.	Descripción del proceso metodológico	53
3.2.5.2	Población	53
3.2.5.3.	Muestreo no probabilístico por conveniencia	53
3.2.5.4.	Encuestas	53
3.2.5.5.	Entrevistas	53
3.2.5.6.	Problemas con recolección de datos	54
3.3.	Análisis de las Encuestas	54
3.4.	Resumen de las Entrevistas	71
3.5.	Resultados metodológicos	72
3.6.	Creación del modelo de Machine Learning	73
3.6.1.	Importación de Datos	77
3.6.2.	Tratamientos de Datos	79
3.6.3	Aplicación de Técnicas	82
3.6.3.1	StopWords	82
3.6.3.2	Tokenización	83
3.6.4.	Elección del Modelo	83
3.6.5.	Evaluación del Modelo	85
3.6.6.	Entrenamiento del Algoritmo	85
3.6.7.	Evaluación de Métricas	85
3.6.8.	Resultados del Modelo	87
3.7.	Metodología de Desarrollo de la Aplicación	88
3.7.1.	Metodología del Prototipado	88
3.7.2.	Prototipo de baja Fidelidad	88
3.7.3.	Construcción de la Aplicación Móvil	88
3.7.4.	Prototipo de baja Fidelidad - Media Fidelidad	91
3.8.	Desarrollo de la aplicación Móvil	91
3.8.1.	Consideraciones antes de unir Módulos (IA) con APP	94
3.8.2.	Importación y exportación del Modelo de Machine Learning	96
3.8.3.	Complicaciones de compatibilidad	96
3.8.4.	Implementación del Módulo (MLP, NLP) dentro de la Aplicación	97
3.8.5.	Testeo y Verificación del proyecto de Software	98
3.9	Conclusiones	101
3.10	Recomendaciones	102

Anexos	102
Bibliografía	132

Índice de Tablas

Nº	Descripción	Pág.
1	Causas y Consecuencias.	2
2	Delimitación del Problema.	3
3	Clasificación de dislalia en función del fonema implicado en el error.	9
4	Clasificación de fonemas.	11
5	Tratamientos de dislalia Infantil.	12
6	Intervención directa.	13
7	Intervención Indirecta.	13
8	Características de Python.	15
9	Características de Java.	15
10	Características de JavaScript.	16
11	Características de C#.	17
12	Características de C/C++.	18
13	Arquitectura de Kivy (Núcleo).	20
14	Arquitectura de Kivy (IU).	21
15	Características de Kivy.	22
16	Características de Flutter.	23
17	Características de Android Studio.	25
18	Arquitectura de Android.	26
19	Comparación Características Kivy, Flutter, Android Studio.	27
20	Comparación Cualidades y Usabilidad Kivy, Flutter, Android Studio.	27
21	Características de Jupyter	31
22	Características de Spyder	32
23	Cronología de la Inteligencia Artificial	35
24	Historia de la Inteligencia Artificial	36
25	Edad.	54
26	Género.	55
27	País de Residencia.	56
28	Provincia de Residencia.	57
29	Ciudad o Cantón de Residencia.	58
30	Tipos de Dislalia.	59
31	Terapias Presenciales y Ejercicios en Casa.	60

32	Tiempo de Duración de los ejercicios.	61
33	Días de Practica de los ejercicios.	61
34	Nivel de comprensión de ejercicios.	62
35	Conocimiento de Aplicaciones dirigidas a Dislalia.	63
36	Aplicaciones Intuitivas.	64
37	Desarrollo de Aplicaciones para Dislalia.	65
38	Aplicaciones dirigidas a Android.	66
39	Uso fácil de aplicaciones móviles.	67
40	Uso de aplicaciones móviles.	68
41	Modalidad de configuración de los ejercicios.	69
42	Consideración de guardar progreso.	70
43	Verbos en forma infinitiva y conjugados que usa el Machine Learning (NLP)	75
44	Modelo de Machine Learning (NLP).	77

Índice de Gráficos

Nº	Descripción	Pág.
1	Clasificación de tipos de Dislalia.	10
2	Mejores lenguajes de programación.	14
3	Artículo Mejores Lenguajes de Programación.	14
4	Frameworks de desarrollo Web de Python	19
5	Frameworks de desarrollo Apps de escritorio y móviles de Python	19
6	Arquitectura de Kivy.	20
7	Lenguaje KV partición de Lógica de programación.	21
8	Implementación de modelos de IA con el framework de Kiv.	22
9	Arquitectura de Flutter.	24
10	Árbol de Widgets de Flutter.	25
11	Arquitectura de la plataforma de Android Studio.	26
12	Microsoft Azure en Ignite	29
13	Entorno de Google Colaboratory.	30
14	Requisitos de Google Colaboratory	30
15	Entorno de Sagemaker	31
16	Entorno de la aplicación Jupyter	31
17	Entorno de la aplicación de Spyder.	32
18	Entorno de la aplicación de Pycharm	33
19	Entorno de la aplicación de VSC.	33
20	Campos que puede cubrir la IA	34
21	Ramas de la IA.	34
22	Pasos para crear un modelo de Machine Learning.	37
23	Tipos de Aprendizaje.	39
24	Regresión Lineal.	40
25	Pasos para creación de un modelo.	41
26	Clasificación de ramas de NLP.	42
27	Pasos para el preprocesamiento de datos de un modelo.	43
28	Fases de creación de Modelos NLP	45
29	Arquitectura de red neuronal del modelo Word2vec.	46
30	Arquitectura del modelo Fastext	47
31	Arquitectura del modelo Transformer	47

32	Arquitectura del modelo ELMo.	48
33	Arquitectura del modelo Bert.	49
34	Metodología de la investigación	52
35	Edad de los encuestados.	55
36	Género de los encuestados.	56
37	Género País de residencia de los encuestados.	56
38	Género Provincia de residencia de los encuestados.	57
39	Ciudad de residencia de los encuestados.	58
40	Tipos de dislalia que presentan los encuestados.	59
41	Ejercicios enviados a casa.	60
42	Tiempo de practica de ejercicios de los encuestados.	61
43	Práctica en días semanales de los encuestados.	62
44	Dificultad de manejo de tecnologías.	63
45	Conocimiento de tecnologías de apoyo.	64
46	Aplicación intuitiva de manejo.	65
47	Opinión sobre desarrollo de la aplicación.	66
48	Desarrollo de aplicaciones para Android	67
49	Capacidad para el uso de tecnologías.	68
50	Tecnología para el apoyo de la dislalia	69
51	Uso de tecnología para los Padres y Madres.	70
52	Información de Proceso del paciente.	71
53	Importación de datos.	78
54	Importación de datos.	78
55	Observación de datos.	79
56	Observación de datos.	79
57	Tratamiento de datos.	80
58	Tratamiento de datos.	80
59	Comparación columnas de texto.	80
60	Comparación columnas de texto.	81
61	Comparación columnas del texto.	81
62	Comparación columnas.	81
63	Uso del StopWord.	82
64	Uso del StopWord.	82

65	Uso de la Tokenización.	83
66	Arquitectura del modelo Básico DNN.	84
67	Arquitectura del modelo Básico DNN.	84
68	Entrenamiento del modelo Básico DNN.	85
69	Evaluación de Accuracy del modelo Básico DNN.	86
70	Evaluación de Model loss del modelo Básico DNN.	86
71	Evaluación de Val_accuracy_precision_recall_auc.	87
72	Evaluación de Val_accuracy_precision_recall_auc.	87
73	Evaluación de Val_accuracy_precision_recall_auc.	88
74	Prototipo de baja fidelidad primera Vista de la APP.	89
75	Prototipo de baja fidelidad Segunda Vista de la APP.	89
76	Prototipo de baja fidelidad Tercera Vista de la APP.	90
77	Prototipo de baja fidelidad Cuarta Vista de la APP.	90
78	Prototipo de baja fidelidad Quinta Vista de la APP.	91
79	Prototipo de Baja-Media fidelidad Diseño general de la APP.	91
80	Instalación de Python.	92
81	Instalación de Kivy.	92
82	Entorno VSC con librerías de Kivy.	93
83	Ejecución de archivo .exe de Kivy.	93
84	Modalidad de Programación en Kivy archivo.kv.	94
85	Modalidad de Programación en Kivy archivo.kv.	94
86	Controladores Gráficos de Nvidia.	95
87	Controladores de Tensorflow	95
88	Lista de controladores.	95
89	Comandos de instalación en el cmd de la librería Tensorflow.	96
90	Importación y exportación del Modelo.	96
91	Importación y exportación del Modelo.	97
92	Importación y exportación del modelo.	97
93	Implementación del modelo.	97
94	Implementación del modelo.	98
95	Implementación del modelo.	98
96	Implementación del modelo.	99
97	Testeo y verificación la Aplicación.	99

98	Testeo y verificación la Aplicación.	99
99	Testeo y verificación la Aplicación.	100
100	Testeo y verificación la Aplicación.	100



**ANEXO XIII.- RESUMEN DEL TRABAJO DE
TITULACIÓN (ESPAÑOL)
FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



**“DESARROLLO DE UN PROTOTIPO DE APLICACIÓN MÓVIL CON
ALGORITMOS DE MACHINE LEARNING Y NLP PARA EL APOYO DE
PERSONAS CON DISLALIA EVOLUTIVA”**

Autor: Vera Vera Juan Andrés

Tutor: Ing. Comp. Acosta Guzmán Iván Leonel. MSIG.

Resumen

El siguiente trabajo aborda sobre el desarrollo del lenguaje comunicativo en infantes, si bien la dislalia es tratada por terapeutas de lenguaje, en muchas ocasiones, los tratamientos son costosos, no se toman las indicaciones por profesionales a falta de uno en los hogares y la falta de importancia sobre los representantes hacia el infante que la padezca.

Por estos motivos, se propone realizar como apoyo tecnológico el desarrollar un software aplicativo móvil que contengan ejercicios de práctica de articulación de fonemas. Para que sirva como complemento de práctica de los ejercicios de reforzamiento de lenguaje en casa. Para la metodología de este trabajo se realizó las metodologías bibliográfica, cualitativa y cuantitativa con el fin de poder recolectar la información, y poder llevar a cabo el presente trabajo investigativo. La presente metodología de desarrollo del software es bajo el método prototipado el cual destaca en realizar versiones betas tanto en diseño como en funcionalidad, debido a que existen variedades de los tratamientos sobre la dislalia.

Como hallazgo durante la investigación se tiene que los tratamientos de dislalia se realizan mediante ejercicios de intervenciones directas e indirectas para así estimular la fonoarticulación y articulaciones de fonemas. Dentro del campo de desarrollo de software, Flutter es la tecnología más usada para aplicaciones nativas.

Como conclusiones las métricas de calidad fueron resultados de acc, 89,29%, precisión, 98,94%, auc, 99,97%, recall, 98,89%. Diseñando un aplicativo móvil con algoritmos NLP, para reforzamiento de lenguaje para dislalia evolutiva.

Palabras claves: Aplicación móvil, Dislalia, fonoarticulación, red neuronal, OpenGL.



**ANEXO XIV.- RESUMEN DEL TRABAJO DE
TITULACIÓN (INGLÉS)
FACULTAD DE INGENIERÍA INDUSTRIAL
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



**“DEVELOPMENT OF A PROTOTYPE MOBILE APPLICATION WITH
MACHINE LEARNING ALGORITHMS AND NLP FOR SUPPORT OF PEOPLE
WITH DEVELOPMENTAL DYSLALIA.”**

Author: Vera Vera Juan Andrés

Advisor: Eng. Acosta Guzmán Iván Leonel. MSIG

Abstract

The following work deals with the development of communicative language in infants, although dyslalia is treated by speech therapists, in many cases, the treatments are expensive, the indications are not taken by professionals due to lack of one at home and the lack of importance of the representatives towards the infant who suffers from it.

For these reasons, it is proposed to develop as a technological support the development of a mobile application software containing phoneme articulation practice exercises. To serve as a complement to the practice of language reinforcement exercises at home.

For the methodology of this work, the bibliographic, qualitative and quantitative methodologies were used in order to collect the information, and to carry out this research work. The present methodology of software development is under the prototyping method which stands out in making beta versions both in design and functionality, because there are varieties of treatments on dyslalia.

As a finding during the research, we have found that dyslalia treatments are performed through direct and indirect intervention exercises to stimulate phonoarticulation and phoneme articulations. Within the field of software development, Flutter is the most used technology for native applications.

As conclusions, the quality metrics were acc, 89.29%, accuracy, 98.94%, auc, 99.97%, recall, 98.89%. Designing a mobile application with NLP algorithms for language reinforcement for developmental dyslalia.

Keywords: Mobile application, Dyslalia, phonoarticulation, neural network, OpenGL.

Introducción

La dislalia es considerada un trastorno en el lenguaje que surge en un infante por múltiples causas entre ellas problemas de articulación de palabras y frases, desarrollo parcial de los órganos periféricos del habla, por lo cual la expresión y comportamiento social puede llegar a verse afectado y convertirse en una situación compleja de remediar, a no ser tratada de manera temprana. Es por ello impulsar procesos de orientación y apoyo para que el infante logre la forma correcta de estimular la comunicación y lenguaje desde las primeras etapas de desarrollo de vida.

En condiciones normales el desarrollo natural permite que los infantes alcancen una correctamente pronunciación de frases, por lo cual pequeñas falencias de pronunciación desaparecen con la practica en el tiempo, debido a la evolución de los órganos periféricos del habla, el mejor control de la lengua, el aprendizaje mental y desarrollo lingüístico sin que sea necesaria la intervención de ningún profesional. (Moreno González, 2020)

El medico por su parte puede adoptar por incorporar innovaciones tecnológicas que hayan sido probadas en otros lugares del mundo, y que estén aceptadas por el marco legal del país, con miras a que estas le permitan elevar los resultados alcanzados por los pacientes durante el periodo que duren las terapias.

En este caso el desarrollar aplicaciones que sean enfocadas en reforzamiento de lenguaje sobre la dislalia son buenas aportaciones y usos debido a que en la actualidad. Existen diversos software y programas de desarrollo de código abierto (Bhoyarkar et al., 2019), para una interfaz amigable y sencilla de usar para los infantes, una de ellas es el framework de Kivy al ser un framework del lenguaje de programación de Python lo hacen una herramienta competente debido a que Python es un lenguaje que ha desarrollado popularidad actualmente por sus aportes en la ciencia de datos y el aprendizaje automático.

El Procesamiento de Lenguaje Natural está siendo empleado en múltiples usos de la vida diaria como el Análisis de Sentimientos de los clientes para con los productos de una empresa, Clasificación de textos como libros de una biblioteca, generación de resúmenes de textos extensos, traducción de textos a otro idioma, resolviendo con ello problemas de la vida cotidiana. (Shamrat et al., 2021).

Capítulo I

El Problema

1.1 Planteamiento Del Problema

En datos recientes se indica que existen múltiples casos de niños hasta el momento en que llegan a ser mayores de edad que nacen con discapacidad del trastorno del lenguaje en los cuales presentan problemas en juntar sonidos que les dificulta al momento de expresarse y comunicarse en el habla. Según (Dager Jacome, L. D., & Tomala Vincas, B. J., 2020) “En Ecuador existe una población infantil de 4’333.264, en edades comprendidas entre 5 a 12 años, de los cuales 1.343.311”, debido a esta cifra marca a niños con problemas de aprendizaje en su lenguaje oral y comunicativo. Por otro lado, existen 2.621 niños con dificultades de aprendizaje dentro del sistema educativo, estos se identifican con una enseñanza pedagógica llevado a cabo por docentes de pedagogía para su respectivo apoyo recalcando así la información del Ministerio de Educación., cifra que será considera como la población objetivo para el presente estudio. (Telegrafo, 2019)

La poca complementación tecnológica como apoyo a los procesos de desarrollo del lenguaje de los infantes en los cuales pueden apoyar en agilizar terapias y evolución y brindar mejoras en los pacientes con dislalia sin estas herramientas los procesos suelen ser más lentos (Flor Alvarado, A. S., & Rodríguez Hurtado, V. E, 2021). En gran medida los pacientes que padecen de este trastorno, no obtienen un buen tratamiento por la falta dedicación y responsabilidad que se presenta dentro del procesos de terapias. Al momento de practicar ejercicios en hogares no se realizan, por los días y tampoco por el tiempo diagnosticado que sugieren los terapeutas de lenguaje enviados hacia los que presentan este trastorno.

Una de las causas por las que este trastorno, se convierte en un problema y se desarrolla es porque, se descuida el desarrollo del lenguaje de comunicación al momento que se presenta discapacidades de producir fonemas dentro de un infante se lo considera normal aun cuando tiene problemas en pronunciarlos afrontando ya los 4 años de edad según (Nacimba, V. M., 2012) delimita que se necesita la consulta de un profesional de terapias de lenguaje un logopeda como tal si los errores en el habla en general persisten más allá de los 4 años, por lo tanto, los terapeutas diagnostican si la persona tiene dislalia o no.

1.2 Causas Y Consecuencias Del Problema

Tabla 1. Causas y Consecuencias

CAUSAS	CONSECUENCIAS
- Problemas económicos y costos altos de las terapias del lenguaje.	- En algunos casos puede presentarse complicaciones con el paciente por lo que requiero una alta demanda económica. - Normalmente cuyos casos no se suele optar por tomar el tratamiento, en otros casos abandonan las terapias.
- Problemas de movilización donde se encuentran los centros terapéuticos.	- Habitantes que viven en zonas rurales se ven afectados en esta cusa debido a que se presenta problemas de movilización y transporte.
- Los centros terapéuticos que tratan de dislalia presentan terapias manuales. En tratamientos para los pacientes.	- Al tener falta de calidad en los tratamientos, generan que sean en casos complejos poco efectivos y limitados para los pacientes.
- Los pacientes además de recibir el tratamiento, lo refuerzan en sus hogares y en este caso no cuentan con la supervisión de un profesional.	- Los pacientes no cuentan con la supervisión de un profesional en sus hogares que los guie, y no disponen de ejemplificaciones de ejercicios a practicar en sus hogares.

Elaborado por: Vera Vera Juan Andrés

1.3 Formulación Del Problema

¿Puede una aplicación móvil con algoritmos de machine learning y NLP usarse como apoyo para realización de ejercicios de terapias de lenguaje enviadas a casa para pacientes que poseen de Dislalia Evolutiva?

1.4 Sistematización del problema

¿Qué tecnologías o frameworks se utilizan para la creación de app móvil?

¿Qué tecnologías se utilizan para la creación de modelos NLP de clasificación de textos?

¿Cuáles son los tipos de ejercicios más comúnmente ejercicios enviados a repaso en casa?

1.5 Delimitación Del Problema

En la siguiente Tabla 2 se procede a delimitar el tema mostrando el campo aspectos y características que se efectuaran en el tema.

Tabla 2. Delimitación del problema

<i>Campo</i>	<i>Aplicación de tecnología de la información</i>
<i>Área</i>	Tecnología de los ordenadores
<i>Aspecto</i>	Asistente Virtual realizado con Idle escritorio compatible con Python
<i>Tema</i>	“Desarrollo de un prototipo de aplicación móvil con algoritmos de machine learning y NLP para apoyo de personas con dislalia evolutiva”

Elaborado por: Vera Vera Juan Andrés

1.6 Objetivos

16.1 Objetivo General

- Desarrollar un prototipo de aplicación móvil como herramienta funcional aplicando algoritmos de Machine Learning y NLP para complemento del tratamiento de personas con dislalia evolutiva vía ejercicios en casa.

1.6.2 Objetivos Específicos

- Realizar una revisión bibliográfica de los conceptos etiológicos y bibliográficos relacionados a la dislalia y sus tratamientos.
- Identificar las herramientas tecnológicas para la elaboración de un modelo NLP, para el procesamiento de texto humano en idioma español.
- Realizar un análisis comparativo de tecnológicas disponibles para la construcción de aplicaciones móviles.
- Construir un modelo NLP como herramienta integrada a una aplicación móvil.
- Evaluar el funcionamiento del modelo NLP a través de métricas de calidad.

1.7 Premisa de la Investigación

El poder realizar un prototipo de aplicación con módulo de Machine Learning (NLP) integrado para peticiones de ejercicios de práctica de articulaciones de fonemas para el apoyo a infantes que padezcan de dislalia evolutiva.

1.8 Operacionalización

Dentro del campo de trastornos del lenguaje especificando la dislalia, no se puede contar con mucho apoyo tecnológico que complemente con ejercicios de practica que pueda reforzar a las terapias presenciales. En los cuales los terapistas del lenguaje tienen delimitado los ejercicios de acuerdo a los fonemas erráticos que se posee el paciente y también de tiempos y días que se requieren de practicar para presentar una mejora con el tratamiento.

1.8.1 Variable Independiente:

- Nivel de comprensión y reproducción de los ejercicios enviados a casa por parte del padre madre o tutor encargado del niño.
- Nivel de accesibilidad a aplicaciones móviles que faciliten el acceso a ejercicios de terapias para ejecución en casa.
- Frecuencia de ejecución de ejercicios en casa.

1.8.2 Variable Dependiente:

- Nivel de eficiencia de los ejercicios de terapias enviados a casa.

1.9 Alcance Del Problema

Para el desarrollo del presente trabajo se llevará a cabo, la investigación de antecedes teóricos y empíricos estudiando así los conceptos que se presentan en anteriores investigaciones sobre la dislalia evolutiva y otros tipos de dislalia para que puedan implementarse dentro del módulo y así pueda desarrollarse técnicas de estimulación en el lenguaje a los niños que comprenden esta discapacidad como lo es la dislalia, así mismo se realizara entrevistas a logopedas obteniendo información y parámetros, en las que se pueda asociar dentro de la herramienta.

En efecto se espera en un futuro de impulsar e implementar investigaciones y desarrollos de estas nuevas tecnologías como apoyo para el desarrollo de la comunicación, lenguaje y desarrollo mental de los infantes, mediante innovaciones tecnológicas que puedan aportar en este problema mediante esta investigación se desarrollara un prototipo de aplicación para infantes de 4 a 10 años de edad. En los cuales se tendrán ejercicios de repaso

de fonemas mediante reproducciones de audios de como decir frases y ejercicios de completar frases según el fonema que se dese practicar.

En el desarrollo de la aplicación solo se tomará en cuenta las técnicas de articulación que realizan los terapeutas del lenguaje como son el repaso de sílabas de los fonemas como reproducción de audios de cómo deben emitirse los mismos. Además de implementar un modelo de inteligencia artificial de procesamiento de lenguaje natural como asignación del fonema que se requiera practicar dentro del aplicativo móvil.

Posteriormente la aplicación se realizará mediante una capa en la cual no tendrá, servidores de alojamiento de la lógica de la programación y tampoco contará con servidores de base de datos.

1.10 Justificación e Importancia

La presente propuesta de trabajo de investigación nace de la importancia en la intervención de la tecnología con el lenguaje y expresión oral, como se sabe la comunicación mediante el lenguaje oral es de muy alta importancia, debido a que es la fuente de interacción que se tiene a diario entre personas, el poder relacionarse con los demás en varios contextos.

Mediante la comunicación las personas adquieren informaciones, compartimientos de expresiones, se puede controlar el comportamiento social y por último se puede organizar los pensamientos que se transmiten al momento de comunicarse. De esta manera se favorece al desarrollo, afecto y el pensamiento con respecto al campo social sobre una persona.

Se debe tener en cuenta que la dislalia que no representa una severidad como tal si es llegado a tratarse una vez la persona es diagnosticada en su etapa infantil, pero si no se tiene los debidos cuidados y atención para quienes los padecen este trastorno evoluciona con el pasar del tiempo, para luego ser más difícil el poder tratarse, esto conlleva a consecuencias adversas dentro del desarrollo intelectual como emocional de la persona que padezca este trastorno.

El poder fortalecer este trastorno mediante terapias una vez se identifica es de vital importancia y por ende se debe tener constancia por parte de los representantes de la persona que lo padezca, como se sabe la tecnología avanza y es un campo que normalmente engloba todo el mundo, para ello es de gran ayuda el poder complementar la tecnología mediante softwares, como apoyo en estas terapias de lenguajes en los cuales son enviados una variantes de ejercicios articulación de fonemas para que las personas que padezcan este

trastorno puedan tener un avance y un desarrollo ágil y mental en su entorno de comunicación social durante su etapa infantil.

Capítulo II

Marco Teórico

2.1 Antecedentes De La Investigación

El estudio de investigación de (Díaz Chávez, K. R., & Miranda Martillo, F. R., 2017, pág. 3)“En el Ecuador las deficiencias en el lenguaje y comunicación son relativamente frecuentes en la infancia, lo cual preocupa a padres y profesionales de la salud.”

Esto menciona que los problemas en la niñez el lenguaje de comunicación es de lo más comunes que se presentan consistiendo en la deficiencia de capacidad que tienen para articulación de lenguaje en los niños, esto repercute en las relaciones sociales de los infantes y también en los centros educativos en los que se establecen y ejercen sus estudios básicos durante su etapa infantil.

En el trabajo de (Cabezas, G. S., & Andrade, B. R., 2019) menciona la siguiente referencia a un estudio realizado en Chile de niños que presentaban problemas de lenguaje y aprendizaje los cuales de entre 5 a 7 años, se hayo que el 21.91% corresponde a dislalias, un 59.37% corresponde a niños, y 40.62% a niñas, delimitaron que el 18.53% pertenecen a dislalia funcional y el 3.37% a dislalia orgánica.” (Cancino, D., Castro, R. y Opazo, K. , 2017)

El presente dato de la investigación consiste en que los problemas de dislalias que más se presentan en la etapa infantil de las personas es especificando de 5 a 7 años de edad tanto como géneros masculinos y femeninos, afectando más al género masculino pertenecen al tipo de dislalia funcional y dislalia orgánica.

Según datos TIC (2020) presentan datos estadísticos el porcentaje de acceso a internet desde los hogares de los ciudadanos es del 53,20%, el porcentaje de habitantes que usan el internet es de 70,2% y el porcentaje de personas que utilizan dispositivos electrónicos especificando el más común que es el smartphone es de 81,8% todos estos resultados fueron analizados y estudiados en el periodo entre 2019 al año 2020.

Según el estudio de (Díaz Chávez, K. R., & Miranda Martillo, F. R., 2017, pág. 8) menciona que no se encuentran diseños de softwares a nivel educativo como herramientas de apoyo en los tratamientos terapéuticos enfocados en la dislalia en el Ecuador.

En el trabajo de desarrollo e investigación de (Dager Jacome, L. D., & Tomala Vines, B. J., 2020) lograron impulsar una aplicación móvil, de modulo paciente con

referencia a la dislalia teniendo muy buena aceptación y muy buen desarrollo del software, de lo cual la idea del desarrollo del software del vigente trabajo de investigación será tomada en cuenta para el presente trabajo y desarrollo de la aplicación.

Por lo tanto, el uso de aplicaciones tecnológicas y dispositivos electrónicos, especificando smartphones además de ordenadores informáticos es muy notable llegando, en lo que puede llegar a encontrarse con al menos un elemento dentro de los hogares, reforzando así tema de la presente investigación, el integrar un prototipo de software interactivo de NLP para que pueda cumplir como un asistente de apoyo en los tratamientos de terapia realizados por los logopedas de la dislalia evolutiva.

2.2 Fundamentación Teórica

2.2.1 Etiología de la dislalia

2.2.1.1 Dislalia

Según investigaciones de delimita que la dislalia es un desorden en la articulación de uno a grupos de fonemas, en las cuales su causa proviene de alteraciones funcionales en órganos periféricos del habla, dentro de sus características se encuentran, la lengua, labios, paladar, mejillas y dientes. (Mindiola Vera, 2020)

La dislalia es una irregularidad del lenguaje oral que consiste en una incorrecta declamación de las palabras, los problemas de dislalia son típicos en infantes menores a siete años. (Mamani LLanque, 2019)

Por ende, la dislalia es considerado como una deformación en la articulación de fonemas, en las cuales se pueden presentar en cierto tipo de fonemas sin que se encuentre una justificación orgánica que determine su causa.

2.2.1.2 Tipos de Dislalia

Según la investigación de (Peña Uranga, 2018) menciona que se clasifica en los siguientes tipos:

- En función del fonema implicado en el error.
- En función de número de errores y su persistencia.
- En función del nivel lingüístico implicado.
- En función de la etiología.

Describiendo el fonema implicado en el error según (Perello, 1995) y de la investigación de (Peña Uranga, 2018), en las cuales datan lo siguiente.

Tabla 3. Clasificación de dislalia en función del fonema implicado en el error

Denominación	Fonema Implicado
Sigmatismo	Alteración en la articulación de las consonantes silibantes.
Rotacismo	Dificultades con la articulación de la /r/.
Gammacismo	Articulación defectuosa de la /g/
Lamdacismo	Articulación de la /l/ en vez de la /ll/
Deltacismo	Articulación de la /d/ en vez de la /t/
Mitacismo	En las bilabiales
Iotacismo	En la /j/
Yeismo	Articulación de la /y/ en lugar de la /ll/
Ceceo	Articulación de la /z/ en lugar de la /s/
Seseo	Articulación de la /s/ en lugar de la /z/
Chinoísmo	Sustitución de la /r/ por la /l/
Hotentonismo	Defecto de la articulación de todos los fonemas

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

Esta clasificación solo hace énfasis en la reproducción del fonema, y no toma en cuenta el sistema fono-articulatorio de la persona.

2.2.2 En función de numero de errores y su persistencia

2.2.2.1 Dislalia Simple

La dislalia simple se presenta en la persona cuando este no puede articular un fonema y solo en este se ve afectado.

2.2.2.2 Dislalia múltiple

La dislalia múltiple se presenta cuando la persona que esta afecta tiene problemas en articulares varios fonemas y los cuales en estos persiste y consiste el problema.

2.2.2.3 Retraso del habla

En este caso, la persona produce varios errores, según (Peña Uranga, 2018), las complicaciones en el retraso del habla no son persistentes, varían en función del contexto, de manera que los fonemas implicados en error se califican de la formalización de palabras y también como todo lenguaje de comunicación su respectiva sintaxis.

2.2.3 En función del nivel lingüístico implicado

2.2.3.1 Dislalias Fonológicas

En este caso, se mantiene en énfasis cuando el afectado no ha establecido una diferenciación entre el pronunciamiento entre fonemas similares, un ejemplo como tal se basa en los fonemas /s/ y /z/.

2.2.3.2 Dislalias Fonéticas

Dificultades a nivel de representación de la articulación, en los cuales este se presenta en la en la expresión de la comunicación aun previamente ya establecido diferencias entre fonemas, presenta aun errores en algunos. Por otro lado, también se juzga la coordinación que tienen los órganos periféricos del habla en niveles articulatorios. (Peña Uranga, 2018),

2.2.4 Clasificaciones de Tipos de Dislalia por su Etiología



Figura 1. Clasificación de tipos de Dislalia. Elaborado por: Vera Vera Juan Andrés

2.2.4.1 Dislalia Evolutiva O Fisiológica

Se destacan en el progreso del lenguaje comunicativo en que los niños desordenan o no articulan ciertos fonemas. Las causas que se muestran son por mal control del soplo, dificultades de articulación, inmadurez o falta de discriminación auditiva. (Peña Uranga, 2018)

Concurre con un progreso en el desarrollo del lenguaje en lo que el infante no estructura o deforma algunos fonemas. (Rodríguez, 2016)

La dislalia evolutiva en caso de que se desarrolle y no se solucione el problema en etapa infantil puede llegar mostrarse como un problema de adaptabilidad en el infante en,

tal caso se debe erradicar mejorando el pronunciamiento desde temprana edad. (ACNUR, 2017)

2.2.4.2 Dislalia Funcional

Es una deformación en el desarrollo de la estructuración de sonidos. Los errores tienen un nivel de similitud a los que se estudia en niños que están en desarrollo de una captación fonética. (Auqui Tineo, Maritza, 2019).

2.2.4.3 Dislalia Orgánica

Las dislalias orgánicas son producidas por lesiones anatómicas en los órganos que intervienen en el habla. Generalmente, los niños utilizan con mayor frecuencia la sustitución, la omisión o la distorsión del sonido que se espera pronunciar. (Logopedia, 2020)

2.2.4.4 Dislalia Auditiva

Es provocada por una insuficiencia auditiva que se complementa normalmente de otros problemas del lenguaje, caracterizando a la voz. (Peña Uranga, 2018)

Para poder tener una evolución de mejora en el afectado se debe desarrollar una adecuada audición para y tener una buena adición, en este caso es complejo debido a que, si el afectado no oye bien, tendrá como resultado una mala expresión del lenguaje.

2.2.5 Clasificación de fonemas por su articulación

En la investigación de (Quintanilla, 2013) se clasifican de la siguiente forma:

Tabla 4. Clasificación de fonemas.

Grupo	Fonemas	Concepto
Bilabiales	(m, b, p, v)	Entra en contacto con los labios.
Labiodental	(f)	Intervienen el labio inferior y dientes superiores.
Linguodentales o dentales	(t)	Se complementa la lengua tocando dientes superiores.
Linguoalveolares o alveolares	(l, n, s, r, rr)	La punta de la lengua toca los alveolos.
Linguovelares o velares	(c, j, g)	Intervienen los paladares.

Linguopalatales o palatales	(ch, ñ, ll, y)	La lengua se incorpora en la mitad retrasada al paladar duro, soltando una abertura de producción de aire.
Linguointerdentales o interdentales	(c, z, d)	La lengua se comunica con los incisivos superiores o inferiores.

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.6 Tratamientos

Es imprescindible llevar el tratamiento de tal manera que el que infante pueda enmendar en una manera más sencilla las deformaciones articulatorias que presenta. Para esto se complementa desarrollos neurológicos y psíquicos. (Logopedia, 2020)

Los tratamientos de este tipo de trastornos se basan en erradicar desde temprana edad del paciente cuando este es un niño y está en proceso de maduración de sus organismos ya sean neurológicos o psíquicos, de esta manera refuerzan estas habilidades de pensamiento e interacción con la sociedad para erradicarlos. Debido a que estos problemas a futuro presentaran un entorpecimiento en el desarrollo del pensamiento del niño.

Por ende, el ser un problema que es tratado desde muy corta edad los logopedas piden colaboraciones de los padres, en los cuales los instruyen como una comunicarse e interactuar con los niños que padecen dislalia en sus hogares después de las terapias.

2.2.6.1 Proceso de tratamientos

En el apartado de (Medina, 2019) se muestra la siguiente información de tratamientos de dislalia infantil.

Tabla 5. Tratamientos de Dislalia Infantil.

1	Estimulación de la capacidad del niño para producir sonidos, reproduciendo movimientos y posturas, experimentando con las vocales y las consonantes. Se le enseñará a comparar y diferenciar los sonidos.
2	Realización de ejercicios donde el niño debe producir el sonido dentro de sílabas hasta que se automatice el patrón muscular necesario para la articulación del sonido.
3	Estimulación de la coordinación de los movimientos necesarios para la pronunciación de sonidos: ejercicios labiales y linguales. Se enseña al niño las posiciones correctas de los sonidos más difíciles.

- 4 Al llegar a este punto, el niño estará preparado para empezar con las palabras completas, a través de juegos.

- 5 Una vez que el niño es capaz de pronunciar los sonidos difíciles, se tratará que lo realice fuera de las sesiones, es decir, en su lenguaje espontáneo y no solo en las sesiones terapéuticas.

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.6.2 Intervenciones

Dentro de los tratamientos se realizan las intervenciones o técnicas que se implementan dentro de los tratamientos para el desarrollo lingüístico del infante.

Según (Rodriguez, 2016) detalla lo siguiente.

- Intervenciones Directas

Se comprenden de practicar de manera específica el fonema, mediante ejercicios articulatorios en las cuales el infante no emita de manera correcta.

Tabla 6. Intervención Directa

Intervención Directa	Ejercicios de respiración, ejercicios de soplo, praxias, Ejercicios de articulación del fonema, y repeticiones de palabras.
----------------------	---

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

- Intervención Indirecta

Se comprenden a las funcionalidades de como emitir un fonema, audición de estos mismos y técnicas de prácticas en los cuales estén presentes los órganos articulatorios y también la respiración para efectuar el fonema.

