



UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE INGENIERÍA DE SISTEMAS

GENERAR UN REPOSITORIO DE MATERIALES DIDÁCTICOS PARA LA
RECOMENDACIÓN DE RECURSOS EDUCATIVOS UTILIZANDO BASES DE DATOS
ORIENTADAS A GRAFOS A NIÑOS DE 5- 7 AÑOS CON Y SIN DISCAPACIDAD

Trabajo de titulación previo a la obtención del
título de Ingeniero de Sistemas

AUTORES: MALKI KATARI YUPANKI MEDINA

GABRIEL LEONARDO CHUCHUCA AREVALO

TUTOR: ING. DIEGO FERNANDO QUISI PERALTA, MSC.

Cuenca - Ecuador

2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Malki Katari Yupanki Medina con documento de identificación N° 0150350668 y Gabriel Leonardo Chuchuca Arevalo con documento de identificación N° 0105662068; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 30 de agosto del 2022

Atentamente,



Malki Katari Yupanki Medina
0150350668



Gabriel Leonardo Chuchuca Arevalo
0105662068

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Malki Katari Yupanki Medina con documento de identificación N° 0150350668 y Gabriel Leonardo Chuchuca Arevalo con documento de identificación N° 0105662068, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto técnico: “Generar un repositorio de materiales didácticos para la recomendación de recursos educativos utilizando bases de datos orientadas a grafos a niños de 5- 7 años con y sin discapacidad”, el cual ha sido desarrollado para optar por el título de: Ingeniero de Sistemas, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercerplenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos entrega del trabajo en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 30 de agosto del 2022

Atentamente,



Malki Katari Yupanki Medina
0150350668



Gabriel Leonardo Chuchuca Arevalo
0105662068

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Diego Fernando Quisi Peralta con documento de identificación N° 0104616461, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **GENERAR UN REPOSITORIO DE MATERIALES DIDÁCTICOS PARA LA RECOMENDACIÓN DE RECURSOS EDUCATIVOS UTILIZANDO BASES DE DATOS ORIENTADAS A GRAFOS A NIÑOS DE 5- 7 AÑOS CON Y SIN DISCAPACIDAD**, realizado por Malki Katari Yupanki Medina con documento de identificación N° 0150350668 y por Gabriel Leonardo Chuchuca Arevalo con documento de identificación N° 0105662068, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requerimientos determinados por la Universidad Politécnica Salesiana.

Cuenca, 30 de agosto del 2022

Atentamente,



Ing. Diego Fernando Quisi Peralta, MSc.
0104616461

DEDICATORIA

Este trabajo esta dedicado a mi madre Maria Medina por sus consejos y su apoyo incondicional para seguir adelante y lograr este objetivo que es muy importante en mi vida. A mis abuelos Manuel Medina y Rosa Seraquive quienes con su apoyo y palabras de aliento me ayudaron a no decaer y seguir esforzándome cada día. A mi compañeros y amigos quienes me ayudaron y apoyaron en diferentes momentos, sin esperar nada a cambio.

Malki Katari Yupanki Medina

Dedico este proyecto de titulación primeramente a Dios por darme la fuerza necesaria para poder luchar día a día y no rendirme ante las adversidades diarias permitiéndome seguir adelante en todo este proceso. A mis padres Juan Fernando Chuchuca Orellana y Maria Mercedes Arevalo Japa por todo ese apoyo incondicional, sus consejos sabios y por luchar día a día por mi bienestar para poder llegar a la meta que hoy en día me he propuesto. A mis hermanos mayores Juan Fernando, Wilson Rodrigo, Johanna Estefania y Christian Patricio por todos los valores inculcados para ser una persona de bien.

Gabriel Leonardo Chuchuca Arevalo

AGRADECIMIENTO

Quiero agradecer primeramente a Dios por su guía y permitirme cumplir una meta mas en mi vida.

Agradezco a mi madre Maria Medina por su cariño y consejos que me han permitido siempre seguir adelante cumpliendo mis sueños y metas en la vida.

Agradezco a toda mi familia que me han apoyado y estado presente en los momentos mas importantes de mi vida, por apoyarme en los momentos que mas los necesite, motivándome a seguir siempre adelante.

Agradezco a la Universidad Politécnica Salesiana y sus docentes que me han guiado y brindado sus conocimientos para poder cumplir esta meta y realizar el presente trabajo de titulación.

Malki Katari Yupanki Medina

Agradezco mucho a mi Dios por darme salud para poder cumplir esa meta por la que estoy luchando.

Agradezco a la Universidad Politécnica Salesiana y a todos los docentes que año tras año nos han inculcado con sus enseñanzas.

Agradezco a mi familia entera por se un apoyo incondicional desde mi primer dia en la Universidad hasta el largo proceso que he pasado para poder terminar el proceso exitosamente.

Agradezco al docente tutor Diego Quisi que nos ha brindado la confianza para poder realizar y desarrollar este trabajo de titulación y sobre todo un amigo dentro de mi carrera universitaria.

Gabriel Leonardo Chuchuca Arevalo

Resumen

En la actualidad la información que se genera cada día esta a disposición en diferentes páginas, bases de datos, libros y archivos. Estos almacenan gran información relevante para cada área. Esta información puede ser obtenida y analizada para ser utilizada de acuerdo a diferentes tópicos, necesidades y usuarios.

Por otro lado, la forma de aprendizaje en niños se complementa con diferentes recursos como son: vídeos, libros, cuentos y juegos lo que permite aplicar una enseñanza más dinámica y efectiva. De igual manera la forma de distribuir información y recursos se utiliza las redes sociales las cuales permiten un acceso más rápido y dinámico. Para fomentar el aprendizaje y desarrollo el uso de los smartphones cada vez es más frecuente por lo que se tienen aplicaciones educativas que deben ser canalizadas de forma adecuada.

En consecuencia, el propósito es crear un repositorio de recursos educativos empleando nuevas tecnologías como una base de datos no relacional y una base de datos orientada a grafos que permita realizar recomendaciones de recursos específicos. Finalmente, a través de una encuesta a 10 usuarios, donde se obtuvo resultados positivos sobre las recomendaciones de recursos presentados para diferentes actividades con el uso de un sistema recomendador basado en contenido se determinó las diferentes relaciones entre los recursos y actividades teniendo como resultado un repositorio que genera recomendaciones de acuerdo a una actividad y el contenido de los diferentes recursos almacenados.

Palabras clave: Base orientada a grafos, sistema recomendador, repositorio, Discapacidad de niños, recursos.

Abstract

Nowadays, the information that is generated every day is available in different pages, databases, books and archives. These store a great deal of information relevant to each area. This information can be obtained and analyzed to be used according to different topics, needs and users.

On the other hand, the way of learning in children is complemented with different resources such as: videos, books, stories and games, which allows to apply a more dynamic and effective teaching. Likewise, the way of distributing information and resources uses social networks which allow a faster and more dynamic access. In order to promote learning and development, the use of smartphones is becoming more and more frequent, so there are educational applications that must be channeled appropriately.

Consequently, the purpose is to create a repository of educational resources using new technologies such as a non-relational database and a graph-oriented database to make recommendations of specific resources. Finally, through a survey of 10 users, where positive results were obtained on the resource recommendations presented for different activities with the use of a content-based recommender system, the different relationships between resources and activities were determined, resulting in a repository that generates recommendations according to an activity and the content of the different resources stored.

Keywords: Network-oriented database, recommender system, repository, child disability, resources.

ÍNDICE

I	Introducción	11
II	Problema	13
2.1	Antecedentes	13
2.2	Importancia y alcances	14
III	Objetivos Generales y Específicos	16
3.1	Objetivo General	16
3.2	Objetivos Específicos	16
IV	Revisión de la literatura o fundamentos teóricos	18
4.1	Sistemas de Recomendación	18
4.1.1	Basado en Contenido	19
4.1.2	Medidas de similitud entre textos	20
4.2	Recursos Educativos y Didácticos	21
4.3	Base de Datos	22
4.3.1	Base de datos NoSql	23
4.3.1.1	MongoDB	23
4.3.2	Base de datos Orientada a Grafos	24
4.3.2.1	Neo4j	25
4.4	Aplicaciones Web y Movil	27

4.4.1	Angular	29
4.4.1.1	Características de Angular	29
4.5	FastApi	31
4.5.1	Características	31
4.6	Docker	32
4.6.1	Dockerfile	34
4.6.2	Docker Compose	34
V	Marco metodológico	35
5.1	ANÁLISIS Y REQUERIMIENTOS	35
5.1.1	Requerimientos Funcionales	35
5.1.2	Requisitos No Funcionales	39
5.2	METODOLOGÍA	40
5.2.1	Metodología de Desarrollo ágil SCRUM	40
5.2.1.1	Roles	41
5.2.1.2	Artefactos	42
5.2.1.3	Actividades	42
5.2.1.4	Fases de SCRUM	43
5.3	Fases	44
5.3.1	Primera fase	46
5.3.1.1	Estado del arte	46
5.3.2	Segunda fase	47
5.3.3	Tercera fase	56
5.3.4	Cuarta Fase	59
5.3.5	Quinta Fase	63
5.4	Arquitectura	68
5.4.1	Capa de base de datos	69

5.4.2	Capa de Negocio	70
5.4.3	Capa de presentación	70
VI	Resultados	73
6.1	Resultados de actividades y recursos	73
6.1.1	Actividades con recursos	74
6.2	Pruebas Unitarias	75
6.2.1	Validación Web y Móvil	76
6.3	Análisis de Encuesta	77
VII	Cronograma	84
VIII	Presupuesto	93
IX	Conclusiones	95
X	Recomendaciones	98
	Anexos	103
A	Formato de la encuesta	104
1.1	Información de la aplicación	104

Índice de tablas

1	Ecuacion de Term frequency	21
2	Ecuacion de Inverse document frequency	21
3	Ecuacion de TF-IDF	21
4	Requisito funcional para registro de usuarios	36
5	Requisito funcional para inicio de sesión	36
6	Requisito funcional para cerrar cesión	37
7	Requisito funcional para visualización de actividades	37
8	Requisito funcional para visualización de recursos educativos	38
9	Requisito funcional para creación de actividades	38
10	Requisito funcional para recomendaciones de actividades	39
11	Requisito no funcional para diseño responsivo	39
12	Requisito no funcional para tiempo de respuesta	40
13	Roles del proyecto según SCRUM	44
14	Conexión a Mongo	60
15	Conexión a Neo4j	60
16	Respuestas de la Pregunta 1	77
17	Respuestas de la Pregunta 2.	78
18	Respuestas de la Pregunta 3	79
19	Respuestas de la Pregunta 4	80

20	Respuestas de la Pregunta 5	81
21	Clasificación de los niveles de fiabilidad según el Alfa de Cronbach	82
22	Tabla de resultados de varianza de la encuesta.	83

Índice de figuras

1	Recomendación basado en contenido (Quanam, sf)	20
2	Clasificación DB-Engines - Tendencia de la popularidad de Graph DBMS (DB-Engines, 2022)	25
3	Nodos y relaciones segun (Neo4j, 2022)	26
4	Nodos como objetos y sus atributos segun (Neo4j, 2022)	27
5	Relaciones entre nodos y sus propiedades según (Neo4j, 2022)	27
6	Arquitectura de Docker	33
7	Sprints del proyecto en base a los objetivos	45
8	Página del Cuaderno de Trabajo de primero año (de Educación del Ecuador, 2016)	48
9	Página del Cuaderno de Trabajo para niños con PCI (Saens and Serna, 2019) . .	49
10	Actividad almacenada en MongoDB	50
11	Actividad almacenada en Neo4j	50
12	Búsqueda de actividad en Google	51
13	Cuento recuperado de (MARYLAND, sf)	52
14	Cuento almacenado en MongoDB	52
15	Juego almacenado en MongoDB	53
16	Aplicación en la versión web de Play Store	53
17	Parámetros para obtener app	54
18	Aplicación almacenada en MongoDB	54

19	Panel de Twitter developer	55
20	Autenticación de Twitter desde Python	55
21	Algoritmo TD-IDF en Python	56
22	Actividades similares en Neo4j	57
23	Librería SentenceTransformer	57
24	Actividades y vídeos similares en Neo4j	58
25	Nodos de actividades y recursos en Neo4j	58
26	Nodo de perfiles y enfermedad en neo4j	59
27	Distribución del proyecto en FastAPI	60
28	Métodos para generar un token	61
29	Dockerfile FastAPI	62
30	Proyecto Angular generado	63
31	Pantalla de inicio de sesión	64
32	Pantalla de registro	64
33	Tabla de Diagnóstico Médico	65
34	Tabla de Diagnóstico Lenguaje	65
35	Pantalla del Listado de actividades	66
36	Pantalla del Listado de Recursos en base a una actividad	66
37	Listado de las actividades recomendadas en base a una actividad	67
38	Pantalla de Creación de una Actividad	67
39	Pantalla de Creación de Actividades mediante un archivo csv	68
40	Arquitectura del sistema	69
41	Diagrama Patron MVVM	71
42	Diagrama Estructura Proyecto	72
43	Estructura Proyecto Angular	72
44	Actividades en MongoDB	73

45	Recursos en MongoDB	74
46	Actividades y sus recursos en Neo4j.	74
47	API-Rest Login del sistema	75
48	API-Rest Lista de recursos	76
49	Representación porcentual pregunta 1	78
50	Representación porcentual pregunta 2	79
51	Representación porcentual pregunta 3.	80
52	Representación porcentual pregunta 4	81
53	Representación porcentual pregunta 5.	82

Capítulo I

Introducción

Actualmente los estudiantes realizan búsquedas de materiales que permitan la resolución de sus actividades o ejercicios, el problema se da que existen gran cantidad de información disponible en la Web, y cada día se genera nueva información. Con base en esto se da la iniciativa de los sistemas que analizan el contenido de los diferentes recursos y permiten generar recomendaciones, lo que se conocen como sistemas de recomendación basados en contenidos.

Según (Tariq and Francesco, 2009) "los sistemas de recomendación son aplicaciones inteligentes que permiten a los usuarios realizar búsquedas con resultados más óptimos, sugiriendo aquellos elementos que más se acerquen a sus necesidades y preferencias".

La información que se genera cada día hace más complejo manejarse por lo que surgen nuevas formas de almacenamiento de datos como son las bases de datos NoSQL. Este tipo de base de datos se ha dado en tendencia en los últimos años permitiendo satisfacer las necesidades como son la escalabilidad, permitiendo proporcionar base de datos más eficientes (Castro et al., 2012).

Dentro de las bases de datos NoSQL se encuentran la base de datos orientada a grafos, este tipo de base de datos permiten el almacenamiento de datos en forma de nodos, una de las características de esta base de datos es que mejora el rendimiento a la hora de realizar consultas y permite aplicar diferentes algoritmos propios de la base de datos orientada a grafos.

Los repositorios permiten el almacenamiento de contenidos de manera digital los recursos educativos existentes en la web, estos recursos son accesibles vía internet y residen en servidores específicos por ejemplo: recursos educativos, objetos de aprendizaje, programas/software, etc. (Mortera and Ramírez, 2013).

Al conocer estas nuevas formas de desarrollo, se pretende construir un sistema que permita emplear las capacidades de las nuevas tecnologías para formar un repositorio. Este sistema consiste en una forma de obtener recursos y almacenarlos para luego ser presentado de forma efectiva aplicando recomendaciones de acuerdo a las necesidades y perfil de un usuario.

Capítulo II

Problema

En este capítulo se presenta el problema que se analizará y el cual se pretende resolver con el desarrollo del proyecto. Se analiza los antecedentes, la importancia del mismo y el alcance que se va a conseguir. Se plantea el uso de los repositorios en la actualidad para el almacenamiento de recursos, el uso de los sistemas recomendadores basado en contenido utilizados para generar recomendaciones así como el almacenamiento y manejo de la información empleando base de datos NoSQL.

Esto nos permite definir el problema del uso de los recursos educativos y los repositorios que nos permitan generar recomendaciones de acuerdo al contenido de los mismos enfocados en mejorar y apoyar el aprendizaje de los estudiantes.

2.1 Antecedentes

Un sistema de recomendación (SR) es considerado una herramienta que genera recomendaciones de un recurso u objeto de estudio, para esto el SR puede realizar una recomendación basándose en el contenido del recurso como el perfil del usuario. (Viedma et al., 2004). Estos sistemas en la actualidad han sido aplicados por diferentes empresas en sus sitios Web como lo hace Google, Amazon y Netflix. Los sistemas de recomendación que emplean las diferentes

empresas permiten generar recomendaciones basándose en los gustos del usuario así como el contenido de sus productos.

Los recursos educativos es cualquier material dentro de un contexto educativo, que sea utilizado con la finalidad de apoyar al desarrollo de cualquier actividad escolar, enseñanza o aprendizaje (González and Luz, 2011). Los recursos educativos en la actualidad pueden encontrarse en la Web, estos pueden o no estar etiquetados o clasificados como educativos.

Existen muchos recursos educativos disponibles en la Web, estos recursos son utilizados tanto por estudiantes como por profesores para actividades educativas y didácticas. No todos los recursos pueden servir para una interacción efectiva de aprendizaje por lo que deben ser producidos y utilizados desde una perspectiva educativa y pedagógica.

Los repositorios permiten el almacenamiento de recursos académicos enfocados a instituciones de educación, que ofrecen el manejo y almacenamiento efectivo de la información y documentación. Estos repositorios son enfocados en almacenamiento de publicaciones, tesis, artículos, libros etc. Cada repositorio tiene un enfoque de acuerdo a las necesidades por las que fue desarrollado (Mortera and Ramírez, 2013).

2.2 Importancia y alcances

Los sistemas recomendadores permiten filtrar de una manera efectiva información, para lo cual implementan diferentes métodos los cuales permiten realizar búsquedas de acuerdo a diferentes parámetros y realizar recomendaciones adecuadas para un usuario.

El crecimiento de la información en la Web cada vez se hace extenso, de igual forma se ve afectada la forma de gestionar los datos que se generan a diario. Este problema se ve reflejado en las búsquedas que se realizan en internet donde los sistemas recomendadores se han empleado como herramientas efectivas para filtrar información (Castro et al., 2012).

El sistema se enfoca en la recuperación de recursos educativos analizando el contenido de acuerdo a actividades definidas en el libro del ministerio de educación para niños de primer año

de educación básica. La recopilación de actividades para personas con capacidades especiales se obtienen de manuales desarrollados para una capacidad especial. En donde los usuarios podrán acceder a las actividades y complementarlas con los recursos educativos.

La formación del repositorio de recursos educativos se realiza sobre la base del sistema que maneja actividades y almacena recursos enfocados en cada actividad, esto permite completar y desarrollar las actividades guardadas en el repositorio. El acceso al repositorio se realiza mediante una aplicación web y móvil para la visualización de las actividades y sus recursos.

Empleando nuevas tecnologías que permiten agilizar el desarrollo del software para una optimización en el proceso de diseño e implementación para alcanzar los resultados con mayor precisión y consultas efectivas. Se empleó una base de datos orientada a grafos que permite obtener recomendaciones empleando algoritmos propios de la base y empleando APIs que posibilitan el acceso y la comunicación con este tipo de base de datos.

Capítulo III

Objetivos Generales y Específicos

A continuación se presentan los objetivos que se van a seguir durante el desarrollo del proyecto. Esta sección está conformada por los objetivos generales y específicos, los cuales nos permitirán resolver y realizar un adecuado desarrollo del proyecto enfocado en resolver el problema definido en el capítulo anterior. Los objetivos del proyecto están enfocados en generar un repositorio de recursos educativos y obtener recomendaciones empleando un sistema recomendador basado en contenido y una base de datos orientada a grafos.

3.1 Objetivo General

Diseñar, desarrollar e implementar un sistema experto para la recomendación de recursos educativos para niños de 5 a 7 años con o sin discapacidad en base a su perfil o contenidos requeridos.

3.2 Objetivos Específicos

- Estudiar el estado del arte sobre sistemas recomendadores basados en contenidos y perfiles de usuario.

- Entender y conocer las bases de datos orientadas a grafos para utilizar dentro de los sistemas recomendadores basadas en contenidos y perfiles de usuario.
- Diseñar e implementar una arquitectura basada en contenedores para soportar procesos de inferencia y recomendación del sistema.
- Generar un repositorio de materiales didácticos educativos para niños con y sin discapacidad.
- Desarrollar un sistema experto para la recomendación y generación de material didáctico en base al perfil o contenido.
- Implementar un sistema móvil y web para el acceso a los recursos en base a lo recomendado por el sistema.
- Diseñar y ejecutar un plan de experimentación que permita validar el sistema recomendador de contenido basado en encuestas y usabilidad del sistema.

Capítulo IV

Revisión de la literatura o fundamentos teóricos

En este capítulo se realiza una revisión del marco teórico referente al problema y las tecnologías a utilizar para el desarrollo del sistema. Se revisan los principales temas para el sistema recomendador basado en contenido, la base de datos NoSQL, el desarrollo de aplicaciones web y móvil y los frameworks de desarrollo para el sistema así como el repositorio.