Tabla 7. Intervención Indirecta

Intervención Indirecta	Ejercicios de respiración, ejercicios de soplo ejercicios del velo paladar, ejercicios de discriminación auditiva.
------------------------	--

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.7 Lenguaje De Programación

Un lenguaje de programación es la codificación que utilizan los programadores a través del de un ordenador informático mediante instrucciones para la comunicación y desarrollo de softwares aplicaciones y tareas automatizadas las cuales pueden ser desarrollo web, scripts y aplicaciones móviles.



Figura 2. Mejores lenguajes de programación. Imagen de trabajo de investigación. Elaborado por: STEPHEN CASS.

Este grafico fue realizado por métricas de un lector Spectrum de lenguajes de programación informático en los cuales no recalca el mejor si no el más popular en el año 2017. Dando como en primer lugar el lenguaje de Python.

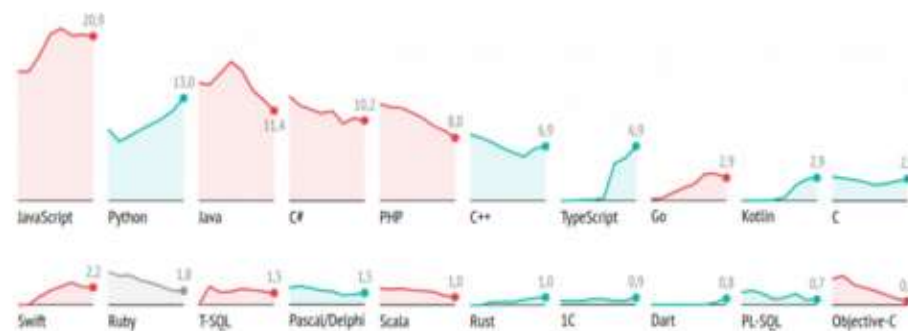


Figura 3. Artículo Mejores Lenguajes de Programación. Elaborado por Medvedev, Maxim A

Este grafico de comparación de tendencias muestra a Python como una curva de crecimiento notable ante los otros lenguajes de programación como lo son Java, C# y JavaScript.



2.2.7.1 Python

Python es un lenguaje adecuado tanto para el aprendizaje como para programar en el campo laboral. Python es un potente lenguaje de programación de alto nivel y orientado a objetos creado por Guido van Rossum.(K. R., 2017)

Es un lenguaje de programación de calidad por tener facilidades en herramientas tales como, estructurar datos como diccionarios, tuplas y listas que estas herramientas permiten facilitar la digitalización de líneas de códigos complejas en pocas líneas de programación.

El estatus de Python como el lenguaje de programación de mayor crecimiento de programación de más rápido crecimiento se debe a un fuerte aumento de su uso para la ciencia de datos.(K. R., 2017)

Tabla 8. Características de Python

Características	Python
Programación de Orientación a Objetos	 
Interfaz de usuario gráfica (GUI)	
Programación para módulos de usos de datos (Inteligencia Artificial)	
Es portable	
Es de código abierto	
Programación de base datos	
Programación en Framework de alta calidad	


Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.7.2 Java

Es un lenguaje de programación muy estructurado que su popularidad fue debido a crear softwares en diferentes campos como ya sean los apartados de Escritorio (Desktop), Videojuegos, Consolas de Video Juegos, Aplicaciones Móviles (App), además de sistemas Operativos.

Tabla 9. Características de Java

Características	Java
-----------------	------

Es robusto	
Tiene arquitectura neutral	
Portable y de código abierto	
Es dinámico	
Programación Orientada a objetos	
Programación Múltiple	

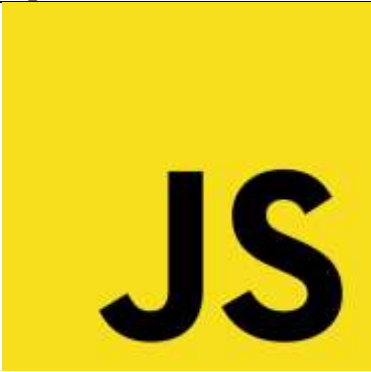
Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.7.3 JavaScript

En el apartado (Ramos, 2019) menciona que JavaScript es el lenguaje utilizado para diseñar la interactividad, este es más usado para desarrollos de entorno a páginas web debido a que los lenguajes de diseño como HTML, CSS y PHP tienen compatibilidad y facilidad en lectura de este lenguaje.

Este lenguaje es muy popular en el desarrollo web en caso de que se desarrollen interactividades en páginas web, como lo son juegos, acciones simultaneas que se obtienen al presionar botones, dinamismo y animación.

Tabla 10. Características de JavaScript

Características	JavaScript
Orientado a Objetos	
Lenguaje Interpretado	
Programación dirigida al cliente	
Sintaxis compatible con otros lenguajes populares. (Java y C/C++)	
Es dinámico	
Lenguaje simple	

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.7.4 C#

Según el artículo publicado por (Tokio, 2018) se menciona que Andrés Hejlsberg determino formar un equipo de trabajo en 1999 para desarrollar un novedoso lenguaje de programación, el cual representa C#.


Desarrollado por la empresa de Microsoft es un lenguaje de programación en cual es dedicado a la orientación de objetos, es usado para compilar codificaciones de diversas

aplicaciones. Debido a sus actualizaciones está abierto a desarrollos de aplicaciones de forma veloz, sin dejar de lado las expresiones y estilo que su lenguaje contemporáneo el C.

Investigaciones realizadas por (Tomic, 2018). En 2002, Microsoft lanzó C# como competidor directo de Java. Como lenguaje alternativo, C# comparte algo de sintaxis con Java, pero tiene más funciones. Tanto C# como C++ se han mejorado significativamente desde su lanzamiento.

Esto recalca que el lenguaje de programación C#(C-Sharp) sea reconocido como un lenguaje muy popular en la actualidad para diseños y desarrollos para aplicaciones de hoy en día.

Tabla 11. Características de C#

Características	C#
Sintaxis sencilla	
Sistema unificado (Operaciones de variables similares a otros lenguajes.)	
Integro con otros lenguajes	
Eficiente	
Adaptable a versiones	
Es un lenguaje que adquiere extensibilidad	

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.7.5 C/C++

Según (Hernandez, 2012) El primer paso hacia C++ se produjo en 1980, cuando se exhibió un primer lenguaje llamado C with classes, descrito por Bjarne Stroustrup. Posteriormente se en 1984 se renombro C++.

En este artículo se menciona que Bjarne Stroustrup desarrollo el lenguaje. Por consiguiente, basándose en otros lenguajes particulares que tenían deficiencias tales como Simula un lenguaje de programación del año 1962 tenía como características buena calidad de programación, pero, tenía deficiencia en rapidez y el lenguaje BCPL lenguaje de programación del año 1966 era rápido, pero de bajo nivel.

El C/C++ es un lenguaje de programación icónico en historia que viene de una versión mejora y alterna del lenguaje C. Este lenguaje ayudo a mejorar e impulsar la


herramienta de programación orientada a objetos. Actualmente muchos lenguajes de programación populares se sostuvieron de las bases de estilo de codificación del C/C++, con las actualizaciones de tiempos modernos es usado para desarrollo de herramientas tales como: Base de datos, Navegadores Web, Sistemas Operativos, Compiladores e incluso Videojuegos, esta última herramienta se la suele usar frente al desarrollo motores gráficos.

Según (Robledano, 2019) menciona que:

Si se requiere programar en una alta categoría, con la opción de poder bajar incluso a ensamblador, C++ sigue siendo una gran opción.

Esto alega que el lenguaje C/C++ con sus constantes actualizaciones no hacen de un programa dejado en el antaño al contrario se lo sigue usando para el desarrollo de herramientas actuales en la programación.

Tabla 12. Características de C/C++

Características	C/C++
Orientado a objetos	
Compatibilidad en bibliotecas	
Lenguaje de programación rápido	
Didáctico es base de otros lenguajes de programación como (Java, JavaScript, PHP, etc.)	
Instrucciones simples de sintaxis como if, do, while	

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.8 Front - End

Es una de las capas dentro del sistema de desarrollo web en la cual es la categoría superior al Back-en, dando a entender que es la comunicación de programación de la interfaz gráfica y visual con el usuario. Para su desarrollo se debe tener conocimiento base de los lenguajes de desarrollo web HTML, CSS y JavaScript.

2.2.9 Frameworks

Es una herramienta que es usada para elaborar un proyecto con un objetivo específico. Para cualquier tipo de desarrollo de softwares nivel web o a nivel de aplicación necesitamos de esta herramienta de programación.



Figura 4. Frameworks de desarrollo Web de Python. Elaborado: elpythonista



Figura 5. Frameworks de desarrollo Apps de escritorio y móviles de Python. Elaborado: Murray Beatty

2.2.9.1 Kivy

Según (Toro, 2017) menciona que Kivy es un framework de Python de código abierto para plataforma que acepta desarrollar aplicaciones. Este es capaz de desarrollar interfaces para el uso de usuario cliente teniendo varias plataformas de uso en las cuales se pueden destacar su flexibilidad en desarrollar aplicaciones multitáctiles.

Como framework de Python, es considerado una potente librería para el desarrollo de aplicaciones móviles debido a su interfaz natural de usuario, según (Bhoyarkar et al., 2019) como multiplataforma puede ser utilizada en diferentes sistemas operativos como lo son IOS, Android, Raspberry Pi, Windows, entre otros, mediante el protocolo de código abierto y software libre.

2.2.9.2 Arquitectura de Kivy

Según (Bhoyarkar et al., 2019) delimita que el entorno gráfico de Kivy se genera mediante la interfaz de programación de aplicación (API) de una abstracción gráfica de OpenGL, al destacarse esto permite acelerar mediante el hardware la emisión de comandos de dibujo.

Mediante esto los Widgets de Kivy utilizan esta herramienta de entorno de interfaz de usuario (IU) que está orientada en el lenguaje C, para favorecer el rendimiento por lo cual genera una ventaja debido a que se pueden optimizar mediante la codificación, si no se tiene conocimiento alguno del entorno de OpenGL.

La siguiente imagen presentara la arquitectura del núcleo de Kivy.

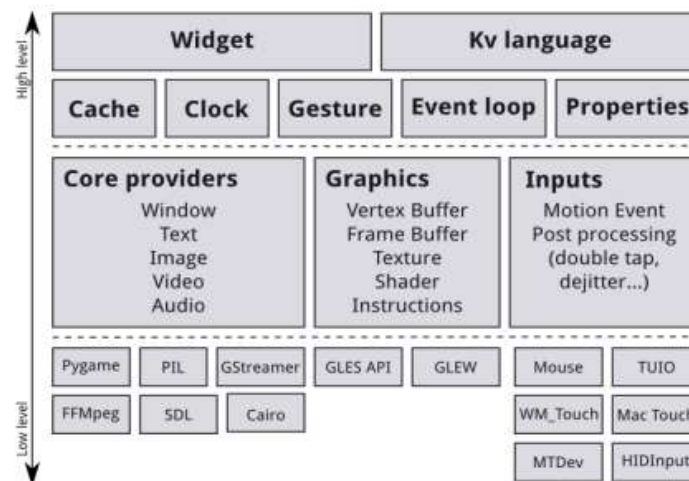


Figura 6. Arquitectura de Kivy tomada de la investigación. Elaborado por: Aman Bhoyarkar 1, Anuja Solanki, Ashutosh Balbudhe

Tabla 13. Arquitectura de Kivy (Núcleo).

Característica	Descripción
Clock	Se puede utilizar la programación por reloj para programar eventos.
Cache	Se pueden utilizar y definir clases almacenar objetos que se utilizan dentro de la programación.
Detección de gestos	Porta con un sencillo reconocimiento de trazos, en los cuales puede ser utilizados para detección de círculos o varias formas de figuras geométricas. Fácil de usar y eficiente para realizar interfaces de usuario.

Lenguaje Kivy	
Propiedades	Se basan en el conocimiento del lenguaje de Python. Las clases propias de Kivy se vinculan con los widgets en la descripción de la programación.

La siguiente tabla dará una breve explicación de las herramientas para crear interfaces entorno de usuario del framework de Kivy.

Tabla 14. Arquitectura de Kivy (IU).

Widgets y Layouts	Descripción
Widgets	Los Widgets son componentes que forman parte de la interfaz de usuario, los cuales se añaden a la programación de Python para producir una funcionalidad.
Layouts	Son los componentes para organizar los Widgets de Kivy, para que estén centralizados en un formato, es posible realizarlo dentro de los Widgets, pero siempre se usan mediante los Layouts.

2.2.9.3 Lenguaje Kivy

En caso de que se realicen proyectos de aplicaciones grandes, suele ser muy difíciles y lentas de desarrollar debido al margen de codificación y declaración de funciones, clases y variables, al momento de construir los árboles de Widgets dentro nuestra aplicación. Por ello Kivy ofrece una solución ante este problema, permitiendo particionar la lógica de programación dentro de un archivo Kv el cual permite vincular las clases y funcionalidades de manera natural, lo cual permite agilizar nuestro desarrollo, para diseñar y crear prototipos de IU. (Bhoyarkar et al., 2019)

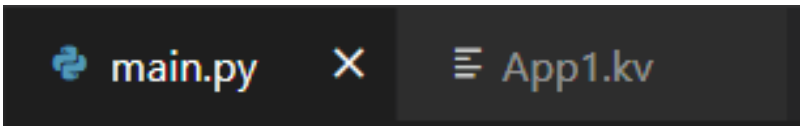


Figura 7. Lenguaje KV partición de Lógica de programación. Elaborado por: Vera Vera Juan Andrés.

Kivy al ser un framework basado en el lenguaje de Python, se han implementado sistemas de redes neuronales convolucionales, como son para el reconocimiento de rostros

danto muy buenos resultados analíticos por trabajos precedentes citando al trabajo de (Aufar & Sitanggang, 2022) se menciona que implementaron un módulo de reconocimiento facial con el framework de Kivy usando la versión 2.0 del mismo, en el cual es la misma versión que se está empleando para el presente trabajo.

En los resultados del trabajo previamente citado, concluyeron que se propusieron a mejorar la eficacia de sistemas de reconocimiento facial aplicando métodos de redes neuronales como lo es la siamesa logrando mejor la eficacia alrededor de una data de 9000 imágenes, en el cual el resultado de precisión fue 98% del modelo que emplearon. Posteriormente delimitaron que el framework de Kivy es un excelente marco para desarrollo e implementación de modelos de redes neuronales, para la demostración de resultado del modelo.

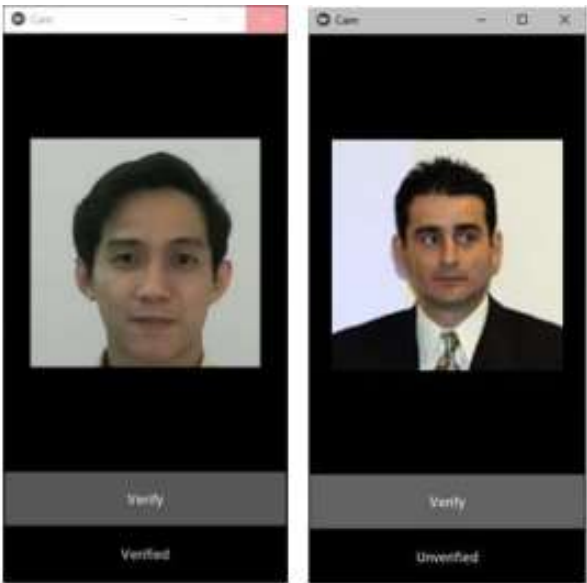


Figura 8. Tomada de la investigación de Implementación de modelos de IA con el framework de Kivy. Elaborado por: Yazid Aufar & Imas Sukaesih Sitanggang

Tabla 15. Características de Kivy

Características	Kivy
Framework para crear aplicativos multi-touch.	
Compatible con MacOSX, Linux, Windows, Android y IOS.	
Soporte para múltiples entradas (Teclado, mouse y multitouch).	

Factibilidad de realizar aplicaciones móviles

Soporte para muchos widgets con KV multitáctil para desarrollo personalizado.




Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.9.4 Flutter

Es un framework de desarrollo para crear diseñar y desarrollar aplicaciones nativas es surgió de Google y es de código abierto. Una de sus cualidades es que puede desarrollar aplicaciones generando una codificación totalmente nativa usando el lenguaje Dart por lo que mantendrá una compatibilidad para varias plataformas diferentes en desarrollo de aplicaciones en etorno Android y IOS. (Devs, 2019)

Tabla 16. Características de Flutter

Características	Flutter
Código 100% nativo	
Desarrollo ágil y rápido, flexible y estructurado usando el lenguaje Dart	
Tiempo de carga de la aplicación Veloz	
Compatibilidad con todos los Frameworks	
Buen rendimiento respecto a interfaces gráficas	

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.9.5 Arquitectura de Flutter

Flutter como herramienta su motor gráfico que usa es entorno a la herramienta denominada Skia. El cual está escrito en el lenguaje programación de C/C++, al momento de codificar en Flutter este mismo en vez de importar a Apis, usa Apis predeterminadas que se crean al inicio de cada proyecto y marco de desarrollo en flutter. (Hernández, 2021) En el siguiente gráfico se mostrarán las capas de desarrollo de Flutter.

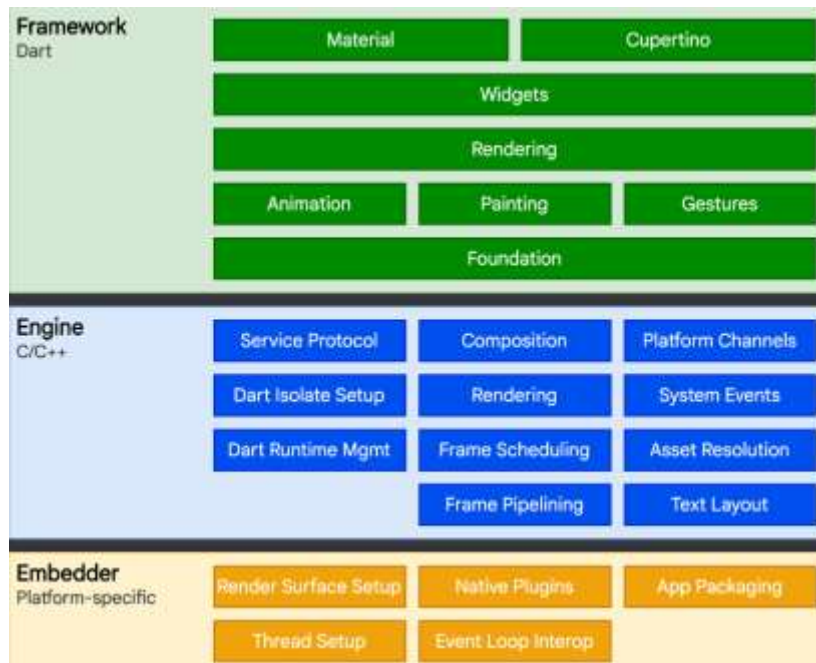


Figura 9. Arquitectura de Flutter. Tomada de la investigación. Elaborado por: Jaime Hernández

Para definir las clases dentro de la programación de entornos IU utiliza también Widgets que son semejantes a controladores nativos de IOS y Android para así facilitar y acelerar el desarrollo de aplicaciones, una de las grandes características de este framework es que utiliza el modelo de programación declarativo el cual este se caracteriza en prever lo que el desarrollador está haciendo para alternativas y sugerencias de como seguir las líneas de codificación que realiza el desarrollador. (Hernández, 2021)

En Flutter sus funcionalidades se componen como delimita en la investigación de (Hernández, 2021) “Los widgets pequeños y de un solo propósito se componen juntos para crear otros más complejos y especializados que representan la interfaz de usuario de su aplicación. Por lo tanto, toda la aplicación está representada por un árbol de widgets”.

En otras palabras, el resultado de este concepto data que las aplicaciones realizadas en Flutter, se componen de un mismo Widget en el cual este conjunta a los demás Widgets combinándolos dentro del entorno raíz con componentes usados dependiendo del proyecto que se desea desarrollar.

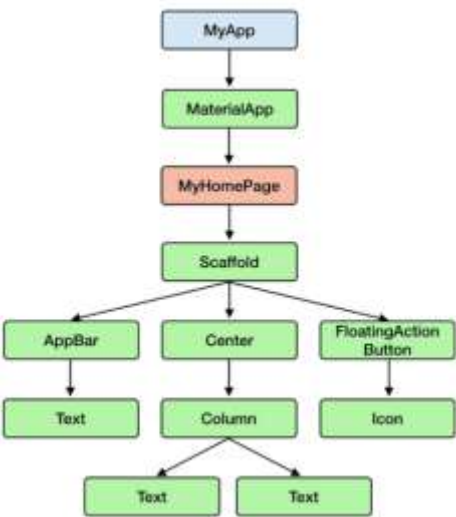



Figura 10. Arbol de Widgets de Flutter. Tomada de la investigación. Elaborado por: Jaime Hernández

2.2.9.6 Android Studio

Es un entorno de desarrollo integro (IDE) para crear aplicaciones novedosas para el sistema operativo de Android, esto se comprende desde el código hasta la interfaz gráfica a desarrollar, normalmente ese entorno tiene guías para ayudar al usuario y crear las aplicaciones con entorno al sistema Android. (Esteban, 2020)

Tabla 17. Características de Android Studio

Características	Android Studio
Flexible automatización de compilación	
Compatibilidad con servicios de nube	
Modificación de partes de código y recursos	
Compatibilidad con cualquier servicio Android	
Variedad de funciones y trabajo de entorno fluido	

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

2.2.9.7 Arquitectura de Android Studio

Android Studio también posee características de uso gratuito como se mostrará en la siguiente ilustración que demuestra la arquitectura de Android formada de cuatro capas se debe enfatizar que estas capas también son gratuitas.

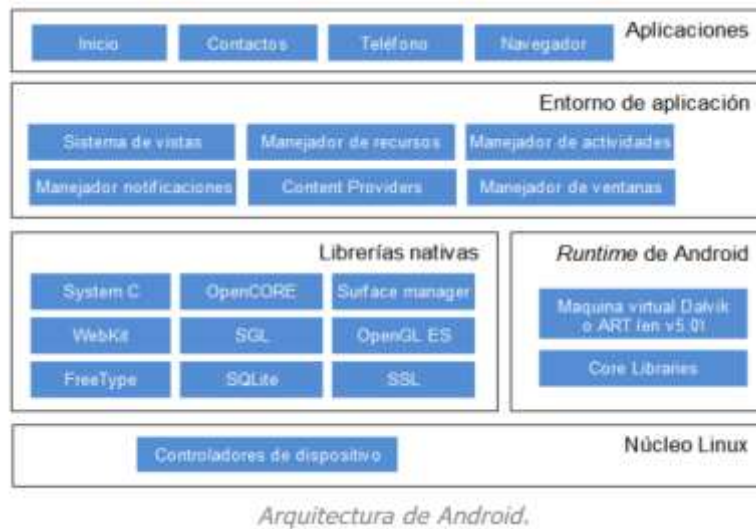


Figura 11. Arquitectura de la plataforma de Android Studio tomada de la investigación. Elaborado por: androidcurso

Tabla 18. Arquitectura de Android.

Característica	Descripción
Núcleo	Se destaca por los servicios de seguridad, usos de memoria para los multiprocesos y sus soportes de drivers usados, el núcleo fue creado por la versión de Linux 2.6
Runtime Android	Utiliza como un entorno de desarrollo una máquina virtual denominada ART, esta última fue un gran avance debido a que permitió generar la API de JAVA y reducir el consumo de recursos en un 33%.
Librerías Nativas	Usa librerías nativas orientadas en C/C++, que estas están desplegadas en la mayoría de herramientas de Android.
Entorno de Aplicaciones	Para el desarrollo de aplicaciones utiliza varias herramientas de IU como notificaciones vistas, localizaciones, sensores etc. todas estas gratuitas.

Aplicaciones	Para el desarrollo usa el lenguaje de Java o Kotlin ejecutando las codificaciones en la máquina Virtual del entorno para su conversión directa en APK o SDK.
--------------	--

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

Tabla 19. Comparación Herramientas. Kivy, Flutter Android Studio.

Tecnología	Kivy	Flutter	Android Studio
Lenguaje de programación	Python, Cython	Dart	Java y Kotlin, C++
Componentes y Bibliotecas	Amplia	Media y creciendo	Amplia
Plataformas Adaptivas	Multiplataforma, Raspberry Pi	Soportes mediante configuración para Android y Mac OS/IOS	Windows, Mac OS, Linux. Mediante Configuración.
Progreso de aprendizaje	Rápido y muy bueno para principiantes	Se requiere tener conocimientos del lenguaje Dart y alta programación en C.	Es imprescindible, tener conocimientos altos en programación debido a que utiliza Java y kotlin.
Desarrollado por	Organización Kivy	Google	Google
Hot Reload	Soportado	Soportado	Soportado
Fecha en que fue lanzado	11 de febrero 2011	26 de marzo del 2015	Mayo del 2017
Disponibilidad	Gratuita	Gratuita	Gratuita

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

Tabla 20. Comparación Cualidades y Usabilidad Kivy, Flutter Android Studio.

Flutter	Android Studio	Kivy
-Usa el lenguaje Dart para programar.	-Usa el lenguaje Kotlin y Java para programar.	-Usa el lenguaje de Python para programar.
-Comparativamente el tamaño en de programación	-Como interprete y uso de dos lenguajes de	-Como interprete se utiliza el lenguaje de Python para el

el lenguaje Flutter es más pequeño.	programación el marco es flexible pero extenso en estructura de programación.	desarrollo de funciones de la aplicación.
-Para aplicaciones con más demandas de desarrollo a nivel de implementación es mejor el marco de Flutter.	-Es recomendado para aplicaciones simples y nivel empresarial.	-Para aplicaciones con un interfaz amigable y simples es recomendado usar Kivy.
-Es menos propenso a problemas de compatibilidad ya que los widgets son parte de la aplicación, y no de la plataforma.	-No tiene problemas de Compatibilidad debido a que maneja todas las versiones programables.	-Tiene Problemas de compatibilidad al momento de importar a archivos APK por asuntos de versiones de relaciones.
-Su marco de desarrollo de interfaces gráficas es muy valorado por la comunidad de desarrollo de aplicaciones móviles.	-No tiene problemas en creación de interfaces gráficas.	-Se pueden llegar a crear hermosas interfaces gráficas competentes con los framework de desarrollo más populares.
-Tiene soporte de elementos nativos compatibles con IOS y Android.	-Tiene soporte de elementos nativos compatibles con IOS y Android.	-Su compatibilidad de desarrollo de Widgets con lenguajes nativos de los Sistemas Operativos de IOS y Android son limitadas.

Información tomada de la investigación elaborado por: Vera Vera Juan Andrés

Basado en las investigaciones y del presente estudio delimitando la flexibilidad y eficacia del lenguaje de Python, y por lo consiguiente los resultados de implementación de modelos de redes neuronales y módulos de Inteligencia Artificial con el framework de

desarrollo de aplicaciones de Kivy, se procederá a realizar el prototipo de desarrollo de la aplicación con el presente framework de Python.

2.2.9.8 Entorno de desarrollo IDE

Un entorno de desarrollo integrado o IDE como más comúnmente conocido por sus siglas es conocido por ser un software informático para el desarrollo y editar códigos de programación, en los cuales tiene la capacidad de ser compatibles con varios lenguajes de programación, para así también compilar aplicaciones desarrolladas por un usuario. (Delgado, 2021)

2.2.10 IDE en la nube

Un IDE en la nube es un entorno de codificación de programación basado en una arquitectura web y optimizado como una aplicación, este mismo consta con las mismas características principales que un IDE de escritorio, como depuración editar y compilador de códigos.

2.2.10.1 Azure Machine Learning

Este es un entorno de desarrollo de programación creado por la empresa de Microsoft, para poder trabajar en este entorno se debe adquirir mediante membresía debido a que este entorno no es gratuito. Según (Microsoft, 2022) este entorno es usado por los profesionales de desarrollo de ciencia de datos, aprendizaje automático, además de ingenieros en informática, tiene variedad para poder usarse creando modelos de inteligencia artificial, o cargarlos desde el almacenamiento previamente desarrollados en entornos de código abierto.

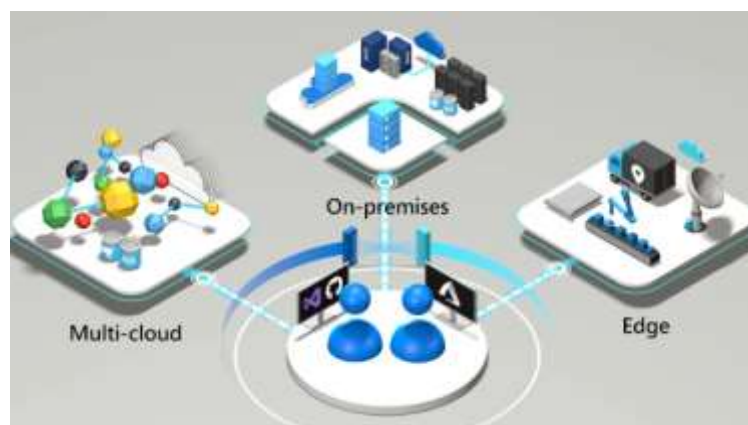


Figura 12. Tomada de Microsoft Azure en Ignite: nube híbrida. Elaborado por el autor

2.2.10.2 Google Colab

Es un entorno de alojado en Google para trabajar con el lenguaje de Python mediante cuadernos de trabajo (Notebooks) y tiene las características de trabajar con datos almacenados en la nube o desde el mismo ordenador del usuario. (López, 2021)



Figura 13. Entorno de Google Colaboratory. Elaborado por: Vera Vera Juan Andrés

En el entorno de Google Colab permite 12 horas de trabajo en los proyectos que se estén realizando dentro del mismo, así mismo este entorno de trabajo permite trabajar con dos modalidades, que son memoria gráfica (GPU), Unidad central del Procesamiento (CPU) y TPU. Podemos visualizar en la Figura 11 las especificaciones de los elementos mencionados.

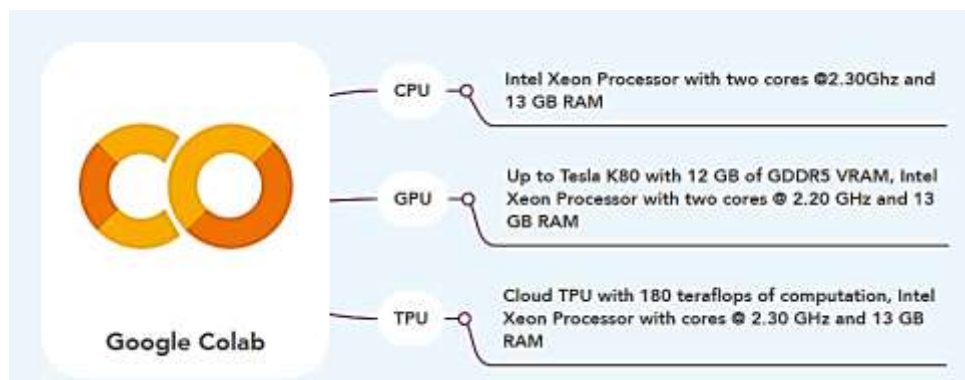


Figura 14. Requisitos de Google Colaboratory. Elaborado por el autor.

Mediante lo previamente investigando, acerca de los softwares en línea de programación, se elegirá el entorno de Google Colaboratory por facilidades, flexibilidades y servicios gratuitos con respecto al desarrollo del modelo de Machine Learning (NLP).

2.2.10.3 SageMaker (SM)

Es un servicio de entorno de desarrollo creado por la empresa de Amazon, en la cual se dispone para crear, entrenar e implementar modelos de inteligencia artificial. Este entorno proporciona un interfaz virtual basado en la web lo que hace realice la productividad de programación de ciencia de datos más rápida de lo normal. (Amazon, 2022)

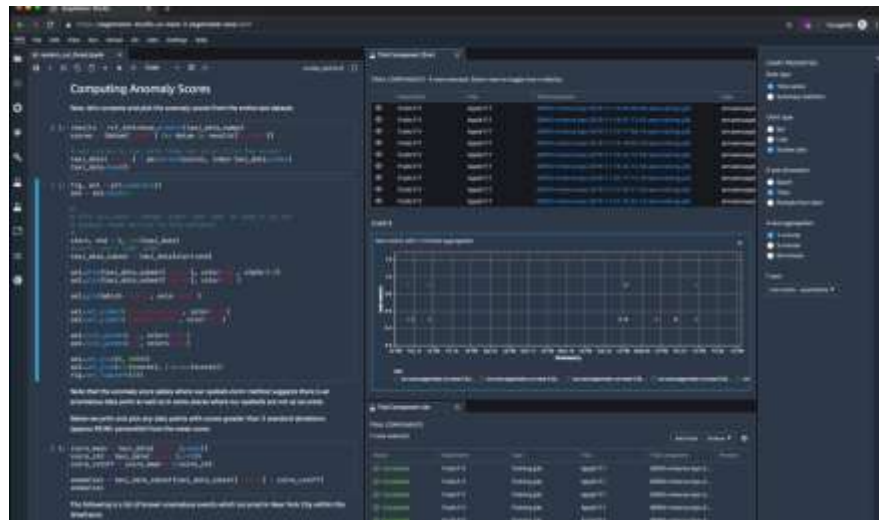


Figura 15. Entorno de Sagemaker. Elaborado por el autor.

2.2.11 IDE de escritorio

2.2.11.1 Jupyter Notebook

Es un entorno interactivo de programación y de código abierto, permite experimentar códigos y compartirlos, porta una arquitectura correspondiente a cliente servidor debido a que es un aplicativo web, además que es compatible con varios lenguajes de programación incluidos el lenguaje de Python y R. (Figueiras, 2021)

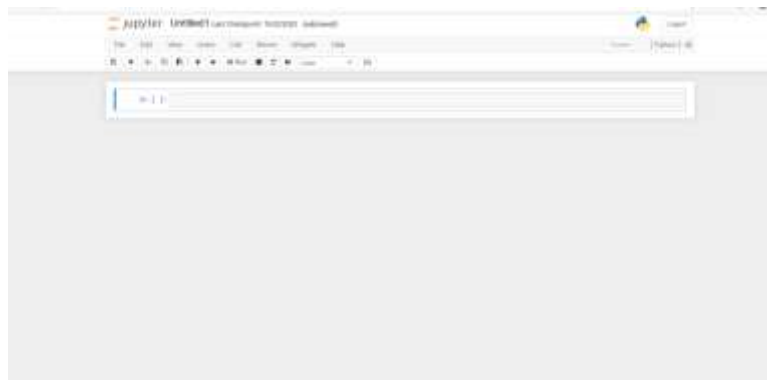



Figura 16. Entorno de la aplicación Jupyter. Elaborado por: Vera Vera Juan Andrés

Tabla 21. Características de Jupyter

Características	Jupyter
Alta compatibilidad en lenguajes de programación	
Ofrece un ambiente Interactivo	
Código abierto y utilizado para programación de ciencia de datos	
Utiliza librerías para realizar modelos de Inteligencia Artificial.	

Información tomada de la investigación Elaborado por: Vera Vera Juan Andrés

2.2.11.2 Spyder

Es un potente entorno de desarrollo interactivo para el lenguaje de Python. Tiene las cualidades de poseer variedades de funciones, bibliotecas y apartados gráficos como el uso de trazado interactivo de 2D/3D. Naturalmente es usado para programación de ciencia de datos en el lenguaje de Python. (Acharya, 2021)

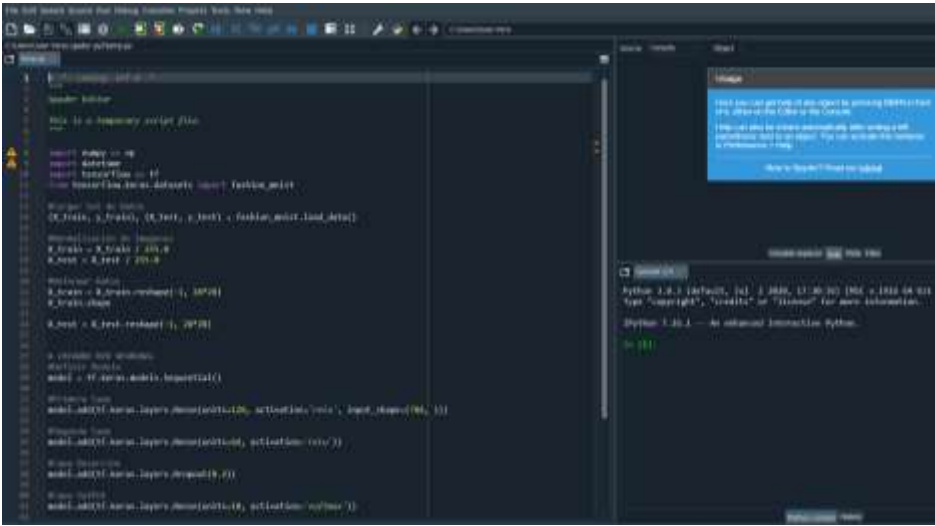


Figura 17. Entorno de la aplicación de Spyder. Elaborado por: Vera Vera Juan Andrés

Tabla 22. Características de Spyder

Características	Spyder
Código abierto y gratuito	
Disponibilidad de Varias funciones	
Apartado de entorno grafico 2D/3D	

Aplicado para la ciencia de datos, además de ser un compilador interactivo y compatible con Python.



Información tomada de la investigación Elaborado por: Vera Vera Juan Andrés

2.2.11.3 Pycharm

Es un IDE dedicado al lenguaje de programación de Python, en el cual facilita su desarrollo y codificación. En general este brinda una facilidad de programación de ciencia de datos y web con librerías familiarizadas para los trabajos específicos, contiene tres entornos de trabajo, uno que es comunal en su versión gratuita para desarrolladores aficionados de softwares, entorno empresarial para proyectos profesionales y su también cuenta con una versión Edu que sirve para aprender herramientas educativas entorno a la tecnología. (School, 2022)

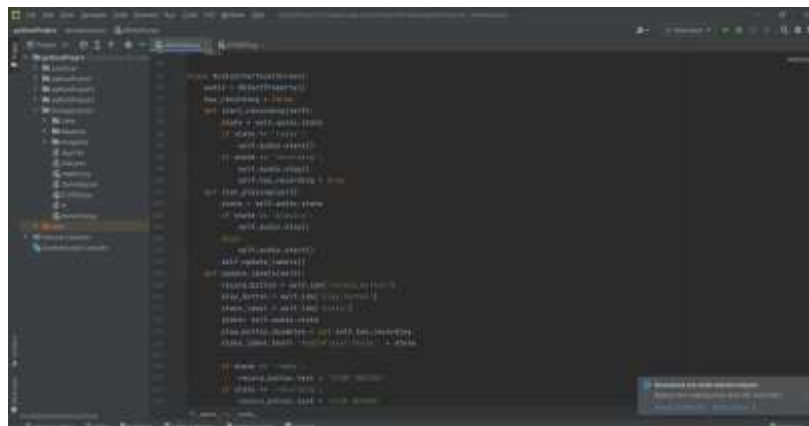


Figura 18. Entorno de la aplicación de Pycharm. Elaborado por Vera Vera Juan Andrés

2.2.11.4 Visual Studio Code

Consiste en un editor de código que fue desarrollado por la empresa de Microsoft. Es caracterizado por ser robusto, de software libre y multiplataforma. Este editor de código es compatible con varios sistemas operativos. El Visual estudio Code cuenta con soporte para depuración de código, y dispone de un sinnúmero de extensiones, por lo que permite programar con una gran variedad de lenguajes de programación.

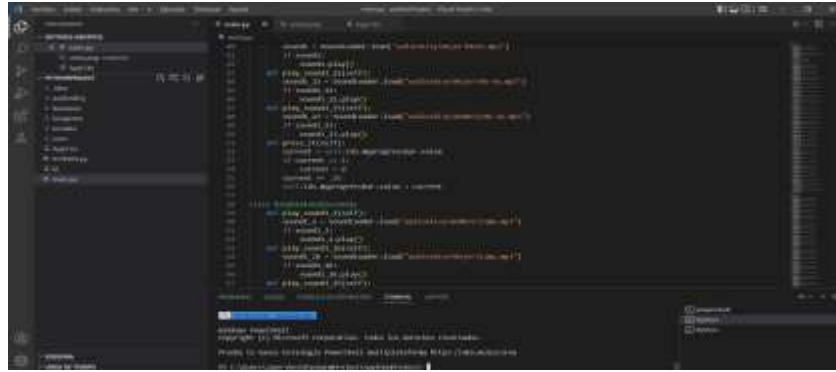


Figura 19. Entorno de la aplicación de VSC. Elaborado por Vera Vera Juan Andrés

2.2.12 Inteligencia Artificial

La inteligencia artificial (IA), es un modelo de razonamiento avanzado por máquinas, esto se origina cuando una máquina elabora ambientes intelectuales y conocedores de los seres humanos donde se relacionan con otras mentes humanas, como percibir, aprender y pensar. (Ferreira, 2022)



Figura 20. Campos que puede cubrir la IA. Elaborado por:

Vera Vera Juan Andrés

Esta herramienta tenía como propósito simular el comportamiento humano inteligente dentro de las máquinas descubriendo las fases que desempeñan en la propia inteligencia humana en tal forma que permitan automatizarse. (Dick, 2019)

La IA se basa en instruir a encomendar los procesos frecuentes que ayuden en las empresas y trabajadores para ahorrar, agilizar trabajos demandados por altos consumos de tiempo. (Apd, 2021)

Mediante estas citas podemos definir que la inteligencia artificial pretende poder automatizar y resolver tareas cotidianas en algunos casos complejas que desarrollan

los seres humanos todo en caso de agilizar los procesos que realiza los seres humanos en sus trabajos diarios.

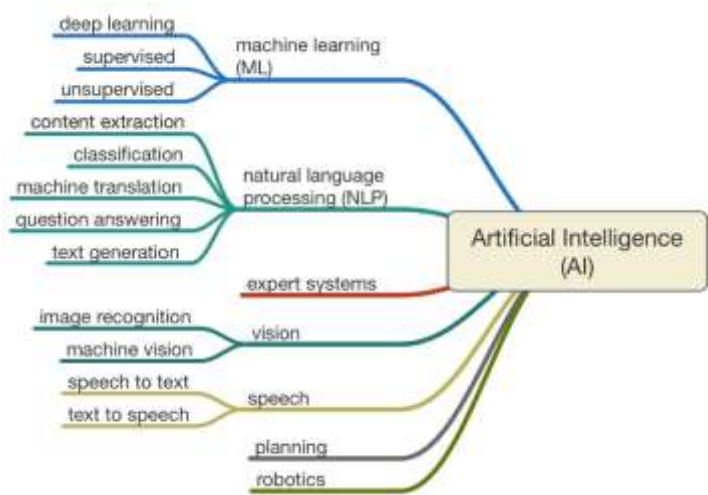


Figura 21. Ramas de la IA. Tomada de la investigación elaborado por: Chethan Kumar GN

2.2.12.1 Orígenes de Fundamentación de la IA

La tecnológica y la programación informática avanzan simultáneamente por ello es importante tener en cuenta el impacto, auge y evolución que ha tenido la Inteligencia Artificial. Mediante esta herramienta se han realizado varios aportes y avances tecnológicos en la vida real, a partir de ello nacieron nuevas ramas las cuales son el lenguaje natural y la percepción.

En el apartado de (Geographic, 2020) se encuentra la siguiente cronología sobre la Inteligencia Artificial

Tabla 23. Cronología de la Inteligencia Artificial

Año	Detalle
1854	Una lógica matemática , el matemático George Boole en este año por primera vez argumento que el razonamiento lógico sería posible estructurarse de forma que resolviera sistema de ecuaciones.
1921	Idea de un robot , en la obra de teatro R.U.R de Karek Apek usa el término “robot” el cual en muchas lenguas esclavas de acuerdo con su etimología significa “trabajo duro”

- 1936 Concepto de algoritmo**, Alan Turing fue considerado el padre de la computación moderna público un artículo introduciendo el concepto de algoritmo el cual constaba de números computables.
- 1941 Z3**, creada por Konrad Zuse siendo la primera computadora automática y programable.
- Leyes de la robótica**, las cuales son 3 y nacieron de un relato denominado “circulo vicioso” escrito por Isaac Asimov.
- 1950 Diferenciación hombre-maquina**, Alan Turing plantea un ensayo que llevo por título Computing Machinery an intelligence el que fue llamado el Test de Turing.
- 1956 Nacimiento del término Inteligencia Artificial**, durante una conferencia del informático John McCarthy indica el termino de Inteligencia artificial la cual fue considerada el germen de la disciplina.
- 1957 Se imito a una mente**, diseñada por primera vez una red neuronal artificial por Frank Rosenblat.
- 1966 Se da voz a las computadoras mediante ELIZA**, siendo así el primer chatbot del mundo creado por Joseph Weizenbaum, incorporando el nlp.
- 1969 Perceptrones**, redactado por el cofundador del MIT Marvin Minsky.
- 1996 Deep Blue**, la cual fue creada por IBM una supercomputadora.
- 1979 Cart de Stanford**, uno de los primeros vehículos que recorrió con éxito en una pista de obstáculos de manera automática
- 2005 Maquinas más inteligentes que el hombre**, Raymond Kurzweil usando la ley de Moore logro predecir que las maquinas llegarían a un nivel de inteligencia mayor en 2029 y para el 2045 superarían la inteligencia de las personas.
-
- 2012 Deep learning el verdadero poder**, se crea un superordenador el cual está capacitado de aprender a través de YouTube a reconocer caras, cuerpos humanos y gatos.

2014 Test de Turing superado por una IA, un Bot capaz de engañar a 30 de los 150 jueces a los que fue sometido durante el Test de Turing haciendo que creyeran que hablaban con un niño de 13 años.

2015 AlphaGo, primera máquina que le gano a un jugador profesional de Go.

Elaborado por: Vera Vera Juan Andrés Fuente: Geographic (2020)

Según Berzal, (2016) delimita lo que se ha generado con inteligencia artificial a lo largo de los años

Tabla 24. Historia de la Inteligencia Artificial

Año	Detalle
2017	Libratus , desarrollado por la Universidad Carnegie Mellon el algoritmo que venció a 4 de los mejores jugadores profesionales de póquer.
2018	En sectores importantes del tejido productivo se asienta la inteligencia artificial.
2019	La utilización de soluciones informáticas basadas en inteligencia artificial se vio con mayor intensidad.
2020	Dado a la situación de la pandemia los avances de la inteligencia artificial son
2021	significativos en el ámbito de la salud y se da el crecimiento del uso del “Low Code”.

Elaborado por: Vera Vera Juan Andrés Fuente: Geographic (2020)

2.2.12.2 Aprendizaje Automático (Machine Learning)

Según Los avances en ML se han logrado desarrollar tecnológicas inteligentes con capacidades de aprendizaje y adquisición de conocimientos mediante la interacción humana, para así agilizar los procesos en la vida laboral y personal relacionados con el ser humano y su actividad diaria. Muchas de estas tecnológicas se han centrado en el mercado electrónico mundial. (Janiesch, C., Zschech, P. & Heinrich, K. , 2021)

El aprendizaje automático es una rama de la ciencia de datos donde corresponde al desarrollo de inteligencia artificial, esta tecnología brinda la facilidad de realizar tareas para agilizar ciertos procesos que son realizados por seres humanos.

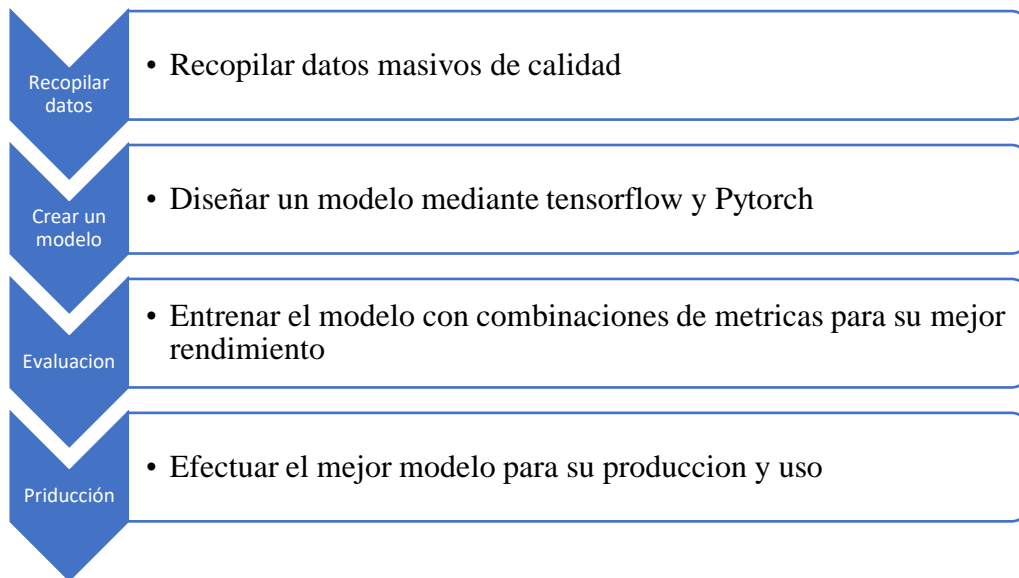


Figura 22. Pasos para crear un modelo de Machine Learning. Elaborado por: Vera Vera Juan Andrés

2.2.12.3 Ciclo de vida del Machine Learning

Esto comprende a la iteración entre el proceso a la mejora de datos, evaluaciones del modelo, que rondan como un ciclo vital en el desarrollo de un modelo de Aprendizaje automático. Siempre se comprende de datos en calidad debido a que se entrena y comprende con sus resultados con esta misma recopilación de datos, es muy importante que esta calidad de datos interfiere en el proceso de iteración de datos. (Hofesmann, 2021)

2.2.13 Librerías que se usan en Machine Learning

2.2.13.1 Pandas

La librería de pandas es usada en programación de datos de ciencia, básicamente su función principal radica en manejar y manipular los datos antes del procesamiento del modelo de Machine Learning a desarrollar. (Onofa, 2021)

2.2.13.2 Numpy

Es un paquete que se usa para el procesamiento de matrices, proporciona entornos multidimensionales y herramientas de trabajo de alta calidad en matrices es muy utilizada para la programación de ciencia de datos en Python. (González, 2020)

2.2.13.3 Scipy

Es un paquete de librería formada de la librería Numpy es de muy utilidad debido ya que espacio de trabajos, como lo es en el campo del algebra lineal, temas de Fourier y finalmente matrices de optimización. (de los Santos, 2018)

2.2.13.4 Matplotlib

Esta librería comprende la parte de graficos estadísticos, en los cuales incluye la manipulación de datos mediante codificación corta, en los cuales se puede apreciar la calidad de los modelos desarrollados. (de los Santos, 2018)

2.2.13.5 Tensorflow

Es una librería del paquete de Python, desarrollado por Google, comprende en la función de realizar cálculos numéricos, mediante datos, esta librería comprende una contradicción debido a que se codifica para desarrollar grafos de operaciones matemáticas entre matrices multidimensionales. (Heras, 2020)

2.2.13.6 Aprendizaje Profundo

El aprendizaje profundo es muy elemental en la ciencia de datos, comprende de modelos predictivos y estadísticos, siendo una subcategoría del Machine Learning, se trata en redes neuronales mejorar cosas como visión artificial, NLP y el reconocimiento de voz. (Banafa, 2016)

2.2.14 Tipos de Aprendizaje

En la presente ilustración se muestran los tipos de aprendizajes en el Machine Learning.

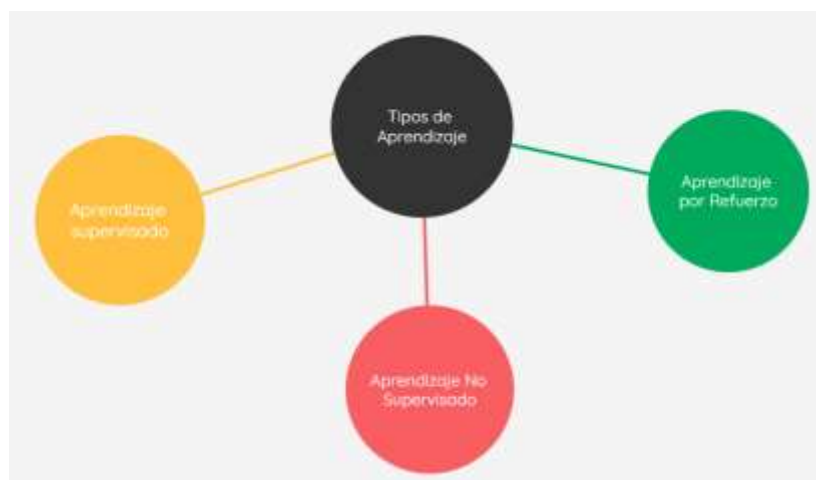


Figura 23. Tipos de Aprendizaje. Elaborado por: Vera Vera Juan Andrés

2.2.14.1 Aprendizaje Supervisado

El aprendizaje supervisado se basa mediante datos de entrenamiento obtener el mejor resultado entre un mapeo de capas de entrada y salida mediante una función en el cual las X son datos de entrada y la Y es una variable de salida esta última puede ser cuantitativa para una regresión, o cualitativa si es un modelo de clasificación. (Alvaro, 2018)

2.2.14.2 Aprendizaje No Supervisado

El aprendizaje no supervisado no consta de dos variables si no de una sola X, su función se basa en la estructura u organización de datos para aprender más de ellos. (Alvaro, 2018)

2.2.14.3 Aprendizaje Por Refuerzo

El aprendizaje por refuerzo se maneja mediante tuplas, se tiene problemas en conjuntos de aprendizajes no supervisados en los cuales el modelo debe aprender de errores para mapear situaciones y así poder lograr definir un resultado optimo en un ambiente dinámico e incierto. (Alvaro, 2018)

2.2.15 Algoritmos de Aprendizaje Supervisado

2.2.15.1 Regresión Lineal

La regresión Lineal es parte del machine learning considerada como una ecuación paramétrica, dando a entender que antes de analizar los datos se definen cuantos factores se van a necesitar. Dando a entender que en base a estos parámetros se efectuara los datos recopilados para así tener la mínima referencia de error. (Heras J. M., 2020)

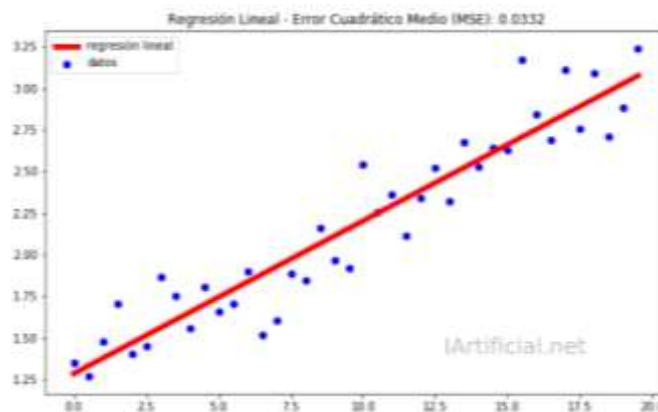


Figura 24. Tomada de la investigación Regresión Lineal. Elaborado por: José Martínez Heras

2.2.15.2 Regresión Polinomial

La regresión polinomial se basa en una regresión lineal que utiliza predictores adicionales, en los cuales producto de este se llega a tener una potencia. La estructura se basa en la siguiente ecuación:

$$x = [a + bx + cx^2 + \dots + Nx^n]$$

2.2.15.3 Regresión de Soporte Vectorial

Se basa en resolver problemas mediante gráficas multidimensionales en los cuales se la considera como un vector de clasificación. Según (Rodrigo, 2017) Los límites de separación lineales generados en el espacio aumentado se convierten en límites de separación no lineales al proyectarlos en el espacio original.

2.2.16 Pasos para la creación de un modelo de Machine Learning

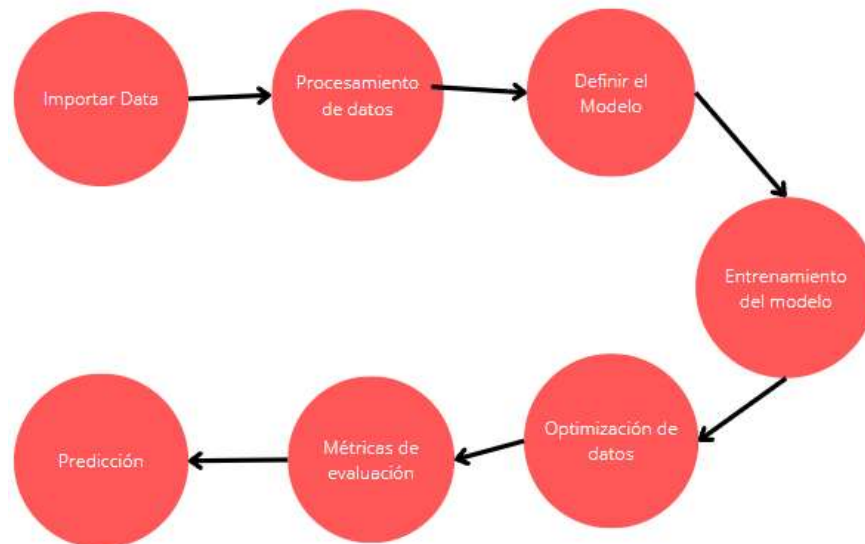


Figura 25. Pasos para creación de un modelo. Elaborado por Vera Vera Juan Andrés

2.2.16.1 Datasets

Se conoce como el conjunto de datos que se posee para el entrenamiento de un modelo de Machine Learning, dependiendo de la data que necesitemos normalmente el portal web de Kaggle dataset, SQuAD datasets, además de proveer la data este portal web tiene modelos de Machine Learning ya entrenados con esos mismos datos para dar una explicación mejor de cómo realizar un modelo.

2.2.17 Natural Language Processing (NLP)

El procesamiento del lenguaje natural comenzó desde la década de 1950 específicamente en 1958 por John McCarthy en el cual desarrollo el lenguaje de programación Lisp. Posteriormente en 1964 se lanzó ELIZA fue el diseño de un algoritmo mediante comentarios y respuestas parecido a un Chatbot la respuesta era de tipo mecanografiado, en el cual cumplía la función de imitar a un especialista psiquiatra en el cual utilizan técnicas de reflexión. (IT, recluit, 2020)

El NLP es una rama de la inteligencia artificial que tiene como función procesar datos de lenguaje una vez este los entienda para así dar una respuesta mediante una asignación pre configurada. Siendo un acontecimiento importante es la herramienta que une a las maquinas con los humanos mediante el procesamiento de lenguaje natural. (Alias, G., & Cassanelli, R, 2019)

El NLP es una herramienta de la inteligencia artificial en la cual comprende el funcionamiento en que los ordenadores desarrollen un procesamiento y análisis del lenguaje humano en los cuales puede realizar características como lo son: traducciones, análisis de sentimientos, reconocimiento de voz, reconocimiento de textos e identificaciones, segmentación de temas y recopilación de noticias. (Egea, 2019)

(Mexico, 2021) Indica que los sistemas de aprendizaje automático almacenan palabras y las maneras en que éstas se unen.

El procesamiento de lenguaje natural (NLP) en la actualidad se ha complementado con la herramienta de machine learning aplicando métodos en los cuales se puedan desarrollar modelos efectivos para el uso en automatización de tareas diarias en las empresas.



Figura 26. Clasificación de ramas de NLP. Elaborado por Vera Vera Juan Andrés

2.2.17.1 Librerías utilizadas en NLP

Son librerías utilizadas para el tratamiento de la datasets para la creación de un modelo de Machine Learning orientado al NLP.

2.2.17.2 Nltk

Es la una herramienta en Python con relación a trabajar y analizar el lenguaje humano, su interfaz se basa en usar más de cincuenta cuerpos léxicos además de recursos tales como WordNet, junto a librerías de clasificación, procesamiento tokenizacion, parseo y razonamiento semántico. (Camacho-Álvarez, M. Á., & Navarro-Álvarez, E., 2020)

2.2.17.3 Sklearn

Es una librería de Python usada en la rama de machine learning para el análisis de datos, surge de las librerías Numpy, SciPy y matplotlib, sus características comprenden de en realizar una fácil sintaxis y variedades de técnicas para la implementación del aprendizaje automático. (Heras J. M., 2020)

En esta librería se pueden usar técnicas de preprocesamiento, métricas, selección de modelos, selección clasificación, agrupación entre otras técnicas que se pueden utilizar para el beneficiar en el entrenamiento del modelo.

2.2.17.4 Pre-Procesamiento

Son pasos en los cuales se preparan previamente los datos, para que en su lógica el modelo pueda utilizarlo como análisis, y predicciones. Se comprende de varios índices para realizar esta herramienta en la creación de un modelo de ML orientado en NLP. (Khanna, 2021)

2.2.17.5 Lista de pasos para el preprocesamiento

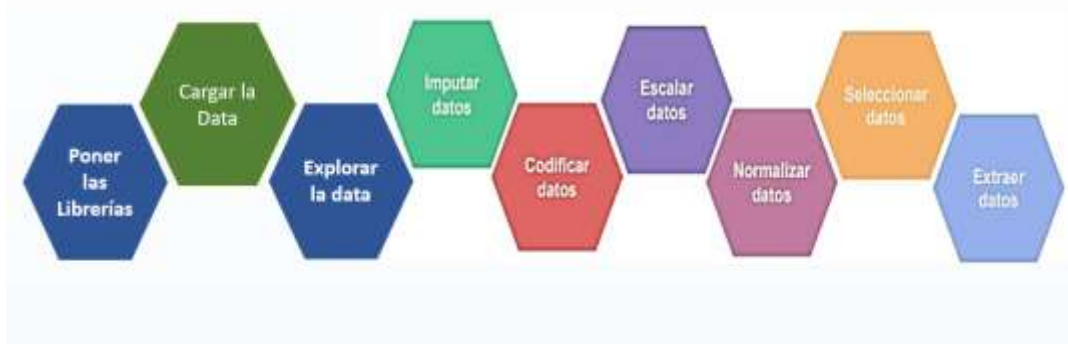


Figura 27. Pasos para el preprocesamiento de datos de un modelo. elaborado por: Vera Vera Juan Andrés

2.2.17.6 Tokenización

La tokenización, es utilizada para dividir el texto ya se en palabra o en oración, para que se pueda trabajar con textos coherentes y más legibles de entendimiento, este el primer paso considerado para estructurar datos, que tienden a ser más fáciles de analizar. (Jablonski, 2021)

(Jonathan, 1992) Menciona que La identificación de palabras sigue siendo un problema de léxico en la PNL china. Al igual que en el caso de las palabras inglesas, sobre todo de los modismos y los compuestos, la fuente de dificultad tiene que ver con la ausencia de identificadores entre los tokens.

Esto recalca lo importante que es el paso de la tokenización al realizar un modelo NLP debido a que los idiomas de diferentes países en el mundo tienen sus propios modismos y expresiones naturales de comunicación.

2.2.17.7 Stop Words

Se utilizan para eliminar palabras, que no son tan relevantes dentro de la información importante que se quiere utilizar para realizar el entrenamiento del modelo, en lo cual dará mayor énfasis en lo importante de la información, cabe recalcar que la eliminación de las palabras normalmente no causa una consecuencia mala. Los modelos de ML lo utilizan reducir la aglomeración de datos y tardar menos tiempo en desarrollar estos modelos puesto que son menos datos de procesar validar y entrenar. (Khanna, 2021)

2.2.17.8 Lematización

La lematización consiste en guardar en juntar un grupo de palabras en su variedad de lema de decir las según el diccionario, para todas agruparlas y tomarlas como una variable única que esta sería la transformación a su lema original o forma original, como es expresada en el diccionario según el idioma, es muy usada en la fase de preprocesamiento de algoritmos de machine learning (ML) enfocados al procesamiento del lenguaje Natural (NLP). (Fernández, 2019)

2.2.17.9 Normalización de palabras

Este proceso se encarga de realizar darle al texto una perspectiva entendible para el usuario en el cual se toma en cuenta la lematización, separación de palabras, corrección de faltas y errores ortográfico, normalmente se usa la conversión con el uso de un diccionario. (Hamdy, 2021)

2.2.17.10 Eliminación de signos de puntuación

En este caso los signos de puntuación no tienen ninguna validez en la comprensión del texto al momento de realizar un modelo NLP al contrario afectan a la hora de realizar el aprendizaje automático en el modelo, resulta ser de gran ayuda en eliminar estas operaciones lingüísticas para el modelo. (Hamdy, 2021)

2.2.17.11 Transformación de símbolos especiales

En este proceso se estudia los emojis para sustituirlos por representaciones de textos que sean entendibles para el desarrollo del modelo, realizando esto se puede delimitar lo que expresa el autor del texto mediante la transformación de estos emojis. (Hamdy, 2021)

2.2.17.12 Bolsa de palabras

Es uno de los procesos más efectivos para la representación de textos considerándose la más efectiva y usada en función de recuperar texto, en su aplicación en los respectivos modelos se destacan características tales como: estructura de la entrada de un texto, párrafos, capítulos oraciones y el formato en el que esta digitalizado. (Díaz, 2019)

2.2.18 Modelos NLP

Son modelos de aprendizaje de aprendizaje automático, basados en Machine Learning (ML), funcionan normalmente mediante predicciones dentro de frases, textos, oraciones basándose en un contexto de palabras previamente cargadas y entrenadas en el mismo. Se basa en descripciones probabilísticas del lenguaje.



Figura 28. Fases de creación de Modelos NLP. Elaborado por: Márquez (2021)

2.2.18.1 Words Embeddings

Representan un enfoque en la semántica de distribución de las palabras o textos como vectores reales, este enfoque resulta muy útil debido a que se presenta agrupamientos útiles de manera semántica sintáctica con características de similitud. (García, 2018)

2.2.18.2 Meta Embeddings

Se basa en juntar la información que se obtiene por la incrustación de elementos generada por palabras o textos mediante diferentes tipos de métodos y fuentes informativas procesar el entrenamiento. (García, 2018)

2.2.18.3 Word2vec (W2V)

Es un modelo de NLP de aprendizaje mediante redes neuronales con características vectoriales continuas, en las cuales se comprenden de una data de alta calidad, además de

tener miles y millones de palabras, el modelo W2V normalmente presentan similitudes semánticas. (Díaz, 2019)

Según (García, 2018) Las capas internas de la red neuronal codifican la representación de la palabra objetivo, es decir, los word embeddings.

Según la investigación de (Rong, 2016) demuestra que el modelo Word2vec tienen gran variedad de utilidad con respecto a significados semánticos en modelos NLP. Delimita que falta materiales didactas de aprendizajes con respecto a modelos de incrustación de textos o palabras en detalles, esto impide que se pueda comprender mecanismos de desarrollo además materiales de redes neuronales con características de NLP.

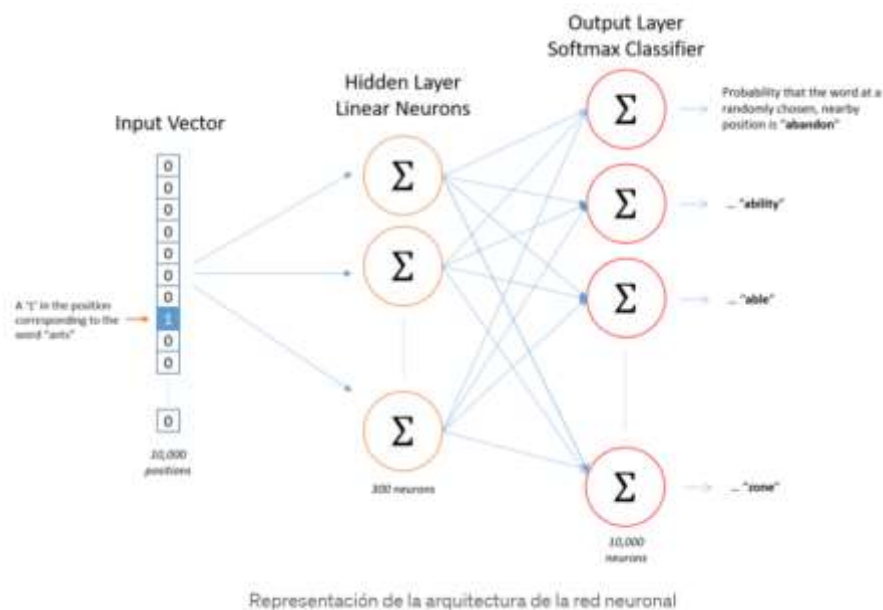


Figura 29. Arquitectura de red neuronal del modelo Word2vec. información tomada de la investigación Elaborado por: Gonzalo Ruiz Villa

2.2.18.4 Global Vectors for Word Representation (GloVe)

Es un modelo basado en conteo en la cual a diferencia de Word2vec crea una matriz grande en volumen permitiendo almacenar la concurrencia entre cualidades lingüísticas tales como textos y palabras. Entonces, cada una de las palabras que se cuenta n veces se visualizará estará en ciertas partes del texto. Su propósito de entrenamiento es aprender datos de vectores de manera que la multiplicación escalar entre las palabras sea equivalente al logaritmo probabilista de coocurrencia entre palabras. (García, 2018)

2.2.18.5 FasText

FastText es una mejora del modelo Word2Vec. En este modelo las palabras son tomadas en cuenta como la suma de su estructuración llamados ngrams. Este vector representara la adición de sus ngramas. (García, 2018)

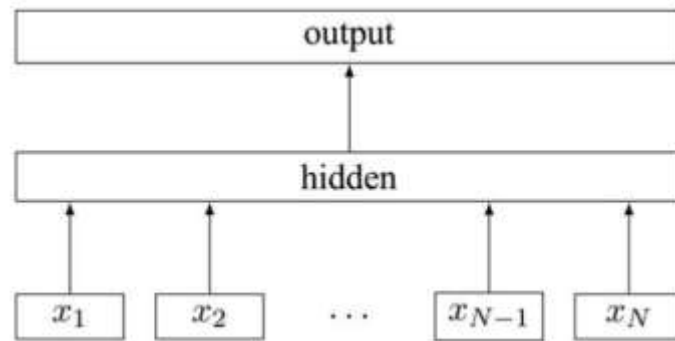


Figura 30. Arquitectura del modelo Fastext. información tomada de la investigación

2.2.18.6 Transformer

Según (Kulshrestha, 2020) menciona que el método transformer NLP es una arquitectura reciente en la cual puede realizar tareas desde una secuencia hasta otra secuencia en tanto maneje dependencias de extenso alcance con facilidad. Se basa por completo en la autoatención para calcular las representaciones de su entrada y salida SIN utilizar RNN alineadas con la secuencia o la convolución.