4.1 Sistemas de Recomendación

Los sistemas de recomendación permiten alcanzar y obtener recursos dentro de todas las posibilidades almacenadas, pues permiten reducir el campo de búsqueda aplicando diferentes técnicas e instrumentos de software para plantear sugerencias y recomendaciones al usuario. (Gómez et al., 2018)

En diferentes ámbitos los sistemas de recomendación se han consolidado como potentes herramientas para ayudar a la sobrecarga de información existente, esto permite filtrar información recuperada de diferentes fuentes, aplicando diferentes métodos y generar recomendaciones a partir de preferencias, contenido o el perfil de un usuario. (Viedma et al., 2004)

Dentro de los sistemas de recomendación existen elementos fundamentales que permiten generar recomendaciones y su buen funcionamiento los cuales son:

- Las entradas y salidas que permiten generar recomendaciones las cuales pueden ser la información del usuario al igual que del contenido de los recursos o ítem .
- El método empleado para producir las recomendaciones que pueden ser filtros colaborativos, basado en contenido o híbridos.
- La personalización del sistema de recomendación que permite formar recomendaciones adecuadas al perfil del usuario.

4.1.1 Basado en Contenido

Los sistemas basados en contenidos emplean la descripción, contenido e información de un ítem o recurso y las datos del perfil de un usuario para generar recomendaciones. Estos sistemas están enfocados en recomendar artículos que contengan información, como documentos, textos, sitios Web (Gómez et al., 2018).

Un sistema recomendador basado en contenido forma recomendaciones similares a los que un usuario visitó en el pasado. Estos sistemas realizan comparaciones con diferentes elementos o recursos previamente consultados por un usuario en donde los mejores y más similares son recomendados (Gómez et al., 2018).

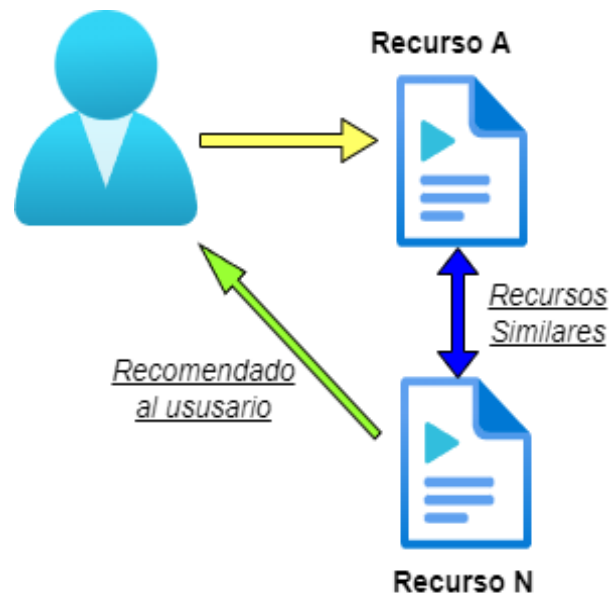


Figura 1. Recomendación basado en contenido (Quanam, sf).

4.1.2 Medidas de similitud entre textos

La similitud o semejanza entre textos en la métrica que existe entre dos o más palabras que se obtiene dependiendo del grado de similitud que exista entre sus significados o contenido semántico. Dentro del procesamiento de lenguaje natural para encontrar la similitud se emplea varias medidas una de ellas es TF-IDF.

La medida TF-IDF que sus siglas derivadas del inglés Term frequency - Inverse document frequency permite establecer un valor numérico a la importancia de una palabra para un documento en una colección.

Para el cálculo de esta medida se determina primero la frecuencia de término que es tf representa la frecuencia de un término o palabra en el documento. Esto determina el número de veces que el término se repite en el documento, esto es conocido como la frecuencia bruta.

$$tf(\text{palabra}, \text{documento}) = \frac{\text{Frecuencia de palabra} \in \text{Documento}}{N^{\circ} \text{palabras} \in \text{Documento}}$$

Tabla 1. Ecuacion de Term frequency.

La frecuencia inversa del documento o IDF es la medida del término si es común o no en una colección de documentos. Si el término no se encuentra en la colección de documentos se produce una división por 0 por lo que se ajusta la fórmula sumando 1 y se obtiene el logaritmo del cálculo.

$$idf(\text{palabra}, \text{Documentos}) = \log_e \left(1 + \frac{N^{\circ} \text{Documentos}}{N^{\circ} \text{Documentos con la palabra}} \right)$$

Tabla 2. Ecuacion de Inverse document frequency.

Finalmente el valor de TF-IDF se obtiene calculando:

$$tdidf(\text{palabra}, \text{Documentos}) = tf(\text{palabra}, \text{documento}) * idf(\text{palabra}, \text{Documentos})$$

Tabla 3. Ecuacion de TF-IDF.

4.2 Recursos Educativos y Didácticos

Un recurso es un medio o elemento que se emplea para resolver un problema o necesidad. Un recurso educativo se define como cualquier material que permite ser empleado en el contexto de la enseñanza, educación o medio didáctico con el objetivo de facilitar la comprensión y el desarrollo de actividades educativas (Jama et al., 2019).

Los recursos educativos son importantes en la enseñanza, estos recursos educativos son empleados y diseñados de acuerdo al quien va dirigido y la finalidad que tiene el mismo. Según (González and Luz, 2011) las funciones de un recurso educativo son:

- Proporcionar información
- Guiar en el proceso de aprendizaje y enseñanza
- Facilitar la comprensión de un tema o actividad
- Motivar el aprendizaje en los estudiantes.

Los recursos y materiales educativos de acuerdo a su representación y el medio que utilizan para estimular y desarrollar el aprendizaje según (Jama et al., 2019) se pueden clasificar como:

- Recursos manipulativos
- Recursos impresos
- Recursos visuales
- Recursos auditivos
- Recursos audiovisuales
- Recursos multimedia

4.3 Base de Datos

La base de datos son un elemento fundamental en la actualidad son empleadas por toda área o disciplina que requieran gestionar y almacenar información. Los datos que se generan a diario son más extensos al igual que presentan ciertas características que aumenta su volumen lo que hace necesario una base de datos y tecnologías que permitan su manipulación y persistencia (Romero et al., 2019).

Según (Mercedes, 2009) "Una base de datos es un conjunto de datos almacenados en memoria externa que están organizados mediante una estructura de datos". Las bases de datos son diseñadas de acuerdo a las necesidades de la empresa, organización o proyecto para satisfacer sus necesidades y requerimientos en el manejo de la información.

4.3.1 Base de datos NoSql

NoSQL significa "Not Only SQL" y se caracteriza por almacenar los datos de forma no relacional. Esto surgió con la necesidad de gestionar grandes volúmenes de información. Según (Cristina et al., 2014) "El término NoSQL fue usado en 1991 en un artículo de UNIX y posteriormente en 1998 se empleo para nombrar una base de datos open-source la cual no tenía un interface SQL, pero si seguía su modelo".

En la actualidad se generan grandes cantidades de datos y esto a dado paso a nuevos conceptos como Big Data, Big User y Cloud Computing lo que ha limitado el uso de las bases de datos relacionales. Google, Facebook, Amazon son algunas empresas que se encontraron con limitaciones en el uso de bases de datos relacionales por lo que generaron su propia base de datos NoSQL (Cristina et al., 2014).

Los sistemas de base de datos NoSQL pueden clasificarse según el modelo de datos que emplea para su almacenamiento como llave/valor, orientado a columnas, orientado a documentos en donde se tiene a MongoDB.

4.3.1.1 MongoDB

MongoDB es una base de datos no relacional orientada a documentos que se caracteriza por almacenar datos en documentos de tipo JSON con un esquema dinámico. MongoDB almacena los datos como documentos binarios conocidos como BSON, estos documentos pueden tener diferentes esquemas que pueden cambiar en el tiempo. MongoDB se caracteriza por ofrecer escalabilidad rendimiento y alta disponibilidad (Edward and Sabharwal, 2015).

Las características de esta base de datos son muy necesarias en la actualidad como seria para el almacenamiento de grandes datos. Debido a su diseño basado en documentos es muy intuitivo y fácil de aprender y usar. Su arquitectura permite que sea utilizado en aplicaciones experimentales y de producción. Segun (Castro et al., 2012) las características de MongoDB se describen a continuación:

- Posee un esquema flexible y dinámico permitiendo que sus campos varíen en los documentos, cada registro puede contener los mismos o diferentes atributos. Esto permite almacenar datos sin la necesidad de realizar un cambio en el esquema del documento.
- El lenguaje de consulta es muy intuitivo, permite realizar consultas simples pero con un alto rendimiento. Las consultas permiten realizar búsquedas de propósito general y específico, al igual que permite filtrar datos según las necesidades requeridas.
- Gracias a sus características le convierte en una base de datos con atomicidad, consistencia, integridad y durabilidad, lo que permite que sus datos se almacenen y se actualicen de forma que permitan mantener su precisión.
- Es una base de datos que permite el alto rendimiento gracias a los documentos en formato BSON.
- Permite alta disponibilidad debido a que admite clústeres distribuidos como mínimo en tres nodos.

4.3.2 Base de datos Orientada a Grafos

La base de datos orientada a grafos se caracteriza por representar la información en forma de nodos y sus relaciones las representa por aristas. Este tipo de base de datos aplica la teoría de grafos que permite realizar su recorrido de forma eficiente (Velásquez, 2014). La base de datos orientada a grafos debe ser empleada de acuerdo a la lógica de negocio para almacenar su información y aprovechar el potencial de este tipo de base de datos.

Una base de datos orientada a grafos está compuesta de los siguientes atributos los cuales son propios de una BDOG:

- Los nodos o vértices que pueden poseer una estructura de acuerdo al caso de estudio al que se emplea la BDOG.

- La relación o arista que posee una dirección y el tipo.
- La propiedad o atributo en los nodos y las relaciones.

Las BDOG están diseñadas para almacenar grafos dirigidos los cuales son aquellos donde la relación apunta a un nodo origen desde un nodo destino, y relaciones no dirigidas las que la relación no tienen un nodo origen si no solo relacionan los nodos. Las relaciones pueden estar definidas por una llave y un valor, esto permite que se puedan aplicar algoritmos para realizar diferentes consultas (Monerris, 2015).

4.3.2.1 Neo4j

Neo4j esta entre las BDOG mas populares en la actualidad según (DB-Engines, 2022) donde se clasifica estos sistemas y su tendencia.

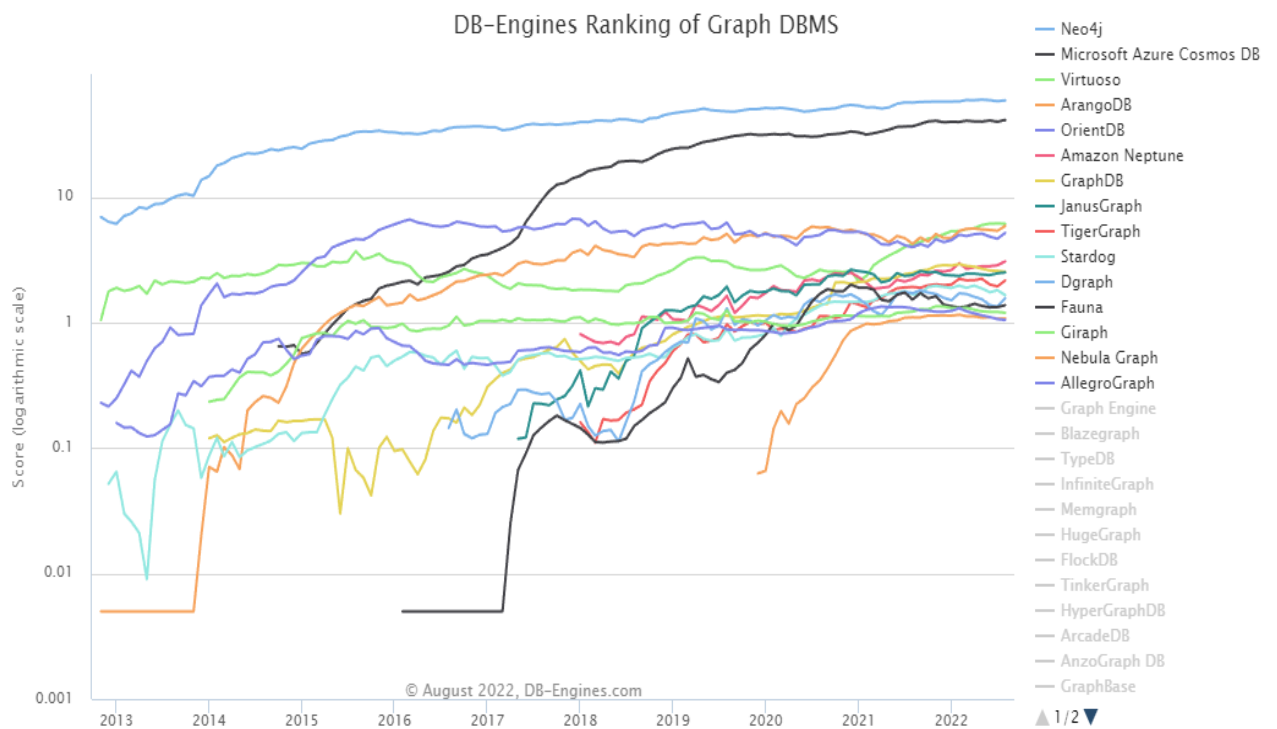


Figura 2. Clasificación DB-Engines - Tendencia de la popularidad de Graph DBMS (DB-Engines, 2022).

Neo4j es una base de datos orientada a grafos líder del mundo la cual fue diseñada para una gestión y recorrido óptimo entre los nodos y sus relaciones. La BDOG adopta un enfoque gráfico lo que permite un alto rendimiento y mejora el tiempo de ejecución de las operaciones de recorrido y búsqueda.

La base de datos Neo4j es una base NoSQL open-source disponible desde el 2017. Ofrece un servicio en la Nube con limitaciones, conocido como AuraDB, Neo4j es multiplataforma y tiene diferentes servicios como su Community Edition o Enterprise Edition. Neo4j está desarrollado en java y scala .En Neo4j un grafo está compuesto por nodos que describen los objetos de la base y las relaciones que poseen etiquetas para definir y poder clasificar los nodos (Neo4j, 2022).

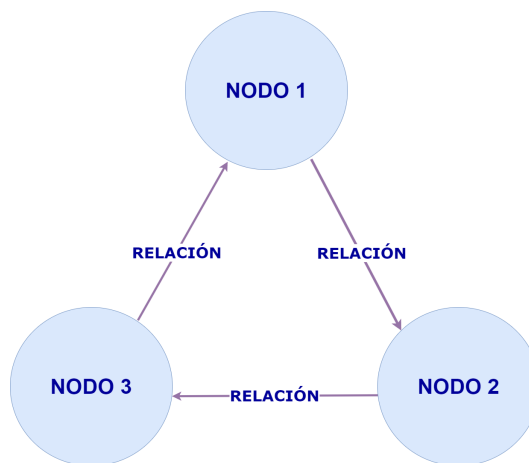


Figura 3. Nodos y relaciones según (Neo4j, 2022).

Los nodos en neo4j representan un objeto o entidad. Estos objetos poseen etiquetas y atributos que son propiedades que caracterizan a cada entidad. Las etiquetas permiten diferenciar o agrupar a los nodos de la base de datos. Las etiquetas se pueden emplear para representar estados temporales de los nodos .

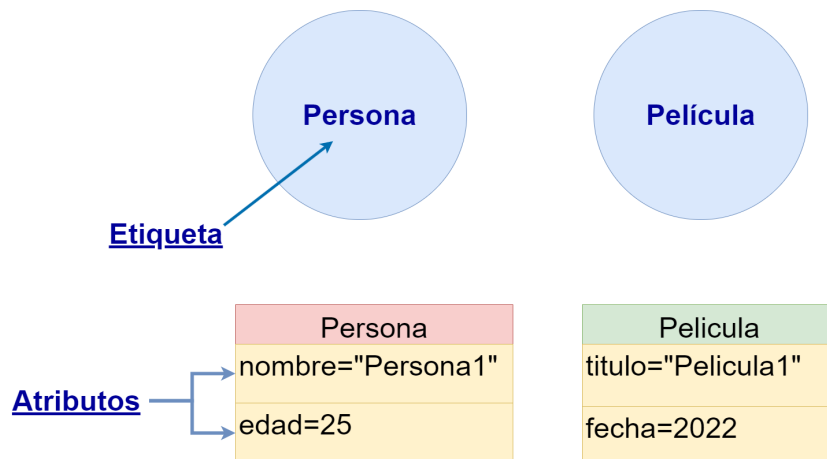


Figura 4. Nodos como objetos y sus atributos según (Neo4j, 2022).

Los nodos se vinculan por relaciones, están se dan entre un nodo origen y un nodo destino. Una relación tiene una dirección y un tipo para definir la clasificación de la relación. Las relaciones pueden poseer propiedades que describen de manera más efectiva como se relacionan entre nodos. Las relaciones permiten formar estructuras lo que conforman o se conoce como un grafo, estos grafos pueden combinarse y formar estructuras complejas.

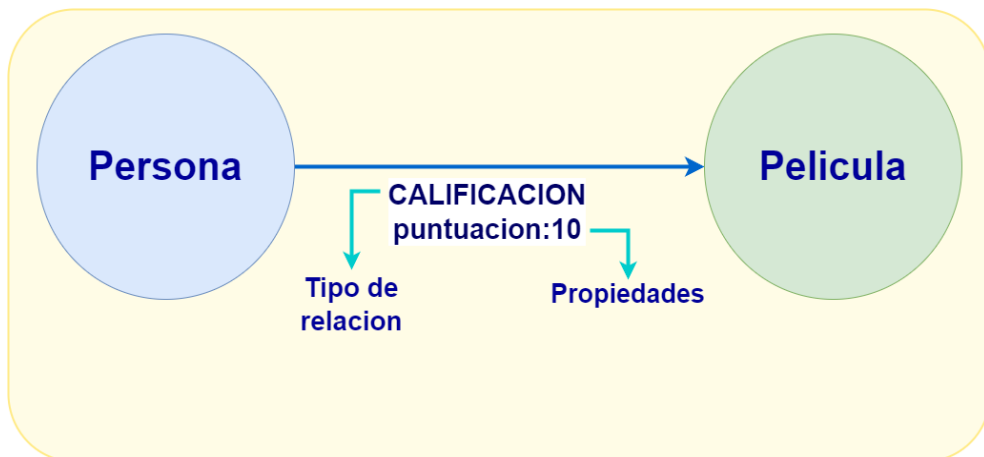


Figura 5. Relaciones entre nodos y sus propiedades según (Neo4j, 2022).

4.4 Aplicaciones Web y Movil

Aplicaciones web son software que se aloja en un servidor web, que los usuarios finales

pueden utilizar a través de internet accediendo con un navegador web para obtener los servicios que ofrece una página web. Hay multitud de aplicaciones web las cuales son gestores de correo, wikis, blogs, e-commerce, etc. Las aplicaciones web tienen un tipo de acceso, las cuales pueden ser:

- **Publicas:** diarios digitales, tiendas virtuales, etc.
- **Restringidas:** como intranets, estos ofrecen servicios mejorando las gestiones internas en una empresa, por ejemplo el control de horas para el personal de la empresa, la gestión tanto de proyectos como de tareas, etc.

Las aplicaciones web tienen la popularidad que se basa en:

- Facilidad de acceso, esto significa que solo es necesario un navegador web para poder acceder a las aplicaciones web.
- Desde cualquier sistema operativo.
- Si la aplicación web se actualiza o se le da mantenimiento, no es necesario redistribuir y reinstalar el software a todos los usuarios finales (Jiménez, 2013).

Aplicaciones móviles denominadas apps son diseñadas para dispositivos móviles, estas aplicaciones están presentes desde tiempos atrás, ya que venían incluidas en sistemas operativos como Blackberry o Nokia. Básicamente una aplicación no deja de ser un software, para un mejor entendimiento lo haremos mediante una analogía siendo que las aplicaciones son para móviles lo que los programas son para los ordenadores. Hoy en día se encuentra una variedad extensa de aplicaciones de todo tipo, ya que en la actualidad las aplicaciones son muy rentables tanto para los desarrolladores de aplicaciones móviles como para las tiendas de aplicaciones como Play Store, App Store y Microsoft Store, existiendo aplicaciones tanto de pago o gratis (Cuello and Vittone, 2013).

4.4.1 Angular

Plataforma o Framework el cual permite el desarrollo de aplicaciones web haciendo uso de HTML y Javascript en la parte del usuario final, dándole al servidor una buena parte del trabajo, logrando que las aplicaciones web tengan un mayor rendimiento en la ejecución. Con el rendimiento mejorado en aplicaciones web, su rendimiento en dispositivos móviles está más optimizada, ya que los ciclos de CPU y memoria son críticos para un funcionamiento óptimo. Para ello permite el desarrollo de aplicaciones móviles en Ionic.

Angular es un framework enfocado en SPA: Single-Page Aplicación que significa aplicación de una sola página haciendo que la carga de datos sea de forma asíncrona. En aplicaciones SPA toda la carga del aspecto de la aplicación recae mayormente en la parte del cliente/usuario final, mientras tanto el servidor ofrece al cliente/usuario final una API de servicios dando acceso a la base de datos la cual se alimenta de la aplicación. Estos datos intercambiados entre cliente/usuario final y servidor están en formato JSON, siendo un formato más óptimo que XML.

Angular se rediseñó por completo en la versión 2.0, cambiando de nombre de AngularJS a Angular simplemente, el encargado de dar mantenimiento y soporte es Google (Oriols and Gutiérrez, 2018).

4.4.1.1 Características de Angular

- **Plantillas:** Angular al permitir la creación de aplicación de una sola página, haciendo posible que se pueda cargar diversas partes de la aplicación sin necesidad de volver a refrescar el contenido en el navegador. Al tener un motor de plantillas logra que se genere contenido dinámico en tiempo real.

Se cuenta con una serie de funciones que nos ayuda con la escritura de plantillas de forma más ordenada haciendo más fácil de leer, también logrando automatizar varias tareas tales como: las iteraciones y condiciones para visualizar el contenido.

- **Estructura MVC:** Con esta estructura se busca tener una mejor organización en el código, donde todo estará totalmente separada, tanto el manejo de datos mediante el Modelo está separado de la Lógica mediante el Controlador de la aplicación, y esta información será presentada al usuario mediante las Vistas. Este proceso es simple donde el usuario con las Vistas de la aplicación, estos se comunican con los Controladores alertando las acciones del usuario, los controladores realizan peticiones a los modelos, luego estos gestionan la solicitud como según corresponda la información brindada. Esta estructura es muy útil al momento de desarrollar aplicaciones de gran escala ya que ayuda a mantenerlas o extenderlas.

- **Vinculación de datos:** El DOM¹ al ser modificable una vez cargada por completo. Esto permite que se pueda modificar el contenido de la página sin necesidad de recargar nuevamente el navegador, siendo esta una de las grandes ventajas que tiene Angular para la vinculación de datos con la vista.

Esto no es nuevo, ya que otras tecnologías ya lo hacían antes, pero lo innovador en Angular es que permite definir que partes de la vista son sincronizadas de forma automática con propiedades de Javascript, ahorrando la cantidad de código que se tiene que escribir para respectivamente mostrar datos del modelo a la vista.

- **Directivas:** Angular se destaca por un complemento muy significativo que son las directivas que permite más allá de seleccionar elementos del DOM, permite extender la sintaxis HTML, con Angular veremos que existe una cantidad extensa de atributos que no son parte de HTML.

Angular dispone de una extensa cantidad de directivas logrando que las plantillas sean fáciles de leer y permitiendo optimizar el código logrando un gran resultado. Angular también nos brinda la oportunidad de hacer nuestras propias directivas para extender HTML haciendo que la aplicación funcione mucho mejor

¹Data Object Model

- **Inyección de dependencia:** Angular al estar basado en un sistema de inyección de dependencias donde los controladores requieren que los objetos que se necesitan se declare a través del constructor. Posteriormente los inyecta de una manera que el controlador pueda hacer uso como sea necesario, sin necesidad de saber como funciona la dependencia ni de que forma entrega los resultados.

Sencillamente el controlador sirve para controlar, esto quiere decir que recibe peticiones y responde con base en las peticiones, cabe mencionar que el controlador no genera el mismo las respuestas. Si se hace un buen seguimiento a los controladores en la aplicación esto hará que sea mas fácil mantener la aplicación incluso si el desarrollo se vuelve a retomar después de un tiempo (Dorta, 2016).

4.5 FastApi

FastAPI es un framework que permite el desarrollo de API RESTful en lenguaje de programación python. Este framework permite el desarrollo ágil, rápido y considerando la sincronización y asincrónica. Se puede emplear desde python 3.6 en adelante.

Este framework moderno se emplea actualmente en el desarrollo de aplicaciones, ya que permite la documentación automática de los diferentes APIs que se desarrollan. Al emplearse con python se puede añadir diferentes librerías que permiten integrarse con otras tecnologías.

4.5.1 Características

El framework de fastAPI posee características que permiten una implementación y desarrollo eficaz. estas características son las siguientes:

- Alto Rendimiento gracias a Starlette y Pydantic
- Programación rápida
- Menos errores

- Intuitivo y con gran soporte y documentación
- Fácil en el diseño y desarrollo de APIs
- Corto permitiendo eliminar código repetido
- Robusto gracias a su documentación automática
- Basado en estándares como OpenAPI y JSON Schema

4.6 Docker

Es una plataforma de código abierto que permite desplegar, subir y ejecutar aplicaciones. Esta herramienta permite empaquetar e iniciar aplicaciones en un entorno aislado (Iliev, 2021). Docker posee herramientas que permiten la administración y el despliegue en contenedores. Estas herramientas permite:

- Desarrollo de la aplicación y despliegue en un contenedor
- El contenedor permite distribuir y probar la aplicación
- Permite el despliegue de la aplicación en un entorno de producción en un contenedor que puede estar en un centro de datos local, o un proveedor en la nube

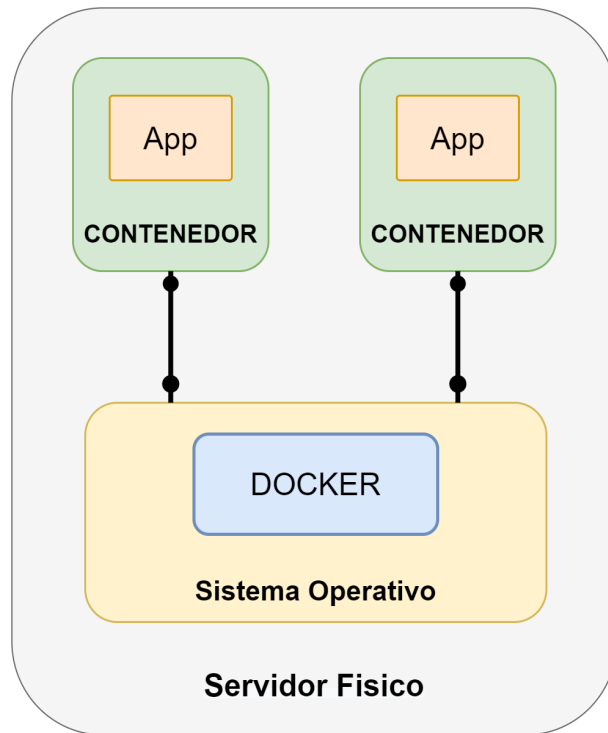


Figura 6. Arquitectura de Docker.

La arquitectura que emplea Docker es cliente-servidor, el cliente de Docker permite comunicarse con el demonio de Docker, esto permite crear, ejecutar y distribuir los contenedores. El cliente de Docker puede ejecutarse en el mismo sistema o cliente remoto de demonio de docker (Docker, 2018a).

Un contenedor es una instancia ejecutable de una imagen de Docker, este contenedor se puede crear, iniciar, detener, mover o eliminar mediante la API de Docker o su Cliente. Se puede adjuntar el almacenamiento al contenedor o incluso crear una imagen nueva basada en su contenedor.

Una imagen de Docker es una plantilla de solo lectura que contiene instrucciones para crear un contenedor, esta imagen se basa en otra imagen con algunas características adicionales. Para la creación de una imagen se emplean archivos conocidos como Dockerfile.

4.6.1 Dockerfile

Dentro de Docker es un archivo de configuración DSL con instrucciones que permite la construcción de nuevas imágenes. Dockerfile permite crear nuevas imágenes adecuadas para una aplicación que se va a ejecutar en un contenedor (Docker, 2018b). La creación de una nueva imagen se realiza de forma que se debe crear un archivo denominado Dockerfile, luego llamar al API de Docker o Docker CLI y ejecutar el archivo. Una vez creada la imagen se puede usar en un contenedor.

4.6.2 Docker Compose

Es una herramienta que facilita el despliegue de aplicaciones en múltiples contenedores, cada aplicación se despliega y se ejecuta en un contenedor dentro de una misma red. Esto permite la comunicación y la interacción entre los contenedores. Para la ejecución de los contenedores Docker compose emplea un archivo con extensión yaml, este archivo permite la configuración de cada contenedor o servicio. Cada servicio puede ser ejecutado por un comando, esto permite crear e iniciar los servicios desde su configuración en el archivo (Docker, 2018c).

Capítulo V

Marco metodológico

En esta sección se describe la metodología empleada para el desarrollo del proyecto según los objetivos generales y específicos. A continuación se presentan el análisis de los requerimientos funcionales y no funcionales para el desarrollo de software, la metodología ágil a seguir para desarrollar el proyecto y cumplir con los objetivos establecidos así como las fases del proyecto de acuerdo a la metodología definida.

5.1 ANÁLISIS Y REQUERIMIENTOS

En el desarrollo de software es la parte principal e importante, permite obtener y definir las bases para iniciar las actividades o tareas para cumplir con los objetivos definidos. Permite comprender de una forma adecuada el problema de estudio y tener conocimiento de lo que se está desarrollando. Los requerimientos bien definidos nos permitirá un buen desarrollo del proyecto cumpliendo con los objetivos planteados.

5.1.1 Requerimientos Funcionales

- Requisitos funcionales según la actividad del usuario.

Código	RE01
Nombre	Registro de perfiles de usuarios
Descripción	Se deberá registrar en el sistema los datos necesarios para el perfil del niño con o sin discapacidad.
Entradas	Cédula, Nombre, Apellido, Fecha de Nacimiento, Si tiene discapacidad, ICD-10, Diagnostico Médico, ICD-10, Diagnostico de Lenguaje
Proceso	Se cuenta con un formulario en donde la persona encargada ingresara los datos necesarios del perfil del niño con o sin discapacidad para su posterior registro en el sistema. Esta información se almacenará en la base de datos.
Salidas	Mensaje exitoso al registrar un perfil de niño. Mensaje de error al no completar todos los campos necesarios en el formulario.
Prioridad	Alta

Tabla 4. Requisito funcional para registro de usuarios.

Código	RE02
Nombre	Inicio de sesión
Descripción	El sistema permitirá realizar un inicio de sesión obligatoria para acceder a las funcionalidades del sistema.
Entradas	Nombre de usuario(nombre y apellido), Contraseña(cédula)
Proceso	Se visualizara un formulario donde el usuario llenara con los datos que se le piden, después el sistema realizara la respectiva verificación de credenciales ingresadas comparándolas con las de la base de datos. En caso de ser exitoso se continuara a la siguiente interfaz, caso contrario pedirá que ingrese las credenciales nuevamente.
Salidas	Mensaje de error al no haber ingresado correctamente los datos.
Prioridad	Alta

Tabla 5. Requisito funcional para inicio de sesión.

Código	RE03
Nombre	Cierre de sesión
Descripción	El sistema permitirá realizar un cierre de sesión para salir de la cuenta
Entradas	
Proceso	El usuario podrá salir de su cuenta al dar clic al botón que cerrara la cuenta y hará redirección al login.
Salidas	
Prioridad	Alta

Tabla 6. Requisito funcional para cerrar cesión.

- Requisitos funcionales correspondientes al manejo de actividades y sus recursos.

Código	RE04
Nombre	Visualización de actividades
Descripción	En el sistema se visualizará actividades basándonos en el perfil.
Entradas	
Proceso	El sistema mostrara las actividades basándonos en el perfil con el que se accede, ya sea un perfil sin discapacidad el cual mostrara todas las actividades disponibles, o en caso de ser un perfil con discapacidad se le mostrara actividades específicas y dedicadas para dicha discapacidad.
Salidas	
Prioridad	Alta

Tabla 7. Requisito funcional para visualización de actividades.

Código	RE05
Nombre	Visualización de recursos educativos con base en una actividad
Descripción	El sistema visualizará recursos educativos con base en una actividad seleccionada.
Entradas	
Proceso	Al momento de escoger una actividad se dirigirá a otra página donde se visualizara recursos educativos que se clasifica en apps, cuentos, juegos, libros, vídeos.
Salidas	
Prioridad	Alta

Tabla 8. Requisito funcional para visualización de recursos educativos.

Código	RE06
Nombre	Creación de una o varias actividades nuevas
Descripción	El sistema permitirá crear una o varias actividades nuevas
Entradas	Nombre de la actividad, Indicadores o Área, Bloque y Competencia, si tiene capacidad especial
Proceso	Se contara con un formulario donde el rol de docente tendrá el acceso a crear una actividad, o en el caso de crear varias actividades se podrá subir un archivo csv con una estructura ya definida.
Salidas	Mensaje de creación de nueva o nuevas actividades con éxito
Prioridad	Alta

Tabla 9. Requisito funcional para creación de actividades.

Código	RE07
Nombre	Recomendación de actividades con base en una actividad
Descripción	El sistema permitirá visualizar actividades recomendadas con base en una actividad.
Entradas	
Proceso	Al momento de visualizar los recursos con base en una actividad, en la parte derecha del listado de recursos, se visualizará un listado aparte de las actividades recomendadas basándonos en dicha actividad
Salidas	
Prioridad	Alta

Tabla 10. Requisito funcional para recomendaciones de actividades.

5.1.2 Requisitos No Funcionales

Código	RE01
Nombre	Diseño responsive.
Descripción	Las interfaces del sistema deben acoplarse a los diferentes tamaños de pantalla en los dispositivos.
Entradas	
Proceso	Al momento de visualizar las interfaces de la aplicación de cualquier pantalla, esta se adaptará y mostrara un diseño ordenado.
Salidas	
Prioridad	Media

Tabla 11. Requisito no funcional para diseño responsive.

Código	RE02
Nombre	Tiempo de respuesta.
Descripción	El sistema no debe tardar más de 10 segundos en brindar una respuesta al usuario.
Entradas	
Proceso	Al momento que el usuario interactúa con el sistema, este debe dar sus respuestas a los diferentes procesos teniendo como tiempo de respuesta 10 segundos máximo.
Salidas	
Prioridad	Media

Tabla 12. Requisito no funcional para tiempo de respuesta.

5.2 METODOLOGÍA

La metodología empleada para el desarrollo del proyecto es "SCRUM", esta metodología ágil permite implementar soluciones para simplificar el desarrollo y asegurando la calidad del producto final cumpliendo los objetivos propuestos.

5.2.1 Metodología de Desarrollo ágil SCRUM

Scrum es un marco de trabajo que permite desarrollar, proporcionar y reducir la complejidad en el proceso de crear o desarrollar proyectos complejos. Scrum permite la gestión de los proyectos aplicando técnicas de gestión que permitan mejorar continuamente el producto, el equipo de trabajo y el entorno (Schwaber and Sutherland, 2017).

Scrum se basa en el control empírico de procesos para la gestión de sistemas adaptativos, y presenta tres principios fundamentales según (Monte, 2016).

- **Transparencia:** todos los participantes deben conocer los aspectos significativos del producto
- **Inspección:** para alcanzar los objetivos los resultados deben ser evaluados de manera continua

- **Adaptación:** si se presenta un error o desviación se debe realizar acciones para garantizar los objetivos.

El marco de trabajo de SCRUM propone una estructura que ayuda a la gestión de proyectos orientados a las personas y los objetivos, estos son los roles, las actividades y los artefactos (Monte, 2016).

5.2.1.1 Roles

Los roles se definen como aquellas responsabilidades que son necesarias en un proyecto para asegurar que se abarcan las necesidades de una adecuada información entre los integrantes y permitir la elaboración y la comunicación de un proyecto ágil. Según (Monte, 2016) los roles de SCRUM se dividen en:

- **Product owner (PO):** Es el designado que permite la comunicación entre el cliente y el equipo de desarrollo. Es el encargado de las estrategias, los objetivos y mantener el product backlog. Define los criterios de aceptación del proyecto.
- **SCRUM máster (SM):** Es la persona con experiencia en el desarrollo y ejecución de la metodología SCRUM, es el encargado de brindar soporte al equipo de desarrollo. Permite la comunicación entre el PO y el DT. Analiza, reporta y registra las mejoras del proceso.
- **Development team (DT):** Es el equipo que se caracteriza porque cada persona desempeña diferentes actividades que permiten alcanzar los objetivos para el producto. El equipo posee habilidades individuales y colectivas que garantizan la ejecución del proyecto.
- **Stakeholder:** Denominados aquellos beneficiarios y receptores del desarrollo y el producto terminado. Los stakeholder realizan la aceptación del producto.

5.2.1.2 Artefactos

Los artefactos permiten que los roles puedan organizarse y trabajar conjuntamente para el desarrollo ágil del producto o proyecto. Según (Monte, 2016) propone los siguientes artefactos, estos no son los únicos que se pueden emplear, se describen a continuación los principales artefactos más importantes que un equipo debe implementar:

- **Product backlog (PB):** Es una lista de operaciones y funcionalidades del producto. El product owner es el encargado de conservar la lista y realizarla de acuerdo a las historias de usuario. Esta lista no debe contener elementos técnicos.
- **Sprint backlog (SB):** Es un grupo de propiedades y elementos del PB que se realizaran y se entregaran al finalizar un sprint. Estos permiten una percepción del trabajo a desarrollar en el sprint actual.
- **Graphs:** Para una mejor visualización del avance del proyecto SCRUM plantea un gráfico. Este gráfico posee dos características principales que permiten observar el avance del proyecto y el avance de los sprints.
- **SCRUM board:** Es una herramienta en la metodología de SCRUM que permite observar el estado de los sprints para el equipo de trabajo. Esta herramienta permite visualizar en tiempo real los sprints, esto permite la actualización de estos cada reunión, esta herramienta permite los cambios requeridos por el DT y con la supervisión del Scrum Master.

5.2.1.3 Actividades

Las actividades son progresos que se realizaron durante el tiempo de desarrollo del producto. Estas actividades son acciones que se ejecutan por el cliente o el equipo de desarrollo las cuales se consideran relevantes para el proyecto. Un sprint es el tiempo que se define para el desarrollo del proyecto aplicando SCRUM. Según (Monte, 2016) se debe realizar las siguientes actividades:

- **Sprint 0:** Es la actividad donde el equipo programa y define reglas para el inicio del proyecto. El sprint inicial es un contenedor de los eventos de SCRUM. Un sprint pueden cancelarse por el product owner o si el objetivo llega a ser obsoleto.
- **Sprint planning:** Esta actividad permite la planificación del sprint donde se recoge las funcionalidades van a desarrollar. Sprint planning determina las reglas y tareas para desarrollarse al inicio de cada Sprint.
- **Daily meeting:** Son reuniones diarias con un tiempo máximo de quince minutos. En estas reuniones están presentes el equipo de desarrollo y el scrum máster. El objetivo principal de las reuniones es determinar el avance del trabajo y como se desarrollan las actividades del proyecto.

5.2.1.4 Fases de SCRUM

El ciclo de desarrollo de SCRUM está definido por el desarrollo ágil, esto se divide en 5 fases. Según (Palacio and Ruata, 2009) el desarrollo ágil de SCRUM está dado por las siguientes fases:

- **Concepto:** Permite tener una perspectiva de lo que se quiere crear. Se define el producto o servicio a desarrollar y se elige el equipo que va a estar a cargo.
- **Especulación:** Basándonos en el concepto el equipo de trabajo establece hipótesis generales para establecer los requisitos, el diseño y el costo del proyecto. Se determinan las limitaciones que se pueden encontrar como son el costo y tiempo de desarrollo.
- **Exploración:** Permite incrementar tareas y acciones al producto.
- **Revisión:** Permite al equipo y los usuarios verificar las tareas, acciones y funcionalidades que se tienen hasta el momento. Permite evaluar la dirección con el objetivo principal.
- **Cierre:** Al momento de la fecha de entrega permite verificar que se cumplan todas las acciones fijadas en las etapas anteriores. El cierre no implica el fin del proyecto se puede

tener mejorar en nuevas versiones.

5.3 Fases

A base de la metodología SCRUM, se procede a implementar en el desarrollo del proyecto, esto permite la entrega de avances de forma periódica. Según la metodología SCRUM implementa roles que permiten una administración y desarrollo ágil del proyecto. Los roles principales son:

Rol	Encargado
Product Owner	Ing. Diego Quisi
Cliente	
SCRUM Máster	Malki Yupanki
Deveploment Team	Gabriel Chuchuca y Malki Yupanki

Tabla 13. Roles del proyecto según SCRUM.

Con base en SCRUM se definen los roles, luego de esto se procede a establecer los sprints para el desarrollo del proyecto. De acuerdo a los objetivos específicos del proyecto se utilizan para definir el inicio del proyecto.