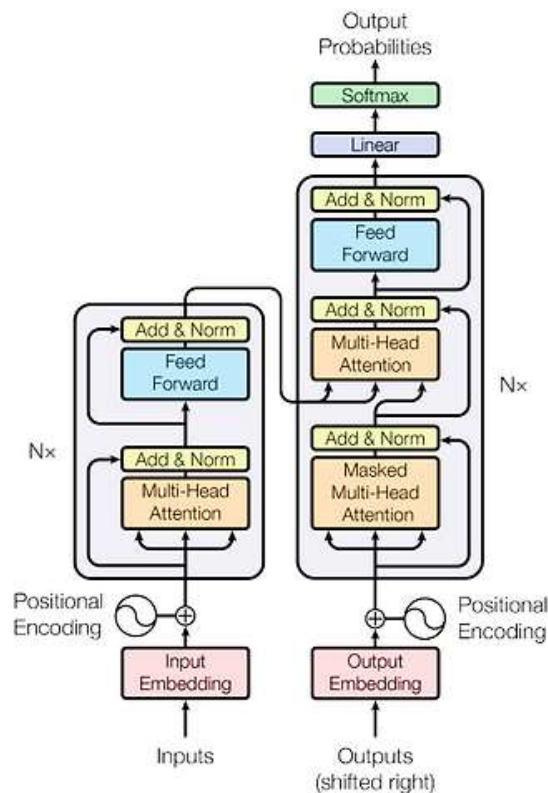


Figura 31. Arquitectura del modelo Transformer. Información tomada del Información tomada de Google. Elaborado por el autor

2.2.18.7 Blazingtext

Es un modelo usado para realizar optimizaciones referentes al modelo Word2vec en lo cual clasifica textos, de manera más ágil y eficiente.

2.2.18.8 Embeddings from Language Models (ELMo)

Según menciona que las incrustaciones de palabras deben incorporar tanto las características a nivel de palabra como la semántica contextual (Ghelani, 2019). El Elmo asigna datos de entrada variados para cada capa de palabra en la red neuronal, en el cual realiza previamente un análisis para definir a cada una de las incrustaciones, para darle más robustez al modelo pequeñas incrustaciones diferentes.

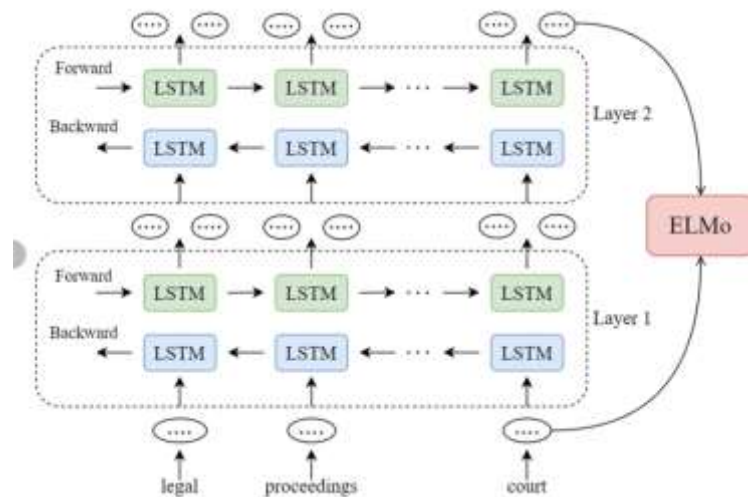


Figura 32. Arquitectura del modelo ELMo. Información tomada del Información tomada de Google. Elaborado por el autor

2.2.18.9 Bert

Es un modelo preentrenado basado en la arquitectura Transformer, en los cuales muestra resultados de vanguardia en variedades de tareas de aprendizaje automático.

Se menciona que el objetivo de esta técnica BERT es mejorar es la utilidad del entrenamiento bidireccional de Transformer, Esto contrasta ciertas maneras de comprensión de textos escritos en las cuales se analizaban de izquierda a derecha o combinaban técnicas para el análisis de derecha a izquierda. (Horev, 2018)

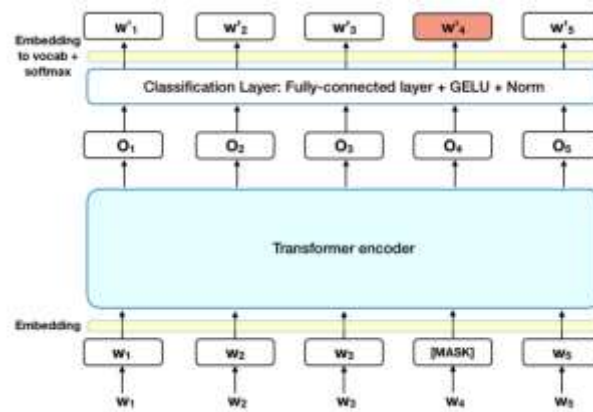


Figura 33. Arquitectura del modelo Bert. Información tomada de Información tomada de Google. Elaborado por el autor

2.2.19 Técnicas NLP

2.2.19.1 Redes Neuronales Recurrentes – RNN

Según (Juan, R. Q., & Mario, C. M., 2011) Son redes neuronales que tienen conexiones formando un ciclo directo. Estos métodos demandan más dinamismo, pero exigen más recursos que redes de capa densa o directas.

Esto alega que las redes neuronales recurrentes son métodos de entrenamiento de modelos con más variedades, pero consumen más recursos de la computadora, es ideal para clasificación de varios objetos dentro de una red neuronal.

2.2.19.2 Multilabel o multi etiqueta

La clasificación multietiqueta se trata de asignaciones de una muestra de datos hacia varias salidas de texto que se representan mediante etiquetas. Este conjunto de etiquetas pertenece al mismo grupo de etiquetas, en las cuales las etiquetas no presencian consistencias algunas. Como objetivo este modelo pretende realizar clasificaciones para muestras que no se vistas. (Wu,Gao, & ,Jiao, 2019)

2.2.19.3 Support Virtual Machine para multilabel o multietiqueta

Este modelo permite definir la clasificación lineal en base a un hiperplano que trabaja como división de un par de tipos de clases.

Según (Vicente, 2012) menciona que el funcionamiento de este modelo en general es el siguiente el SVM busca un hiperplano que divida a las clases y mediante estas alternativas se pueda tener en versión de hiperplano un máximo margen en que las clases se puedan separar.

2.3 Marco Legal

2.3.1 Constitución de la República del Ecuador

El Presente trabajo comprende de un marco Legal el cual será delimitado en el **Anexo 1** del presente trabajo de investigación.

Capítulo III

Metodología

3.1 Propuesta tecnológica

En la siguiente sección se demostrará acorde con las metodologías empleadas a lo largo de la vigente investigación, el desarrollo de la aplicación y la arquitectura del modelo de Machine Learning empleado usados con cada una de las predicciones del mismo. Por consiguiente, se demuestran las poblaciones y muestreo con respecto a personas que poseen dislalia la metodología de investigación fue mixta debido a que se empleó, revisiones bibliográficas, encuestas y entrevistas a expertos en áreas de desarrollo de software aplicativos, expertos en inteligencia artificial y expertos especialistas en terapias del lenguaje.

3.1.2 Diseño de la Investigación tecnológica

En el presente trabajo de investigación se desarrolla una metodología mixta comprendiendo así una metodología cuali-cuantitativa y metodología bibliográfica, el metodillo bibliográfico será utilizado porque se requiere identificar información sobre la dislalia de documentaciones de alto impacto, como también de las tecnologías y herramientas que se usan para el desarrollo de modelos de IA y desarrollo de software.

Debido a que se investigara en el campo cualitativo por medio de entrevistas a profesionales de terapias de lenguaje debido a que se desconoce este campo médico en los cuales puedan brindar información acerca del proceso de diagnóstico y terapias que llevan a cabo a sus pacientes, para así comprender estos campos, y así poder realizar el prototipo de aplicación del cual está enfocado la presente investigación. Por otro lado, se considera entrevistas a profesionales con experiencia en desarrollo de softwares, porque se requiere obtener información acerca tecnologías de desarrollo de aplicaciones móviles, para complementar con la investigación la comparación de las tecnologías de desarrollo de aplicativos. Mientras que la metodología cuantitativa se utilizara porque se necesita obtener información justificada de desarrollar un prototipo de aplicación móvil para personas con familiares que tengan dislalia como apoyo y así, refuercen en sus casas mediante ejercicios de practicas una vez terminadas sus sesiones de terapias con el terapeuta.

3.2 Metodologías de Investigación

Son estrategias que se utilizan para determinar el rumbo de la investigación y que tenga como resultado un efecto positivo y exitoso, por lo que la elección de las metodologías de investigaciones suelen ser decisiones difíciles debido a su alta importancia dentro del proyecto o trabajo de investigación.

3.2.1 Metodología Mixta

Se basa en integrar de metodologías de recopilación y análisis de datos con referencia al problema que se está planteando dentro del marco de investigación en las cuales se presentan datos cuantitativos como también cualitativos.



Figura 34. Información obtenida de Metodología de la investigación (p.535). Elaborado por Hernández, Fernández y Baptista

3.2.2 Metodología para la revisión Bibliográfica

La metodología Bibliográfica se realiza a cualquier trabajo de investigación, en lo cual recalca la información de mayor importancia y relevancia acerca de este, para así mantener originalidad del tema de investigación. (Gómez-Luna et al., 2014)

3.2.3 Metodología Cualitativa

La metodología cualitativa se trata de un proceso sistemático en el cual se busca indagar en técnicas especializadas en recaudar datos con una perspectiva profunda, por lo que se están buscando informaciones de que las personas piensan y sienten con respecto a un fenómeno de investigación. (Suárez L. C. ; Escudero C. S; Margarita., 2018)

3.2.4 Metodología Cuantitativa

La metodología cuantitativa se basa en la recolección de datos de forma numérica, con indagación estadística, en el cual se pretenden estudiar y analizar las variables con

entorno a la investigación refiriéndose a la hipótesis y variables numéricas de la misma. (Arias, 2021)

3.2.5 Técnicas de Investigación

A lo largo de la investigación el proceso de técnicas de investigación se realizó mediante recopilación de datos se realizó mediante encuestas y entrevistas, las encuestas se llevaron a cabo mediante la plataforma de Google Forms para facilidad de obtención y recopilación de los datos cuantitativos, y con respecto a las entrevistas, fue mixta hubo cooperación por parte de profesionales en terapias de lenguaje con apoyo a entrevistas presenciales, como también se llevó a cabo una segunda entrevista por medio de la plataforma de Google Forms. En cambio, a los expertos en desarrollo de software las entrevistas se realizaron por medio de la ya nombrada plataforma de Google.

3.2.5.1 Descripción del proceso metodológico

3.2.5.2 Población

De acuerdo con el estudio realizado 2621 niños, correspondiente a estudiantes de sistema educativo que poseen dificultad de aprendizaje y que son guiados por pedagogos de acuerdo con datos de ministerio educación. (Telégrafo, 2022), los cuales serán considerada como la población objetivo de este trabajo.

3.2.5.3 Muestreo no probabilístico por conveniencia

El muestreo no probabilístico por conveniencia se aplica cuando se encuentra casos que deseen ser incluidos dentro de la investigación. (Otzen & Manterola, 2017)

Para el presente trabajo de investigación no se realizó un muestro probabilístico, por la dificultad de acceso a personas que tienen familiares con dislalia debido a que estos, procuran mantener en privado esta situación médica, por lo cual para el presente estudio se aplicara muestreo no probabilístico por conveniencia.

3.2.5.4 Encuestas

La encuesta es una técnica que se usa de manera de cuestionario, está relacionado a personas para poder obtener información sobre sus opiniones, percepciones y comportamientos ante un fenómeno. Las encuestas pueden tener datos salientes tanto cualitativos como numéricos, siempre manteniendo un orden lógico para su implementación y comprensión del fenómeno que se está llegando a investigar, analizar y estudiar. (Gonzalez, 2020)

Las encuestas se realizaron por medio de la plataforma Google Forms, debido a la dificultad de encontrar los casos de personas que tengan discapacidades en el habla especificando así la dislalia, además de la falta de colaboración por parte de las personas para el desarrollo en la metodología cuantitativa para el presente trabajo de investigación.

La encuesta fue dirigida, a padres y madres de niños que residen en la ciudad de Guayaquil, sin embargo, se logró la cooperación de otras provincias de Ecuador, y por lo cual se logró recopilar un total de 36 con la información facilitada por personas que tienen hijos con dislalia.

3.2.5.5 Entrevistas

La entrevista es una técnica altamente conocida dentro de la metodología cualitativa, como indica (Meneses, J., & Rodríguez-Gómez, D., 2011) la intención de la entrevista es obtener información de un profesional o usuario dentro del campo determinado al cual se está investigando.

Con el fin de llevar a cabo datos cualitativos en el presente trabajo de investigación se realizó, entrevistas a 3 profesionales en el campo de ingeniería de software, para obtener información sobre el desarrollo de aplicaciones móviles, por otro lado, por parte de tecnologías de la inteligencia artificial se tomó información de las entrevistas pasadas realizados por investigaciones parte del FCI 010-2021, y por último en el campo externo de la tecnología como lo es la dislalia se entrevistó a 3 profesionales de terapias de lenguaje 2 se llevaron a cabo de manera presencial y 1 entrevista fue mediante la plataforma de Google Forms, esta última se realizó por este medio debido a la poca disponibilidad que presentaba la profesional.

3.2.5.6 Problemas en recolección de datos

Durante la investigación vigente, hubo problemas en recopilación de datos en la sección de encuestados y entrevistas a especialistas en terapeutas del lenguaje debido a falta de cooperación, negación a brindar información sobre la metodología de sus terapias, para brindar información para el proceso de desarrollo del software.

3.3 Análisis de las Encuestas

Pregunta 1. ¿Cuál es la edad del niño, niña o joven que presenta dislalia?

Tabla 25. Edad.

Opciones de Respuestas	Encuestados	Porcentajes
------------------------	-------------	-------------

De 3 a 6 años	9	25,0%
De 6 a 9 años	12	33,3%
De 9 a 12 años	5	13,9%
De 12 a 15 años	3	8,3%
De 15 a 18 años	3	8,3%
De 18 a 20 años	4	11,1%
Total	36	100%

Información obtenida de Google Forms de la investigación directa. Elaborado por Vera Vera Juan Andrés

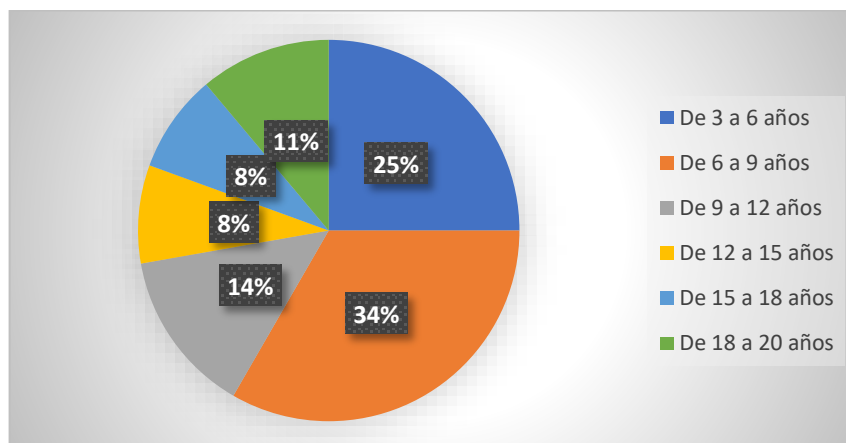


Figura 35. Edad de los encuestados. Información obtenida de Google Forms de la investigación directa. Elaborado por Vera Vera Juan Andrés

Se identificó que el 58,3% de los encuestados poseen niños con dislalia entre edades inferiores a 9 años, el 14% pertenece a edades entre 9 a 12 años y por último, el 8% de los encuestados pertenecen al rango de edades de 12 a 15 años, otro 8% de encuestados entre 15 y 18 años y por último el 11% pertenecen a rango de 18 a 20 años de edad

Pregunta 2. ¿Cuál es el género del niño, niña o joven que presenta dislalia?

Tabla 26. Género.

Opciones de respuestas	Encuestados	Porcentajes
Masculino	25	69,4%
Femenino	11	30,6%
Total	36	100%

Información obtenida de Google Forms de la investigación directa. Elaborado por Vera Vera Juan Andrés

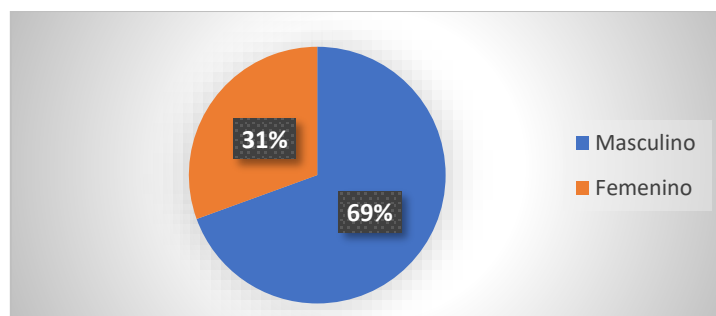


Figura 36. Género de los encuestados. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Analizando los resultados del gráfico de la pregunta 2 de la encuesta notamos que los pacientes con complicaciones que presentan dislalia, pertenecen en un 69,4% al género masculino, mientras que el otro 30,6% corresponde a género femenino.

Pregunta 3. ¿En qué país reside el niño, niña o joven con dislalia?

Tabla 27. País de residencia.

Opciones de Respuestas	Encuestados	Porcentajes
Argentina	1	2,8%
Cuba	1	2,8%
Ecuador	32	88,9%
México	2	5,6%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

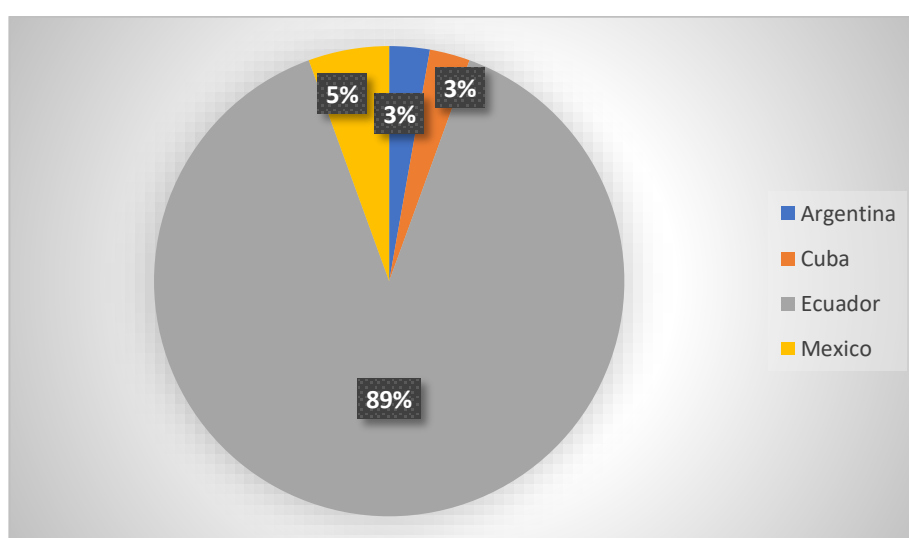


Figura 37. País de residencia de los encuestados. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Según los resultados de la encuesta el lugar donde residen las personas que poseen dislalia se presentan con un 88,9% pertenecen al país de Ecuador, un 5,6% pertenecen a México y por último tenemos dos valores de 5,6% que pertenecen entre Argentina y Cuba.

Pregunta 4. ¿En qué provincia reside el niño, niña o joven con dislalia?

Tabla 28. Provincia de Residencia

Opciones de		
Respuestas	Encuestados	Porcentajes
Bolívar	1	2,8%
Bs As Burzaco	1	2,8%
Ciudad México	3	8,3%
Guayas	24	66,7%
La Habana	1	2,8%
Manabí	3	8,3%
Pichincha	2	5,6%
Santo Domingo	1	2,8%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

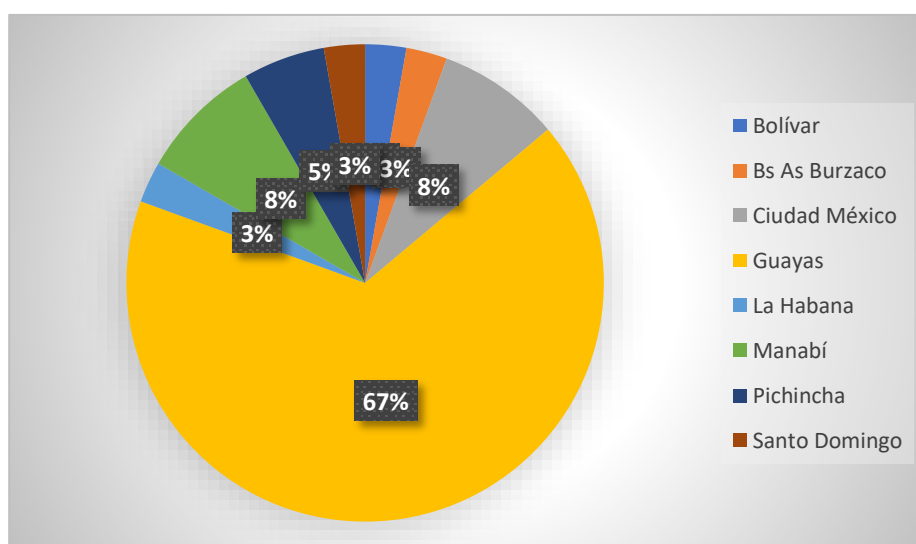


Figura 38. Género Provincia de residencia de los encuestados. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Como se puede observar en la pregunta 4, la provincia de residencia de los encuestados que poseen dislalia, pertenece en un 66,7% a la Provincia del Guayas, un 8,3% a

la Provincia Manabita y Ciudad de México, un 11,2% equivalente entre las Provincias de Bolívar, Santo Domingo, La Habana y Burzaco y por ultimo un 6% correspondiente a la provincia de Pichincha.

Pregunta 5. ¿En qué ciudad o cantón reside el niño, niña o joven con dislalia?

Tabla 29. Ciudad o Cantón de Residencia.

Opciones de Respuestas	Encuestados	Porcentajes
Aurora	1	2,8%
Balzar	1	2,8%
Burzaco	1	2,8%
Ciudad de México	2	5,6%
Duran	1	2,8%
Guayaquil	20	55,6%
Manta	3	8,3%
Milagro	1	2,8%
Miramar	1	2,8%
Quito	3	8,3%
San Miguel	1	2,8%
Santo Domingo	1	2,8%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

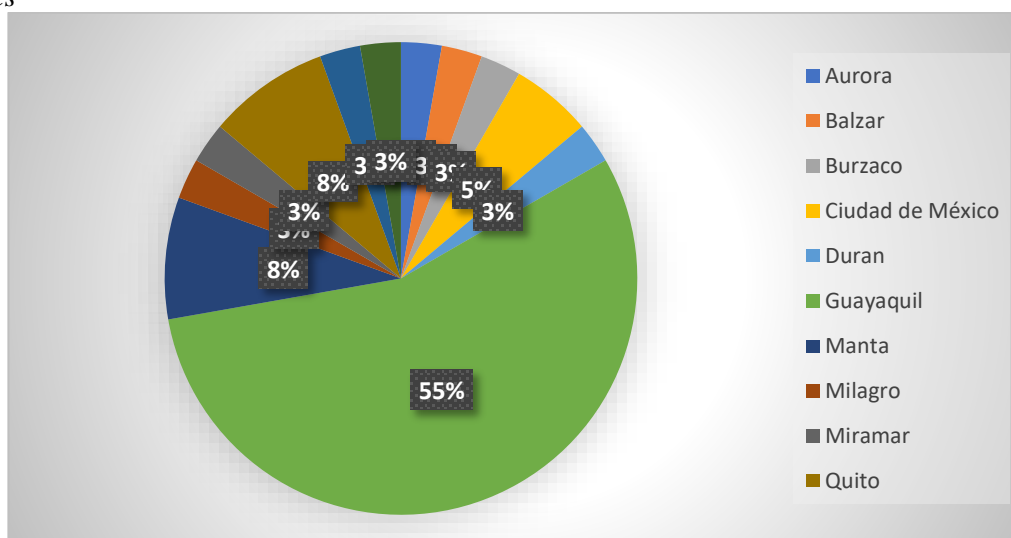


Figura 39. Ciudad de residencia de los encuestados. Información obtenida de formulario de Google y de la investigación directa. Elaborado por Vera Vera Juan Andrés

De la muestra obtenida los casos que mas se presentaron de la dislalia fue ne la Ciudad de Guayaquil obteniendo el 55,6%, por consiguiente la ciudad de Manta y Quito corresponden a los porcentajes de 8,3%, un porcentaje equivalente de 22,4% entre Aurora, Balzar, Burzaco, Duran, Milagro, San Miguel, Santo Domingo y por ultimo un 5,6% perteneciente a Ciudad de México.

Pregunta 6. Mencione usted. ¿cuál de los tipos de dificultades del habla presenta su familiar (niño o niña, joven)?

Tabla 30. Tipos de Dislalia.

Opciones de respuestas	Encuestados	Porcentajes
Dislalia Funcional	14	31,8%
Dislalia Evolutiva	8	18,2%
Dislalia Orgánica	1	2,3%
Dislalia Auditiva	4	9,1%
Poseo familiar con dislalia, pero no conozco el tipo específico	9	20,5%
No poseo esa información (No Aplica)	8	18,2%
Total	44	100,0%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

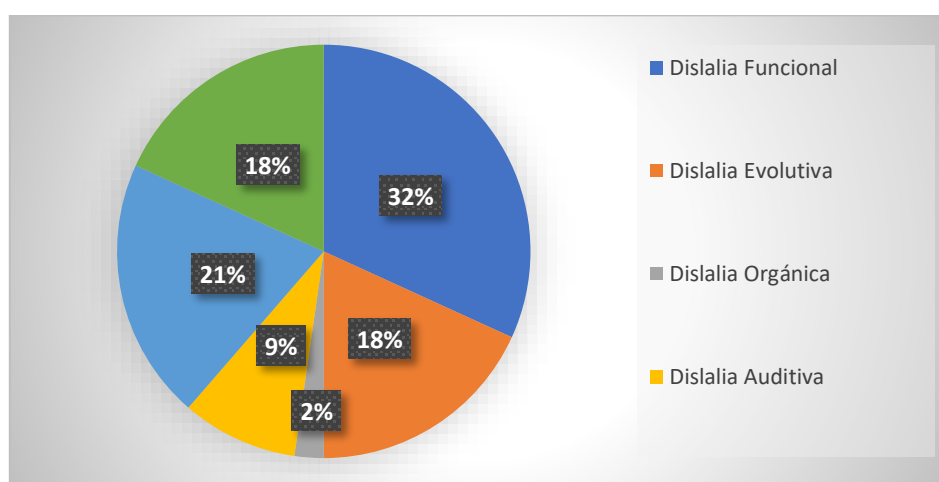


Figura 40. Tipos de dislalia que presentan los encuestados. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Según los datos de la encuesta realizada datan que el 31,8% presentaron dislalia funcional, el 18,2% a dislalia evolutiva, un 2,3% a dislalia orgánica, un 9,1% son de

dislalia auditiva y por último un 21% tienen familiares con dislalia, pero no saben cuál es el tipo que presenta.

Pregunta 7. ¿Considera usted que las terapias de lenguaje presenciales en los centros médicos o unidades de terapias se complementa con ejercicios de terapias de lenguajes enviadas a practicar en casa del niño, niña o joven con dislalia?

Tabla 31. Terapias presenciales y ejercicios en casa.

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	31	86,1%
Parcialmente de Acuerdo	5	13,9%
Ni en acuerdo ni en desacuerdo	0	0,0%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

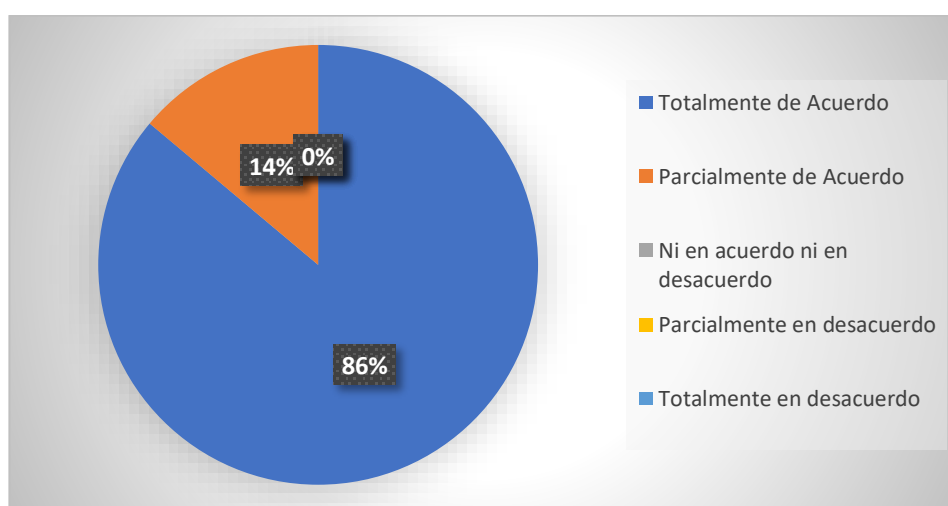


Figura 41. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Del resultado de las encuestas obtenidas de la pregunta 7, se puede apreciar que el 86,1% de los encuestados está totalmente de acuerdo y un 13,9% esta parcialmente de acuerdo.

Pregunta 8. ¿Cuánto tiempo en minutos diarios su familiar (niño, niña o joven) práctica los ejercicios enviados a casa?

Tabla 32. Tiempo de duración de los ejercicios.

Opciones de Respuestas	Encuestados	Porcentajes
De 0 a 5 minutos	4	11,1%
De 5 a 10 minutos	0	0,0%
De 10 a 15 minutos	1	2,8%
De 15 a 20 minutos	0	0,0%
De 20 a 25 minutos	9	25,0%
De 25 a 30 minutos	22	61,1%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

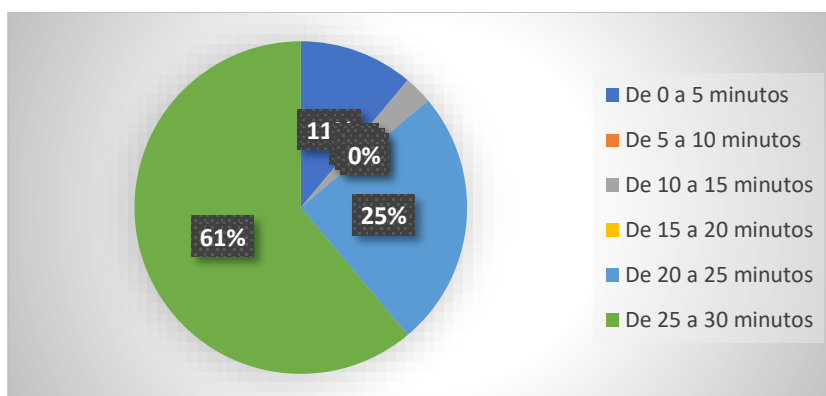


Figura 42. Tiempo de practica de ejercicios de los encuestados. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

De los datos de la encuesta realizada en la pregunta 8, se puede observar que los tiempos diarios que se más se realizan prácticas de ejercicios en hogares son de 25 a 30 minutos que pertenecen al 61,1%, seguidamente de un 25% que pertenecen de 20 a 25 minutos, un 2,8% practican menos hasta 15 minutos y por último un 11,1% practican de 5 minutos a nada.

Pregunta 9. ¿Cuántos días en la semana su familiar (niño, niña o joven) práctica los ejercicios enviados a casa?

Tabla 33. Días de practica de los ejercicios.

Opciones de Respuestas	Encuestados	Porcentajes
De 0 a 1 Días	2	5,6%
De 1 a 2 Días	1	2,8%
De 2 a 3 Días	1	2,8%
De 3 a 4 Días	18	50,0%

De 4 a 5 Días	3	8,3%
De 5 a 6 Días	5	13,9%
De 6 a 7 Días	6	16,7%
Total	36	100%

Información obtenida de formulario de Google y de la investigación directa. Elaborado por Vera Vera Juan Andrés

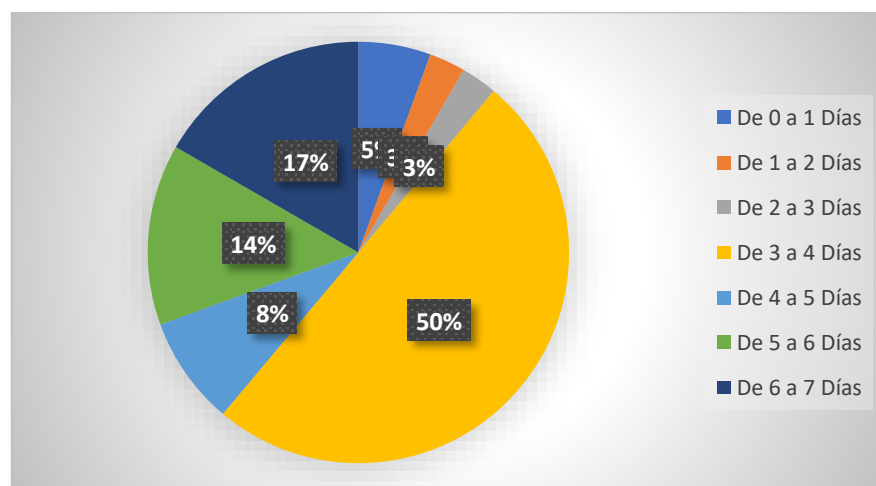


Figura 43. Práctica en días semanales de los encuestados. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

De los resultados de la encuesta en la pregunta 9, se puede observar que el 50% de los encuestados practican ejercicios de 3 a 4 días, el 16,7% respondieron que practican de 6 a 7 días, un 13,9% practican de 5 a 6 días, un 8% practican de 4 a 5 días y por último tenemos un porcentaje sumado del 12% practican 3 días o menos en la semana.

10. ¿Referente a los ejercicios de terapias de lenguajes enviadas a practicar en casa, que nivel de dificultad de comprensión y reproducción del ejercicio enfrenta el familiar (papá, mamá o tutor) al tener que reproducir el ejercicio en casa?

Tabla 34. Nivel de comprensión de los ejercicios

Opciones de Respuestas	Encuestados	Porcentajes
Muy fácil comprensión y reproducción	9	25,0%
Fácil comprensión y reproducción	17	47,2%
Ni fácil ni difícil	10	27,8%
Medianamente difícil de comprender y reproducir	0	0,0%
Muy difícil comprensión reproducción	0	0,0%

Total	36	100%
--------------	-----------	-------------

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

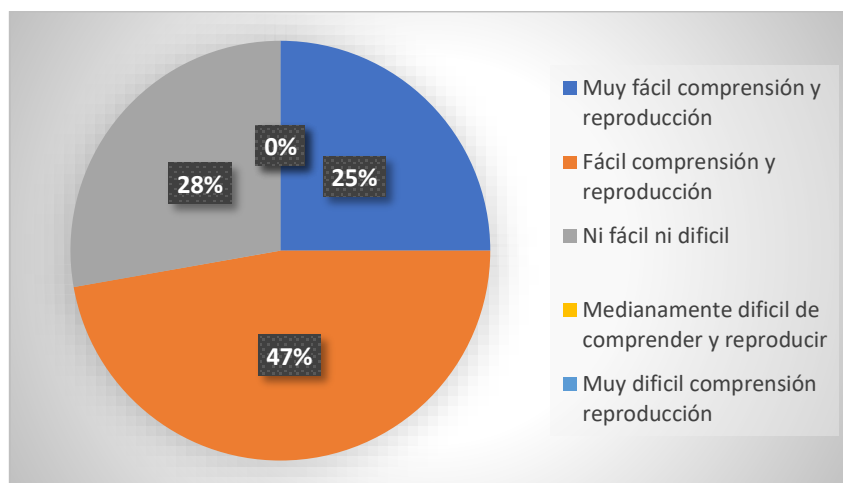


Figura 44. Dificultad de manejo de tecnologías. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Según los resultados obtenidos de la encuesta, muestra que el 47,2% de los encuestados respondieron que tienen fácil comprensión y reproducción de ejercicios en casa, un 27,8% respondieron ni fácil ni difícil y el 25% respondieron que es de muy fácil comprensión y reproducción.

Pregunta 11. ¿Sabía usted que existen en la actualidad aplicaciones móviles (APP) para el apoyo en las terapias de lenguaje para niños

Tabla 35. Conocimiento de aplicaciones referentes a la dislalia.

Opciones de Respuestas	Encuestados	Porcentajes
Si tengo mucho conocimiento	7	19,4%
Tengo poco conocimiento	14	38,9%
No tengo conocimiento	15	41,7%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

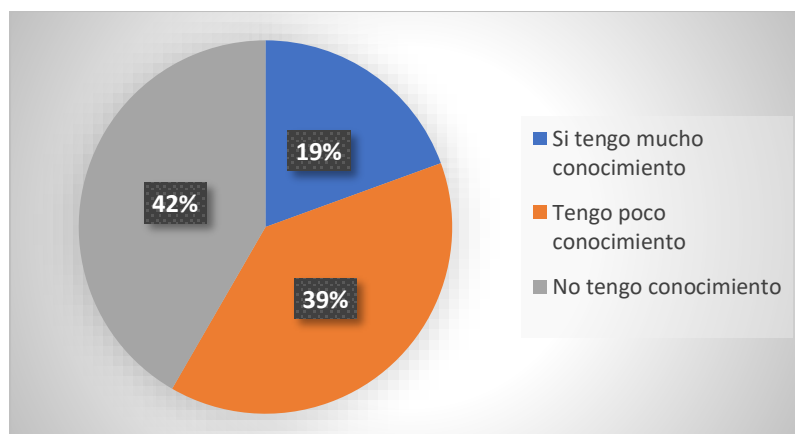


Figura 45. Conocimiento de tecnologías de apoyo. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Según los datos recolectados de las encuestas, se demuestra que se tiene pocos conocimientos de aplicaciones que complementen ejercicios de terapias de lenguaje acerca de la dislalia, marcando así un 41,7% que no tienen conocimiento, un 38,9% de poco conocimiento, un 19,4% que si tienen conocimiento de las aplicaciones.

Pregunta 12. ¿Considera usted que una aplicación móvil (APP) intuitiva de fácil uso sería un apoyo para las familias cuyos niños o jóvenes necesitan realizar ejercicios de lenguaje de comunicación en sus hogares?

Tabla 36. Aplicaciones Intuitivas

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	21	58,3%
Parcialmente de Acuerdo	13	36,1%
Ni en acuerdo ni en desacuerdo	2	5,6%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

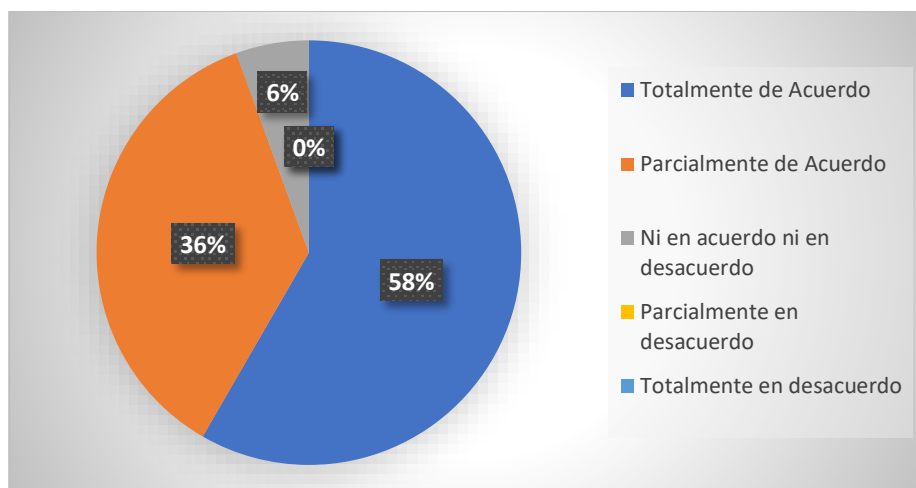


Figura 46. Aplicación intuitiva de manejo. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Analizando los resultados de la encuesta, se aprecia que un 58,3% de los encuestados está de acuerdo con aplicaciones que sean intuitivas, un 36,1% respondieron que están parcialmente de acuerdo y por último un 5,6% que no están de acuerdo ni en desacuerdo.

Pregunta 13. ¿Estaría usted de acuerdo en que se desarrollen nuevas aplicaciones móviles de acceso gratuito para apoyo a las terapias de niños o jóvenes con dislalia?

Tabla 37. Desarrollo de aplicaciones para dislalia.

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	30	83,3%
Parcialmente de Acuerdo	5	13,9%
Ni en acuerdo ni en desacuerdo	1	2,8%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

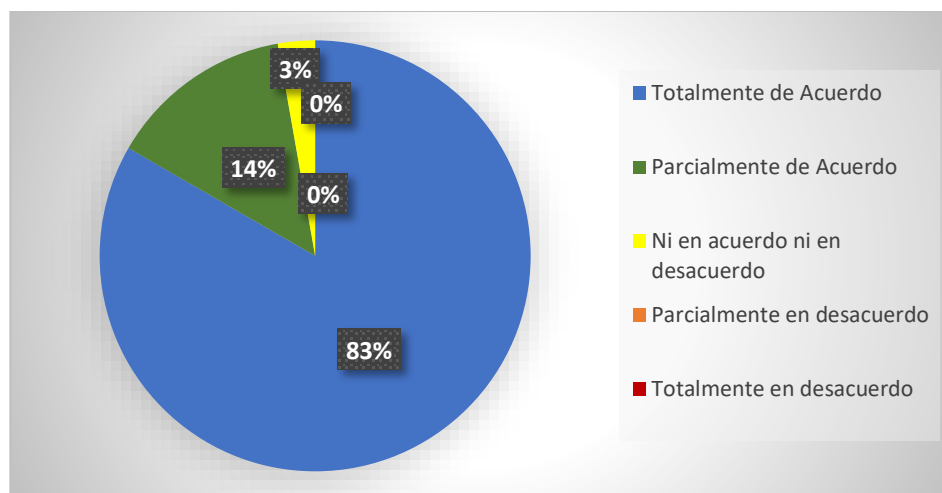


Figura 47. Opinión sobre desarrollo de la aplicación. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Se demuestra en los resultados de la encuesta, que el porcentaje de aceptación porque se realicen nuevas aplicaciones móviles que traten la dislalia es muy alto alcanzando un 83,3% que están totalmente de acuerdo y un 13,9% que están parcialmente de acuerdo y por último un 2,8% que no están ni acuerdo ni en desacuerdo.

Pregunta 14. ¿Estima usted que el desarrollo de una aplicación móvil (APP) para el sistema operativo móvil de Android sea recomendable por su facilidad en uso para los niños o jóvenes?

Tabla 38. Aplicaciones dirigidas a Android

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	21	58,3%
Parcialmente de Acuerdo	14	38,9%
Ni en acuerdo ni en desacuerdo	1	2,8%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

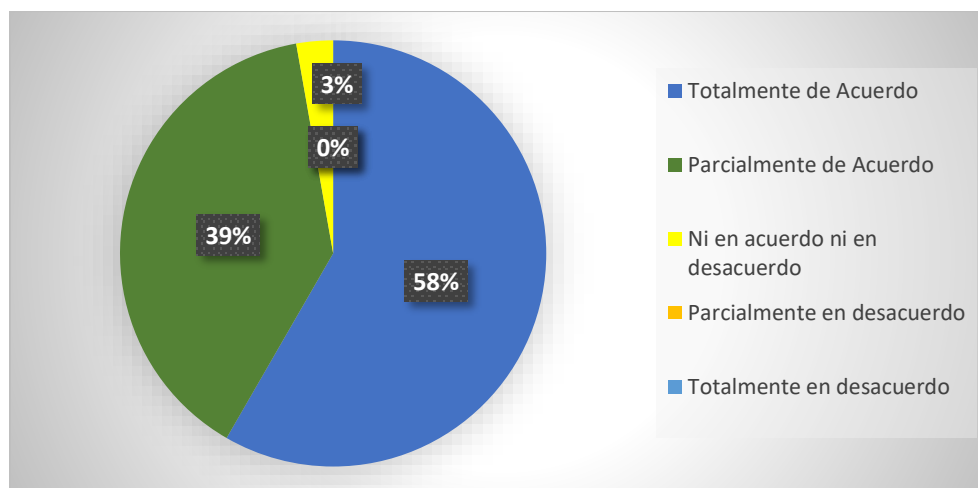


Figura 48. Desarrollo de aplicaciones para Android. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

De la información recopilada de las encuestas, se demuestra que el 58,3% de los encuestados prefieren el desarrollo de aplicaciones gratuitas para el sistema Operativo Móvil de Android teniendo alta aceptación, un 38,9% está parcialmente de acuerdo y un 2,8% tiene una opinión neutral.

15. ¿Estima usted que su familiar (niño, niña o joven) con dislalia cuentan con las habilidades motrices para interactuar fácilmente con una aplicación móvil (APP) para el apoyo en sus sesiones de terapias?

Tabla 39. Uso fácil de aplicaciones móviles.

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	20	55,6%
Parcialmente de Acuerdo	13	36,1%
Ni en acuerdo ni en desacuerdo	3	8,3%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

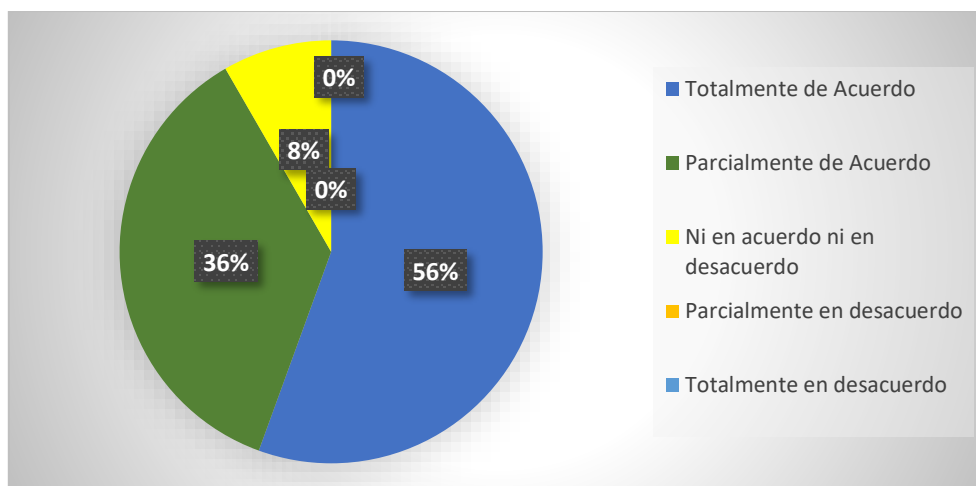


Figura 49. Capacidad para el uso de tecnologías. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

De acuerdo al resultado de las encuestas realizadas, se observa que las personas que poseen dislalia no tienen una complejidad motriz el cual les impida que no puedan usar aplicaciones móviles para su aprendizaje en ejercicios y uso de la tecnología, remarcando un 55,6% en totalmente de acuerdo, un 36,1% en parcialmente de acuerdo y un 8,3% respondieron que no están de acuerdo ni en desacuerdo.

Pregunta 16. ¿Considera usted que sería de utilidad el poder contar con una aplicación móvil (APP) en la cual el especialista terapéutico pueda preconfigurar un conjunto de ejercicios recomendados para practicarse en casa de tal forma que estos se habiliten para su uso?

Tabla 40. Uso de aplicaciones móviles.

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	28	77,8%
Parcialmente de Acuerdo	7	19,4%
Ni en acuerdo ni en desacuerdo	1	2,8%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

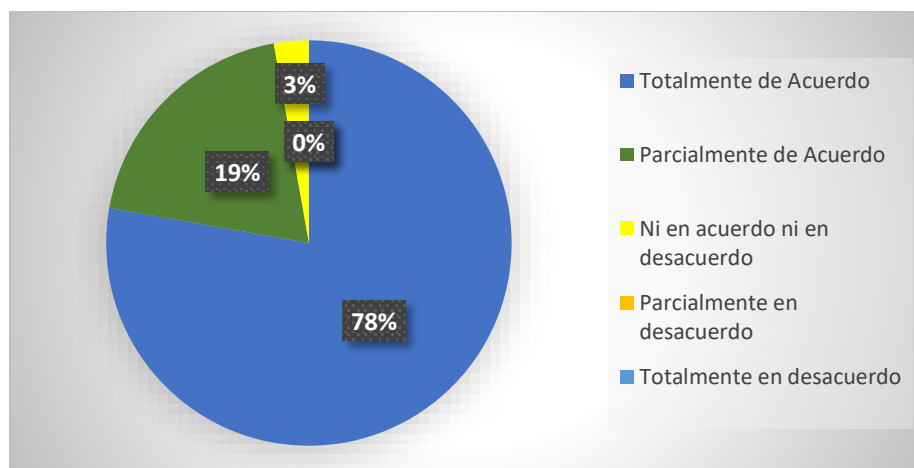


Figura 50. Tecnología para el apoyo de la dislalia. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

El resultado de la encuesta, muestra que el 77,8% están totalmente de acuerdo que el especialista, les cite y defina los ejercicios a practicar en casa desde una aplicación móvil y un 19,4% están parcialmente de acuerdo y por ultimo

Pregunta 17. ¿Considera usted que sería de utilidad que el padre o la madre puedan contar con una aplicación móvil (APP) en la cual en casa puedan elegir el conjunto de ejercicios recomendados por el especialista para practicarse en casa?

Tabla 41. Modalidad de configuración de los ejercicios.

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	27	75,0%
Parcialmente de Acuerdo	8	22,2%
Ni en acuerdo ni en desacuerdo	1	2,8%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

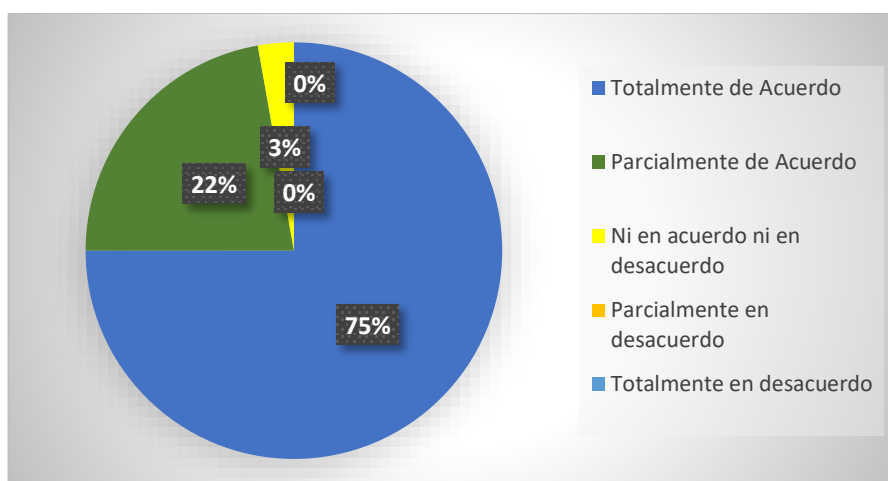


Figura 51. Uso de tecnología para los Padres y Madres. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

Según el resultado de las encuestas, demuestra que un 75% de los encuestados respondieron que están totalmente de acuerdo, un 22% parcialmente de acuerdo y un 3% no están en acuerdo ni en desacuerdo en que se incluya interactividad en peticiones de ejercicios a practicar para una rápida navegación mediante una aplicación.

Pregunta 18. ¿Considera usted que la App guarde un progreso diario que el niño o joven va alcanzado en el tiempo?

Tabla 42. Consideración de guardar progreso.

Opciones de Respuestas	Encuestados	Porcentajes
Totalmente de Acuerdo	29	80,6%
Parcialmente de Acuerdo	6	16,7%
Ni en acuerdo ni en desacuerdo	1	2,8%
Parcialmente en desacuerdo	0	0,0%
Totalmente en desacuerdo	0	0,0%
Total	36	100%

Información obtenida de formulario de Google y de la investigación directa. Elaborado por Vera Vera Juan Andrés

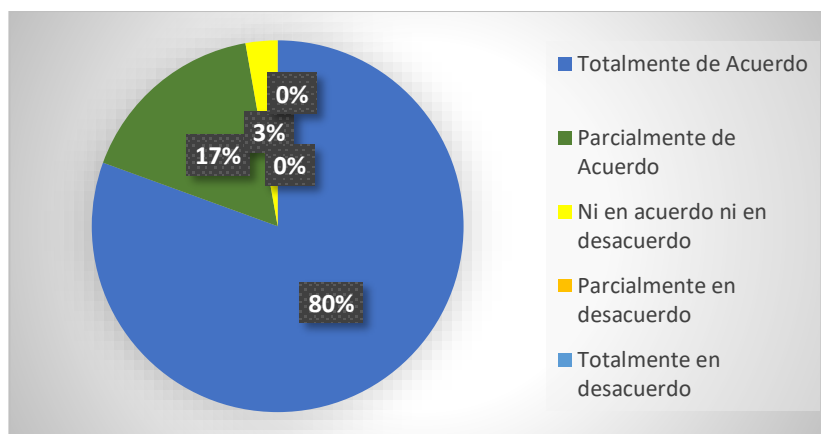


Figura 52. Información de Proceso del paciente. Información obtenida de Google Forms y de la investigación directa. Elaborado por Vera Vera Juan Andrés

El resultado de las encuestas demuestra que el 80,6% de los encuestados prefieren una aplicación que tenga guardado un registro de actividades, para monitorear la realización de ejercicios dentro de una aplicación móvil. El 16,7% de los encuestados están parcialmente de acuerdo y el 2,8% respondieron que están ni en acuerdo ni en desacuerdo.

3.4 Resumen de la entrevista

Considerando en primer punto las entrevistas a profesionales en desarrollo de software se entrevistó a los profesionales: Ingeniero. Cristian Calero, Juan Heredia Ingeniero en Sistemas y José Luzarraga delimitaron que los framework de Flutter y Android Studio son para el desarrollo de aplicaciones a nivel empresarial, o con fines comerciales que serán en un futuro implementadas para que se sustenten en el mercado, por otro lado mostraron pocos conocimientos sobre el framework de desarrollo de aplicaciones denominado Kivy, pero dieron una aceptación del software para el diseño de aplicaciones rápidas y con interfaces amigables como también si son elaboraciones para prototipos de desarrollo de software.

En cambio, en el ámbito de profesionales de terapias de Lenguaje se entrevistó a las Licenciadas Cecilia Calero, Dominga Villamar y Verónica Vélez, en los cuales de estas entrevistas tuvieron como objetivo, el poder profundizar en el ámbito de la dislalia, apreciaciones por parte de la tecnología para apoyo en las terapias de lenguaje y modalidades de terapias que realizan para, así poder recaudar información y obtener una iniciativa del contenido que se mostrara en el desarrollo del software. Por ende, se delimitaron por las entrevistas que las terapias presenciales se llevan a cabo tanto de mediante la estimulación

de la fono articulación en lo cual esto engloba, tanto la dificultad que presentan los pacientes para articular frases y palabras mediante fonemas como también lo son la pronunciación de los mismos, su modalidad de proceso de llevar a cabo las terapias son en movimientos de ejercicios vocales en parte de musculatura facial y una vez se entrena los órganos periféricos del habla se proceden a realizar ejercicios de pronunciación de frases mediante fonemas, con ejercicios de articulación mediante repetición de frases y palabras ejercicios además de ejercicios que estimulen nominaciones de objetos a través de imágenes. Los cuales los ejercicios de nominaciones de imágenes relacionados con objetos y/o animales además de repeticiones de palabras mediante audios y completar palabras fueron considerados para el prototipo. Los ejercicios de articulación fueron sugeridos y facilitados por plantillas de palabras con fonemas y nominaciones de imágenes las cuales se mostrarán en el **Anexo 5** de la investigación, además de usar imágenes de plantillas de portales web tales como: <https://palabras-con.org/d/>.

De parte a entrevistas con orientación a profesionales de inteligencia artificial se revisó la data del FCI-010 2021, en las cuales se delimito por parte de la Ing. En Networking y Telecomunicaciones Viviana Vera que los modelos más factibles en uso para sistemas que contengan algoritmos NLP de clasificación de textos, son los modelos mediante los Generative Pre-trained Transformer 3 (GPT3) y redes neuronales recurrentes (RNN) y Fastext por el hecho que estos demandan menos almacenamiento en el campo de implementación dentro del sistema por lo cual se reduce el riesgo de que el sistema sufra bajas de rendimiento por sobrecargas de almacenamiento. Además, que las técnicas de NLP que se desarrollan más son la detección de oraciones, palabras, frases y segmentación de palabras para clasificar y realizar un análisis profundo para incrustar las repuestas dependiendo del modelo que se esté creando.

3.5 Resultados metodológicos

De los resultados entre las metodologías empleadas para la investigación se obtuvieron los siguientes datos importantes los cuales se consideraron para realizar el prototipo de software en general de la presente investigación. Según investigaciones anteriores de FCI 010-2021, citando en específico a la investigación de (Solorzano, 2022), delimito en su investigación que el modelo más efectivo en métricas de calidad fue el modelo de red neuronal de capa densa (DNN), para clasificadores multi-etiqueta el cual este será utilizado para crear el modelo de aprendizaje automático.

De resultados de las entrevistas en el campo médico, se tomó los datos de las siguientes de las preguntas en las cuales las terapistas delimitaron lo siguiente puntos.

- Sus tratamientos y diagnósticos se realizan mediante nominaciones de imágenes, ejercicios de reproducción de audios, y ejercicios de repaso de palabras referentes a estos los fonemas, que el paciente presenta errores de pronunciación.
- La dislalia es tratada por terapistas de acuerdo a cuál o cuáles fonemas presentan sus pacientes dificultades o desordenes de articulación.
- Sus terapias se basan en estimular primero el aparato fonoarticulatorio para así pasar al nivel de articular frases y palabras completas.

De los resultados de entrevistas a profesionales con experiencia en desarrollo de software delimitaron que, Flutter y Android Studio son las herramientas tecnológicas más usadas debido a que presentan mejores resultados en el campo empresarial debido a que presentan compatibilidad y mejores acoplamientos al momento de desarrollar aplicaciones para sistemas completos.

De los resultados de la metodología cuantitativa en base a los encuestados se obtuvieron los siguientes resultados:

- El 75% y el 25% presentan opiniones parciales y neutros referentes a estar de acuerdo de los encuestados desean elegir de forma inteligente, la asignación de ejercicios para así practicar mediante softwares aplicativos.
- El 83% de los encuestados contestaron estar totalmente de acuerdo en desarrollar aplicaciones como complemento de sus terapias para presentar mayor agilidad en progreso en sus tratamientos.

3.6 Creación del modelo de Machine Learning

Para la creación de un modelo de Machine Learning (NLP) siempre se requiere de una dataset, en este requerimiento podrá usarse para la creación de la red neuronal para si el este obtenga aprendizaje y por lo consiguiente pueda efectuar de manera correcta las predicciones. La dataset que se implementó para la creación de la red neuronal fue una data, en la cual se realizaban validaciones mediante procesamiento de lenguaje natural, en el cual usuario pueda realizar interactividad mediante peticiones de los ejercicios, en base al fonema que quiere practicar dentro de la aplicación.

Los datos se almacenan en un archivo de Excel con el respectivo formato xlxs. Almacenado en el ordenador. Debido a que se trabajó en el entorno de Google Colaboratory,

cada vez que se presentaba un trabajo prolongado o se consistía una inactividad en la plantilla del trabajo, se debía cargar la dataset del archivo Excel nuevamente para poder continuar con el desarrollo del modelo de red neuronal.

Entre los verbos que el modelo puede detectar en la variable de entrada se encuentran considerados la siguiente tabla:

Tabla 43. Verbos en forma infinitiva y conjugados que usa el Machine Learning (NLP)

Verbo	el fonema	la letra	Carácter	Símbolo	consonante	la vocal
se quiere	se quiere el fonema	se quiere la letra	se quiere carácter	se quiere símbolo	se quiere consonante	se quiere la vocal
Quiero	quiero el fonema	quiero la letra	quiero carácter	quiero símbolo	quiero consonante	quiero la vocal
queremos	queremos el fonema	queremos la letra	queremos carácter	queremos símbolo	queremos consonante	queremos la vocal
Usa	usa el fonema	usa la letra	usa carácter	usa símbolo	usa consonante	usa la vocal
Usar	usar el fonema	usar la letra	usar carácter	usar símbolo	usar consonante	usar la vocal
Usemos	usemos el fonema	usemos la letra	usemos carácter	usemos símbolo	usemos consonante	usemos la vocal
preséntame	preséntame el fonema	preséntame la letra	preséntame carácter	preséntame símbolo	preséntame consonante	preséntame la vocal
Presenta	presenta el fonema	presenta la letra	presenta carácter	presenta símbolo	presenta consonante	presenta la vocal
mostrar	mostrar el fonema	mostrar la letra	mostrar carácter	mostrar símbolo	mostrar consonante	mostrar la vocal
Muestra	muestra el fonema	muestra la letra	muestra carácter	muestra símbolo	muestra consonante	muestra la vocal
muéstranos	muéstranos el fonema	muéstranos la letra	muéstranos carácter	muéstranos símbolo	muéstranos consonante	muéstranos la vocal
muéstrame	muéstrame el fonema	muéstrame la letra	muéstrame carácter	muéstrame símbolo	muéstrame consonante	muéstrame la vocal
Denos	denos el fonema	denos la letra	denos carácter	denos símbolo	denos consonante	denos la vocal
Danos	danos el fonema	danos la letra	danos carácter	danos símbolo	danos consonante	danos la vocal
Pido	pido el fonema	pido la letra	pido carácter	pido símbolo	pido consonante	pido la vocal
te pido	te pido el fonema	te pido la letra	te pido carácter	te pido símbolo	te pido consonante	te pido la vocal
te pedimos	te pedimos el fonema	te pedimos la letra	te pedimos carácter	te pedimos símbolo	te pedimos consonante	te pedimos la vocal
se pide	se pide el fonema	se pide la letra	se pide carácter	se pide símbolo	se pide consonante	se pide la vocal
haz me el favor de presentar	haz me el favor de presentar el fonema	haz me el favor de presentar la letra	haz me el favor de presentar carácter	haz me el favor de presentar símbolo	haz me el favor de presentar consonante	haz me el favor de presentar la vocal

podrías presentar	podrías presentar el fonema	podrías presentar la letra	podrías presentar carácter	podrías presentar símbolo	podrías presentar consonante	podrías presentar la vocal
puedes presentar	puedes presentar el fonema	puedes presentar la letra	puedes presentar carácter	puedes presentar símbolo	puedes presentar consonante	puedes presentar la vocal
Deseo	deseo el fonema	deseo la letra	deseo carácter	deseo símbolo	deseo consonante	deseo la vocal
recupera	recupera el fonema	recupera la letra	recupera carácter	recupera símbolo	recupera consonante	recupera la vocal
recuperarme	recuperarme el fonema	recuperame la letra	recuperame carácter	recuperame símbolo	recuperame consonante	recuperame la vocal
Repasa	repasa el fonema	repasa la letra	repasa carácter	repasa símbolo	repasa consonante	repasa la vocal
Repasar	repasar el fonema	repasar la letra	repasar carácter	repasar símbolo	repasar consonante	repasar la vocal

Información Tomada de la investigación directa. Elaborado por Vera Vera Juan Andrés

Tabla 44. Modelo de Machine Learning (NLP).

Variable de entrada	Fonema detectado	Variables de salida
x1	por el modelo NP	y1 a y25
Frase Tipeada	Fonemas /m	1 y 0
Frase Tipeada	Fonemas /p	1 y 0
Frase Tipeada	Fonemas /b	1 y 0
Frase Tipeada	Fonemas /d	1 y 0
Frase Tipeada	Fonemas /t	1 y 0
Frase Tipeada	Fonemas /z	1 y 0
Frase Tipeada	Fonemas /s	1 y 0
Frase Tipeada	Fonemas /l	1 y 0
Frase Tipeada	Fonemas /n	1 y 0
Frase Tipeada	Fonemas /r	1 y 0
Frase Tipeada	Fonemas /rr	1 y 0
Frase Tipeada	Fonemas /ch	1 y 0
Frase Tipeada	Fonemas /ll	1 y 0
Frase Tipeada	Fonemas /y	1 y 0
Frase Tipeada	Fonemas /ñ	1 y 0
Frase Tipeada	Fonemas /k	1 y 0
Frase Tipeada	Fonemas /g	1 y 0
Frase Tipeada	Fonemas /j	1 y 0
Frase Tipeada	Fonemas /x	1 y 0
Frase Tipeada	Alveolar	1 y 0
Frase Tipeada	Bilabial	1 y 0
Frase Tipeada	Labiodental	1 y 0
Frase Tipeada	Dental	1 y 0
Frase Tipeada	Palatal	1 y 0
Frase Tipeada	Velar	1 y 0

Información Tomada de la investigación directa. Elaborado por Vera Vera Juan Andrés

3.6.1 Importación de datos

Se importa los datos desde el ordenador, para trasladarlo al entorno de trabajo de Google Colaboratory para el proyecto que se realizara para el presente trabajo.

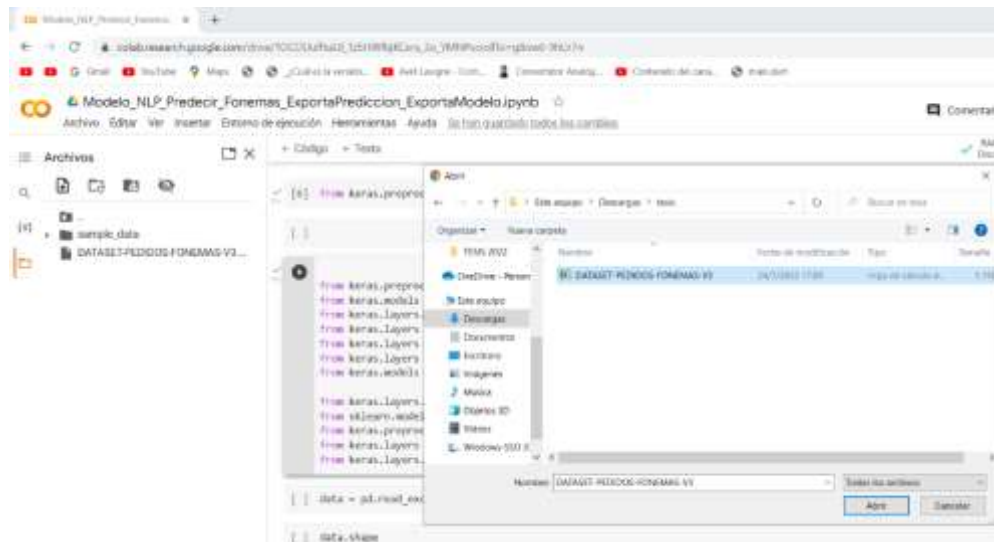


Figura 53. Importación de datos. Elaborado por: Vera Vera Juan Andrés

Se puede observar en la imagen el archivo cargado del dataset al entorno de Google Colaboratory.

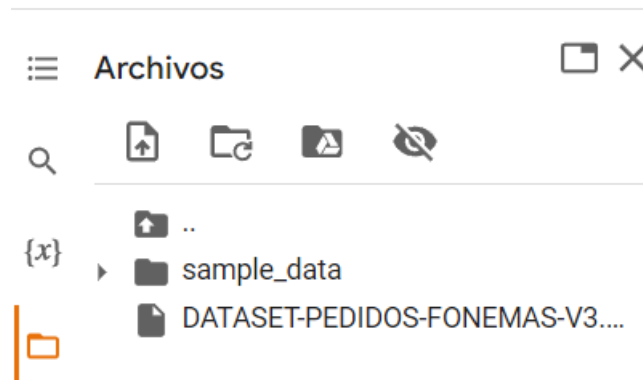


Figura 54. Importación de datos. Elaborado por: Vera Vera Juan Andrés

En la imagen se puede observar la data cargada dentro del entorno de Google Colaboratory, para desarrollar el modelo de red neuronal a usar, el archivo llamado DATASET-PEDIDOS-FONEMAS-V3 el cual tiene la extensión xlsx en otras palabras, un archivo Excel. Por lo tanto, ahora se podrá llamar a los datos dentro de la dataset.

Se debe importar la librería Panda como pd, para agregar una abreviatura al momento de llamar en las líneas de programación a la librería. Una vez realizado este paso realizamos los datos, se convertirán en “dataframes”, por lo consiguiente usamos 2 comandos para la visualización del dataset usando el comando “shape” mostramos cuantas columnas tiene y

que tipo de datos presenta la data, que esta almacenada una vez para luego proceder a mostrarlas en pantallas con el comando “info”.



Figura 55. Observación de datos. Elaborado por: Vera Vera Juan Andrés

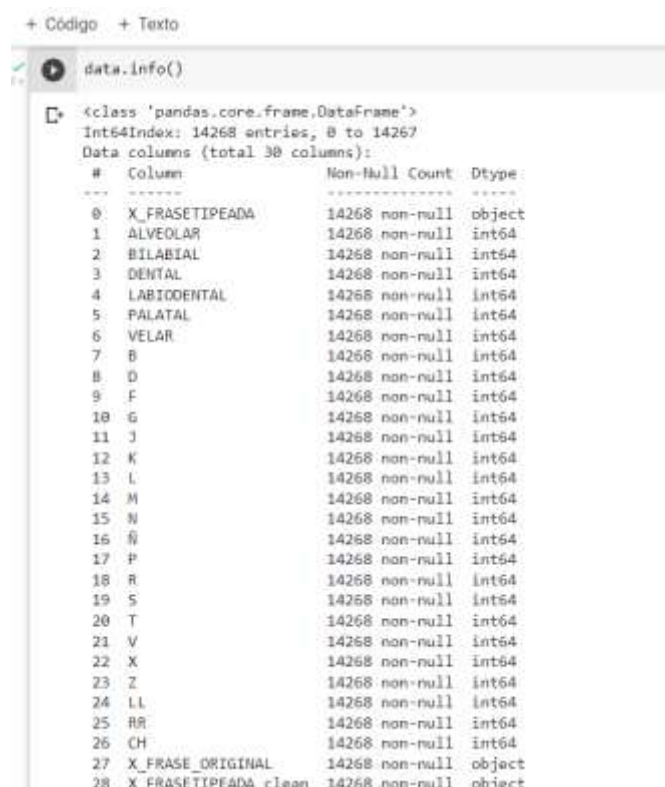


Figura 56. Observación de datos. Elaborado por: Vera Vera Juan Andrés

Como se puede observar el dataset tiene una data de 14268 filas. Posteriormente se puede observar las columnas de la data formada por: X_FRASETIPEADA, ALVEOLAR, BILABIAL, DENTAL, LABIODENTAL, PALATAL, VELAR, B, D, F, G, J, K, L, M, N, Ñ, P, R, S, T, V, X, Z, LL, RR, CH, X_FRASE_ORIGINAL, X_FRASETIPEADA_clean.