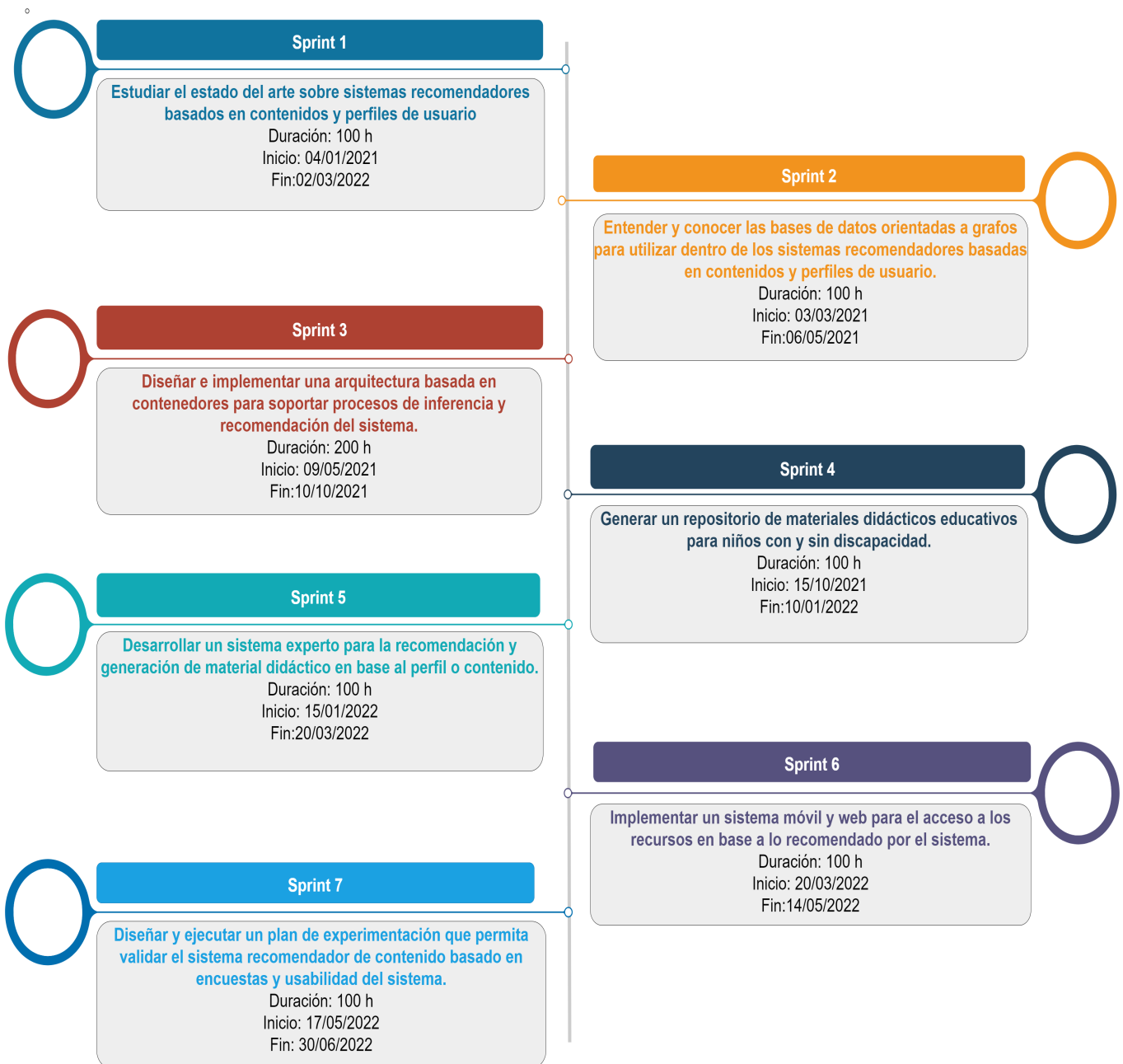


Figura 7. Sprints del proyecto en base a los objetivos.

El desarrollo de las actividades propuestas para cada sprint se define un tiempo el mismo que se puede observar en la Figura 7. Gracias a los sprints se puede planificar la distribución de tiempo adecuada que permite el desarrollo, revisión, verificación y reedificación de cada sprint.

5.3.1 Primera fase

En esta fase se realiza una investigación y estudio de los sistemas de recomendación basado en contenido, su implementación y características.

Se realizó una investigación acerca de las tecnologías a usar y los sistemas de recomendación basados en contenido, se realiza un análisis y extracción de la información relevante la cual se va a emplear para el desarrollo del proyecto.

5.3.1.1 Estado del arte

- De acuerdo a (Viedma et al., 2004) en su trabajo "**Sistemas De Recomendaciones: Herramientas para el Filtrado de Información en Internet**" describe el uso de los sistemas de recomendación empleados en la actualidad debido al crecimiento de información en la Web, al igual que el uso de internet lo que ha complicado la gestión de la excesiva cantidad de información generada. Los sistemas de recomendación han tomado valor como herramientas que permiten reducir el exceso de información que se puede encontrar al momento de realizar una búsqueda de información, los SR permiten emplear diferentes técnicas permitiendo filtrar ítems de información que cumplan con los criterios o necesidades del usuario. En este trabajo se describe la estructura que conforma un sistema recomendador como son las entradas y salidas, el método empleado para generar una recomendación y la personalización que se da al sistema para tener mejores resultados.
- En el artículo de (Carrillo and Ochoa, 2013a) señala la representación y el almacenamiento de información representado por ítems presenta complicaciones, donde se debería tratar de normalizar la información de acuerdo a la estructura de la información, que pueden ser datos estructurados que presentan atributos limitados y datos no estructurados como páginas web, noticias, artículos etc. donde estos deben pasar por un proceso que permita normalizar dicha información. La representación del perfil de los usuarios se

da de acuerdo a procedimientos de recopilación de información, donde el perfil puede ser formado mediante un test o realizar un modelo de acuerdo a las preferencias de los usuarios.

- En el artículo de (Vargas, 2017) define el uso de los materiales didácticos y recursos educativos como medios que facilitan el proceso de aprendizaje y enseñanza. Estos recursos en la actualidad se han complementado con las nuevas tecnologías de información. Establece la importancia de un recurso educativo está en su influencia en el aprendizaje y la función que cumple. Las funciones de un material educativo debe proporcionar información que contribuya al proceso de aprendizaje y facilite la comprensión de un tema mediante ideas, motivaciones o implique el uso de los sentidos para una mejor comprensión por parte del estudiante.
- En el artículo (Carrillo and Ochoa, 2013b) establece los sistemas de recomendación que buscan calcular una recomendación utilizando los perfiles de los usuarios, estos sistemas de recomendación se basan en la descripción del perfil y los intereses del usuario. Los aspectos más importantes para los sistemas de recomendación basados en contenido de acuerdo al perfil de un usuario se compone de tres aspectos importantes como el modelamiento del perfil de usuario y sus preferencias, formar el ítem de recomendación y el algoritmo empleado para realizar una recomendación.

5.3.2 Segunda fase

En esta fase se procede a desarrollar la capa de acceso a datos. Para el diseño de la capa de acceso a datos se procede a obtener actividades de un libro de trabajo del ministerio de educación para niños de primer año, estas actividades se almacenaran en una base de datos NoSQL y una base de datos orientada a grafos. Esta fase permite el inicio y establece la base para el desarrollo del proyecto.

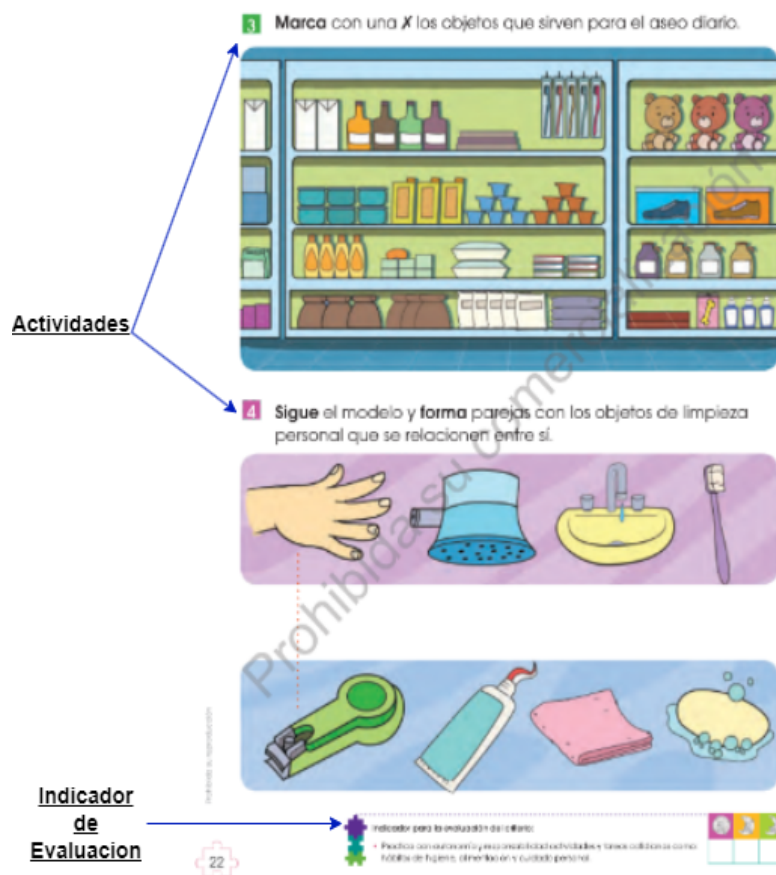


Figura 8. Pagina del Cuaderno de Trabajo de primero año (de Educación del Ecuador, 2016).

Con base en los objetivos definidos se procede a almacenar las actividades en una base de datos NoSQL, la base de datos utilizada para el almacenamiento de las actividades será MongoDB. La base de datos MongoDB permite almacenar grandes cantidades de información no estructurada por lo que no es necesario definir un esquema de almacenamiento. Las actividades base que nos permitirán formar el repositorio de recursos educativos son las actividades del cuaderno de trabajo de primer año de educación básica. Para poder generar nuevos recursos más adelante se podrán agregar nuevas actividades que no pertenecen al cuaderno de trabajo.

Las actividades que se guardan en la base de datos MongoDB se proceden a extraer del libro en formato PDF, este libro se encuentra disponible en la página del ministerio de educación. La

estructura del libro está compuesta por actividades e indicadores de evaluación como se muestra en la Figura 8 en la cual observamos como está estructurada una página del libro.

Para los niños con capacidades diferentes se procede a obtener actividades desarrolladas para cada capacidad especial. Estas actividades se encuentran en diferentes textos que contienen actividades que son diseñadas para que los niños con sus tutores puedan desarrollarlas.

➤ **CONTICUENTOS**, Lectura de cuentos cortos y llamativos que centren la atención de los niños, con ilustraciones grandes.

ACTIVIDAD




Imagen 1

OBJETIVO

Objetivo de la actividad

➤ Fortalecer en los niños la estimulación del pensamiento e imaginación, el sentido auditivo y visual.

Materiales

- Libro de cuentos
- Narrador
- Aula de clase o rincón de lectura

Figura 9. Página del Cuaderno de Trabajo para niños con PCI (Saens and Serna, 2019).

Los libros para diferentes capacidades especiales están compuestas por una actividad y los objetivos de la misma. La actividad y los objetivos son los principales ítems que serán almacenados, ya que estas nos permitirán obtener recursos que apoyen al desarrollo de la actividad.

Para la extracción de las actividades se procede a crear scripts en Python que permite la lectura del libro de cuaderno de trabajo, y los libros que contienen actividades para personas con capacidades especiales.

```

1 {
2   _id: '62c5e5d3748e9bd1d0ab9b6f',
3   pagina: 4,
4   N_Actividad: 1,
5   titulo: 'Actividades',
6   actividades: 'Mi escuela, mi aula Completa las líneas punteadas para llegar al destino.',
7   analisis: 'escuela, aula, Completa, líneas, punteadas, llegar, destino',
8   indicadores: '• Interpreta la silueta y los paratextos.'
9 }

```

Figura 10. Actividad almacenada en MongoDB.

Una vez que las actividades se encuentran almacenadas en MongoDB, se proceden a crear una base de datos orientada a grafos. Para ello se utiliza neo4j, esta base de datos permite el almacenamiento de las actividades que serán utilizadas más adelante en el sistema recomendador.

La base de datos de neo4j está conformado por nodos de actividades. Los nodos de las actividades contienen el ID del documento de la base de datos de MongoDB. Esto nos permitirá realizar búsquedas y recomendaciones empleando algoritmos propios de neo4j, esto se aplicará en el sistema recomendador para generar recomendaciones de actividades con base en las relaciones de la base de datos Neo4j.

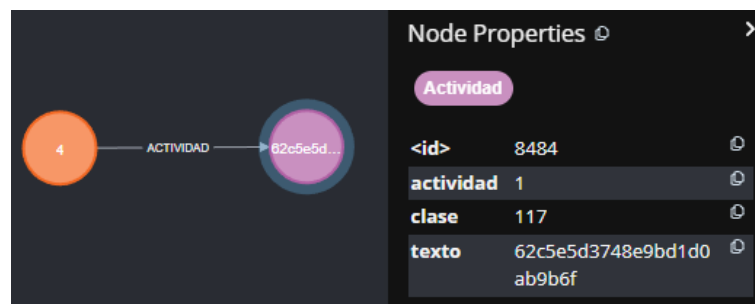


Figura 11. Actividad almacenada en Neo4j.

Una vez que se obtiene las actividades se procede desarrollar un webcrawler que nos permitirá obtener recursos multimedia para las actividades. El webcrawler está desarrollado en Python esto nos permite obtener vídeos de acuerdo a las actividades almacenados en la base de datos MongoDB.

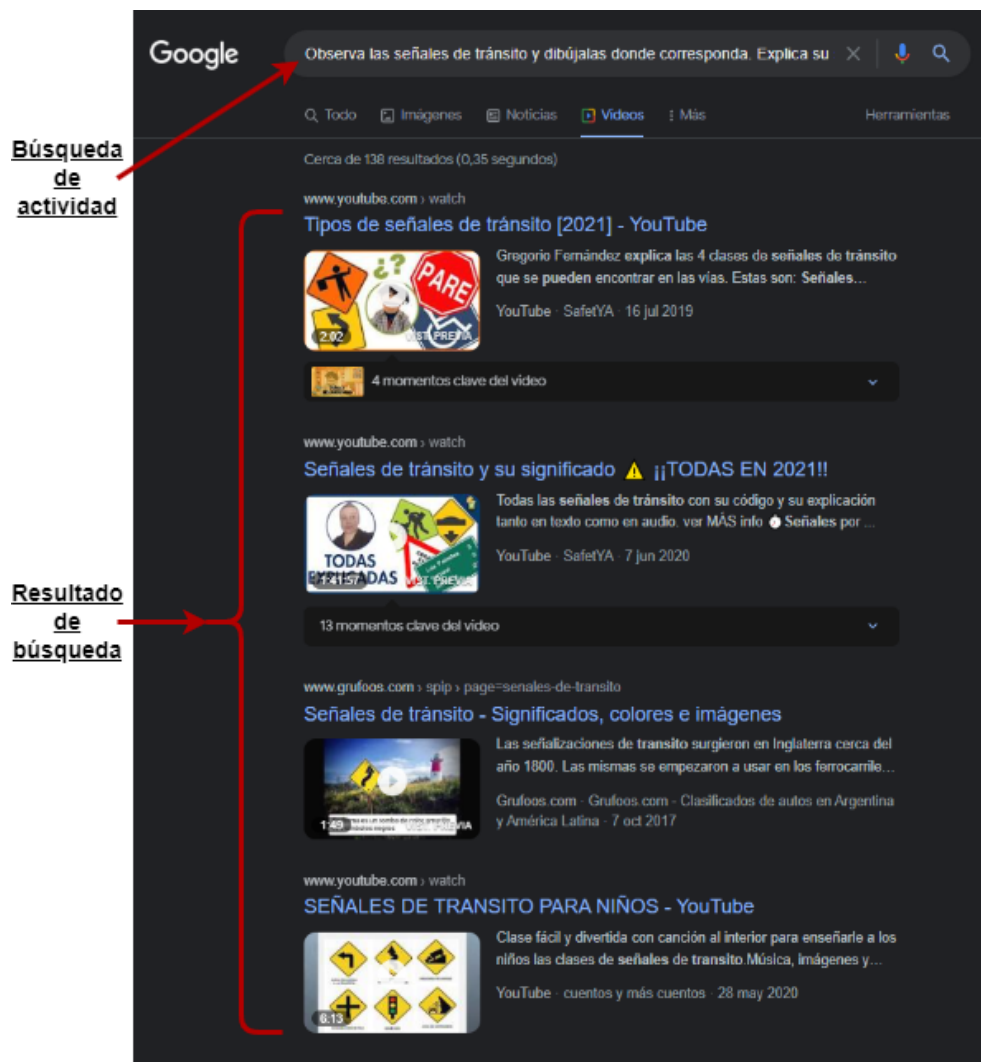


Figura 12. Búsqueda de actividad en Google.

El webcrawler procede a realizar una búsqueda en Google de las actividades como se muestra en la Figura 12 . Para obtener mejores resultados se realizará una búsqueda de la actividad como se encuentra en el libro, y una búsqueda después de realizar un proceso de tokenización a la actividad obteniendo las principales palabras de la actividad. Esto nos permite obtener vídeos de acuerdo al criterio de las actividades, el webcrawler almacena los links de los principales videos de la búsqueda. Una vez que se obtiene los links de los vídeos se procede a obtener el texto presente en los subtítulos de los vídeos lo que nos permitirá analizar su contenido para ser utilizado en el sistema recomendador.

Otro recurso que se procede a obtener son libros para niños, estos libros se obtienen mediante un webcrawler que recupera los títulos y el link del libro de una página web. La página web donde se encuentran los libros es (MARYLAND, sf).



Figura 13. Cuento recuperado de (MARYLAND, sf).

El webcrawler almacena en MongoDB el título, el abstract y la url del cuento. Estos atributos serán usados por el sistema recomendador. En la página de (MARYLAND, sf) se encuentran los libros escaneados por lo que la url almacenada nos permitirá acceder y leer el libro.

Este webcrawler permite extraer otros recursos como cuentos de páginas, los ítems recuperados de las páginas de los cuentos son el título, el cuento en texto y el link donde se encuentra el cuento.

```
1 {  
2   _id: ObjectId('62e4b385767642fa87cf3ed0'),  
3   titulo: 'La manía de la rana rockera, un cuento sobre Aceptarse',  
4   categoria: 'aceptarse',  
5   url: 'https://cuentosparadormir.com/infantiles/cuento/la-mania-de-la-rana-rockera',  
6   cuento: 'La rana Paca cambió su vida el día que vio la televisión por primera vez.'  
7 }
```

Figura 14. Cuento almacenado en MongoDB.

Otro recurso que se procedió a recuperar son juegos en la web, estos juegos son clasificados como educativos, ya que su finalidad es reforzar temas académicos. Los ítems recuperados de los juegos fueron el título, la url del juego y su descripción.

```
1 {
2   _id: '62c323663875d2dd910fccdb',
3   titulo: ' Pic-a-Pix: Mosaicos del Abecedario ',
4   url: 'https://www.cokitos.com/pic-a-pix-mosaicos-del-abecedario/play/',
5   descripcion: 'Pic-a-pix es un juego educativo con mosaicos de letras del
6 }
```

Figura 15. Juego almacenado en MongoDB.

Mediante un web crawler se procede a extraer información de una tienda de aplicaciones. Para esto se realiza una búsqueda en la tienda de Play Store en la web con las actividades que se encuentran en MongoDB, el web crawler nos permitirá analizar y extraer información de los datos de las aplicaciones que se encuentran en la versión web de la Play Store.

Primero se obtiene las actividades almacenadas en la base de datos MongoDB, luego se procede a realizar una búsqueda de las actividades en la versión web de la Play Store añadiendo una palabra clave la cual es niños o kids. En la figura 16 se muestra un ejemplo de la búsqueda de una aplicación:

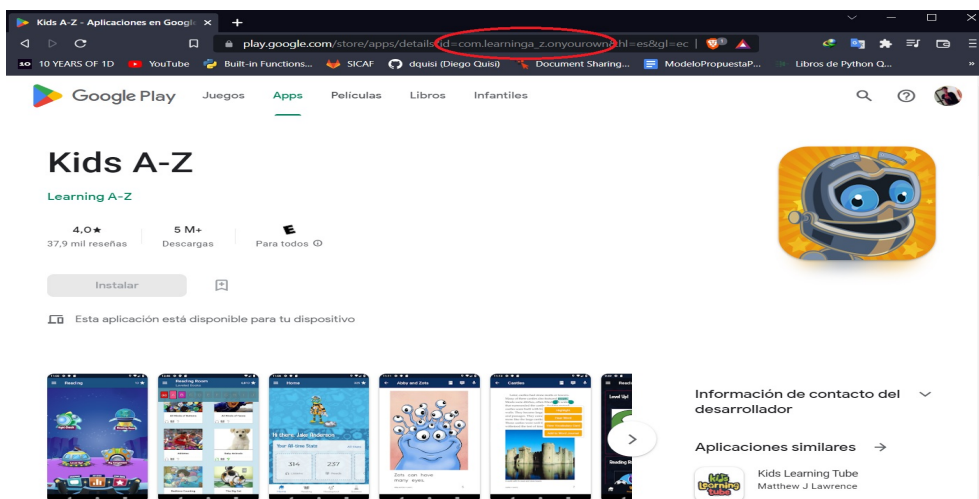


Figura 16. Aplicación en la versión web de Play Store.