3.6.2 Tratamiento de Datos

Una vez que ya se importan las librerías y se visualizan cuantas columnas tiene procedemos con el comando “`isna_shape`” cuantas filas vacías tenemos, y con el comando “`not.null`”, mostramos la data presentada con sus filas y columnas.


```
data[ pd.isna( data["X_FRASETIPEADA"] )].shape
```

```
(0, 28)
```

Figura 57. Tratamiento de datos. Elaborado por: Vera Vera Juan Andrés

Como se puede apreciar en la imagen no presenta filas vacías para luego usar el comando y presentar la data con columnas y filas sin vacíos como se demostrará en la siguiente imagen.

```
data.loc[ data["X_FRASETIPEADA"].notna()]
```

	X_FRASETIPEADA	ALVEOLAR	BILABIAL	DENTAL	LABIODENTAL	PALATAL	VELAR	B	D	F	...	R	S	T	V	X	Z	LL	RR	CH	X_FRASE_ORIGINAL
0	vibrante	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	vibrante
1	vibrante /a	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	vibrante /a
2	/a	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	/a
3	zona central	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	zona central
4	zona central /a	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	zona central /a
...
14263	recuperame consonante palatal /i	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	recuperame consonante palatal /i
14264	recuperame consonante /i	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	recuperame consonante /i
14265	recuperame consonante palatal	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	recuperame consonante palatal
14266	recuperame consonante palatal /ch	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	recuperame consonante palatal /ch
14267	recuperame consonante /ch	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	recuperame consonante /ch

14268 rows x 28 columns

Figura 58. Tratamiento de datos. Elaborado por: Vera Vera Juan Andrés

Posteriormente se realizó una comparación de datos crudos con datos depurados en los tratamientos de datos, por lo tanto, se usaron funciones definidas y se presentaron mediante el comando “head” para mostrar las 20 primeras filas, por último, se presentarán los datos de las X y las Y en forma individual luego de la depuración.

```
[35] data.iloc[:, 1:40]
```

	ALVEOLAR	BILABIAL	DENTAL	LABIODENTAL	PALATAL	VELAR	B	D	F	...	T	V	X	Z	LL	RR	CH	X_FRASE_ORIGINAL	X_FRASETIPEADA_clean	Num_Palabras
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	vibrante	vibrante	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	vibrante /a	vibrante /a	2
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	/a	/a	1
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	zona central	zona central	2
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	zona central /a	zona central /a	3
...
14263	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	recuperame consonante palatal /i	recuperame consonante palatal /i	4
14264	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	recuperame consonante /i	recuperame consonante /i	3
14265	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	recuperame consonante palatal	recuperame consonante palatal	3
14266	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	recuperame consonante palatal /ch	recuperame consonante palatal /ch	4
14267	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	recuperame consonante /ch	recuperame consonante /ch	3

14268 rows x 29 columns

Figura 59. Comparación columnas de texto. Elaborado por: Vera Vera Juan Andrés

df_depurada

	X_FRASETIPEADA_clean	ALVEOLAR	BILABIAL	DENTAL	LABIODENTAL	PALATAL	VELAR	B	D	F	...	P	R	S	T	V	X	Z	LL	RR	CH
0	vibrante	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	vibrante /a	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	/a	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	zona central	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	zona central /a	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
...
14263	recuperame consonante palatal /l	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	1	0	0
14264	recuperame consonante /l	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	1	0	0
14265	recuperame consonante palatal	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
14266	recuperame consonante palatal /ch	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1
14267	recuperame consonante /ch	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1

14268 rows × 27 columns

Figura 60. Comparación columnas de texto. Elaborado por: Vera Vera Juan Andrés

X_ini

	X_FRASETIPEADA_clean
0	vibrante
1	vibrante /a
2	/a
3	zona central
4	zona central /a
...	...
14263	recuperame consonante palatal /l
14264	recuperame consonante /l
14265	recuperame consonante palatal
14266	recuperame consonante palatal /ch
14267	recuperame consonante /ch

14268 rows × 1 columns

Figura 61. Comparación columnas del texto. Elaborado por: Vera Vera Juan Andrés

y

	ALVEOLAR	BILABIAL	DENTAL	LABIODENTAL	PALATAL	VELAR	B	D	F	G	...	P	R	S	T	V	X	Z	LL	RR	CH
7	0	1	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	1	0	0	...	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	1	0	0	...	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	1	0	0	...	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
...
14263	0	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	1	0	0
14264	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	1	0	0
14265	0	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
14266	0	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1
14267	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	1

8816 rows × 26 columns

Figura 62. Comparación columnas. Elaborado por: Vera Vera Juan Andrés

3.6.3 Aplicación de Técnicas

3.6.3.1 StopWords

Se realizó la técnica de frases de parada (StopWords) a la columna de X_FRASETIPEADA, primero importando la librería nltk, y posteriormente se descargó el tokenizador nombrado “Punkt”, para aplicarlo en la columna y posteriormente crear una nueva columna sin StopWords. Por último, se utiliza la función lambda para aplicarlo en la columna correspondiente y así presentar el resultado la técnica, en la columna X_FRASETIPEADA_clean.

X_FRASETIPEADA_clean	
2905	repasar el fonema alveolar
157	se quiere el fonema /g
11113	recuperame simbolo velar
12967	danos consonante palatal /l
6069	queremos caracter /j
...	...
9286	presentame simbolo alveolar /n
8412	repasa caracter alveolar
8727	quiero simbolo palatal
1402	muestrame el fonema alveolar /l
11771	usa consonante alveolar /s

7052 rows × 1 columns

Figura 63. Uso del StopWord. Elaborado por: Vera Vera Juan Andrés

Se consultó en percentiles cuantas palabras diferentes se obtienen en la variable x_train, la cual es correspondiente a los valores de entrada del modelo.

```
count    7052.000000
mean      4.251276
std       1.234792
min       1.000000
25%       4.000000
50%       4.000000
75%       5.000000
85%       5.000000
90%       5.000000
95%       6.000000
max       10.000000
Name: X_FRASETIPEADA_clean, dtype: float64
```

Figura 64. Uso del StopWord. Elaborado por: Vera Vera Juan Andrés

3.6.3.2 Tokenización

La segunda técnica que se utilizó dentro del modelo es la tokenización, y esta se aplicó a la columna X_FRASETIPEADA_clean, que es la columna donde anteriormente se aplicó la técnica de StopWords, para realizar una conversión de lo tipeado en frases a números por lo que a cada frase tipiada la tokenizacion le asigna un numero a cada palabra para su correspondiente procesamiento, debido a que el lenguaje de las maquina lo que entiende son números, este paso es de vital importancia debido a que con esta técnica se podrá llevar a cabo la implementación del modelo a un módulo externo. En este caso es el prototipo de aplicación ya desarrollada.

```
frase_lst
['quiero el fonema /b']

#Bloque 2 kivy - Esta linea realiza la conversion de palabras a codigos numericos
frase_numerica = tokenizer1.texts_to_sequences(frase_lst)

frase_numerica
[[34, 1, 6, 35]]
```

Figura 65. Uso de la Tokenización. Elaborado por: Vera Vera Juan Andrés

3.6.4 Elección del Modelo

Según investigaciones anteriores de FCI 010-2021, citando en específico a la investigación de (Solorzano, 2022) delimita que se probaron diversos algoritmos tales como Long Short-Term Memory (LSTM), Modelo básico red neuronal de capa densa (DNN), Modelo NLP con clasificador Random Forest para multilabel o multi etiqueta y KNeighborsClassifier (kNN) k Vecino cercano. En los cuales luego de realizar la comparativa como resultado el modelo con mejores resultados de calidad es Modelo básico DNN, por lo cual este modelo también se aplicó para la red neuronal en este modelo.

La red neuronal se compone por una capa de entrada de 11 neuronas, una capa intermedia mejorada por el algoritmo Long Short-Term Memory (LSTM) que incluye una capa intermedia de 15 neuronas, mientras que la capa de salida es de 26 neuronas. En el cual demuestra que se utiliza un entrenamiento para 30 epochs.

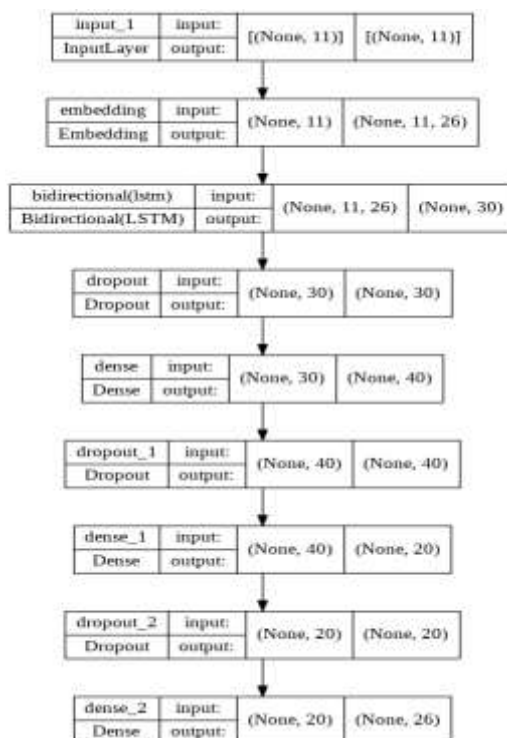


Figura 66. Arquitectura del modelo Básico DNN. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

En el siguiente gráfico se mostrará la arquitectura, de la red neuronal entrenada y también el total de parámetros de salida.

Model: "sequential"		
Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 11, 26)	1924
bidirectional (BidirectionalLSTM)	(None, 30)	5040
dropout (Dropout)	(None, 30)	0
dense (Dense)	(None, 40)	1240
dropout_1 (Dropout)	(None, 40)	0
dense_1 (Dense)	(None, 20)	820
dropout_2 (Dropout)	(None, 20)	0
dense_2 (Dense)	(None, 26)	546
Total params: 9,570		
Trainable params: 9,570		
Non-trainable params: 0		

Figura 67. Arquitectura del modelo Básico DNN. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.6.5 Evaluación del modelo

Una vez realizado el pre y post procesamiento de los datos se procede a usar el modelo Básico DNN, entrenándolo para 30 epochs, en donde se ajustan parámetros de tamaño de datos en este caso como usamos `tam_batch_size = 6` en el cual permite un nivel de salto de tamaño intermedio para que el algoritmo pueda llegar a encontrar la solución óptima durante el entrenamiento, se usaron técnicas de modelamiento como lo son “sigmoid” y “Adam, para optimizar la neurona y por último se usó un entrenamiento de validación mediante el parámetro “`tam_validation_split = 0.2`”, permite poner a prueba el modelo ante datos de entrada desconocidos y por último el parámetro “`verbose`” permite visualizar el procesamiento de entrenamiento que se realiza en la red neuronal.

Una vez establecidos todos los parámetros de configuración realizamos el entrenamiento con el método “`model.fit`” el resultado del entrenamiento se guarda en una variable llamada “`history`” esto permitirá presentar la evolución del entrenamiento de manera gráfica como se mostrará en la figura 68.

3.6.6 Entrenamiento del Algoritmo

Se ajusto los parámetros previamente dichos al modelo básico de red neuronal DNN.

```
Epoch 1/30
941/941 [=====] - 18s 13ms/step - loss: 0.2141 - accuracy: 0.1585 - precision: 0.0795 - recall: 0.0136 - auc: 0.0300 - val_loss: 0.1518 - val_accuracy: 0.2378 - val_p
Epoch 2/30
941/941 [=====] - 11s 12ms/step - loss: 0.1326 - accuracy: 0.4315 - precision: 0.3433 - recall: 0.2164 - auc: 0.8824 - val_loss: 0.8886 - val_accuracy: 0.6634 - val_p
Epoch 3/30
941/941 [=====] - 11s 12ms/step - loss: 0.0701 - accuracy: 0.6907 - precision: 0.8182 - recall: 0.4829 - auc: 0.9767 - val_loss: 0.8411 - val_accuracy: 0.8988 - val_p
Epoch 4/30
941/941 [=====] - 12s 13ms/step - loss: 0.0473 - accuracy: 0.8243 - precision: 0.8938 - recall: 0.7387 - auc: 0.9917 - val_loss: 0.8238 - val_accuracy: 0.9238 - val_p
Epoch 5/30
941/941 [=====] - 11s 12ms/step - loss: 0.0329 - accuracy: 0.8685 - precision: 0.8958 - recall: 0.8392 - auc: 0.9980 - val_loss: 0.8238 - val_accuracy: 0.9134 - val_p
Epoch 6/30
941/941 [=====] - 10s 12ms/step - loss: 0.0208 - accuracy: 0.9668 - precision: 0.9053 - recall: 0.8872 - auc: 0.9976 - val_loss: 0.8005 - val_accuracy: 0.9043 - val_p
Epoch 7/30
941/941 [=====] - 11s 12ms/step - loss: 0.0135 - accuracy: 0.8715 - precision: 0.8894 - recall: 0.9825 - auc: 0.9977 - val_loss: 0.8885 - val_accuracy: 0.9823 - val_p
Epoch 8/30
941/941 [=====] - 12s 13ms/step - loss: 0.0288 - accuracy: 0.8736 - precision: 0.9233 - recall: 0.9233 - auc: 0.9981 - val_loss: 0.8887 - val_accuracy: 0.8597 - val_p
Epoch 9/30
941/941 [=====] - 11s 12ms/step - loss: 0.0178 - accuracy: 0.8780 - precision: 0.9280 - recall: 0.9380 - auc: 0.9987 - val_loss: 0.8858 - val_accuracy: 0.8838 - val_p
Epoch 10/30
941/941 [=====] - 11s 12ms/step - loss: 0.0187 - accuracy: 0.8745 - precision: 0.9342 - recall: 0.9389 - auc: 0.9987 - val_loss: 0.8812 - val_accuracy: 0.9134 - val_p
Epoch 11/30
941/941 [=====] - 11s 12ms/step - loss: 0.0151 - accuracy: 0.8671 - precision: 0.9418 - recall: 0.9443 - auc: 0.9988 - val_loss: 0.8898 - val_accuracy: 0.9248 - val_p
Epoch 12/30
941/941 [=====] - 12s 13ms/step - loss: 0.0168 - accuracy: 0.8631 - precision: 0.9380 - recall: 0.9482 - auc: 0.9887 - val_loss: 0.8851 - val_accuracy: 0.8872 - val_p
Epoch 13/30
941/941 [=====] - 11s 12ms/step - loss: 0.0134 - accuracy: 0.8754 - precision: 0.9473 - recall: 0.9534 - auc: 0.9981 - val_loss: 0.8848 - val_accuracy: 0.8985 - val_p
Epoch 14/30
941/941 [=====] - 11s 12ms/step - loss: 0.0127 - accuracy: 0.8888 - precision: 0.9483 - recall: 0.9555 - auc: 0.9991 - val_loss: 0.8843 - val_accuracy: 0.8751 - val_p
Epoch 15/30
941/941 [=====] - 12s 13ms/step - loss: 0.0128 - accuracy: 0.8832 - precision: 0.9482 - recall: 0.9585 - auc: 0.9989 - val_loss: 0.8843 - val_accuracy: 0.8858 - val_p
```

Figura 68. Entrenamiento del modelo Básico DNN. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.6.7 Evaluación de métricas

Realizado el entrenamiento de épocas, se procede a mostrar los resultados mediante gráficas de métricas bajo las librerías de `matplotlib`, usando en este caso para el modelo de clasificación multi etiqueta con variables discretas para dos tipos de clases. Para este modelo se utilizaron las métricas de Accuracy, Loss, Val_Accuracy, Precision, Recall, AUC.

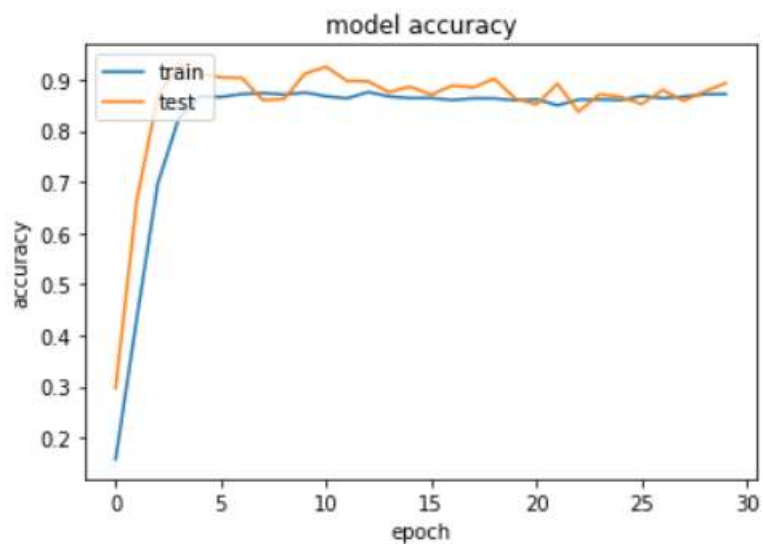


Figura 69. Evaluación de Accuracy del modelo Básico DNN. Elaborado por Vera Vera Juan Andrés

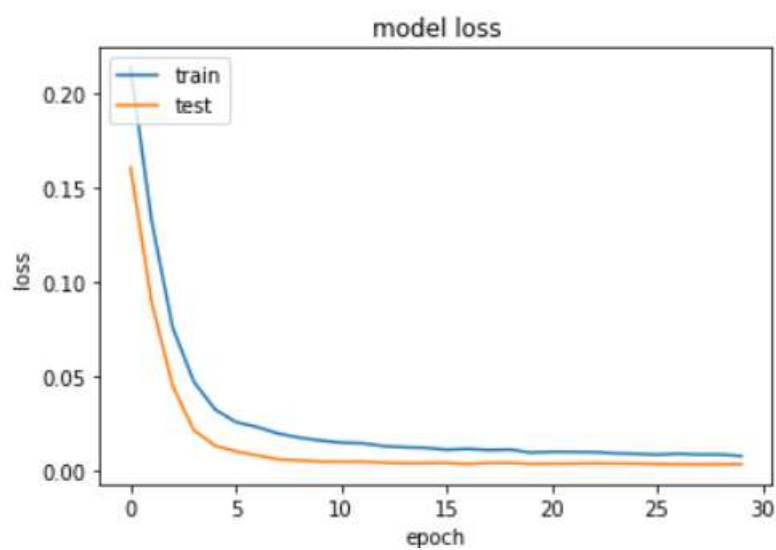


Figura 70. Evaluación de Model loss del modelo Básico DNN. Elaborado por Vera Vera Juan Andrés

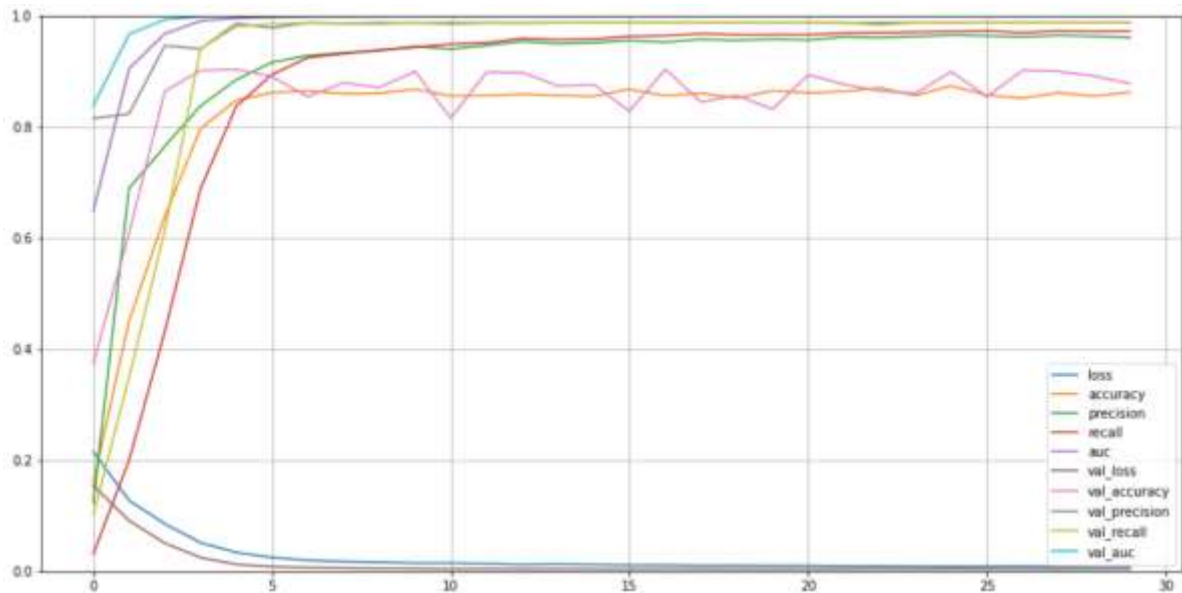


Figura 71. Evaluación de Val_accuracy_precision_recall_auc. Elaborado por Vera Vera Juan Andrés

```

loss_test:      0.31 %
acc_test:       89.29 %
preci_test:     98.94 %
recall_test:    98.89 %
auc_test:       99.97 %
loss_train:     0.3 %
acc_train:      87.83 %
preci_train:    98.89 %
recall_train:   98.9 %
auc_train:      99.99 %

```

Figura 72. Evaluación de Val_accuracy_precision_recall_auc. Elaborado por Vera Vera Juan Andrés

3.6.8 Resultados del modelo

Una vez terminado el modelo además de comprobar su calidad, se realizan pruebas de uso del modelo mostrando así las banderas, que deben activarse según lo digitado y analizado por el tokenizador para la detección de estas banderas.


```

#SETERO LAS ETIQUETAS QUE DEBEN DETECTARSE
# 'quiero el fonema /b'

frase2_str = "quiero el fonema /b"
frase2_lst=[]
frase2_lst.append(frase2_str)

# Levanto las banderas que deberian detectarse
frase2_banderas = [0, 0, 0, 0, 0, 0,
                   1, 0, 0, 0, 0, 0,
                   0, 0, 0, 0, 0, 0,
                   0, 0, 0, 0, 0, 0,
                   0, 0]

```

Figura 73. Evaluación de Val_accuracy_precision_recall_auc. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.7 Metodología de Desarrollo de la Aplicación

Para lo que es el diseño de la aplicación se escogió la metodología de desarrollo de software basado en prototipado, debido a que esta presenta las siguientes características necesarias del impulso del siguiente proyecto.

3.7.1 Metodología del Prototipado

Esta metodología está relacionada en la creación de un software que a futuro previo constara de mejoras continuas según (Sánchez y Gereá, 2021) consiste en un proceso iterativo enfocado en diseñar, implementar, medir y ajustar un plan.

3.7.2 Prototipo de baja Fidelidad

En este caso, esta subrama de la metodología del prototipado se usa cuando se requieren realizar prototipos rápidos, en los cuales se enfocan en una básica interacción y no en diseñar en totalidad los detalles de implementación de la idea. Por lo general este método permite lanzar de manera más rápida un software, debido a que no se consumen muchos recursos. (Sánchez y Gereá, 2021)

3.7.3 Construcción de la aplicación Móvil

Antes de comenzar la codificación se utilizaron diseños básicos mediante la plataforma web de Uizard en los cuales se facilitaron la realización de los mockups de las vistas de pantalla que tiene la aplicación.

En esta primera vista es la pantalla de inicio donde está instalado el módulo de Machine learning (NLP), además está incluido un botón en la sección inferior para navegación manual de la aplicación.

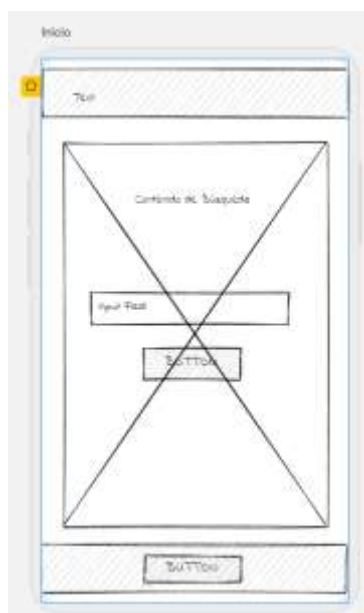


Figura 74. Prototipo de baja fidelidad primera Vista de la APP. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

En esta Segunda vista se presentarán la familia de fonemas de articulación de frase en las cuales se puede seleccionar cual familia desea acceder.

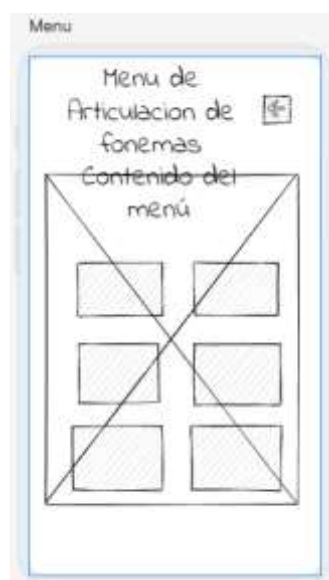


Figura 75. Prototipo de baja fidelidad Segunda Vista de la APP. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

En esta tercera vista se obtiene una tercera vista en donde se presenta una lista de contenido de cuales vocales o letras pertenecen a la familia de fonemas que se desean practicar.

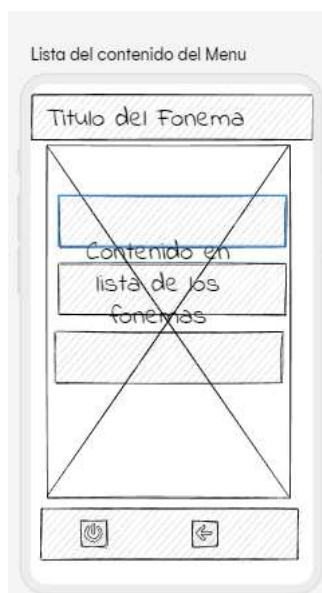


Figura 76. Prototipo de baja fidelidad Tercera Vista de la APP. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

En esta cuarta vista se presentará en el contenido de pantalla una imagen del objeto y audios reproductivos, en modalidad de frase completa y separada en sílabas.



Figura 77. Prototipo de baja fidelidad Cuarta Vista de la APP. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

En la siguiente imagen se mostrará una quinta vista de la aplicación agregando una variedad de ejercicio más en los cuales se podrá reproducir audios para completar la frase del objeto con la imagen que se representa en la pantalla.



Figura 78. Prototipo de baja fidelidad Quinta Vista de la APP. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.7.4 Prototipo de Baja Fidelidad – Media Fidelidad

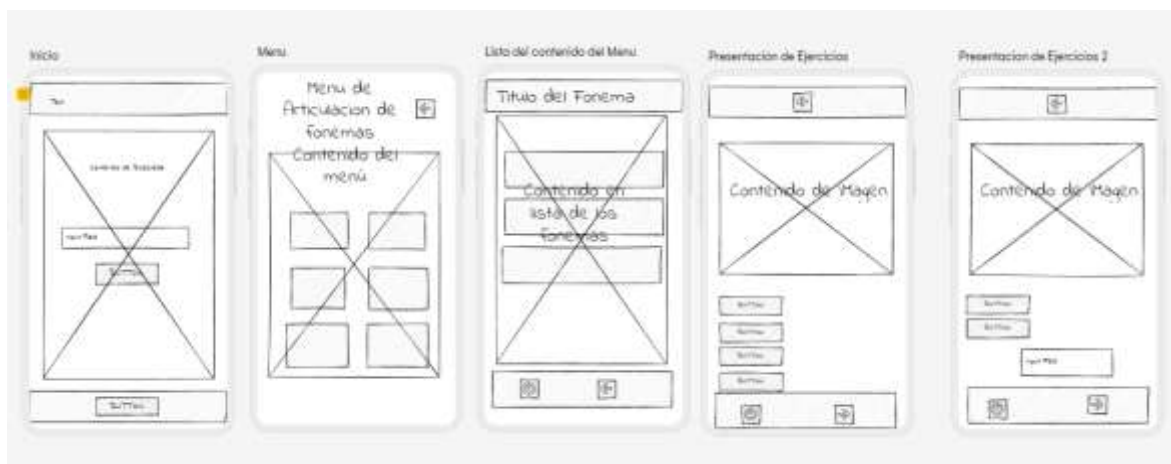


Figura 79. Prototipo de Baja-Media fidelidad Diseño general de la APP. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.8 Desarrollo de la aplicación Móvil

Se investigaron tecnologías que sean sencillas de realizar para el desarrollo de softwares más en el ámbito de unir dos módulos externos, los cuales son el módulo de machine learning y el módulo de la aplicación en general. Por lo tanto, se determinó usar tecnologías de programación en base al lenguaje de programación de Python, por su facilidad en desarrollo de programación y flexibilidad, se utilizó este lenguaje para el desarrollo del modelo de machine learning y por consiguiente se investigó cuales frameworks en el desarrollo de aplicaciones móviles, por lo que el framework Kivy de

Python basado en desarrollo de aplicaciones móviles fue el indicado para utilizar y elaborar el software.

Siguiendo con el avance del desarrollo en la siguiente fase, se usará el IDLE de Visual Studio Code, y como lenguaje de programación se usará la versión de Python 3.10.5, y para la codificación de la aplicación móvil se utilizará el framework de desarrollo de aplicaciones móviles denominado Kivy en la versión 2.0 es importante tener en cuenta las versiones debido a que los fabricantes liberan versiones en forma continua. Debido a que la interacción entre Kivy un rápido desarrollo y navegación sencilla y la creación de una aplicación sencilla con navegación sencilla que se construye a partir de estas dos herramientas de programación.



Figura 80. Instalación de Python. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

Una vez instalado Python se debe instalar el módulo de desarrollo de aplicaciones móviles Kivy, abriendo la ventana del “cmd” del sistema mediante el comando “pip install”, se instalarán por defecto la última versión del módulo de Kivy y el contenido de sus librerías.

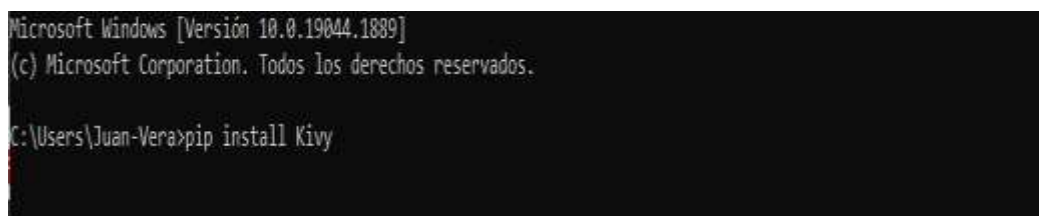


Figura 81. Instalación de Kivy. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

Se instala el módulo de Kivy para proceder a ingresar al entorno IDLE de programación en este caso el Visual Studio Code, para empezar la programación se debe importar las librerías del módulo previamente instalado.

```
from kivy.core.audio import SoundLoader
from kivy.core.window import Window
from kivy.uix.button import ButtonBehavior
from kivy.uix.image import Image
from kivy.uix.screenmanager import Screen
from kivy.uix.screenmanager import ScreenManager
from kivy.uix.popup import Popup
from kivy.uix.floatlayout import FloatLayout
from kivymd.app import MDApp
from kivymd.toast import toast
```

Figura 82. Entorno VSC con librerías de Kivy. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

Como todo framework Kivy tiene particularidades para ejecutar el entorno de aplicación una de estas es que se debe declarar el nombre de la App mediante una clase y una función de run () para ejecutar el archivo exe.

```
class App1(MDApp):
    def build(self):
        self.theme_cls.theme_style="Light"
        self.theme_cls.primary_palette='LightBlue'
        self.theme_cls.accent_palette="Blue"
        pass

if __name__ == '__main__':
    window.show_cursor = True
    window.size = (360, 680)
    App1().run()
```

Figura 83. Ejecución de archivo .exe de Kivy. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

Luego de esto se debe tener en cuenta que el desarrollo de aplicación móvil se realiza mediante la codificación de Python y las librerías de Kivy se utilizan para programar las interfaces gráficas. Por lo tanto, se ejecutan 2 archivos el archivo .py que es el entorno de programación de funciones, y el archivo kv que es donde se alojará toda la interfaz gráfica de la aplicación.

```

class Pantallainicio(Screen):
    def fonema(self):
        fonemas = self.ids['fonemas'].text
        print(fonemas)
        if not fonemas:
            toast("Debe insertar una Búsqueda")

```

Figura 84. Modalidad de Programación en Kivy archivo.py. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

```

main.py  App1.kv  X
App1.kv
3316     elevation: 10
3317     MDFloatingActionButton:
3318         icon: 'backburger'
3319         elevation: 0
3320         pos_hint: {'center_x': 0.5, 'center_y': 0.5}
3321         on_release:
3322             root.manager.current="Pantalla1"
             root.manager.transition.direction="right"

```

Figura 85. Modalidad de Programación en Kivy archivo.kv. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.8.1 Consideraciones antes de unir Módulos (IA) con APP

Para trabajar y unir módulos de machine learning con una aplicación de software se debe tener en cuenta que la librería Tensorflow, es la que realiza este proceso por lo cual esta estará trabajando ya sea con la tarjeta gráfica (GPU) del ordenador, en caso de no poseer alguna, trabaja con el procesador del ordenador (CPU).

Para que la librería de Tensorflow pueda trabajar y comunicarse con la GPU se debe instalar los controladores gráficos (Drivers) de la correspondiente, y por consiguiente instalar la librería de Tensorflow dentro de la tarjeta gráfica. Además de controladores que utiliza la librería de Tensorflow estos son CUDA y CUDNN los cuales se mostraran a continuación. Es muy importante en esta instalación tener en cuenta las versiones de nuestro Hardware en totalidad como también la versión de Python con la que se está trabajando.

Los drivers se descargan de la misma página oficial de NVIDIA.

Descarga de controladores NVIDIA

Selecciona en la lista desplegable siguiente para identificar el controlador adecuado para tu producto NVIDIA.

Tipo de producto: GeForce ▼

Serie del producto: GeForce 10 Series ▼

Familia del producto: GeForce GTX 1050 ▼

Sistema operativo: Windows 10 64-bit ▼

Tipo de descarga: Controlador Game Ready (GRD) ▼ ?

Idioma: Español (España) ▼

Figura 86. Controladores Gráficos de Nvidia. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés Fuente: (Nvidia, 2022)

Debes instalar el siguiente software de NVIDIA® en tu sistema:

- [Controladores de GPU NVIDIA®](#) [🔗](#): CUDA® 11.2 requiere la versión 450.80.02 o una posterior.
- [Kit de herramientas CUDA®](#) [🔗](#): TensorFlow es compatible con CUDA® 11.2 (TensorFlow >= 2.5.0).
- [CUPTI](#) [🔗](#) incluye el kit de herramientas CUDA®.
- [SDK de cuDNN 8.1.0](#) [🔗](#) (versiones de cuDNN [🔗](#)).
- (Opcional) [TensorRT 6.0](#) [🔗](#) para mejorar la latencia y la capacidad de procesamiento para la inferencia en algunos modelos.

Figura 87. Controladores de Tensorflow. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés (citar de tensorflow)

A continuación, se muestra la lista de controladores que deben estar preinstalados para el correcto funcionamiento del software ya desarrollado.

	NVIDIA Controlador de audio HD 1.3.39.14	12/8/2022
	NVIDIA Controlador de gráficos 516.94	12/8/2022
	NVIDIA CUDA Development 11.7	13/8/2022
	NVIDIA CUDA Documentation 11.7	13/8/2022
	NVIDIA CUDA Night NVTX 11.7	13/8/2022
	NVIDIA CUDA Runtime 11.7	13/8/2022
	NVIDIA CUDA Visual Studio Integration 11.7	13/8/2022
	NVIDIA FrameView SDK 1.2.7521.31103277	13/8/2022
	NVIDIA GeForce Experience 3.25.1.27	13/8/2022
	NVIDIA Nsight Compute 2022.2.1	832 MB 12/11/2022
	NVIDIA Nsight Systems 2022.1.3	834 MB 12/8/2022
	NVIDIA Nsight Visual Studio Edition 2022.2.1.22136	698 MB 13/8/2022

Figura 88. Lista de controladores. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

Por consiguiente, se procede a instalar la librería Tensorflow con el comando “pip install” dentro del “cmd”

```
$ pip install tensorflow==1.15 # CPU
$ pip install tensorflow-gpu==1.15 # GPU
```

Figura 89. Comandos de instalación en el cmd de la librería Tensorflow. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés Fuente: Propia

Una vez realizado este entorno de instalación se procede a unir los dos módulos ya programados.