El web crawler necesita parámetros como se muestra en la Figura 17 para obtener una aplicación.

```
result = app(  
  'com.learninga_z.onyourown'  
  lang='es', # defaults to 'en'  
  country='ec' # defaults to 'us'  
)
```

Figura 17. Parámetros para obtener app.

Luego obtener las aplicaciones se procede a guardar en los datos en MongoDB. Los ítems que se almacenan de la aplicación son el título, la descripción y la url de la aplicación.

```
1 {  
2   _id: ObjectId('62d7229aee3655f06f61ebe8'),  
3   titulo: 'Sonidos de animales',  
4   descripcion: 'Juego educativo para aprender los sonidos de animales, per  
5   url: 'https://play.google.com/store/apps/details?id=net.fagames.android.  
6 }
```

Figura 18. Aplicación almacenada en MongoDB.

El uso de las redes sociales permite compartir información y recursos de aprendizaje, por lo cual se procedió a trabajar con Twitter. Las publicaciones que se encuentran en Twitter denominados tweets contienen información y algunos comparten diferentes tipos de recursos como vídeos, imágenes y links de páginas para ayudar el desarrollo de actividades y el aprendizaje.

Para poder recuperar los tweets se trabajó con la API de Twitter que permite realizar consultas y extracción de información. Para el uso de esta API se debe crear una cuenta de desarrollador y crear una aplicación dentro del panel de usuarios para tener acceso a las credenciales.

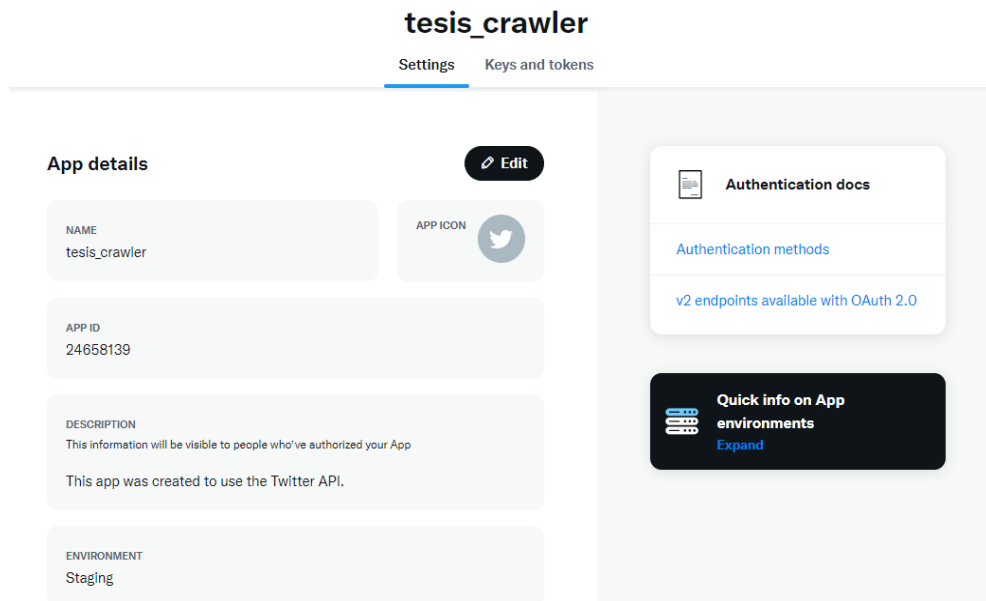


Figura 19. Panel de Twitter developer.

Una vez que se tiene las credenciales se puede acceder a la API de Twitter para obtener los tweets, para esto se procedió a desarrollar scripts en Python que permiten obtener los tweets y almacenarlos en la base de datos MongoDB. Para el acceso a la API desde Python es necesario obtener las credenciales desde el panel de desarrollador de Twitter.

TWITTER

```

1 auth = tweepy.OAuthHandler("c438mUNde3fV8AAsPctZVphKo", "Cb8Bo94Yf1GkAORqprUbuqUxPnNLCUht9PBcG0sRM5gaswv70gf")
2 auth.set_access_token("250372770-zK1UdwQScmKtjYzJuNtuH5yJ8DLafMMztG5nBCQO", "DwcEEXY3927FZoxyjsicjuBDUMrB6egPSZr1zWlgmfvGh")
3 api = tweepy.API(auth, wait_on_rate_limit=True)
4
5 try:
6     api.verify_credentials()
7     print("Authentication OK")
8 except:
9     print("Error during authentication")

```

Authentication OK

Figura 20. Autenticación de Twitter desde Python.

Una vez que se realiza la autenticación el script en Python procede a obtener diferentes tweets de cuentas relacionadas a la educación en Twitter. Luego de obtener los tweets de

las cuentas se procede a guardar en la base de datos MongoDB. Los ítems principales que se guardan son la cuenta a la que pertenece el tweet y el texto del tweet.

5.3.3 Tercera fase

Esta fase está conformada por el diseño y desarrollo del sistema recomendador basado en contenidos y el almacenamiento de los recursos educativos en la base de datos orientada a grafos para la aplicación de diferentes algoritmos que permitan generar búsquedas y recomendaciones aplicando algoritmos propios de Neo4j.

Para poder realizar recomendaciones con base en las actividades obtenidas del cuaderno de trabajo de primer año y los documentos desarrollados para los niños con capacidades especiales se procede a realizar el análisis de las actividades empleando TF-IDF que permite obtener la similitud de los documentos empleando la similaridad del coseno.

El sistema recomendador se desarrolla en Python para esto se crean scripts que permitan obtener las actividades y analizarlas para calcular la similaridad de cada una. En Python se utiliza la librería sklearn que permite aplicar TF-IDF y la similaridad del coseno de los vectores de los documentos para generar recomendaciones de actividades.

```
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics.pairwise import cosine_similarity
from stop_words import get_stop_words
from sklearn.feature_extraction.text import TfidfVectorizer

actividades = lista_actividades['actividades'].tolist()
stop = get_stop_words('spanish')
tfidf = TfidfVectorizer(stop_words=stop)
vector_actividades = tfidf.fit_transform(actividades)
resultadoActividades = []
for actividad in range(len(lista_actividades)):
    sim_aux = []
    #similaridad del coseno entre la actividad y toda la matriz
    cos_sim = cosine_similarity(vector_actividades[actividad],vector_actividades)
```

Figura 21. Algoritmo TD-IDF en Python.

Una vez que se tiene la similaridad de las actividades se procede a crear las relaciones en la base de datos Neo4j. Las relaciones de la base de datos Neo4j permitirá aplicar algoritmos de

búsqueda para poder generar recomendaciones de actividades.

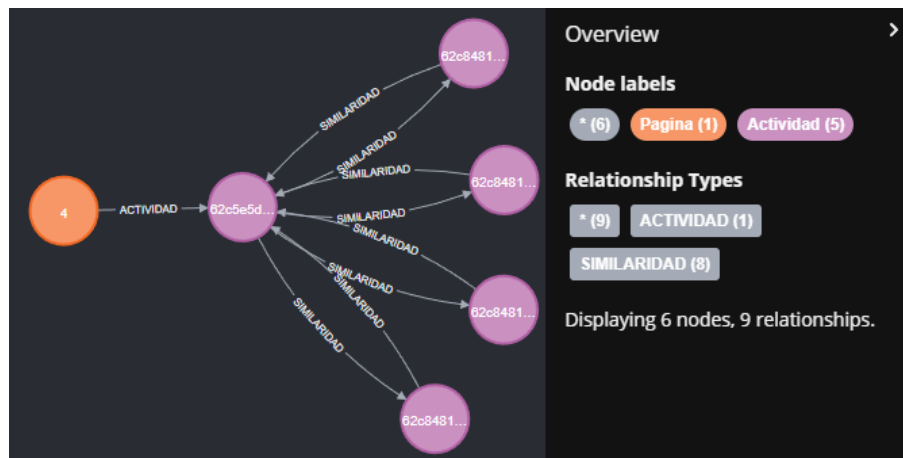


Figura 22. Actividades similares en Neo4j.

Los recursos multimedia recuperados son vídeos de YouTube, de estos recursos se tiene el título, el texto de los subtítulos y el link del vídeo. Para poder vincular a una actividad se procede a utilizar una librería open source que permite obtener la similaridad del texto. Esta librería se encuentra disponible en (Face, sf). La librería SentenceTransformer al igual que TF-IDF convierte los documentos en vectores y aplica la similaridad del coseno, pero emplea modelos de aprendizaje profundo. Los modelos se pueden encontrar en la página de (Face, sf). A continuación se presenta la importación de un modelo de la librería SentenceTransformer:

```
from sentence_transformers import SentenceTransformer, util
model = SentenceTransformer('sentence-transformers/all-MiniLM-L6-v2')
```

Figura 23. Librería SentenceTransformer.

Para vincular una actividad con un vídeo obtenido con el web crawler se procede a realizar una comparación entre la actividad y el título del vídeo o su contenido en texto recuperado mediante los subtítulos. Una vez que se tiene la similitud entre el vídeo y la actividad se procede a crear un nodo de tipo vídeo y vincularlo a la actividad en la base de datos Neo4j como se observa en la Figura 24.

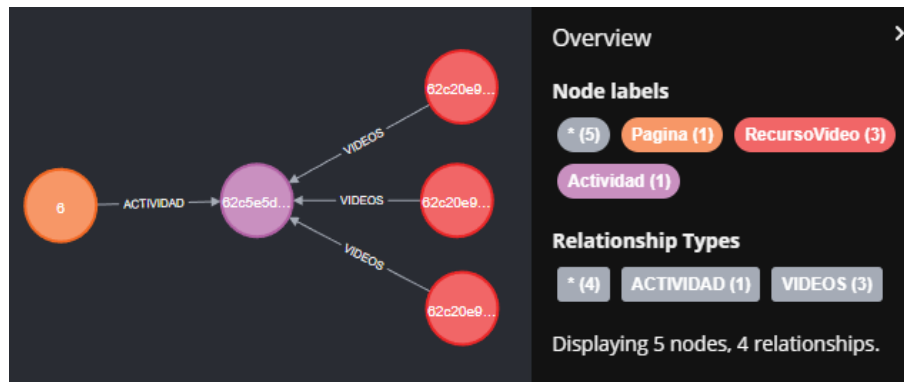


Figura 24. Actividades y vídeos similares en Neo4j.

Esto se aplica con todos los recursos para libros, cuentos, juegos, aplicaciones y tweets. Para cada recurso se creó un nodo de su mismo tipo. Cada relación tiene un peso que es la similaridad calculada con base en el contenido por TF-IDF y la similaridad del coseno para actividades y utilizando la librería de (Face, sf) para los recursos obtenidos.

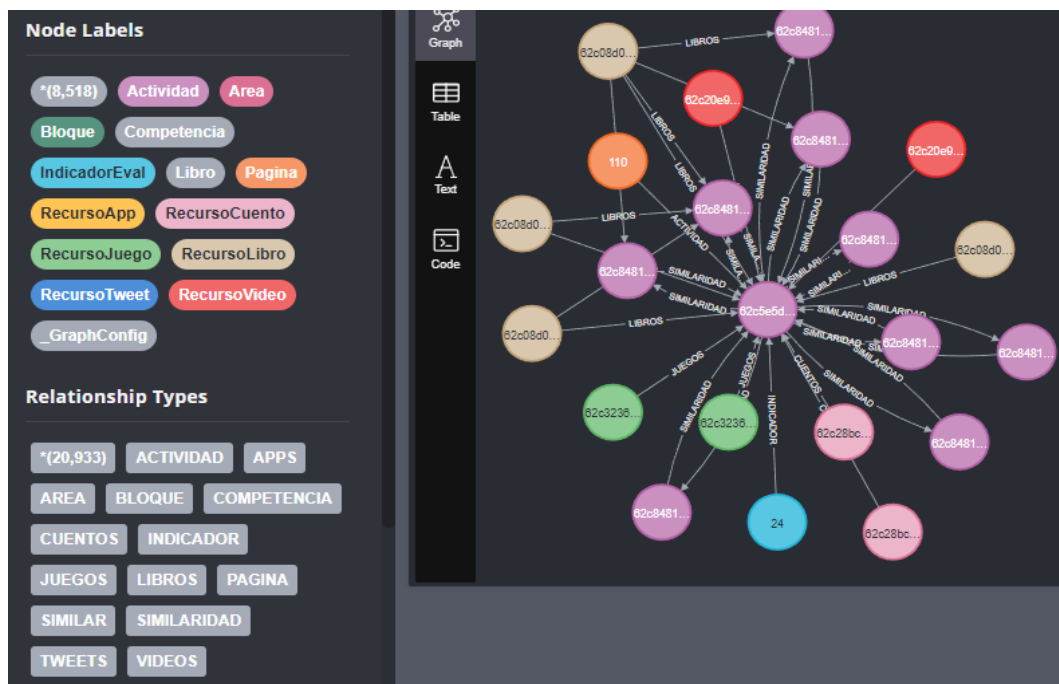


Figura 25. Nodos de actividades y recursos en Neo4j.

Luego de obtener los recursos se procede a establecer las diferentes enfermedades relacionadas a las capacidades especiales. Para obtener las enfermedades basadas en ontologías

en Neo4j se utiliza la librería neosemantics, esta librería permite trabajar con archivos RDF y ontologías. Para establecer las capacidades especiales se trabaja con la base de datos de Human Disease Ontology, se establecen las diferentes enfermedades para poder crear relaciones entre el nodo perfil del usuario y una capacidad especial derivada de una enfermedad clasificada dentro de la ontología.

El perfil del usuario con capacidades especiales está definido por un experto en la salud. La clasificación de la capacidad especial es con base en el código ICD10CM la cual es una clasificación de enfermedades autorizadas por la organización mundial de la salud. El código ICD10CM establecido en el perfil del usuario se vincula a la enfermedad establecida en la base de datos Neo4j.

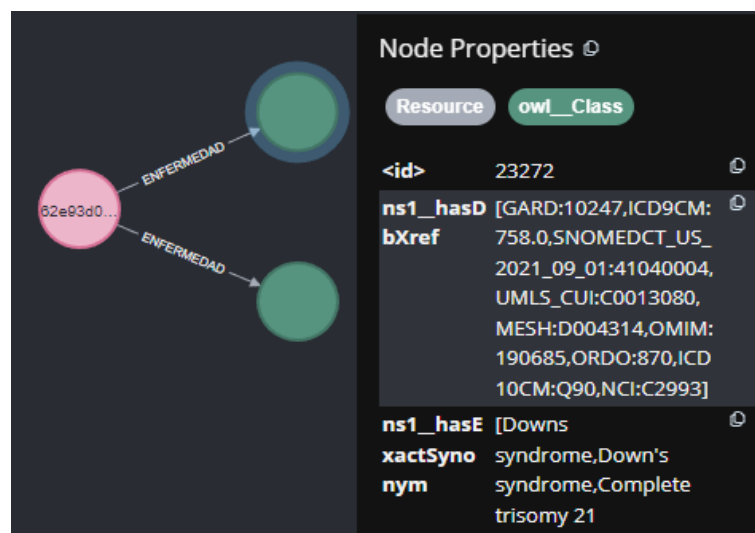


Figura 26. Nodo de perfiles y enfermedad en neo4j.

5.3.4 Cuarta Fase

En esta fase se utiliza FastAPI, para la parte del registro, autenticación de usuarios y otras API's necesarias para el funcionamiento del sistema.

Para el manejo de cada una de las partes de la API se estableció la creación de archivos de Python como se muestra en la figura 27.

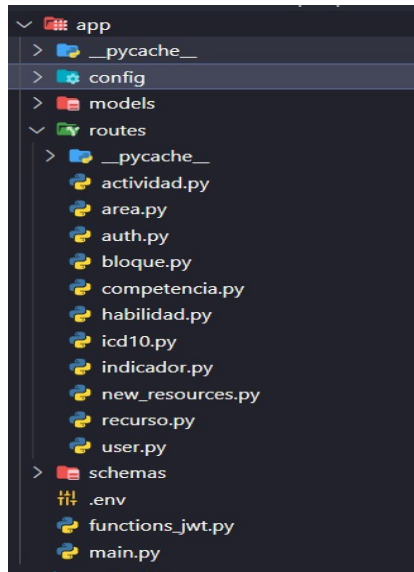


Figura 27. Distribución del proyecto en FastAPI.

Para la funcionalidad de las diferentes API's se conectará con la base de datos no relacional MongoDB donde se tiene almacenada toda la información que requiere la FastAPI, también haremos consultas a la base de datos orientada a grafos Neo4j.

A continuación se presenta la conexión a la base de datos no relacional que requiere FastAPI.

```
conn = MongoClient("mongodb://repoups:repo.2022@repositorioups.online:443/")
```

Tabla 14. Conexión a Mongo.

El framework FastApi nos permite realizar la conexión a la base de datos orientada a grafos Neo4j, a continuación se presenta la línea de la conexión:

```
gds = GraphDataScience("bolt://repositorioups.online/", auth=("neo4j", "test"))
```

Tabla 15. Conexión a Neo4j.

Una de las API's más relevantes es para el registro la cual recibe los datos personales de una

persona con o sin capacidad especial. Esto se verifica con la conexión a MongoDB para validar si el usuario a registrar ya existe caso contrario se procede a registrar con éxito.

Otras de las API's más relevantes es la que se muestra en la tabla 28 que es para la respectiva autenticación, para esto haremos uso de JWT por sus siglas JSON Web Token que simplemente nos recibe unas credenciales y verifica si el usuario existe, si existe nos devolverá un token. Este token sirve para poder mantener una sesión de usuario en la aplicación web que posteriormente se explicara en las siguientes fases.

```
JWT_SECRET = config('secret')
JWT_ALGORITHM = config('algorithm')

def token_response(token: str, usr: str):
    us = conn.TestTesis1.user.find_one({'username': usr})
    us['_id'] = str(us['_id'])
    return {
        'resultado': us,
        'jwt': token
    }

def signJWT(userId: str):
    payload = {
        'userId': userId,
        'expiry': time.time() + 86400
    }
    token = jwt.encode(payload, JWT_SECRET, algorithm=JWT_ALGORITHM)
    return token_response(token, userId)
```

Figura 28. Métodos para generar un token.

Al trabajar con Docker se define el archivo que construirá la imagen de las API's mediante el archivo Dockerfile, que se visualiza en la Figura 29

```
FROM python:3.10.5

ENV PYTHONUNBUFFERED=1

WORKDIR /code

COPY ./requirements.txt /code/requirements.txt

RUN python -m pip install --upgrade pip

RUN pip install --no-cache-dir --upgrade -r /code/requirements.txt

COPY ./app /code/app
```

Figura 29. Dockerfile FastAPI.

Una vez que se tiene almacenado los recursos como vídeos, libros, cuentos, juegos, aplicaciones y tweets con sus respectivas relaciones se procede a emplear algoritmos de búsqueda para obtener los recursos que se pueden recomendar de acuerdo a cada actividad, al igual que determinar actividades similares que pueden recomendarse de acuerdo a las relaciones que se tienen en la base de datos Neo4j.

Para aplicar la recomendación más importante del sistema haremos uso de la librería oficial de Neo4j Graph Data Science, el algoritmo que empleamos se llama Degree Centrality, el principio de este algoritmo es calcular las relaciones que entran o salen con base a un nodo, para la recomendación se necesita poder acceder a la base de datos orientada a grafos Neo4j desde FastAPI llamando al algoritmo, este algoritmo necesita una de las propiedades del nodo en cuestión para poder procesarlo dentro del algoritmo y el resultado del algoritmo se procesa dentro del sistema recomendador para poder generar recomendaciones con base en una actividad.

Para otra de las recomendaciones aplicaremos el algoritmo PageRank, el principio de este algoritmo es medir la influencia de las direcciones entre los nodos, esto significa que toma en cuenta las relaciones de los nodos y los vecinos de un nodo para así tener una mayor influencia en la red de nodos.

5.3.5 Quinta Fase

En esta fase consiste en el desarrollo de la aplicación web y móvil usando el framework Angular. Lo primero sería crear un nuevo proyecto angular con un nombre que distinga a nuestra aplicación como se visualiza en la figura 30.

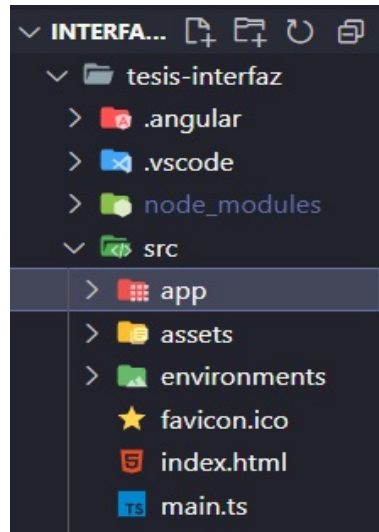


Figura 30. Proyecto Angular generado.

Esta fase hará uso de la fase anterior en el cual todas las API's generadas por FastAPI serán consumidas por el framework Angular. Para poder hacer las demás funcionalidades de la aplicación web es necesario generará más componentes como para el registro, login de usuarios, el listado de las actividades, visualización de recursos con base a una actividad y creación de nuevas actividades.