3.8.2 Importación y exportación del Modelo de Machine Learning (NLP)

Con el modelo de inteligencia artificial ya entrenado se procede a importarlo y exportarlo para así descargarlo y usarlo, para este proceso se debe realizar tanto al Tokenizador del modelo como también al archivo del modelo ya entrenado, este proceso se realizó en dos formatos diferentes los cuales son: pkl y h5. Para el Tokenizador se descargó en el formato json.

```
[ ] import pickle as pkl

[ ] modelo2 = pkl.load(open("clasificador_Multietiqueta_Fonemas_NND_Model.pkl", 'rb'))

[ ] tokenizer2 = pkl.load(open("Tokenizador1_Fonemas_NND_Model.pkl", 'rb'))

[ ] frase_str = "quiero el fonema /b"
frase_lst=[]
frase_lst.append(frase_str)
frase_numerica = tokenizer2.texts_to_sequences(frase_lst)

[ ] frase_numerica

[[34, 1, 6, 35]]

[ ] tokenizer2.texts_to_sequences(frase_lst)

[[34, 1, 6, 35]]
```

Figura 90. Importación y exportación del Modelo. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.8.3 Complicaciones de compatibilidad

En esta sección se detalla que el modelo tanto como el Tokenizador se descargaron e implementaron con el formato pkl, dado los resultados de incompatibilidad, se procedió a importar y exportarlos modelos, el modelo de inteligencia artificial se lo exporto mediante formato h5 y el Tokenizador por último se lo descargo en formato json.

```

▶ modelo2.save('modeloredneuronalnlp.h5')
#Sirve para exportar el modelo

[ ] modelo2 = load_model('modeloredneuronalnlp.h5')
#sirve para importar y trasladar el modelo

[ ] tokenizer_json = tokenizer1.to_json()
with io.open('tokenizer.json', 'w', encoding='utf-8') as f:
    f.write(json.dumps(tokenizer_json, ensure_ascii=False))
# exportar el tokenizer

[ ] with open('tokenizer.json') as f:
    data = json.load(f)
    tokenizer2 = tokenizer_from_json(data)
#importar para el software

```

Figura 91. Importación y exportación del Modelo. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

- 📄 Clasificador_Multietiqueta_Fonemas_NND_Model.pkl
- 📄 DATASET-PEDIDOS-FONEMAS-V3.xlsx
- 📄 Tokenizador1_Fonemas_NND_Model.pkl
- 📄 model.png
- 📄 modeloredneuronalnlp.h5
- 📄 tokenizer.json

Figura 92. Importación y exportación del Modelo. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.8.4 Implementación del Módulo (ML, NLP) dentro de la Aplicación

Para la implementación dentro de la aplicación se debe traer los archivos exportados previamente del Google Colaboratory dentro de la carpeta del proyecto por lo que el entorno de la programación solo, puede identificar archivos que estén dentro de la carpeta del proyecto de programación.

```

▼ modelo
  ≡ Clasificador_Multietiqueta_Fonemas_NND_Mo...
  📄 DATASET-PEDIDOS-FONEMAS-V3.xlsx
  ≡ desktop.ini
  {} mitokenizer.json
  🖼️ model.png
  ≡ modeloredneuronalnlp.h5
  ≡ Tokenizador1_Fonemas_NND_Model.pkl

```

Figura 93. Implementación del Modelo. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

Con los procedimientos previamente realizados se pueden importar las librerías de Tensorflow dentro del entorno de VSC y posteriormente manejar las variables para efectuar y trabajar con el modelo de IA dentro de la programación, se deben realizar una función de llamada a los archivos del modelo una vez importadas las librerías dentro de la programación.

```
from keras.preprocessing.sequence import pad_sequences
#from keras.utils.data_utils import pad_sequences
from keras.models import load_model
import json
from keras.preprocessing.text import tokenizer_from_json

#Bloque 1
modelo2 = load_model("modelo\modeloredneuronalnlp.h5")

#sirve para importar y trasladar el modelo
with open("modelo\mitokenizer.json") as f:
    data = json.load(f)
    tokenizer2 = tokenizer_from_json(data)
```

Figura 94. Implementación del Modelo. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

```
frase_str = fonemas
# creamos una lista en la que se va a guardar la frase digitada
frase_lst = []
frase_lst.append(frase_str)
# Bloque 2 - kyvy - Esta línea realiza la conversión de palabras a códigos numéricos
frase_numerica = tokenizer2.texts_to_sequences(frase_lst) # tokenizer2

# Bloque 3 - kyvy - completamos el código en la entrada del modelo nlp
frase_numerica_pad = pad_sequences(frase_numerica, padding='post', maxlen=11)
# Toma la frase numérica (formateada a 11 números por lo que si no hay 11 caracteres se llenarán con 0)
# prueba de recuperación de 1 texto digitado a partir de los códigos del tokenizer
#tokenizer2.sequences_to_texts(frase_numerica)

# Bloque 4 - kyvy - Formateo la arreglo a las dimensiones que necesita el modelo a la entrada
#np.expand_dims(frase_numerica_pad[0], 0)

# Bloque 5 - kyvy - hacemos la predicción de lo pedido
V_predicho1_index = modelo2.predict(np.expand_dims(frase_numerica_pad[0], 0))

# Bloque 6 - kyvy - Redondeamos la probabilidad a 0 dígitos decimales para hallar las etiquetas predichas
V_predicho1_index_round = np.round(V_predicho1_index, 0)

if (V_predicho1_index_round[0, 6] == 1):
    print("Aquí se muestra la ventana B")
    self.change_screen("BilabialesB")
else:
    if (V_predicho1_index_round[0, 7] == 1):
```

Figura 95. Implementación del Modelo. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

3.8.5 Testeo y Verificación del proyecto del Software

Una vez realizado correctamente la implementación del modelo dentro de la codificación la terminal del IDLE ofrecerá una confirmación de lectura en la cual confirmará el uso del módulo asociado con la GPU.

```

/replica:0/task:0/device:GPU:0 with 1893 MB memory: -> device: 0, name: NVIDIA GeForce GTX 1050, pci bus id: 0000
:01:00:0, compute capability: 6.1
[INFO ] [GL      ] NPOT texture support is available
[INFO ] [GL      ] Unpack subimage support is available
[INFO ] [Base    ] Start application main loop
[INFO ] [Base    ] Leaving application in progress...

```

Figura 96. Implementación del Modelo. Tomada de investigación directa. Elaborado por Vera Vera Juan Andrés

Por último, se procede a usar el modelo dentro de la Aplicación desarrollada realizando el respectivo testeo y verificación mediante las terminales del código concluyendo con la implementación del modelo y unión de los 2 módulos.

```

object(wenv\Lib\site-packages\kivy\__init__.py')
2022-08-25 11:56:38.532681: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimiz
ed with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical
operations: AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-08-25 11:56:39.366345: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1532] Created device /job:localhost
/replica:0/task:0/device:GPU:0 with 1893 MB memory: -> device: 0, name: NVIDIA GeForce GTX 1050, pci bus id: 0000
:01:00:0, compute capability: 6.1
[INFO ] [GL      ] NPOT texture support is available
[INFO ] [GL      ] Unpack subimage support is available
[INFO ] [Base    ] Start application main loop
Quiero practicar el fonema /m
2022-08-25 11:59:57.431559: I tensorflow/stream_executor/cuda/cuda_dnn.cc:384] Loaded cuDNN version 8400
Aquí se muestra la ventana M

```

Figura 97. Testeo y verificación de la Aplicación. Elaborado por Vera Vera Juan Andrés

```

[INFO ] [logger   ] Record log in C:\Users\Juan-Vera\kivy\logs\kivy_22-08-25_2.txt
[INFO ] [deps     ] Successfully imported "kivy_deps.angle" 0.3.2
[INFO ] [deps     ] Successfully imported "kivy_deps.glew" 0.3.1
[INFO ] [deps     ] Successfully imported "kivy_deps.sdl2" 0.4.5
[INFO ] [kivy     ] v2.1.0
[INFO ] [kivy     ] Installed at "C:\Users\Juan-Vera\Python\Projects\pythonProject\venv\Lib\site-packages\kiv
y\__init__.py"
[INFO ] [Python  ] v3.10.5 (tags/v3.10.5:f377151, Jun 6 2022, 16:14:11) [MSC v.1929 64 bit (AMD64)]
[INFO ] [Python  ] Interpreter at "C:\Users\Juan-Vera\Python\Projects\pythonProject\venv\Scripts\python.exe"
[INFO ] [logger   ] Purge log fired. Processing...
[INFO ] [logger   ] Purge finished!
[INFO ] [Audio    ] Providers: audio_sdl2 (audio_ffpyplayer ignored)
[INFO ] [Factory  ] sfp symbols loaded
[INFO ] [Image    ] Providers: img_tex, img_dds, img_sdl2, img_pil (img_ffpyplayer ignored)
[INFO ] [Window   ] Provider: sdl2
[INFO ] [GL        ] Using the "OpenGL" graphics system
[INFO ] [GL        ] GLEW initialization succeeded
[INFO ] [GL        ] Backend used: glew
[INFO ] [GL        ] OpenGL version: '4.6.0 NVIDIA 516.04'
[INFO ] [GL        ] OpenGL vendor: 'NVIDIA Corporation'
[INFO ] [GL        ] OpenGL renderer: 'NVIDIA GeForce GTX 1050/PCIe/SSE2'
[INFO ] [GL        ] OpenGL parsed version: 4, 6
[INFO ] [GL        ] Shading version: '4.60 NVIDIA'
[INFO ] [GL        ] Texture max size: <12768>
[INFO ] [GL        ] Texture max units: <32>
[INFO ] [Window   ] auto add sdl2 input provider
[INFO ] [Window   ] virtual keyboard not allowed, single mode, not docked
[INFO ] [Text     ] Provider: sdl2
[INFO ] [KivyMD   ] 1.0.2, git-30f0356, 2022-08-11 (installed at "C:\Users\Juan-Vera\Python\Projects\pythonProject\venv\Lib\site-packages\kivy\

```

Figura 98. Testeo y verificación de la Aplicación. Elaborado por Vera Vera Juan Andrés



Figura 99. Testo y verificación de la Aplicación. Elaborado por Vera Vera Juan Andrés

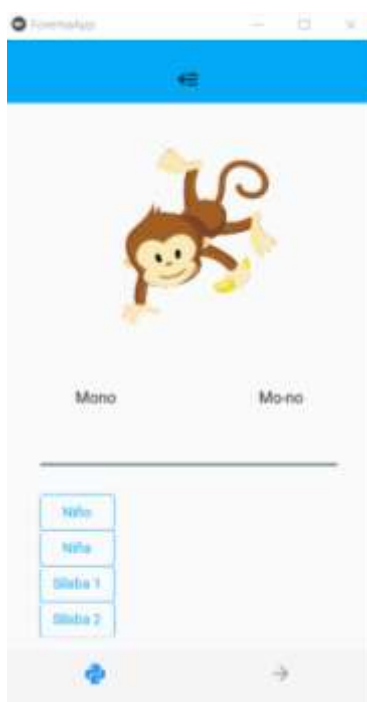


Figura 100. Testo y verificación de la Aplicación. Elaborado por Vera Vera Juan Andrés

3.9 Conclusiones

Se concluye que:

- Mediante la revisión bibliográfica y entrevistas a profesionales de logopedia se muestra que la dislalia en su etiología se divide en evolutivas, funcionales, orgánicas y auditivas y en prácticas médicas se dividen en simples, complejas y retrasos del habla. Sus tratamientos comprenden de intervenciones indirectas correspondientes a ejercicios de fono articulación para estimular el pronunciamiento e intervenciones directas las cuales corresponden prácticas de palabras y frases con respecto a los fonemas.
- Se identificó usar el lenguaje de Python por sus características flexibles y especialidades y en desarrollos de modelos IA y la el entorno de Google Colab por su gratuidad y facilidad en importaciones y exportaciones de los modelos además de no descargar componentes adicionales para la elaboración del modelo y e entorno ordenado que ofrece entre las ejecuciones del programa.
- Se identificó en el análisis comparativo mediante las entrevistas y revisión bibliográfica que las tecnologías de Flutter y Android Studio, son más utilizadas por sus características de compatibilidad para sistemas completos dentro del campo a nivel empresarial.
- Se logro construir un modelo NLP como herramienta integrada en la aplicación instalando drivers correspondientes y las librerías compatibles dependiendo del tipo del GPU que porta el ordenador, para su correcto funcionamiento y uso.
- Obteniendo los resultados de evaluaciones de métricas bajo los parámetros Val_Accuracy, AUC, Precision, Recall dan resultados arriba del 90% demostrando una alta calidad en modelo de clasificación multi etiqueta usando una red neuronal DNN.
- Se identifico usar el framework de Kivy dedicado a Python para la programación de la aplicación debido a su rápido desarrollo de aplicaciones, características multiplataforma, amplio contenido de Widgets personalizados para generaciones de interfaces gráficas de alta definición amigables e integraciones de modelos de IA.

3.10 Recomendaciones

Se recomienda:

- En caso de actualizar el software se recomienda mejorar el software aplicativo a nivel de 3 capas. Estas Incluye montar un servidor web, montar un servidor de base de datos para registro y autenticación, y así diseñar un registro de actividades y guardar el progreso de los usuarios que utilicen el aplicativo, y por último un servidor a nivel de aplicaciones para que se pueda alojar toda la lógica de la programación dentro del servidor.
- Para futuras investigaciones se recomienda crear un módulo de visión artificial que permita mediante el reconocimiento de imágenes evaluar la correcta postura a nivel bucofacial y praxia lingual de ejercicios de fonoarticulación con respecto a intervenciones indirectas.
- Para futuros trabajos de investigación se recomienda crear otro modelo NLP que tenga capacidades de diversidad en petición de interactividad con los audios reproductivos en los ejercicios de la aplicación.

ANEXOS

Anexo 1

Artículos del Marco Legal

Constitución de la República del Ecuador.

Artículo 350.- "El sistema de educación superior tiene como finalidad la formación académica y profesional con visión científica y humanista; la investigación científica y tecnológica; la innovación, promoción, desarrollo y difusión de los saberes y las culturas; la construcción de soluciones para los problemas del país, en relación con los objetivos del régimen de desarrollo."

Art. 262.- Los gobiernos regionales autónomos tendrán las siguientes competencias exclusivas, sin perjuicio de las otras que determine la ley que regule el sistema nacional de competencias:

6. Determinar las políticas de investigación e innovación del conocimiento, desarrollo y transferencia de tecnologías, necesarias para el desarrollo regional, en el marco de la planificación nacional.

Art. 277.- Para la consecución del buen vivir, serán deberes generales del Estado:

6. Promover e impulsar la ciencia, la tecnología, las artes, los saberes ancestrales y en general las actividades de la iniciativa creativa comunitaria, asociativa, cooperativa y privada

Art. 385.- El sistema nacional de ciencia, tecnología, innovación y saberes ancestrales, en el marco del respeto al ambiente, la naturaleza, la vida, las culturas y la soberanía, tendrá como finalidad:

1. Generar, adaptar y difundir conocimientos científicos y tecnológicos.

3. Desarrollar tecnologías e innovaciones que impulsen la producción nacional, eleven la eficiencia y productividad, mejoren la calidad de vida y contribuyan a la realización del buen vivir.

Artículo 386.- El Sistema Nacional, de Ciencia, Tecnología, Innovación, y; Saberes Ancestrales "Comprenderá programas, políticas, recursos, acciones, e incorporará a instituciones del Estado, universidades y escuelas politécnicas, institutos de investigación públicos y particulares, empresas públicas y privadas, organismos no gubernamentales y personas naturales o jurídicas, en tanto realizan actividades de investigación, desarrollo tecnológico, innovación y aquellas ligadas a los saberes ancestrales."

2.3.2 Reglamento de Régimen Académico

Art. 78.- "Se entenderá como pertinencia de carreras y programas académicos a la articulación de la oferta formativa, de investigación y de vinculación con la sociedad, con el régimen constitucional del Buen Vivir, el Plan Nacional de Desarrollo, los planes regionales y locales los requerimientos sociales en cada nivel territorial y las corrientes internacionales científicas y humanísticas de pensamiento.

2.3.3 Ley Orgánica de Comunicación

Art. 35.- Derecho al acceso universal a las tecnologías de la información y comunicación. - Todas las personas tienen derecho a acceder, capacitarse y usar las tecnologías de información y comunicación para potenciar el disfrute de sus derechos y oportunidades de desarrollo.

2.3.4 Ley Orgánica de Telecomunicaciones

Art. 88.- Promoción de la Sociedad de la Información y del Conocimiento.

5. Promover el desarrollo y masificación del uso de las tecnologías de información y comunicación en todo el territorio nacional.

6. Apoyar la educación de la población en materia de informática y tecnologías de la información, a fin de facilitar el uso adecuado de los servicios o equipos.

2.3.5 Revisión de Normas Internacionales del uso de la IA

Debido a que la normativa referente al manejo, entrenamiento y uso de la IA a nivel de legislación ecuatoriana aún no se encuentra desarrollada se verifican las leyes que están en vigencia en otros continentes.

2.3.6 Resolución del Parlamento Europeo sobre uso policial de la Inteligencia Artificial

El pasado 6 de octubre de 2021 fue aprobada con 377 votos a favor, 248 en contra y 62 abstenciones, la Resolución del Parlamento Europeo sobre la IA en el Derecho Penal y su utilización por las autoridades policiales y judiciales. El presente Reglamento también debe aplicarse a las instituciones, oficinas y organismos de la Unión cuando actúen como proveedores o usuarios de un sistema de IA. Los sistemas de IA desarrollados o utilizados exclusivamente con fines militares deben quedar excluidos del ámbito de aplicación del

presente Reglamento cuando dicho uso sea competencia exclusiva de la Política Exterior y de Seguridad Común regulada en el título V del Tratado de la Unión Europea (TUE).

Art. 6 Normas de clasificación de los sistemas de IA de alto riesgo

Independientemente de si un sistema de IA se introduce en el mercado o se pone en servicio independientemente de los productos mencionados en las letras a) y b), dicho sistema de IA se considerará de alto riesgo cuando se cumplan las dos condiciones siguientes:

(a) el sistema de IA está destinado a ser utilizado como componente de seguridad de un producto, o es en sí mismo un producto, cubierto por la legislación de armonización de la Unión enumerada en el anexo II;

(B) el producto cuyo componente de seguridad es el sistema de IA, o el propio sistema de IA como producto, debe someterse a una evaluación de la conformidad por un tercero con vistas a la comercialización o puesta en servicio de dicho producto de conformidad con la armonización de la Unión legislación enumerada en el Anexo II.

Art. 10 Datos y gobernanza de datos

Los sistemas de IA de alto riesgo que utilicen técnicas que impliquen el entrenamiento de modelos con datos se desarrollarán sobre la base de conjuntos de datos de entrenamiento, validación y ensayo que cumplan los criterios de calidad.

2.3.7 Reglamento general de protección de datos

Art. 22 En función de los datos que se utilizan para entrenar los sistemas de IA, sus resultados pueden estar sesgados. La utilización de la IA para crear obras puede tener repercusiones sobre los derechos de propiedad intelectual y plantear cuestiones en relación, por ejemplo, con la patentabilidad, los derechos de autor y los derechos de propiedad.

Anexo 2

Manual de Usuario

Pantalla Vista 1



A continuación, se presenta la pantalla de inicio donde el usuario puede realizar su búsqueda del fonema que desea practicar, adicionalmente se encuentra un botón en la parte inferior en la cual se puede acceder al botón de menú.

Pantalla Vista 2



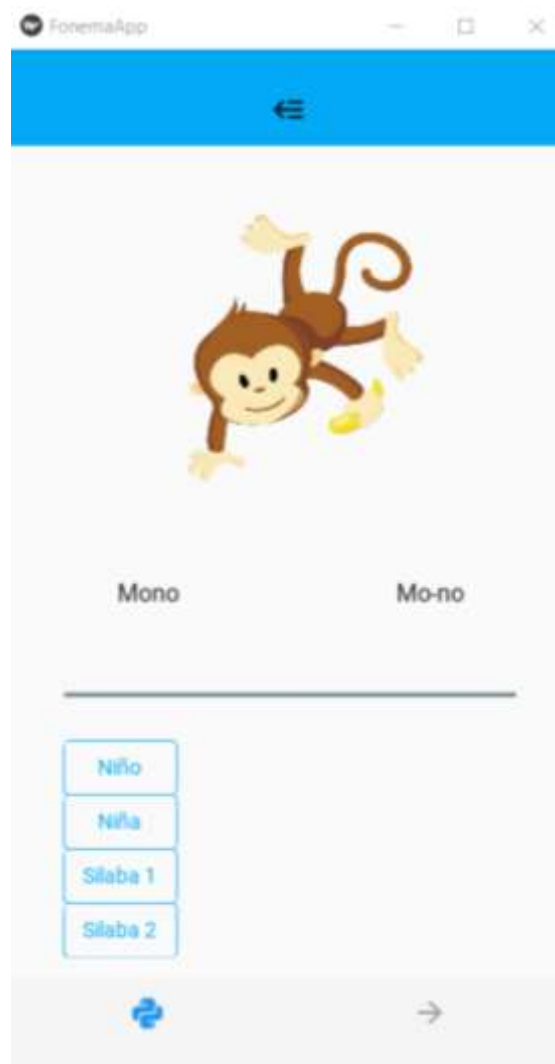
En la vista del menú de la aplicación se podrá observar un entorno familiar en la cual cada caja tiene contenido de los fonemas agrupados en la familia indicada a la cual pertenecen una vez se acceda a cualquiera de ellos aparecerá la siguiente vista.

Pantalla vista 3

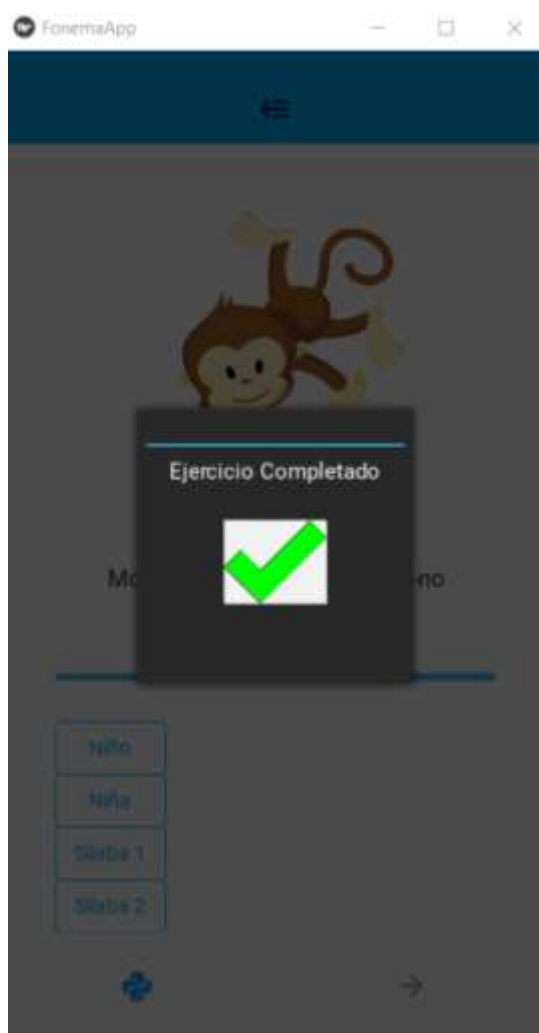


En la tercera vista se accede a los fonemas clasificados por el grupo que el usuario selecciono anteriormente adicionalmente tiene un botón de volver el cual servirá para regresar a la vista del menú de la aplicación.

Pantalla Vista 4



En la vista de los ejercicios se podrá observar la imagen del objeto y debajo de él se presentará la frase completa y adicionalmente la frase separada por silabas, los botones que se presentan reproducirán audios de como decir la frase de forma correcta. Adicionalmente habrá una barra inferior que presentara dos botones uno es el botón para volver a la pantalla de inicio y el siguiente botón accederá al siguiente ejemplo de la sección de fonema.

Pantalla vista 5

En la misma vista de ejercicios una vez se reproduzcan los cuatro audios aparecerá un mensaje de alerta que contendrá el mensaje de ejercicio completado.

Pantalla vista 6

En la siguiente Variante de Ejercicios se puede reproducir la frase de la imagen que se presenta adicionalmente se tendrá que completar la frase de acuerdo a lo escuchado si se responde de manera incorrecta aparecerá un mensaje de vuelve a intentarlo, caso contrario aparecerá el mensaje de alerta de ejercicio completado.

Diagrama de caso de Uso

A continuación, se presenta un Diagrama de Caso de uso de la aplicación.

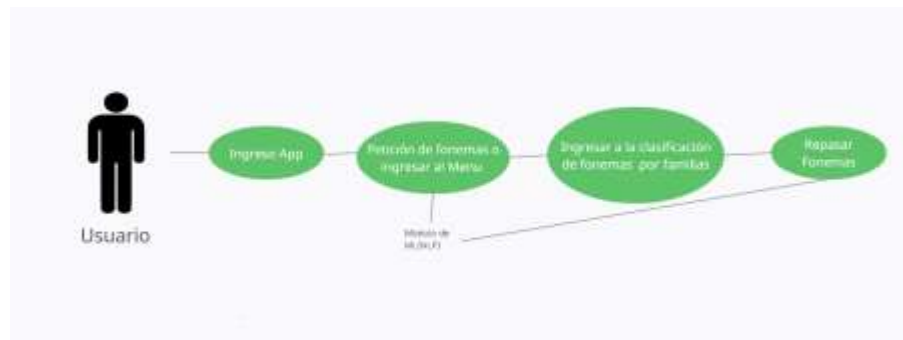
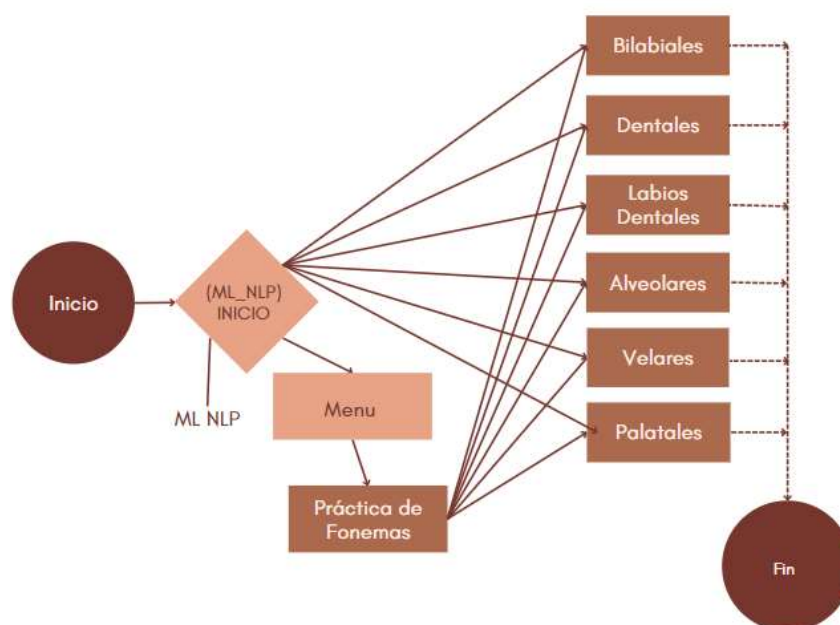


Diagrama de Flujo

Diagrama de Flujo de la aplicación



Anexo 3

Entrevista realizada a Logopedas (Terapistas del Lenguaje)



1. **Fecha de la entrevista:** 04/05/2022
2. **Nombre:** Cecilia Calero
3. **Profesión:** Licenciada en Terapeuta de Lenguaje
4. **¿Como Logopeda cree usted que una aplicación móvil (App) con ejercicios orales, auditivos, gráficos y vocales aportaría una mejora en los pacientes tratados con dislalia evolutiva?**

Parcialmente de acuerdo

5. **¿Estima usted que los pacientes con dislalia evolutiva pueden apoyarse o cuenta con la destreza para usar una aplicación móvil (APP) sin mayor dificultad?**

Totalmente de acuerdo

6. **¿Estima usted que una aplicación móvil con ejercicios complementarios para realizarse en casa de manera didáctica, ilustrativa y fácil de usar podrían ser de ayuda al paciente con dislalia?**

Totalmente de acuerdo

7. **¿Considera usted que una aplicación móvil (APP) podría servir como comunicación continua entre familias de niños con dislalia y especialistas para comunicación del progreso del paciente?**

Totalmente de acuerdo

8. **¿Cómo terapeuta basado en su experiencia cuantos niveles de dislalia evolutiva considera que se puedan interpretar?**

Si el paciente registra problemas con 1 solo fonema lo considero una dislalia normal o simple, cuando el paciente presenta problemas con dos o varios fonemas la considero una dislalia múltiple, en caso que no tenga problemas en el lenguaje en su

lenguaje espontaneo y no pueda articular ninguna palabra bien, presenta un retraso en su comunicación o tartamudez y este debe ser atendido con mucha frecuencia.

9. ¿Cómo Terapeuta de pacientes de dislalia evolutiva que ejercicios habituales sugiere usted para brindar un diagnóstico de dislalia evolutiva a los pacientes y con qué frecuencia debe realizarlos?

En su lenguaje, en su conversación y su lenguaje oral, repetición de palabras considerando cada una con las consonantes y vocales ya sea al si estas vocales se presentan al principio, medio y final de la palabra, además de nominación de imágenes de los fonemas articulados en las palabras (inicio, medio, final).

10. ¿Cuáles son los ejercicios que usted como logopeda usa en el proceso de tratamiento de dislalia?

Repetición de palabras de sonido, Labiales, Bilabiales, linguopalatales, praxias linguales, linguodentales, linguoalveolares, linguovelares, linguointerdentales.

11. ¿Cuáles son los ejercicios que más sugiere usted para una dislalia leve intermedia o crítica?

Depende del fonema que el paciente presenta el error, solo procuro trabajar en el fonema que el paciente tenga problemas en articular.

12. ¿Qué ejercicio sugiera usted repasar en casa a sus pacientes dependiendo del nivel de dislalia que presenten ya sea nivel leve intermedio o critico?

Presencialmente usamos ejercicios con juegos, e indicamos que siga a los padres estas dinámicas en casa, prácticas de lenguaje espontaneo frente al espejo, nominaciones de imágenes además de brindarles plantillas con ejercicios de repaso de palabras según donde el paciente presente los problemas articulatorios.

13. ¿Cuál es la modalidad que recomienda a sus pacientes en el repaso de frases y palabras en casa?

Dependiendo de las evaluaciones del paciente que registra, 3 – 4 días por semanas en un lapso de 30 minutos, pero recomendando practicar a mis pacientes diariamente.



1. **Fecha de la entrevista:** 11/05/2022
2. **Nombre:** Dominga Villamar
3. **Profesión:** Licenciada en Terapeuta de Lenguaje

4. **¿Como Logopeda cree usted que una aplicación móvil (App) con ejercicios orales, auditivos, gráficos y vocales aportaría una mejora en los pacientes tratados con dislalia evolutiva?**

Totalmente de acuerdo

5. **¿Estima usted que los pacientes con dislalia evolutiva pueden apoyarse o cuenta con la destreza para usar una aplicación móvil (APP) sin mayor dificultad?**

Totalmente de acuerdo

6. **¿Estima usted que una aplicación móvil con ejercicios complementarios para realizarse en casa de manera didáctica, ilustrativa y fácil de usar podrían ser de ayuda al paciente con dislalia?**

Totalmente de acuerdo

7. **¿Considera usted que una aplicación móvil (APP) podría servir como comunicación continua entre familias de niños con dislalia y especialistas para comunicación del progreso del paciente?**

Totalmente de acuerdo

8. **¿Cómo terapeuta basado en su experiencia cuantos niveles de dislalia evolutiva considera que se puedan interpretar?**

Considerando cuantos fonemas el paciente no puede articular, como dislalias simples si solo son un par de fonemas y complejas si presenta varios problemas de articulación.

9. ¿Cómo Terapeuta de pacientes de dislalia evolutiva que ejercicios habituales sugiere usted para brindar un diagnóstico de dislalia evolutiva a los pacientes y con qué frecuencia debe realizarlos?

Me baso en su expresión oral haciéndole al niño practicar lecturas de libros infantiles como el pastorcito, la liebre y la tortuga, además de repetir palabras después de las evaluaciones en las lecturas para aterrizar más en cuáles son sus problemas específicos en la articulación.

10. ¿Cuáles son los ejercicios que usted como logopeda usa en el proceso de tratamiento de dislalia?

Uso repeticiones de frases, repeticiones de palabras reconocimiento y nominación de imágenes considerando en estas vocales y consonantes además de ejercicios bucofaciales y praxias linguales.

11. ¿Cuáles son los ejercicios que más sugiere usted para una dislalia leve intermedia o crítica?

Uso más ejercicios orientados a la movilización facial con ejercicios bucofaciales y praxias linguales para que entrene su lengua y tener mejorías al momento de repetir palabras y en las lecturas.

12. ¿Qué ejercicio sugiera usted repasar en casa a sus pacientes dependiendo del nivel de dislalia que presenten ya sea nivel leve intermedio o crítico?

Los ejercicios procuro mandar ejercicios de palabras lectura de libros y en caso de ser necesario ejercicios de movimientos vocales previamente indicándole al padre que debe tomar en cuenta para estas actividades.

13. ¿Cuál es la modalidad que recomienda a sus pacientes en el repaso de frases y palabras en casa?

Repasar diariamente los ejercicios que se le indica entre 20 a 30 minutos.



1. **Fecha de la entrevista:** 18/08/2022
2. **Nombre:** Verónica Vélez
3. **Profesión:** Licenciada en Terapeuta de Lenguaje
4. **¿Como Logopeda cree usted que una aplicación móvil (App) con ejercicios orales, auditivos, gráficos y vocales aportaría una mejora en los pacientes tratados con dislalia evolutiva?**

Parcialmente de acuerdo

5. **¿Estima usted que los pacientes con dislalia evolutiva pueden apoyarse o cuenta con la destreza para usar una aplicación móvil (APP) sin mayor dificultad?**

Parcialmente de acuerdo

6. **¿Estima usted que una aplicación móvil con ejercicios complementarios para realizarse en casa de manera didáctica, ilustrativa y fácil de usar podrían ser de ayuda al paciente con dislalia?**

Parcialmente de acuerdo

7. **¿Considera usted que una aplicación móvil (APP) podría servir como comunicación continua entre familias de niños con dislalia y especialistas para comunicación del progreso del paciente?**

Parcialmente de acuerdo

8. **¿Cómo terapeuta basado en su experiencia cuantos niveles de dislalia evolutiva considera que se puedan interpretar?**

Puede haber simples como su mismo nombre lo indica puede ser en uno o más fonemas y complejas es cuando es en muchos fonemas y en especial las sílabas trabadas.

- 9. ¿Cómo Terapeuta de pacientes de dislalia que ejercicios habituales sugiere usted para brindar un diagnóstico de dislalia evolutiva a los pacientes y con qué frecuencia debe realizarlos?**

Praxias linguales deben ser realizadas en su mayoría de casos diariamente para que haya una mejor evolución y este sea rápido el proceso de mejora en su articulación.

- 10. ¿Cuáles son los ejercicios que usted como logopeda usa en el proceso de tratamiento de dislalia?**

Praxis linguales Y para diagnosticar se utiliza los diferentes test sean de articulación o zimmerman u otros dependiendo el caso a Evaluar.

- 11. ¿Cuáles son los ejercicios que más sugiere usted para una dislalia leve intermedia o crítica?**

Praxis linguales. El éxito radica en poder tener una mejor movilidad lingual. Ya que cada fonema tiene un punto y modo de articulación diferente.

- 12. ¿Qué ejercicio sugiera usted repasar en casa a sus pacientes dependiendo del nivel de dislalia que presenten ya sea nivel leve intermedio o crítico?**

Los ejercicios dependen de los resultados de la Evaluación. Se le indica al familiar que se debe hacer y ellos en casa.

- 13. ¿Cuál es la modalidad que recomienda a sus pacientes en el repaso de frases y palabras en casa?**

Diariamente. En un lapso de unos 30 a 46 minutos.

Anexo 4

Entrevista a Profesionales en desarrollo de software (Ingenieros, Licenciados)



1. **Fecha de la entrevista:** 14/08/2022
2. **Nombre:** Cristian Calero
3. **Profesión:** Ingeniero en sistemas administrativos computarizados
4. **Posee conocimiento en desarrollo de aplicaciones móviles.**
Poseo medio conocimiento en este campo
5. **Indique los años de experiencia trabajando en temas o proyectos de Desarrollo de aplicaciones móviles.**
Tengo experiencia de 3 años.
6. **Describa brevemente su experiencia o trayectoria laboral en el ámbito de Desarrollos de Aplicaciones Móviles.**
Trabaje 3 años en desarrollo de aplicaciones móviles bajo la herramienta de Aplicaciones Zero Code.
7. **¿Posee conocimientos sobre el lenguaje de Python?**
Poseo conocimientos altos en el lenguaje de programación de Python
8. **¿Conoce usted el framework de kivy del lenguaje de programación Python para desarrollar aplicaciones móviles?**
Poseo bajo conocimientos sobre este framework.
9. **¿Considera usted que el framework de kivy es apropiado para realizar aplicaciones móviles?**
No estoy en acuerdo ni en desacuerdo en desarrollo de aplicaciones mediante este framework.
10. **De haber indicado la opción anterior, indique el motivo de su respuesta.**
No ha habido oportunidad en proyectos de producción, usar esta tecnología.

11. ¿Cómo profesional, en desarrollo de aplicaciones que plataformas sugiere usted que sean las indicadas para realizar aplicaciones móviles?

Flutter, Android studio y React Native

12. De haber indicado la opción anterior, indique el motivo de su respuesta

Mayor acogida en el campo empresarial, y muy bueno para sistemas, informáticos completos.



1. **Fecha de la entrevista:** 15/08/2022
2. **Nombre:** Juan Heredia
3. **Profesión:** Ingeniero en Sistemas
4. **Posee conocimiento en desarrollo de aplicaciones móviles.**
Poseo medio conocimiento en este campo
5. **Indique los años de experiencia trabajando en temas o proyectos de Desarrollo de aplicaciones móviles.**
Tengo experiencia de 1 año.
6. **Describa brevemente su experiencia o trayectoria laboral en el ámbito de Desarrollos de Aplicaciones Móviles.**
1 año de experiencia trabajando con equipos de trabajo mediante células scrum.
7. **¿Posee conocimientos sobre el lenguaje de Python?**
Poseo conocimientos medios en el lenguaje de programación de Python
8. **¿Conoce usted el framework de kivy del lenguaje de programación Python para desarrollar aplicaciones móviles?**
Poseo bajo conocimientos sobre este framework.
9. **¿Considera usted que el framework de kivy es apropiado para realizar aplicaciones móviles?**
No estoy en acuerdo ni en desacuerdo en desarrollo de aplicaciones mediante este framework.
10. **De haber indicado la opción anterior, indique el motivo de su respuesta.**
Tengo bajos conocimientos sobre este framework de desarrollo de aplicaciones
11. **¿Cómo profesional, en desarrollo de aplicaciones que plataformas sugiere usted que sean las indicadas para realizar aplicaciones móviles?**
Flutter, Android studio

12. De haber indicado la opción anterior, indique el motivo de su respuesta

Mayores usos por parte de las empresas por su lenguaje nativo en Flutter y lenguaje Java por parte de Android studio ya que este lenguaje se utiliza mayormente en el desarrollo de sistemas a nivel empresarial.

Anexo 5

Encuesta realizados a padres madres que tienen familiares con dislalia



La Universidad de Guayaquil a través de sus investigadores impulsa la creación de soluciones tecnológicas que buscan ayudar a la comunidad en el corto o mediano plazo, ofreciendo herramientas manera gratuita que apoyen a la comunidad en general.

Por este motivo se solicita su apoyo, con la siguiente encuesta que busca recopilar información necesaria para la construcción de una aplicación móvil (APP) para apoyo a familiares (niño, niñas o joven) con dificultades en el habla.

1. **¿Cuál es la edad del niño, niña o joven que presenta dislalia?**
2. **¿Cuál es el género del niño, niña o joven que presenta dislalia?**
3. **¿En qué país reside el niño, niña o joven con dislalia?**
4. **¿En qué provincia reside el niño, niña o joven con dislalia?**
5. **¿En qué ciudad o cantón reside el niño, niña o joven con dislalia?**
6. **Mencione usted cuál de los tipos de dificultades del habla presenta su familiar (niño o niña, joven)**
7. **¿Considera usted que las terapias de lenguaje presenciales en los centros médicos o unidades de terapias se complementa con ejercicios de terapias de lenguajes enviadas a practicar en casa del niño, niña o joven con dislalia?**
 - Totalmente de Acuerdo
 - Parcialmente de Acuerdo
 - Ni en acuerdo ni en desacuerdo
 - Parcialmente en desacuerdo
 - Totalmente en desacuerdo
8. **¿Cuánto tiempo en minutos diarios su familiar (niño, niña o joven) práctica los ejercicios enviados a casa?**
9. **¿Cuántos días en la semana su familiar (niño, niña o joven) práctica los ejercicios enviados a casa?**

10. ¿Referente a los ejercicios de terapias de lenguajes enviadas a practicar en casa, que nivel de dificultad de comprensión y reproducción del ejercicio enfrenta el familiar (papá, mamá o tutor) al tener que reproducir el ejercicio en casa?

- Muy Fácil comprensión y reproducción
- Fácil comprensión y reproducción
- Ni fácil ni difícil
- Medianamente difícil de comprender y reproducir
- Muy difícil comprensión y reproducción

11. ¿Sabía usted que existen en la actualidad aplicaciones móviles (APP) para el apoyo en las terapias de lenguaje para niños?

- Si tengo mucho conocimiento
- Tengo poco conocimiento
- No tengo conocimiento

12. ¿Considera usted que una aplicación móvil (APP) intuitiva de fácil uso sería un apoyo para las familias cuyos niños o jóvenes necesitan realizar ejercicios de lenguaje de comunicación en sus hogares?

- Totalmente de acuerdo
- Parcialmente de acuerdo
- Ni en acuerdo ni en desacuerdo
- Parcialmente en desacuerdo
- Totalmente en desacuerdo

13. ¿Estaría usted de acuerdo en que se desarrollen nuevas aplicaciones móviles de acceso gratuito para apoyo a las terapias de niños o jóvenes con dislalia?

- Totalmente de Acuerdo
- Parcialmente de Acuerdo
- Ni en acuerdo ni en desacuerdo
- Parcialmente en desacuerdo
- Totalmente en desacuerdo

14. ¿Estima usted que el desarrollo de una aplicación móvil (APP) para el sistema operativo móvil de Android sea recomendable por su facilidad en uso para los niños o jóvenes?

- Totalmente de Acuerdo
- Parcialmente de Acuerdo

- Ni en acuerdo ni en desacuerdo
- Parcialmente en desacuerdo
- Totalmente en desacuerdo

15. ¿Estima usted que su familiar (niño, niña o joven) con dislalia cuentan con las habilidades motrices para interactuar fácilmente con una aplicación móvil (APP) para el apoyo en sus sesiones de terapias?

- Totalmente de Acuerdo
- Parcialmente de Acuerdo
- Ni en acuerdo ni en desacuerdo
- Parcialmente en desacuerdo
- Totalmente en desacuerdo

16. ¿Considera usted que sería de utilidad el poder contar con una aplicación móvil (APP) en la cual el especialista terapéutico pueda preconfigurar un conjunto de ejercicios recomendados para practicarse en casa de tal forma que estos se habiliten para su uso?

- Totalmente de Acuerdo
- Parcialmente de Acuerdo
- Ni en acuerdo ni en desacuerdo
- Parcialmente en desacuerdo
- Totalmente en desacuerdo

17. ¿Considera usted que sería de utilidad que el padre o la madre puedan contar con una aplicación móvil (APP) en la cual en casa puedan elegir el conjunto de ejercicios recomendados por el especialista para practicarse en casa?

- Totalmente de Acuerdo
- Parcialmente de Acuerdo
- Ni en acuerdo ni en desacuerdo
- Parcialmente en desacuerdo
- Totalmente en desacuerdo

18. ¿Considera usted que la App guarde un progreso diario que el niño o joven va alcanzado en el tiempo?

- Totalmente de Acuerdo
- Parcialmente de Acuerdo
- Ni en acuerdo ni en desacuerdo

- Parcialmente en desacuerdo
- Totalmente en desacuerdo

Anexo 6

EXÁMEN LOGOPÉDICO DE LA ARTICULACIÓN ELA - ALBOR

NOMBRE Y APELLIDOS:		
EDAD:	FECHA:	
VALORACION:		
SU = SUSTITUCIÓN	AS= ASIMILACIÓN	SI= SIMPLIFICACIÓN
MODO DE EJECUCIÓN:		
() REPETICIÓN DEL SONIDO		
() IDENTIFICACIÓN DEL OBJETO		
() LECTURA DE LA PALABRA		

	FONEMAS	LISTA DE PALABRAS			INICIAL	MEDIA	FINAL
BILABIALES	/m/	MONO	CAMA	LAMPARA			
	/p/	PELOTA	ESPADA	STOP			
	/b - v/	BOLO VACA	CUBO UVAS	SUBMARINO			
DENTALES	/d/	DADO	NIDO	RED			
	/t/	TORO	BOTA	ROBOT			
	/z/	ZAPATO CERILLAS	LAZO	PEZ			
LARINGO DENTALES	/f/	FOCA	ELEFANTE				
ALVEOLARES	/s/	SILLA	MESA	VELAS CESTA			
	/l/	LUNA	PALA	SOL BOLSO			
	/n/	NARIZ	MANO	PAN MONTAÑA			
	/r/		PERA	TENEDOR ARBOL			
	/rr/	RUEDA	RATÓN	PERRO			
PALATALES	/ch/	CHUPETE	CUCHARA				
	/ll/	LLAVE YOGUR	GALLETA PAYASO				
	/ñ/	ÑU	NIÑA				
VELARES	/k/	CAMPANA QUESO	PICO CARACOL	ACTOR			
	/g/	GATO	TORTUGA				
	/j/	JIRAFÁ	OJO	RELOJ			
	/x/	XILOFONO	TAXI				



T



Z



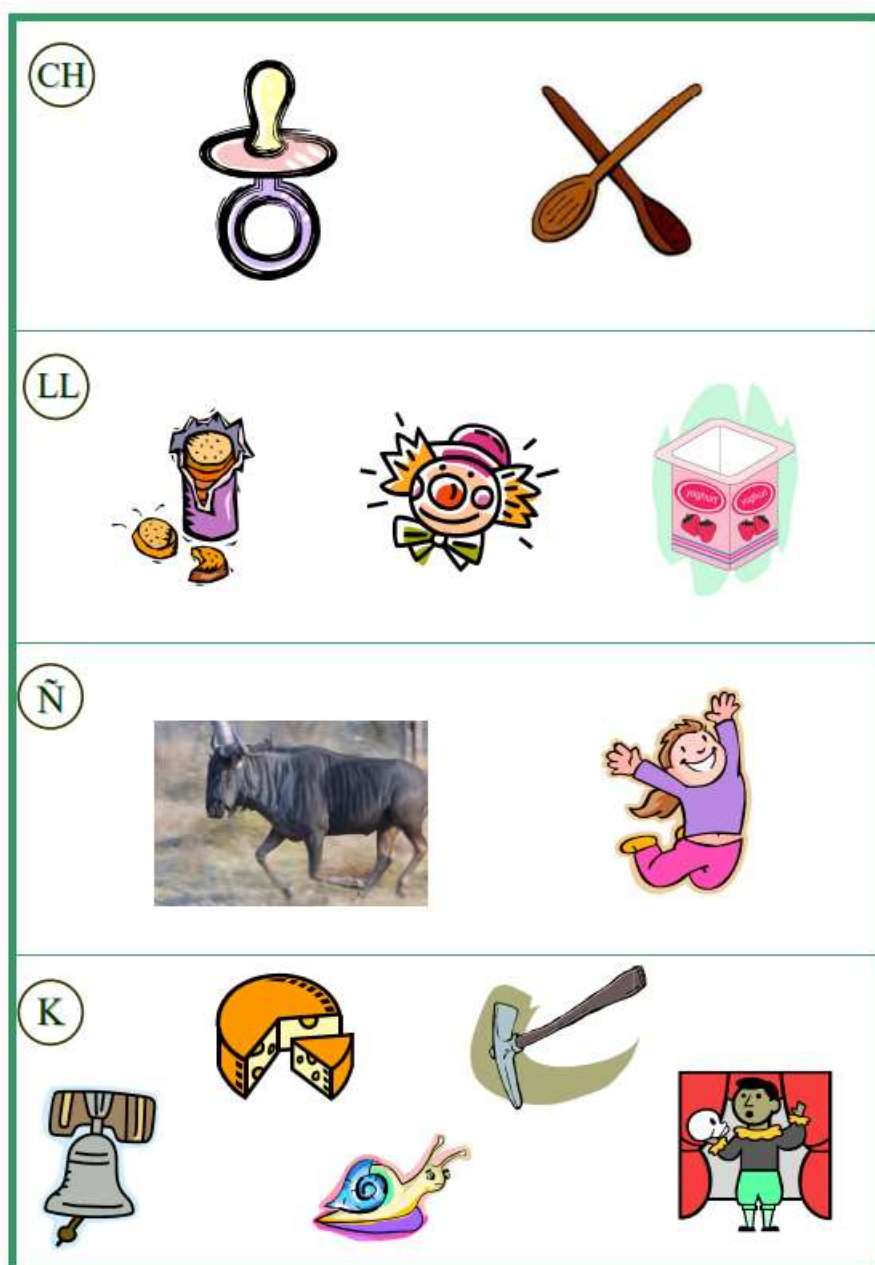
F

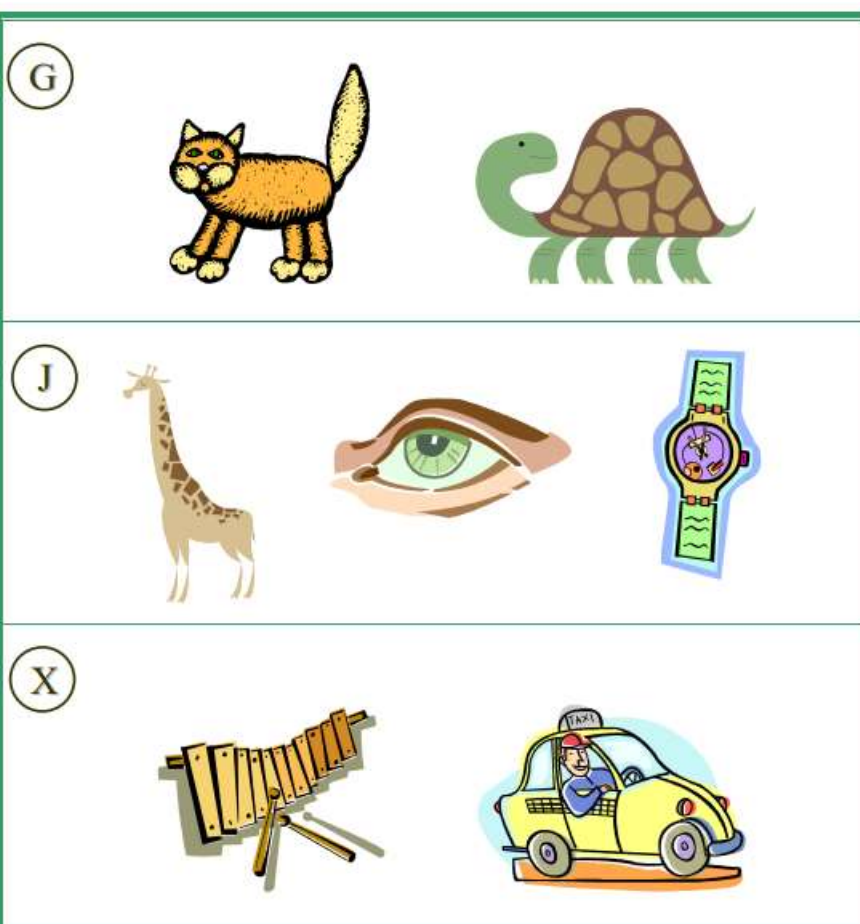


S









Bibliografía

- Acharya, D. P. (2021, 5 noviembre). *Los 11 mejores IDE de Python para potenciar el desarrollo y la depuración*. Geekflare. <https://geekflare.com/es/best-python-ide/>
- Amazon *Primer entorno de desarrollo integrado (IDE) para machine learning - Amazon SageMaker Studio*, Amazon Web Services. (2020, julio 29). Amazon Web Services, Inc. <https://aws.amazon.com/es/sagemaker/studio/>
- Alias, G., & Cassanelli, R. (2019). *NLP aplicado a análisis de texto* (Doctoral dissertation, Universidad Nacional de Mar del Plata. Facultad de Ingeniería. Argentina). Obtenido de: <http://rinfi.fi.mdp.edu.ar/xmlui/handle/123456789/354>
- Alvarado F., A. S., & Rodríguez H, V. E. (2021). Métodos de recuperación para sistema de razonamiento basado en casos de pacientes con dislalia. <http://repositorio.ug.edu.ec/handle/redug/56989>
- Alvaro, (2018, mayo 26). *Tipos de aprendizaje automático* Machinelearningparatodos.com. <https://machinelearningparatodos.com/tipos-de-aprendizaje-automatico/>
- Aponte M., G. & Gómez L, E., & Fernando, N., D. & Betancourt B., L (2014). Metodología para la revisión bibliográfica y la gestión de información de temas científicos, a través de su estructuración y sistematización. Dyna, 81(184),158-163. [fecha de Consulta 2 de septiembre de 2022]. ISSN: 0012-7353. <https://www.redalyc.org/articulo.oa?id=49630405022>
- Apd, R. (2021, 1 diciembre). *El impacto de la inteligencia artificial en las empresas*. APD España. <https://www.apd.es/el-gran-impacto-de-la-inteligencia-artificial-en-las-empresas/>
- Arias Gonzáles, J. L. (2020). Técnicas e instrumentos de investigación científica. <https://repositorio.concytec.gob.pe/handle/20.500.12390/2238>
- Aufar, Y., & Sitanggang, I. S. (2022). Face recognition based on Siamese convolutional neural network using Kivy framework. *Indonesian Journal of Electrical Engineering and Computer Science*, 26(2), 764–772. <https://doi.org/10.11591/ijeecs.v26.i2.pp764-772>
- Banafa, A. (2018, 1 agosto). *¿Qué es el aprendizaje profundo?* OpenMind. <https://www.bbvaopenmind.com/tecnologia/mundo-digital/que-es-el-aprendizaje-profundo/>

- Bhoyarkar, A., Solanki, A., & Balbudhe, A. (2019). Application Development using Kivy Framework. *IJARCCCE*, 8(2). <https://doi.org/10.17148/ijarcce.2019.8209>
- Build from source on windows. (22, mayo 2022). TensorFlow. https://www.tensorflow.org/install/source_windows
- Cabezas, G. S., & Andrade, B. R. (2019). *Propuesta de acciones psicológicas para el afrontamiento de ansiedad en niños con dislalia. Riobamba, 2018-2019.* <http://dspace.unach.edu.ec/handle/51000/6131>
- Cancino, D., Castro, R. y Opazo, K. (2017). Estudio de prevalencia de las dislalias en escolares de 5 a 7 años 11 meses asistentes a la escuela municipal Oscar Castro Zúñiga, al Colegio Particular Subvencionado Liceo Técnico Profesional Jorge Sánchez Ugarte y al Colegio Particular pagado de los Sagrados Corazones, provenientes de Concepción y Hualpén, durante el año 2017 (Tesis de Pregrado). Universidad del Desarrollo, Chile. <http://hdl.handle.net/11447/2097>
- Cass, S. (2017, julio 18). The 2017 top programming languages. IEEE Spectrum. <https://spectrum.ieee.org/the-2017-top-programming-languages>
- Censos, E. D. N. I. Y. (2020). *Tecnologías de la Información y Comunicación-TIC*. Instituto Nacional de Estadística y Censos. <https://www.ecuadorencifras.gob.ec/tecnologias-de-la-informacion-y-comunicacion-tic/>
- Camacho Á., M. Á., & Navarro A., E. (2020). Procesamiento del lenguaje natural con Python Natural language processing with Python. *Procesamiento del lenguaje natural*, 4(13), 24-28. https://www.researchgate.net/publication/346152605_Procesamiento_del_lenguaje_natural_con_Python
- Dager, L., D., & Tomalá B. (2020). Prototipo de aplicativo móvil de terapias del lenguaje, para el tratamiento de Dislalia modulo paciente mediante procesamiento de voz. <http://repositorio.ug.edu.ec/handle/redug/48888>
- de los Santos, P. R. (2018, abril 2). Python para todos (3): ScyPy, NumPy, Pandas ... ¿Qué librerías necesitamos? - Think Big Empresas. Think Big. <https://empresas.blogthinkbig.com/python-todos-3-librerias/>

- de Villa, G. R. (2018, mayo 18). Introducción a Word2vec (skip gram model) - Gonzalo Ruiz de Villa. Medium. <https://gruizdevilla.medium.com/introducci%C3%B3n-a-word2vec-skip-gram-model-4800f72c871f>
- Delgado, C. (2021, 29 marzo). *¿Qué es un IDE (entorno de desarrollo integrado)?* Our Code World. <https://ourcodeworld.co/articulos/leer/1469/que-es-un-ide-entorno-de-desarrollo-integrado>
- Devs, Q. (2019, 8 julio). *Qué es Flutter y por qué utilizarlo en la creación de tus apps.* Quality Devs. <https://www.qualitydevs.com/2019/07/05/que-es-flutter/>
- Descargar los últimos controladores oficiales de NVIDIA.* (2022, agosto 9). Nvidia.es. <https://www.nvidia.es/Download/index.aspx?lang=es>
- Díaz C, K. R., & Miranda M, F. R. (2017). *Dislalia funcional y su relación con la lectoescritura estudio realizado en el Centro de Atención Comunitaria de las Carreras de Tecnológicas Médicas.* <http://repositorio.ug.edu.ec/handle/redug/36812>
- Díaz, A. A., de la Rosa Adriana, M. R., & Esaú, G. D. V. T. (2019, septiembre). Herramienta automática para diferenciar zonas dialectales de México en Twitter. <https://gabyrr.github.io/students/>
- Dick, S. (2019, 1 julio). *Artificial Intelligence · Issue 1.1, Summer 2019.* Harvard Data Science Review. <https://hdsr.duquduq.org/pub/0aytgrau/release/1>
- Español, A. C. (2020, 9 diciembre). *Dislalia evolutiva: cómo afecta una situación de crisis.* ACNUR. https://eacnur.org/blog/dislalia-evolutiva-tc_alt45664n_o_pstn_o_pst/
- Esteban. (2021, 27 abril). Android Studio: esto es lo que puedes hacer con este programa. El Español. https://www.elespanol.com/elandroidelibre/tutoriales/20200117/android-studio-puedes-hacer-programa/460455295_0.html
- Fernández, L. A. U. (2021, 9 diciembre). *Reducir el número de palabras de un texto: lematización y radicalización (stemming) con Python.* Medium. <https://medium.com/qu4nt/reducir-el-n%C3%BAmero-de-palabras-de-un-texto-lematizaci%C3%B3n-y-radicalizaci%C3%B3n-stemming-con-python-965bfd0c69fa#:~:text=La%20lematizaci%C3%B3n%20es%20un%20proceso,algunos%20casos%20obtendremos%20resultados%20inesperados.>
- Ferreira, J. G. (2022, 27 mayo). *Análisis de la inteligencia artificial en las relaciones laborales / CES Derecho.* Revista CES Derecho. <https://revistas.ces.edu.co/index.php/derecho/article/view/6395>

- Figueiras, S. (2022, 20 septiembre). *¿Conoces Jupyter Notebook?* Ceupe.mx. Disponible en: <https://www.ceupe.mx/blog/conoces-jupyter-notebook.html>
- García F, I. (2018). Estudio de Word Embeddings y métodos de generación de Meta Embeddings. Obtenido de: <https://addi.ehu.es/handle/10810/29088>
- Geographic. (2020, 2 diciembre). *Breve historia visual de la inteligencia artificial*. www.nationalgeographic.com.es.
https://www.nationalgeographic.com.es/ciencia/breve-historia-visual-inteligencia-artificial_14419
- Gn, C. K. (2018, agosto 31). Artificial intelligence: Definition, types, examples, technologies. Medium. <https://chethankumargn.medium.com/artificial-intelligence-definition-types-examples-technologies-962ea75c7b9b>
- González, L. (2021, 8 septiembre). *Librería NumPy*. Aprende IA. <https://aprendeia.com/libreria-numpy/>
- Heras, J. M. (2019, enero 31). Regresión Lineal: teoría y ejemplos en Python. Iartificial.net. <https://www.iartificial.net/regresion-lineal-con-ejemplos-en-python/>
- Ghelani, S. (2021, 10 diciembre). *From Word Embeddings to Pretrained Language Models — A New Age in NLP — Part 2*. Medium. <https://towardsdatascience.com/from-word-embeddings-to-pretrained-language-models-a-new-age-in-nlp-part-2-e9af9a0bdcd9>
- Hamdy, E. (2021, 30 mayo). *Neural Models for Offensive Language Detection*. arXiv.Org. <https://arxiv.org/abs/2106.14609>
- Heras, J. M. (2020, 10 octubre). *15 librerías de Python para Machine Learning*. IArtificial.net. <https://www.iartificial.net/librerias-de-python-para-machine-learning/>
- Hernández, R., Fernández, C., & Baptista, P. (2017). Metodología de la investigación. 6ta Edición Sampieri. Soriano, RR (1991). *Guía para realizar investigaciones sociales*. Plaza y Valdés. <https://www.uca.ac.cr/wp-content/uploads/2017/10/Investigacion.pdf>
- Hernández, E. G. (2012). *Historia de C++*. Programacion C++. <http://ejercicioscpp.blogspot.com/2012/09/historia-de-c.html>
- Hofesmann, E. (2021, 28 diciembre). *The Machine Learning Lifecycle in 2021 - Towards Data Science*. Medium. <https://towardsdatascience.com/the-machine-learning-lifecycle-in-2021-473717c633bc>

- Horev, R. (2018, 17 noviembre). *BERT Explained: State of the art language model for NLP*. Medium. <https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>
- Jablonski, J. (2021, 18 febrero). *Natural Language Processing With Python's NLTK Package*. Real Python. <https://realpython.com/nltk-nlp-python/>
- Janiesch, C., Zschech, P. & Heinrich, K. Machine learning and deep learning. *Electron Markets* **31**, 685–695 (2021). <https://doi.org/10.1007/s12525-021-00475-2>
- Juan, R. Q., & Mario, C. M. (2011). Redes neuronales artificiales para el procesamiento de imágenes, una revisión de la última década. *RIEE&C, Revista de Ingeniería Eléctrica, Electrónica y Computación*, 9(1), 7-16. Obtenido de: https://www.itson.mx/publicaciones/rieeyc/Documents/v9/art2vf_redes_neuronales_artificiales_para_el_procesamiento_de_imagenes-una_revision_de_la_ultima_decada.pdf
- K. R., S. (2017). Python -The Fastest Growing Programming Language. *International Research Journal of Engineering and Technology*, 4(12). <https://www.irjet.net/archives/V4/i12/IRJET-V4I1266.pdf>
- Khanna, C. (2022, 7 enero). *Sampling Techniques in Statistics | Chetna | Towards Data Science*. Medium. <https://towardsdatascience.com/sampling-techniques-in-statistics-9c77a39e0948>
- Kulshrestha, R. (2021, 15 diciembre). *Transformers in NLP: A beginner friendly explanation | Towards Data Science*. Medium. <https://towardsdatascience.com/transformers-89034557de14>
- Lojo V, J. D. (2012). Clasificación automática de documentación clínica. <http://hdl.handle.net/2183/10051>
- López, B. R. (2021, 21 junio). *¿Qué es Google Colab? Cursos GIS | TYC GIS Formación*. <https://www.cursosgis.com/que-es-google-colab/>
- programador clic. Programmerclick.com. <https://programmerclick.com/article/79291185590/>
- Lloret E., J. (2019). Procesamiento del Lenguaje Natural (PLN) | Natural Language Processing (NLP). 10.13140/RG.2.2.30635.39205. Obtenido de: https://www.researchgate.net/publication/337172055_Procesamiento_del_Lenguaje_Natural_PLN_Natural_Language_Processing_NLP

- Los mejores frameworks web en Python.* (2020, agosto 12). El Pythonista. <https://elpythonista.com/los-mejores-frameworks-web-en-python>
- Mamani LL, M. R. (2019). Capacitación en diagnóstico e intervención psicopedagógica de la dislalia funcional. *Horizontes. Revista de Investigación En Ciencias de La Educación*, 3(10). <https://doi.org/10.33996/revistahorizontes.v3i10.73>
- Más, L. Y. (2022, 19 junio). *¿Qué es la Dislalia?* Logopedia y Más. <https://www.logopediaymas.es/blog/dislalia/>
- Medina, V. (2019, 24 julio). *Tratamiento de la dislalia infantil.* Guiainfantil.com. <https://www.guiainfantil.com/1044/tratamiento-de-la-dislalia-infantil.html>
- Meneses, J. (2011b, septiembre 22). *El cuestionario y la entrevista.* Julio Meneses. <https://femrecerca.cat/meneses/publication/cuestionario-entrevista/>
- México, R. C. (2021, 7 mayo). *Procesamiento del Lenguaje Natural: ¿qué es, ¿dónde se aplica, ¿dónde aprenderlo y cuánto gana un especialista en PNL?* CIO MX. <https://cio.com.mx/procesamiento-del-lenguaje-natural-que-es-donde-se-aplica-donde-aprenderlo-y-cuanto-gana-un-especialista-en-pnl/>
- Microsoft. (2022, 25 agosto). *¿Qué es Azure Machine Learning? - Azure Machine Learning.* Microsoft Docs. <https://docs.microsoft.com/es-es/azure/machine-learning/overview-what-is-azure-machine-learning>
- Mindiola V, C. (2020). Guía metodológica para la atención a escolares con dislalia funcional en el subnivel básica elemental. *Ciencia Digital*, 4(4.1). <https://doi.org/10.33262/cienciadigital.v4i4.1.1452>
- Moreno G, R. (2020). Las habitaciones de la dislalia. *ReiDoCrea: Revista Electrónica de Investigación Docencia Creativa.* <https://digibug.ugr.es/handle/10481/21941>
- Nacimba, V. (2012). Como afecta la dislalia en el desarrollo de los prerrequisitos para el aprendizaje de la lecto-escritura en los niños de primero de básica del centro infantil EMDI de La ciudad de Quito, Parroquia el Tingo, durante el año lectivo 2010-2011. <http://www.dspace.uce.edu.ec/handle/25000/286>
- Otzen, T., & Manterola, C. (2017). Técnicas de Muestreo sobre una Población a Estudio. *International Journal of Morphology*, 35(1). <https://doi.org/10.4067/S0717-95022017000100037>

- Palabras. (2016, junio 14). Palabras. Recuperado el 9 de septiembre de 2022, de <https://palabras-con.org/>
- Polivio, O. (2021, 30 junio). *Librería Pandas y Python*. IngeniusWorlds. <https://www.ingeniusworlds.com/libreria-pandas-y-python/>
- Peña U, I. (2018). *Propuesta de intervención para un niño con dislalias funcionales*. <http://uvadoc.uva.es/handle/10324/32371>
- Ramos, R. (2021, 25 septiembre). *¿Qué es JavaScript y para qué sirve?* Agencia de Marketing Digital Sevilla - Rafa Ramos. <https://soyrafamos.com/que-es-javascript-para-que-sirve/>
- Reclu IT. (2020, 18 diciembre). *¿Qué es el Procesamiento Natural del Lenguaje?* Reclu iT. <https://recluit.com/que-es-el-procesamiento-natural-del-lenguaje/#.YxK1V3bMJPb>
- Robledano, A. (2021, 24 agosto). *Qué es C++: Características y aplicaciones*. OpenWebinars.net. <https://openwebinars.net/blog/que-es-cpp/>
- Rodriguez, E. C. (2016). *Alumnos con dislalia: evaluación e intervención*. España: Area de Educacion Especial. <https://educrea.cl/alumnos-con-dislalia-evaluacion-e-intervencion/>
- Rodrigo, J. A. (2017, abril). *Máquinas de Vector Soporte (Support Vector Machines, SVMs)*. [cienciadedatos.net](https://www.cienciadedatos.net). https://www.cienciadedatos.net/documentos/34_maquinas_de_vector_soporte_support_vector_machines
- Rong, X. (2014). word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*. <https://arxiv.org/abs/1411.2738>
- Rytikova, V. O., & Medvedev, M. A. (2021). Statistical data and programming languages analysis for developing a chatbot in telegram. THERMOPHYSICAL BASIS OF ENERGY TECHNOLOGIES (TBET 2020).
- Shamrat, F. M. J. M., Chakraborty, S., Imran, M. M., Muna, J. N., Billah, M. M., Das, P., & Rahman, M. O. (2021). Sentiment analysis on twitter tweets about COVID-19 vaccines using NLP and supervised KNN classification algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*, 23(1). <https://doi.org/10.11591/ijeecs.v23.i1.pp463-470>

- Solorzano, M. M. (Abril de 2022). *Repositorio Universidad de Guayaquil*. Obtenido de repositorio.ug.edu.ec:<http://repositorio.ug.edu.ec/bitstream/redug/59539/1/Solorzano%20Monserrate%20Mirian%20Estefania-1.pdf>
- School, T. (2022, 29 julio). *Cómo programar en PyCharm*. Tokio School. <https://www.tokioschool.com/noticias/como-programar-en-pycharm/>
- Suárez L. C. ; Escudero C. S; Margarita. (2018). *Técnicas y métodos cualitativos para la investigación científica*. Utmach. <http://repositorio.utmachala.edu.ec/bitstream/48000/12501/1/Tecnicas-y-MetodoscualitativosParaInvestigacionCientifica.pdf>
- Tineo, A., Angelica, M., Castillo J., C., Santos L., S., Z., Giannina Q. P., N., & Lima -Perú, E. (2019). *INVESTIGACIÓN FACULTAD DE TECNOLOGIA MÉDICA*. <http://repositorio.unfv.edu.pe/handle/UNFV/4356>
- Tomić, J. (2018, 14 febrero). *C# vs. C++: What's at the Core?* Toptal Engineering Blog. <https://www.toptal.com/c-sharp/c-sharp-vs-c-plus-plus>
- Tokio. (2021, 30 agosto). *C#, ¿qué es? ¿Quién lo inventó? ¿Conoce este lenguaje de programación!* Tokio School. <https://www.tokioschool.com/noticias/c-que-es/#:%7E:text=Andr%C3%A9s%20Hejlsberg%20decidi%C3%B3%20formar%20un,de%20programaci%C3%B3n%20orientado%20a%20objetivos.>
- Toro, L. (2017, 7 septiembre). *Kivy: Un framework para Python que permite desarrollar aplicaciones de manera rápida*. Desde Linux. <https://blog.desdelinux.net/kivy-framework-para-python/>
- Test ELA: Examen Logopédico de ARTICULACION en Tests-gratis.com. (2020, octubre 13). Tests-gratis.com. https://www.tests-gratis.com/tests_lenguaje/test-ela-examen-logopedico-de-articulacion.htm
- Telegrafo. (25 de septiembre de 2019). 2.621 niños superan las dificultades de aprendizaje en los planteles. *Diario el Telegrafo*, pág. 1. <https://www.eltelegrafo.com.ec/noticias/sociedad/6/aprendizaje-planteles-ecuador>
- Walton, A. (2020, 26 junio). *Breve Historia de Java: Características y Aplicaciones*. Java desde Cero. <https://javadesdecero.es/fundamentos/breve-historia-caracteristicas-y-aplicaciones/>

- Webster, J. J., & Kit, C. (1992). Tokenization as the initial phase in NLP. In *COLING 1992 volume 4: The 14th international conference on computational linguistics*. Obtenido de: https://www.researchgate.net/publication/221102283_Tokenization_as_the_initial_phase_in_NLP
- Wu, Gao, & Jiao, (2019). Multi-Label Classification Based on Random Forest Algorithm for Non-Intrusive Load Monitoring System. *Processes*. 7. 337. 10.3390/pr7060337.