- **Login:** es la primera pantalla que se visualizara al momento de cargar el sistema, en esta pantalla permitirá acceder al sistema por medio de sus credenciales, en caso de no tener una cuenta se le presentará otro botón que direccionará al proceso de registro.

Bienvenido

Username:

Password:

[Conectarse...](#)

Tiene una cuenta? [Regístrase](#)

Figura 31. Pantalla de inicio de sesión.

- **Registro de usuario:** es la pantalla donde nos permitirá realizar el proceso de registro de la cuenta para el sistema, para esto nos solicita datos personales como cédula, nombre, apellido, fecha de nacimiento, unidad educativa, si tiene una capacidad especial. En caso de existir el usuario se presentará un respectivo mensaje.


Registro

Cedula:

Nombre:

Apellido:

Fecha de Nacimiento:

Unidad Educativa:

Capacidad Especial:

 SI

Figura 32. Pantalla de registro.

En caso de tener una capacidad especial se mostrará unas tablas para el diagnóstico

médico como se visualiza en la figura 33 y el diagnóstico de lenguaje como se visualiza en la figura 34.

Diagnostico Medico:

Busque por codigo.....		
Codigo	Descripcion	Seleccionado
ICD10CM:A00	cholera	<input type="checkbox"/> TIENE
ICD10CM:A00-A09	intestinal infectious disease	<input type="checkbox"/> TIENE
ICD10CM:A01.0	typhoid fever	<input type="checkbox"/> TIENE
ICD10CM:A01.1	paratyphoid fever	<input type="checkbox"/> TIENE
ICD10CM:A01.2	paratyphoid fever	<input type="checkbox"/> TIENE

« Previous **1** 2 3 4 5 ... 733 Next »

Figura 33. Tabla de Diagnóstico Médico.

Diagnostico Lenguaje:

Busque por codigo.....		
Codigo	Descripcion	Seleccionado
ICD10CM:E75.2	hypomyelinating leukodystrophy 3	<input type="checkbox"/> TIENE
ICD10CM:E75.4	neuronal ceroid lipofuscinosis 7	<input type="checkbox"/> TIENE
ICD10CM:E75.4	neuronal ceroid lipofuscinosis 8	<input type="checkbox"/> TIENE
ICD10CM:E77.1	aspartylglucosaminuria	<input type="checkbox"/> TIENE
ICD10CM:F80.1	expressive language disorder	<input type="checkbox"/> TIENE

« Previous **1** 2 3 4 5 6 Next »

Figura 34. Tabla de Diagnóstico Lenguaje.

- **Visualización de actividades:** en esta ventana se logra ver un listado de actividades, estas

actividades se visualizarán con base en el perfil con el cual el usuario ingrese al sistema. Cada actividad tendrá un enlace el cual nos dirigirá a otra ventana en donde se visualiza los recursos educativos en base a una actividad.



Figura 35. Pantalla del Listado de actividades.

- **Visualización de recursos:** en esta ventana se logra ver todos los recursos educativos en la misma ventana en base a la actividad seleccionada en la ventana anterior.



Figura 36. Pantalla del Listado de Recursos en base a una actividad.

- **Recomendación de actividades:** en el momento de mostrar los recursos educativos en

base a una actividad, en la parte derecha se visualiza en un listado las actividades recomendadas en base a la actividad seleccionada.



Figura 37. Listado de las actividades recomendadas en base a una actividad.

- **Creación de una o varias actividades:** para poder ver esta ventana se necesita acceder al sistema con el rol de docente, en esta ventana nos permite crear una actividad como se visualiza en la figura 38 o subir varias actividades de un archivo csv como se visualiza en la figura 39, estas actividades pueden ser para usuarios con o sin capacidad especial.

A screenshot of a web application interface for creating a new activity. The page has a white background with a navigation bar at the top. The navigation bar includes a home icon, the text "INICIO", a green icon for "Crear Actividad", a person icon for "Crear Usuario", and a user profile section showing "Usuario: gaviellalx@hotmail.com - Edad Usuario: 25". On the left and right sides of the main content area, there are vertical orange and yellow bars respectively. The main content area is titled "Nueva Actividad" and contains a checkbox labeled "Para Capacidad Especial". Below this, there are two tabs: "Crear una actividad" (selected) and "Subir archivo csv". There is a text input field for "Nombre de la Actividad/Habilidad:". Below that, there are two radio buttons for "Actividad" and "Habilidad". At the bottom of the form, there is a "Crear" button.

Figura 38. Pantalla de Creación de una Actividad.

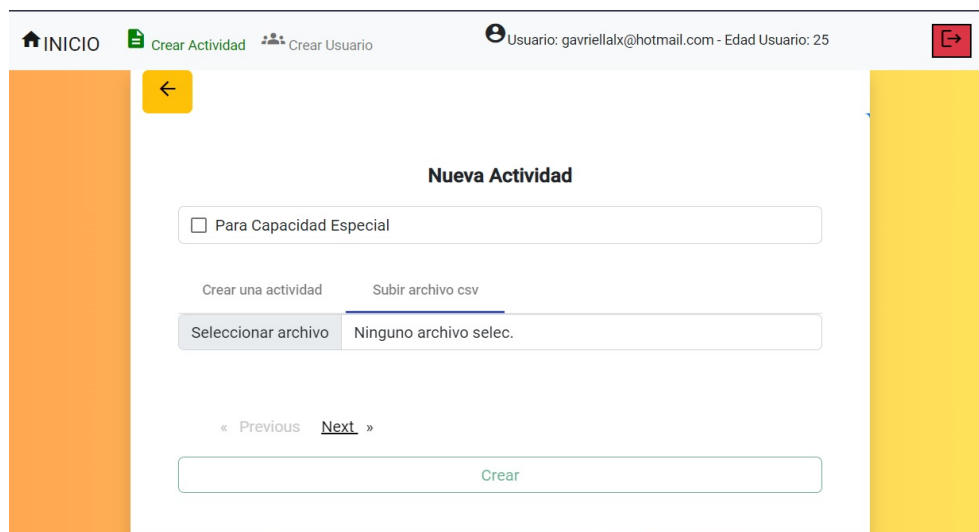


Figura 39. Pantalla de Creación de Actividades mediante un archivo csv.

5.4 Arquitectura

Aquí se presenta el diseño y estructura del proyecto, se especifican la distribución, comunicación y los recursos y tecnologías empleadas para el desarrollo. Nos permite especificar de forma general la distribución del sistema final. En la Figura 40 se muestra la arquitectura desarrollada en el proyecto.

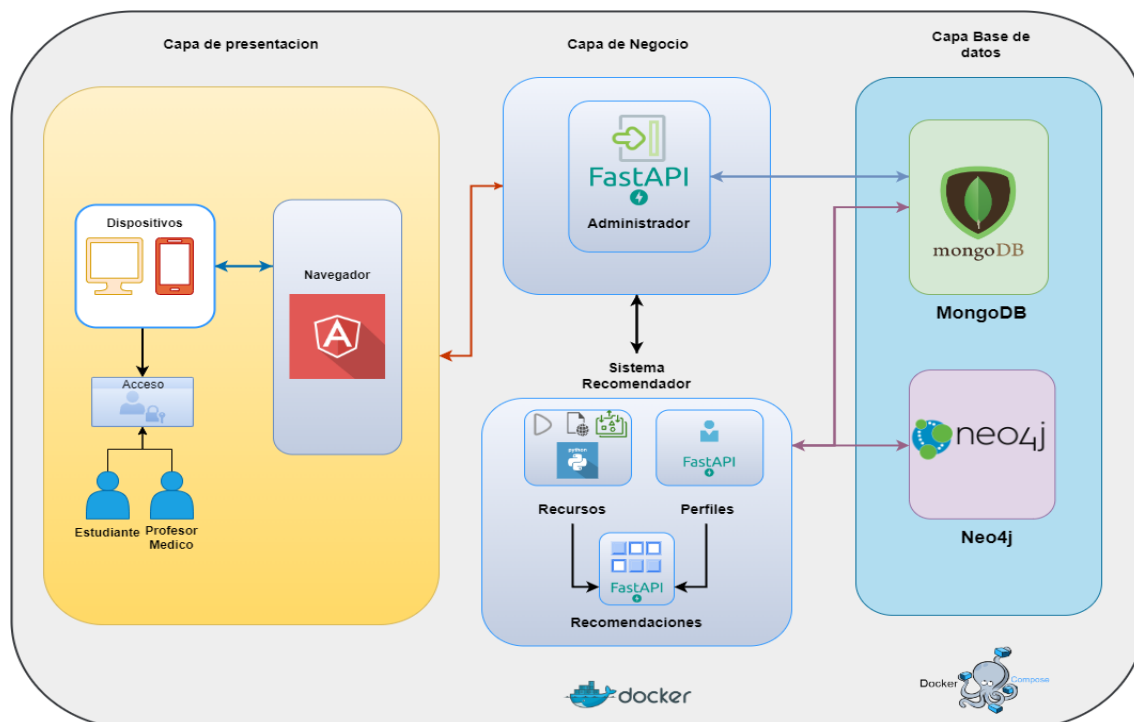


Figura 40. Arquitectura del sistema.

5.4.1 Capa de base de datos

Aquí se encuentran las bases de datos NoSQL del sistema que permite el almacenamiento de los recursos y perfiles del usuario. A continuación se presentan los componentes que comprende esta capa.

- **MongoDB:** La base de datos MongoDB nos permite crear colecciones y almacenar diferentes datos en documentos. La base de datos MongoDB almacena los recursos recuperados mediante los web crawler.
- **Neo4j:** Esta base nos permite almacenar los recursos creando nodos e identificándolos con el ID de MongoDB para crear relaciones entre los nodos y aplicar algoritmos para realizar búsquedas y generar recomendaciones de los diferentes recursos.

5.4.2 Capa de Negocio

Es donde se encuentra toda la funcionalidad del sistema y realiza las principales operaciones. Esta capa permite la interacción con las base de datos. Esta comprende los siguientes módulos:

- **Lógica de Negocio:** Esto permite determinar las reglas de comunicación y el flujo de datos entre la base de datos y la aplicación. Esto permite el registro de los usuarios, almacenamiento de las actividades y sus respectivos recursos.
- **Sistema Recomendador:** Esta parte permite generar las inferencias y recomendaciones con base en el contenido de las actividades y sus recursos. En esta parte se complementa con la base de datos Neo4j para emplear algoritmos de búsqueda que permitan obtener los recursos y generar recomendaciones.

5.4.3 Capa de presentación

Es parte fundamental del sistema en donde se hace la presentación visual al usuario final conteniendo las funcionalidades establecidas, tanto para el registro, acceso o otras funcionalidades, para el acceso se lo hace obteniéndolo con algún tipo de autenticación.

Para la construcción de la aplicación con Angular se sabe que este no usa MVC y que no tiene establecido un patrón de diseño propio, pero se puede decir que utiliza MVVM que significa Modelo-Vista Vista-Modelo donde el modelo tiene mucha relación con la vista, esto significa que cuando se sincroniza los datos con la vista y el modelo-vista son totalmente dependientes, esto quiere decir que estando en la vista se puede modificar el modelo y cuando se está en el modelo se puede modificar la vista.

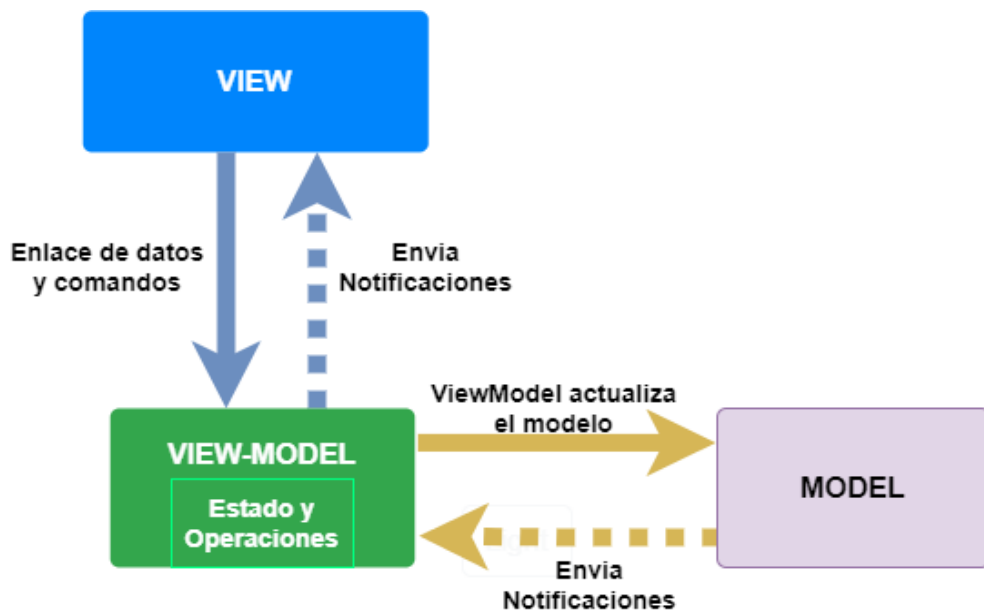


Figura 41. Diagrama Patron MVVM.

La estructura del proyecto con la cual se construyo el software se hizo para poder tener escalabilidad y dar mantenimiento, con esto se logra que al modificar o agregar cambios al software se vuelva sencillo y no de problemas entre los demás componentes.

- **main:** está compuesta por los componentes que conforman el sistema, cada componente cuenta con su propio modelo-vista.
- **models:** está compuesta por las clases que se usaran dentro de los modelos para declaración de objetos,
- **services:** está compuesta por los servicios de API's externas que requiere la aplicación para poder hacer las funcionalidades.
- **environments:** se encuentran declaradas todas las url's de las API's necesarias.

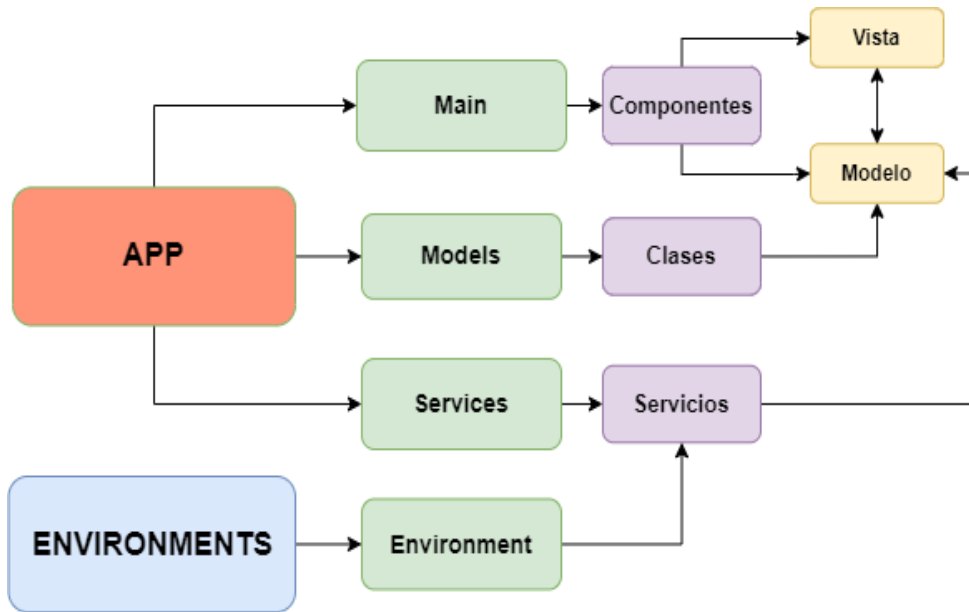


Figura 42. Diagrama Estructura Proyecto.

Al manejar esta estructura del proyecto, las carpetas del proyecto quedan ordenadas de la siguiente manera como se visualiza en la figura 43.

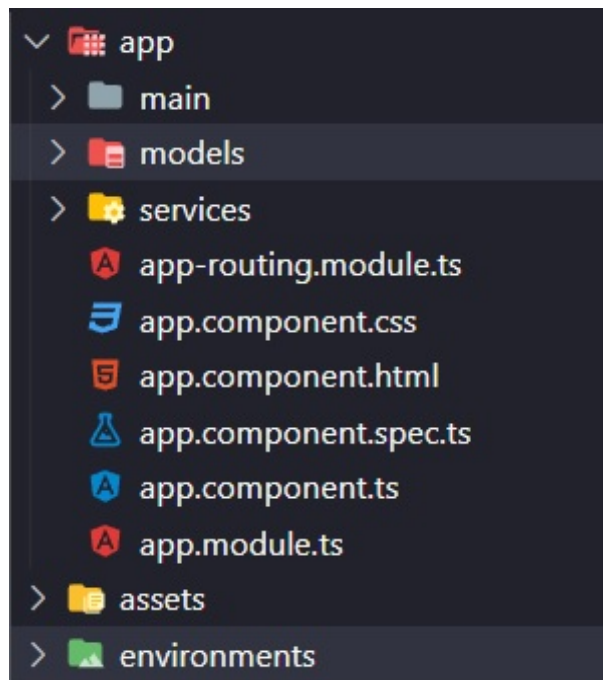


Figura 43. Estructura Proyecto Angular.

Capítulo VI

Resultados

A continuación se muestran las pruebas para obtener los recursos y generar recomendaciones. Se aplicaron pruebas unitarias para la validación de la aplicación web y móvil.

6.1 Resultados de actividades y recursos

Se obtiene actividades del libro de primer año de educación básica. Las actividades se obtienen mediante scripts, luego se procede a guardar las actividades en la base de datos MongoDB. Las actividades obtenidas tienen una página, título, actividad y un indicador de evaluación.




_id	pagina	N_Actividad	titulo	actividades	analisis	indicadores
 62c5e5d3748e9bd1d0ab9b6f	4	1	Actividades	Mi escuela, mi aula Completa las líneas punteadas...	escuela, aula, Completa, líneas, punteadas, llega...	• Interpreta la silueta y los paratextos.
 62c5e5d3748e9bd1d0ab9b70	5	1	Actividades	Traza las líneas punteadas y pinta los útiles esc...	Traza, líneas, punteadas, pinta, útiles, escolare...	• Interpreta la silueta y los paratextos.
 62c5e5d3748e9bd1d0ab9b71	6	1	Actividades	Listen and repeat. book eraser chair teacher penc...	Listen, repeat, book, eraser, chair, teacher, pen...	• Understand familiar words about school.
 62c5e5d3748e9bd1d0ab9b72	7	1	Recuerda los personajes del cuento El caso de las...	Juega y nombra de uno en uno a los animales de la...	Juega, nombra, animales, ilustración. Ganará, nom...	Completar Actividad
 62c5e5d3748e9bd1d0ab9b73	7	2	Recuerda los personajes del cuento El caso de las...	Identifica los sonidos que hacen estos animales. ...	Identifica, sonidos, animales, Pidan, profesor, e...	Completar Actividad
 62c5e5d3748e9bd1d0ab9b74	8	1	Actividades	Escucha la frase que dirá tu profesor y aplaude p...	Escucha, frase, dirá, profesor, aplaude, palabra,...	• Registra, expresa y comunica ideas mediante sus...

Figura 44. Actividades en MongoDB.

6.1.1 Actividades con recursos

Seguidamente de obtener las actividades se procede a obtener recursos. Los recursos obtenidos son para cada actividad, estos recursos son vídeos, libros, cuentos, juegos, aplicaciones y tweets. Los recursos complementan la actividad para una mejor comprensión y desarrollo.

View	Export	[JSON]	Import	RecursoCuentosCatg	Del
View	Export	[JSON]	Import	RecursoJuegos	Del
View	Export	[JSON]	Import	RecursoLibros	Del
View	Export	[JSON]	Import	RecursoVideos	Del
View	Export	[JSON]	Import	RecursosAppsR	Del
View	Export	[JSON]	Import	RecursosTweets	Del

Figura 45. Recursos en MongoDB.

Los recursos obtenidos se almacenan en la base de datos Neo4j que permiten crear relaciones con las actividades para generar búsquedas, aplicar algoritmos, obtener recomendaciones de actividades y recursos en base a sus relaciones.

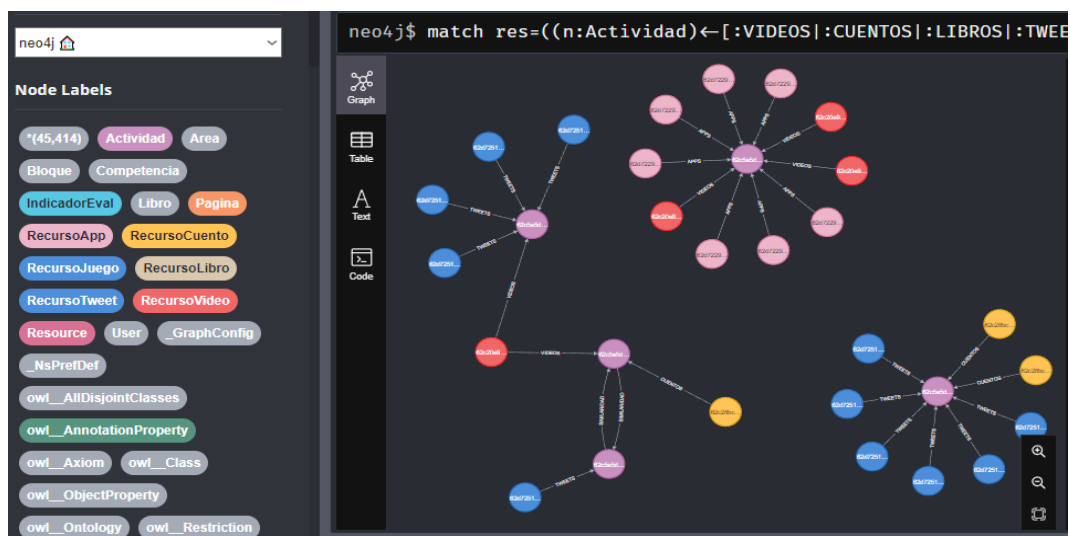
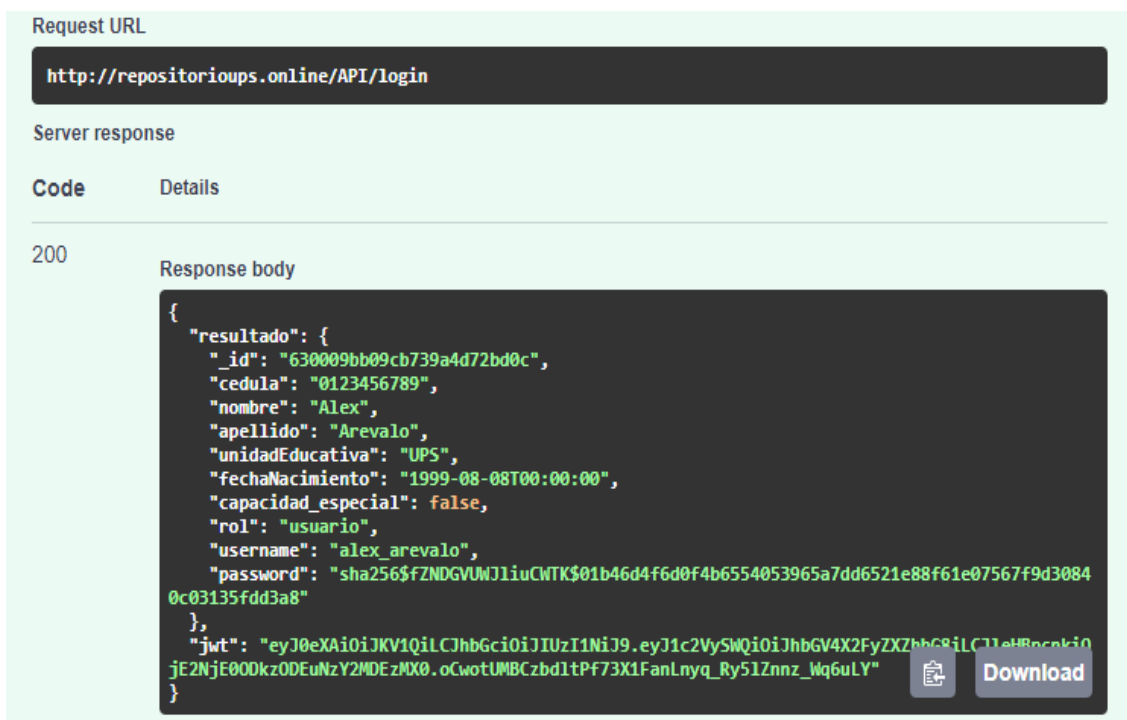


Figura 46. Actividades y sus recursos en Neo4j..

6.2 Pruebas Unitarias

Para la validación del sistema se utilizó pruebas unitarias. Las pruebas unitarias nos permiten verificar el correcto funcionamiento del código del sistema. A continuación se muestran pruebas realizadas al sistema mediante la documentación de FastAPI.

Se realizó una prueba de acceso, esto permite verificar el correcto funcionamiento de acceso con sus respectivas credenciales.



The screenshot displays an API-REST client interface. At the top, the 'Request URL' is set to 'http://repositorioups.online/API/login'. Below this, the 'Server response' section shows a 'Code' of 200 and a 'Response body' containing a JSON object with user details and a JWT token. The JSON object includes fields for '_id', 'cedula', 'nombre', 'apellido', 'unidadEducativa', 'fechaNacimiento', 'capacidad_especial', 'rol', 'username', and 'password'. The JWT token is also visible. A 'Download' button is present next to the response body.

```
{
  "resultado": {
    "_id": "630009bb09cb739a4d72bd0c",
    "cedula": "0123456789",
    "nombre": "Alex",
    "apellido": "Arevalo",
    "unidadEducativa": "UPS",
    "fechaNacimiento": "1999-08-08T00:00:00",
    "capacidad_especial": false,
    "rol": "usuario",
    "username": "alex_arevalo",
    "password": "sha256$fZNDGVUMJ1iuCWTk$01b46d4f6d0f4b6554053965a7dd6521e88f61e07567f9d30840c03135fdd3a8"
  },
  "jwt": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1c2VySWQiOiJhbGV4X2FyZXZhbGciLCJ1eHRocm9kaWkiOiJlE2NjE0ODkzODEuNzY2MDEzMX0uOGwotUMBCzbd1tPf73X1FanLnyq_Ry5lZnZz_Wq6uLY"
}
```

Figura 47. API-Rest Login del sistema.

Para la obtención de las actividades y sus recursos se tiene un servicio que permite obtener y listar los recursos de una actividad en base a su identificador único.

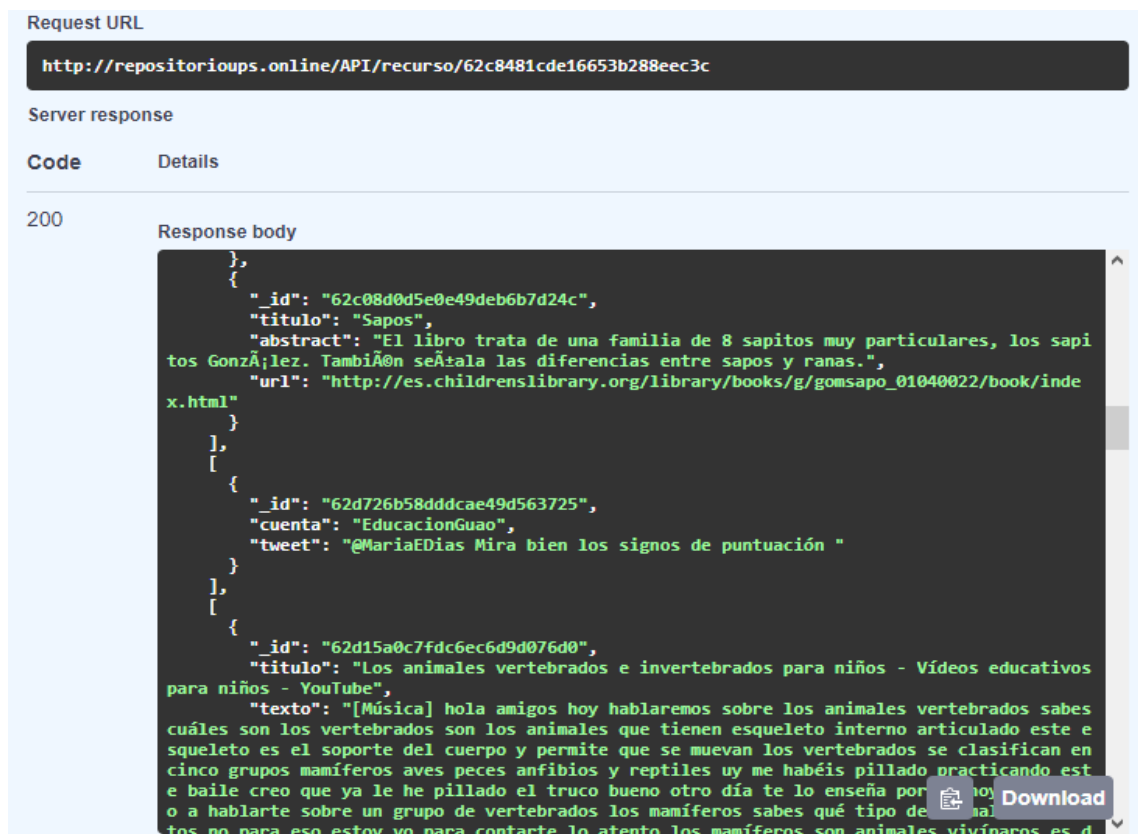


Figura 48. API-Rest Lista de recursos.

6.2.1 Validación Web y Móvil

Para validar la aplicación web y móvil la cual serán las mismas para las dos formas, lo primero es explicar como funciona cada componente como es el registro de usuarios con opción a dos roles, poder autenticarse con las credenciales que se guardan al momento de registrarse, buscar y observar actividades educativas, ver recursos educativos de cada actividad, recomendación de actividades en base a una actividad, en base a rol docente se procede a la creación de una o más actividades.

- Tanto como para el niño como para el doctor/docente, crearon sus cuentas para posteriormente ingresar al sistema.

- Al ingresar con las credenciales correctas, se visualizará actividades educativas dependiendo si el niño tiene capacidades especiales, El doctor/docente podrá ver todas las actividades.
- Una vez visualizado las actividades educativas, tanto como usuario niño o doctor/docente podrán ver recursos educativos de varios tipos.
- El usuario docente tiene permitido crear actividades nuevas para los perfiles tanto con o sin capacidad especial.

Todas estas funcionalidades explicadas tienen la misma forma de interactuar tanto en web como móvil.

6.3 Análisis de Encuesta

El análisis de la encuesta se presenta a continuación, esta encuesta se realizó a 10 estudiantes con la ayuda de sus representantes.

Pregunta 1. ¿Cuál es su posición frente al uso de nuevas tecnologías como internet y smartphones para el uso en la educación?

Alternativas	Respuesta	Porcentaje
Muy importante	4	40%
Importante	4	40%
Moderadamente importante	2	20%
Poco importante	0	0%
Nada importante	0	0%

Tabla 16. Respuestas de la Pregunta 1.

Pregunta 1

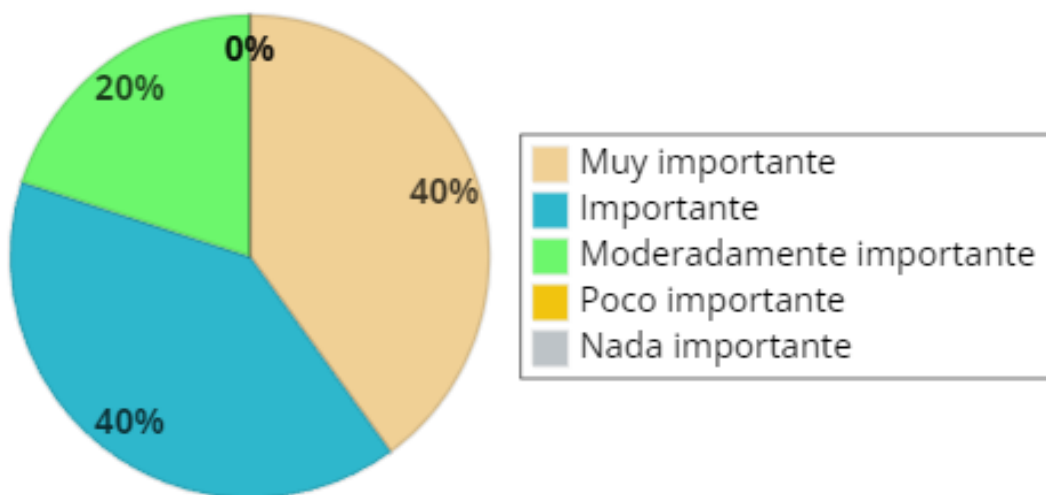


Figura 49. Representación porcentual pregunta 1.

En la encuesta del 100% de los usuarios en la pregunta uno, el 40% de los encuestados consideran muy importante el uso de las nuevas tecnologías, el 40% consideran importante y el 20% moderadamente importante. Esto significa que en la educación es muy importante el uso de los smartphones y el internet.

Pregunta 2. ¿Estás de acuerdo en que un sistema ofrezca sugerencias?

Alternativas	Respuesta	Porcentaje
Totalmente de acuerdo	6	60%
De acuerdo	4	40%
Ni de acuerdo ni en desacuerdo	0	0%
En desacuerdo	0	0%
Totalmente en desacuerdo	0	0%

Tabla 17. Respuestas de la Pregunta 2..

Pregunta 2

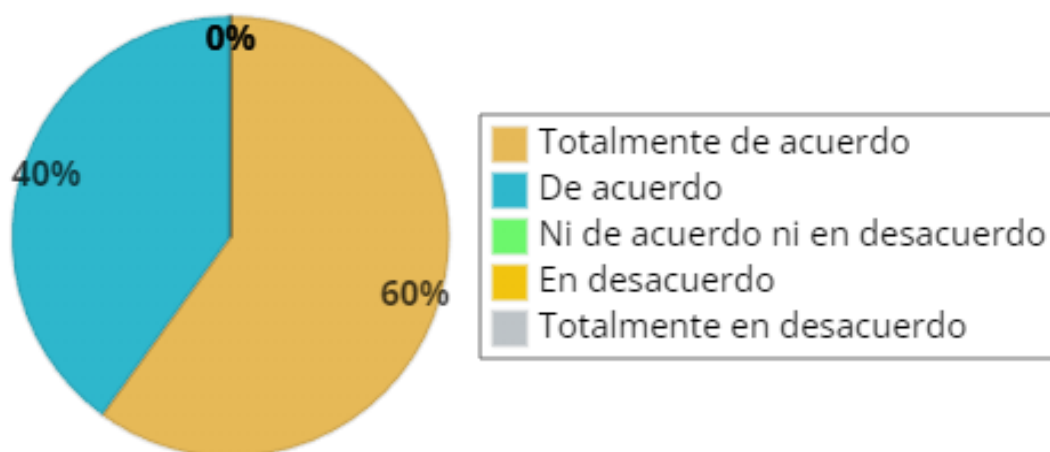


Figura 50. Representación porcentual pregunta 2.

En la encuesta del 100% de los usuarios en la pregunta dos, el 60% de los encuestados esta totalmente de acuerdo con que se ofrezca recomendaciones, el 40% esta de acuerdo. Esta representa que los sistemas que ofrecen recomendaciones son muy aceptados por los beneficios que ofrecen a los usuarios.

Pregunta 3. ¿Estás de acuerdo con las recomendaciones de actividades a realizar?

Alternativas	Respuesta	Porcentaje
Totalmente de acuerdo	2	20%
De acuerdo	7	70%
Ni de acuerdo ni en desacuerdo	1	10%
En desacuerdo	0	0%
Totalmente en desacuerdo	0	0%

Tabla 18. Respuestas de la Pregunta 3.

Pregunta 3

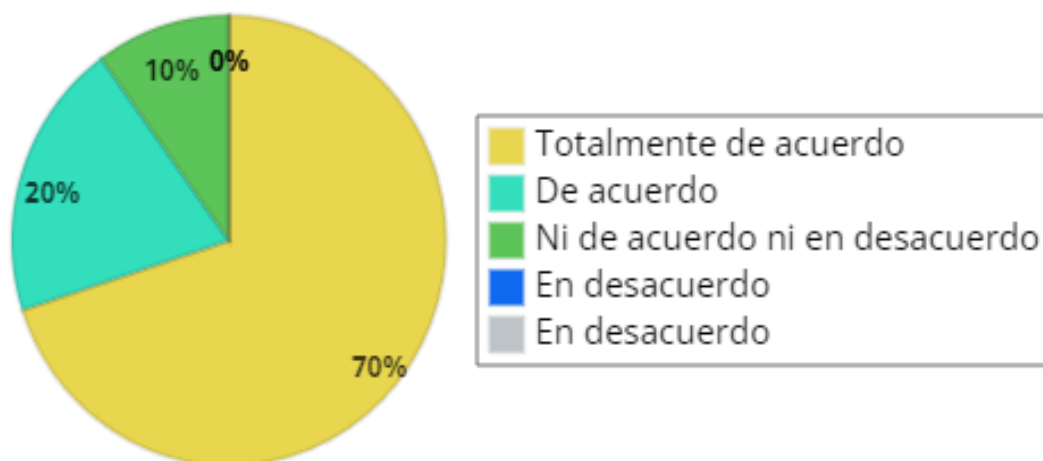


Figura 51. Representación porcentual pregunta 3..

En la encuesta del 100% de los usuarios en la pregunta tres, el 20% de los encuestados esta totalmente de acuerdo con las recomendaciones, el 70% esta de acuerdo y el 10% no esta de acuerdo ni en desacuerdo. Esto representa que las recomendaciones que genera el sistema son aceptadas por los usuarios.

Pregunta 4. ¿Estás de acuerdo con las recomendaciones de recursos para las actividades a realizar?

Alternativas	Respuesta	Porcentaje
Totalmente de acuerdo	3	30%
De acuerdo	6	60%
Ni de acuerdo ni en desacuerdo	1	10%
En desacuerdo	0	0%
Totalmente en desacuerdo	0	0%

Tabla 19. Respuestas de la Pregunta 4.

Pregunta 4

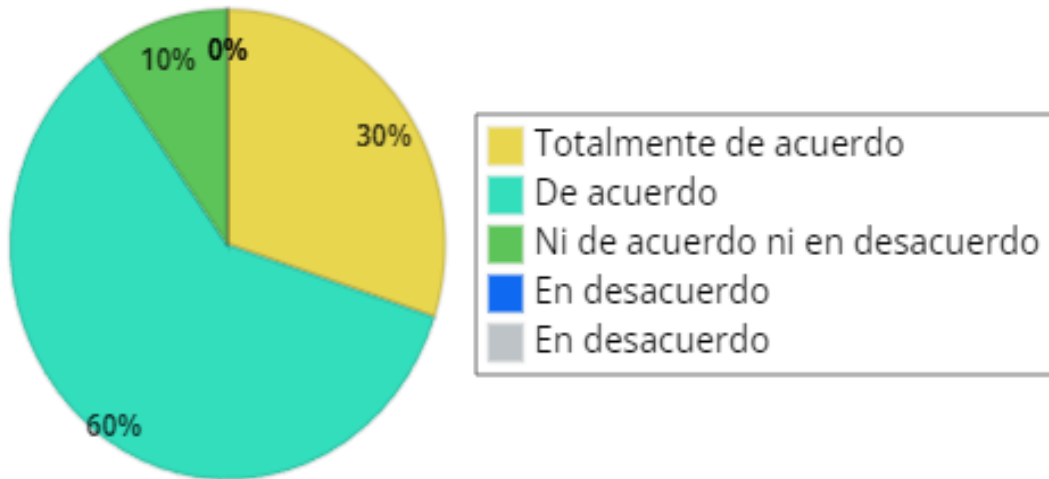


Figura 52. Representación porcentual pregunta 4.

En la encuesta del 100% de los usuarios en la pregunta cuatro, el 30% esta totalmente de acuerdo con las recomendaciones de los recursos, el 60% esta de acuerdo y el 10% no esta de acuerdo ni desacuerdo. Esto significa que los recursos recomendados para las actividades son de gran ayuda para los usuarios del sistema.

Pregunta 5. ¿Cuál es tu nivel de satisfacción con el uso de la aplicación?

Alternativas	Respuesta	Porcentaje
Altamente satisfecho	3	30%
Muy satisfecho	6	60%
Satisfecho	1	10%
Poco satisfecho	0	0%
Completamente insatisfecho	0	0%

Tabla 20. Respuestas de la Pregunta 5.

Pregunta 5

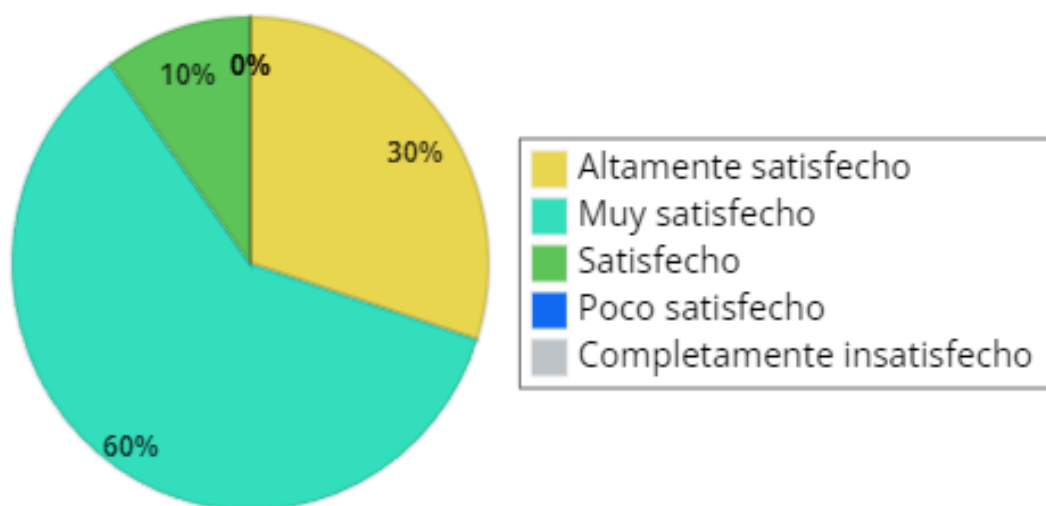


Figura 53. Representación porcentual pregunta 5..

En la encuesta del 100% de los usuarios en la pregunta cinco, el 30% está altamente satisfecho con el uso de la aplicación, el 60% está muy satisfecho y el 10% satisfecho. El uso de la aplicación cumple con las expectativas y necesidades del usuario.

Luego de realizar un análisis de la encuesta y los resultados obtenidos se procedió a calcular el nivel de fiabilidad de la encuesta según Alfa de Cronbach.

Indice	Nivel de fiabilidad	Valor de Alfa de Cronbach
1	Excelente	[0.9,1]
2	Muy Bueno	[0.7,0.9]
3	Bueno	[0.5,0.7]
4	Regular	[0.3,0.5]
5	Deficiente	[0,0.3]

Tabla 21. Clasificación de los niveles de fiabilidad según el Alfa de Cronbach.

Para el calculo de se determina la varianza de los resultados de cada pregunta de acuerdo a los resultados obtenidos

	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Total
Varianza	0,56	0,24	0,29	0,36	0,36	3,61

Tabla 22. Tabla de resultados de varianza de la encuesta..

$$AlfadeCronbach = \left(\frac{N^0 Preguntas}{N^0 Preguntas - 1} \right) * \left(1 - \frac{\sum Varianza}{Total} \right)$$

$$AlfadeCronbach = \left(\frac{5}{5 - 1} \right) * \left(1 - \frac{1.81}{3.61} \right) = 0.63$$

Luego de realizar el calculo se obtienen que el nivel de fiabilidad de la encuesta realizada a diez usuarios de la aplicación esta en el rango Alfa de Cronbach de buena.

Capítulo VII

Cronograma

A continuación se detalla el cronograma de actividades que se siguió para el desarrollo del proyecto. Se describe las actividades desarrolladas, la fecha y las horas requeridas para su cumplimiento.

Actividad Desarrollada	Recursos, Materiales, Conocimientos Requeridos-Adquiridos/Resultados	Fechas	Horas Requeridas	Realizado
Estudio de los fundamentos de los Sistemas de Recomendación.	Se realizo una investigación y estudio sobre sistemas recomendadores y la estructura base a seguir para la investigación.	mar 05/01/2021	Tutor: 2 horas Estudiante: 4 horas	SI
Investigación de las últimas tecnologías referente a Sistemas de Recomendación.	Se investigo las tecnologías actuales sobre los sistemas de recomendación y su estado actual.	mar 05/02/2021	Tutor: 2 horas Estudiante: 3 horas	SI

Revisar las técnicas basadas en el contenido.	Se reviso técnicas limítrofes para el análisis del contenido. Se realizo un análisis documental de los textos educativos propuestos por el ministerio de educación para los niños de 5 y 6 años.	mie 06/01/2021	Tutor: 2 horas Estudiante: 4 horas	SI
Conocer los enfoques de los Sistemas de Recomendación para perfiles de usuarios.	Se reviso los diferentes modelos de sistemas de recomendación, En el cual emplearemos un sistema de recomendación basada en contenidos.	mie 06/01/2021	Tutor: 2 horas Estudiante: 4 horas	SI
Identificar algoritmos aplicados en sistemas de recomendación basados en contenidos y perfiles.	Se reviso diferentes algoritmos para recomendación de contenidos, Para ello es utilizará la base de datos orientada a grafos Neo4J la cual nos permitirá implementar algoritmos de búsqueda y recomendaciones.	jue 07/01/2021	Tutor: 2 horas Estudiante: 5 horas	SI
Estudiar los fundamentos de las bases de datos orientadas a grafos.	Se realizo una investigación sobre bases de datos orientadas a grafos para el almacenamiento de contenido y perfiles de usuario.	jue 07/01/2021	Tutor: 3 horas Estudiante: 6 horas	SI

Investigar sobre los procesos de inferencia.	Acerca de los procesos de inferencia se investigó los diferentes enfoques y se llevará un enfoque basado en contenidos en donde se recogerá información, se tomó diferentes muestras para su respectiva integración al sistema de recomendación.	vie 08/01/2021	Tutor: 4 horas Es- tudiante: 11 horas	SI
Estudiar sobre el impacto de las generaciones que tendrán los materiales educativos con el pasar de los años.	Se estudio sobre el impacto que tienen las generaciones con los materiales educativos que al pasar de los años actualiza sus contenidos en base a los modelos de enseñanza de acuerdo al Ministerio de Educación.	mar 12/01/2021	Tutor: 3 horas Es- tudiante: 6 horas	SI
Estudiar e investigar conceptos relacionados con materiales didácticos para niños de 5 a 7 años.	Se investigo los materiales didácticos y recursos educativos que utilizan los maestros en la actualidad para la enseñanza de acuerdo al ministerio de educación.	mie 13/01/2021	Tutor: 4 horas Es- tudiante: 10 horas	SI

Investigar materiales didácticos para niños de 5 a 7 años con discapacidad.	Se investigó sobre los materiales didácticos que se emplean para la enseñanza a los niños con discapacidad de acuerdo al ministerio de educación.	jue 14/01/2021	Tutor: 4 horas Estudiante: 13 horas	SI
Conocer repositorios educativos externos de materiales educativos.	Se busco e interactuó con repositorios que están alojados en la web para su análisis y obtención de recursos para formar nuestra base de datos.	sab 16/01/2021	Tutor: 4 horas Estudiante: 15 horas	SI
Analizar perfiles de niños de 5 a 7 años.	Se investigo los perfiles de los niños de 5 a 7 años para su análisis y almacenamiento en la base de datos orientada a grafos.	mar 19/01/2021	Tutor: 4 horas Estudiante: 10 horas	SI
Comprender los perfiles de niños con discapacidad de 5 a 7 años.	Se investigo los perfiles de los niños de 5 a 7 años con discapacidades para su análisis y almacenamiento en la base de datos orientada a grafos.	mie 20/01/2021	Tutor: 4 horas Estudiante: 16 horas	SI

Desarrollar una estructura basada en contenedores.	Se desarrollo una estructura de contenedores con la base de datos orientada a grafos Neo4j y el Sistema de recomendación empleando los algoritmos de Neo4j.	sab 23/01/2021	Tutor: 8 horas Es- tudiante: 70 horas	SI
Definir contenedores para los procesos de inferencia.	Se desarrollo diferentes contenedores para cada uno de los procesos de inferencia del sistema recomendador utilizando Docker.	dom 09/05/2021	Tutor: 8 horas Es- tudiante: 20 horas	SI
Implementar procesos de inferencia a través de contenedores.	Se desarrollo un sistema recomendador para cada uno de los procesos que permiten generar recomendaciones, los procesos se encuentran alojados en los contenedores de Docker.	lun 12/07/2021	Tutor: 6 horas Es- tudiante: 25 horas	SI
Integrar con el sistema recomendador basado en contenidos.	Se desarrollo un sistema recomendador basados en contenidos, este sistema se integra con los diferentes procesos de inferencia.	vie 01/10/2021	Tutor: 6 horas Es- tudiante: 20 horas	SI

Desarrollar un sistema inteligente para la recomendación de materiales didácticos.	El sistema permite recomendaciones empleando el sistema recomendador basado en contenidos y los diferentes procesos de inferencia y obteniendo recursos educativos del repositorio.	vie 15/10/2021	Tutor: 10 horas Estudiante: 73 horas	SI
Implementar un módulo para el análisis de perfil y recuperación de recursos educativos.	El sistema permite ingresar y analizar diferentes perfiles para recuperar y generar recomendaciones de recursos educativos.	lun 29/11/2021	Tutor: 10 horas Estudiante: 60 horas	SI
Desarrollar e implementar un módulo para generar y recomendar materiales didácticos.	Se desarrollo un módulo que permite obtener recursos educativos y realizar recomendaciones en base a una actividad y perfil de usuario.	lun 03/01/2022	Tutor: 10 horas Estudiante: 73 horas	SI
Diseñar y desarrollar un repositorio para almacenamiento de materiales educativos.	Se diseño y desarrollo un repositorio para almacenar actividades y recursos educativos.	sab 15/01/2022	Tutor: 6 horas Estudiante: 35 horas	SI
Recuperar y formar un repositorio de materiales educativos.	Se obtuvo recursos educativos como cuentos, libros, juegos, videos, aplicaciones y tweets.	mar 18/01/2022	Tutor: 6 horas Estudiante: 33 horas	SI

Diseñar de la estructura de la base de datos para los ingresos de usuarios con su perfil respectivo recolectando todos sus atributos.	El almacenamiento de los diferentes usuarios con sus atributos se realiza empleando la base de datos MongoDB.	lun 07/02/2022	Tutor: 4 horas Es- tudiante: 24 horas	SI
Desarrollar la relación de la BDOG con el sistema recomendador.	El sistema permite obtener recursos de acuerdo al sistema recomendador basado en contenido, la base de datos Neo4j almacena los recursos generados y sus relaciones.	lun 21/02/2022	Tutor: 6 horas Es- tudiante: 25 horas	SI
Implementar el sistema recomendador con la BDOG.	El sistema recomendador basado en contenido se comunica e integra con la BDOG para realizar recomendaciones empleando diferentes algoritmos propios de Neo4j.	lun 07/03/2022	Tutor: 8 horas Es- tudiante: 43 horas	SI
Definir herramientas de desarrollo web y móvil.	Se definió el framework Angular para el desarrollo de aplicaciones web y móvil.	dom 20/03/2022	Tutor: 4 horas Es- tudiante: 12 horas	SI

Desarrollar la estructura de la aplicación web y móvil.	Se desarrollo la estructura de la aplicación web siguiendo el patrón Modelo-Vista Vista-Modelo del framework Angular.	lun 28/03/2022	Tutor: 10 horas Es- tudiante: 84 horas	SI
Implementar el ingreso de perfiles de los niños.	Se implemento el ingreso respectivo de los perfiles de los niños con o sin capacidades especiales.	lun 11/04/2022	Tutor: 6 horas Es- tudiante: 24 horas	SI
Desarrollar e implementar la recuperación y visualización de las recomendaciones.	Se desarrollo e implemento la visualización de recomendaciones en base a las actividades.	lun 11/04/2022	Tutor: 6 horas Es- tudiante: 24 horas	SI
Ejecutar pruebas del sistema.	Se realizó pruebas unitarias a las API's más destacadas del sistema para determinar los tiempos de respuestas y su correcto funcionamiento.	mar 17/05/2022	Tutor: 6 horas Es- tudiante: 25 horas	SI
Diseñar un plan de experimentación para saber el impacto que tendrá el producto final.	Se diseño una encuesta la cual se realizó a 10 personas para determinar el impacto de la aplicación final.	lun 30/05/2022	Tutor: 10 horas Es- tudiante: 56 horas	SI

Recopilar datos sobre el plan de experimentación.	Una vez realizada la encuesta se realiza el cálculo de Alpha de CronBach que permite calcular la fiabilidad de la encuesta realizada a los usuarios.	lun 13/06/2022	Tutor: 5 horas Es- tudiante: 15 horas	SI
Hacer mantenimiento y retroalimentación del sistema.	Se hizo el respectivo mantenimiento en cuanto a validaciones para el sistema.	lun 27/06/2022	Tutor: 5 horas Es- tudiante: 12 horas	SI
Total de horas de trabajo:			Tutor: 180 horas Estu- diente: 1600 horas	

Capítulo VIII

Presupuesto

En esta sección se detalla el presupuesto requerido para el desarrollo del proyecto. Se detalla los recursos, la cantidad y el costo unitario y total de cada uno.

Denominación	Cantidad (unidades)	Costo Unitario (dolares)	Costo Total (dolares)
1. Bienes			
Copias	10	0.01	0.1
Impresiones	30	0.05	1.5
2. Tecnológico			
Computador Portátil	2	700	1400
Celular Inteligente	2	300	600
Servidor en Nube	1	80	80
Dominio	1	2	2
3. Servicios			

Servicio de transporte	50	0.3	15
Servidor de Internet	6	35	210
Alimentación	130	2.5	325
4. Personal			
Estudiante Investigador	2	1000	2000
Asesoría Especializada	1	800	800
5. Otros			
Imprevistos	1	500	500
Total	235	3419.86	5933.6

Capítulo IX

Conclusiones

- En base al estado del arte de los sistemas de recomendación y sus técnicas empleadas para generar recomendaciones como la recomendación basado en contenidos, se comprobó su importancia y la ayuda que brindan a los usuarios en las aplicaciones actuales en las que son empleados, permitiendo facilitar resultados en búsquedas de acuerdo a un contenido o perfil del usuario. La información recuperada se encuentran los principales conceptos, así como ejemplos de implementación de los mismos permitiendo formar y entender la importancia y el enfoque de los sistemas de recomendación basado en contenidos.
- El almacenamiento de la información y diferentes recursos se realizó empleando bases de datos NoSQL, dentro de este tipo de base de datos tenemos a MongoDB y Neo4j. La base de datos MongoDB permite trabajar con colecciones y documentos permitiéndonos almacenar datos sin establecer un esquema, esto posibilita la agilidad, robustez y fiabilidad del almacenamiento. La base de datos Neo4j permite trabajar con nodos, y emplear diferentes algoritmos propios de la base de datos que permiten realizar inferencias y predicciones, esta base de datos orientada a grafos permite agilizar el trabajo en la hora de aplicar recomendaciones gracias sus algoritmos lo que permite mejorar el tiempo de

búsqueda y aplicar lenguaje Cypher que facilita la interacción con la base.

- El sistema posee una arquitectura basada en contenedores empleando Docker, esto se realizó a través de la creación de imágenes con DockerFile y sus servicios. La tecnología de Docker nos permitió crear y desplegar contenedores que permitan la interacción de la base de datos NoSQL para el almacenamiento de los recursos educativos recuperados y un sistema recomendador. Cada contenedor contiene y realiza procesos específicos de acuerdo a la configuración y la aplicación que se emplea a cada uno de los contenedores. Los contenedores se desarrollaron para cada capa que conforma el sistema, esto facilita y ayuda al despliegue del repositorio y el sistema recomendador, el sistema se encuentra desarrollado con el framework de fastAPI que permite desarrollar los procesos de inferencia y recomendaciones y ofrecerlos mediante servicios REST.
- El repositorio de recursos académicos permite complementar las actividades del aula, para ello es necesario la extracción de información, esto se realizó mediante scripts o crawlers que permiten recuperar información para niños de 5 a 7 años, para esto se trabajó con el libro del ministerio de educación y con manuales desarrollados para personas con capacidades especiales. Seguidamente se realizó el análisis del contenido de las actividades, empleando Term frequency – Inverse document frequency (TF-IDF) y la similaridad del coseno se determinó las actividades con mayor similaridad entre sí y entre nuevas actividades. Para cada actividad se procedió a obtener recursos mediante un crawler. Los recursos que se obtienen son vídeos, cuentos, libros, juegos, aplicaciones y publicaciones en Twitter, estos recursos para poder ser almacenados cumplen con un umbral de similitud en base al análisis de su contenido y el perfil del usuario.
- En base a la información obtenida en el estudio se los sistemas de recomendación basado en contenido y perfiles de usuario, se desarrollo un sistema recomendador mediante la recuperación de actividades de libros y recursos disponibles en la web empleando cr-

walers, para generar recomendaciones, el sistema trabaja con Neo4j que almacena toda la información del repositorio y permite emplear búsquedas y algoritmos para generar recomendaciones. Los algoritmos empleados por Neo4j como el Degree Centrality que permite determinar la influencia de relaciones entrantes y salientes de un nodo y Page Rank que permite determinar aquellos nodos con mayor influencia dentro de la base de datos gracias a sus relaciones y la importancia de los nodos origen ayuda a generar inferencias y ofrecer recomendaciones de acuerdo a las relaciones y los resultados de cada algoritmo.

- La implementación de aplicación se realizó en el framework Angular, se eligió esta herramienta porque cuenta con una documentación extensa con ejemplos de componentes adaptables, reutilizables para la interfaz de usuario y librerías disponibles que son usadas por la aplicación, con el uso de la librería HttpClient permite el consumo de servicios con el protocolo HTTP que residen en un backend y posteriormente ser presentado en la interfaz de la aplicación. Para la implementación de la aplicación móvil usaremos el SDK Ionic basado en Angular, lo que se hace es a partir de la aplicación web clonar y generar una APK, esto permitiendo no repetir código en Ionic.
- En base a lo desarrollado se procedió a diseñar una encuesta que permita determinar el alcance y validar las recomendaciones y recursos obtenidos, se obtuvo resultado buenos donde la mayoría de los usuarios aprobaron las recomendaciones de actividades y recursos que se ofrecen, permitiendo el desarrollo y una mejor comprensión de la actividad. Del total de los usuarios encuestados el 80% está de acuerdo con las recomendaciones que se presentan.

Capítulo X

Recomendaciones

Para mejorar los recursos de personas con capacidades especiales se necesitan trabajar con especialistas con perfiles específicos que permitan generar y obtener recursos que puedan ayudar a otros usuarios del sistema con un perfil similar.

En las actividades se deben trabajar con textos y manuales educativos desarrollados para maximizar el aprendizaje y ayudar a la enseñanza de los niños. Las actividades deben tener un indicador de evaluación, área, bloque o competencia a la que estén enfocadas para una mayor precisión en la obtención de los recursos educativos.

Finalmente, el sistema recomendador se debe tener actividades con mayor texto que permitan obtener recursos y actividades con un umbral determinado de similitud. Los recursos deben tener los ítems necesarios para un análisis de contenido adecuado y poder vincular a las actividades de acuerdo a un grado de similitud.

REFERENCIAS

- Aggarwal, C. (2016). *Recommender Systems*. Springer International.
- Carrillo, G. and Ochoa, X. (2013a). Recomendación de objetos de aprendizaje basado en el perfil del usuario y la información de atención contextualizada. *Centro de Tecnologías de Información, Escuela Superior Politécnica del Litoral*.
- Carrillo, G. and Ochoa, X. (2013b). Recomendación de objetos de aprendizaje basado en el perfil del usuario y la información de atención contextualizada. *Escuela Superior Politécnica del Litoral*.
- Castro, A., Gózales, J., and Callejas, M. (2012). Utilidad y funcionamiento de las bases de datos nosql. *Universidad Pedagógica y Tecnológica de Colombia*.
- Cristina, B., Claudia, D., Juan, G., Maria, H., and Julio, P. (2014). *Bases de datos NoSQL. LATIn*.
- Cuello, J. and Vittone, J. (2013). *Diseñando apps para móviles*. José Vittone.
- DB-Engines (2022). Db-engines ranking - trend of graph dbms popularity. https://db-engines.com/en/ranking_trend/graph+dbms. Online; recuperado 08 de Agosto de 2022.
- de Educación del Ecuador, M. (2016). *Texto Preparatoria 1 de EGB Cuaderno de Trabajo*. Editorial Don Bosco.

- Docker (2018a). Docker overview. <https://docs.docker.com/get-started/overview/>. Online; recuperado 08 de agosto de 2022.
- Docker (2018b). Dockerfile reference. <https://docs.docker.com/engine/reference/builder/>. Online; recuperado 08 de agosto de 2022.
- Docker (2018c). Overview of docker compose. <https://docs.docker.com/compose/>. Online; recuperado 08 de agosto de 2022.
- Dorta, M. R. (2016). *AngularJs Paso a Paso*.
- Edward, S. G. and Sabharwal, N. (2015). *Practical MongoDB: Architecting, Developing, and Administering MongoDB*. Apress.
- Face, H. (s.f). Sentence similarity. <https://huggingface.co/tasks/sentence-similarity>. Online; recuperado 08 de agosto de 2022.
- González, C. and Luz, M. (2011). Recursos educativos tic de informaciòn, colaboraciòn y aprendizaje. *Pixel-Bit. Revista de Medios y Educaciòn*.
- Gómez, P., Guarda, T., Cedeño, J., Benavides, A., Alejandro, C., Mosquera, G., Garcia, T., and Benavides, V. (2018). Sistemas de recomendaciòn: un enfoque a las técnicas de filtrado. *Universidad Estatal Península de Santa Elena*.
- Iliev, B. (2021). *Introduction to Docker*. Ibis Ebook.
- Isinkaye, F., Folajimi, Y., and Ojokoh, B. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16:261–273.
- Jama, V., Cornejo, J., Navia, T., Parraga, M., Arteaga, K., and Alava, J. (2019). *Recursos educativos y tecnológicos en la educaciòn*. Casa Editora del Polo. Online; recuperado 08 de agosto de 2022.

Jiménez, J. Z. (2013). *Aplicaciones Web*. Macmillan Iberia, S.A.

Marquès, P. (2011). Los Medios didàcticos y los recursos educativos. <http://www.peremarques.net/medios.htm#:~:text=%2D%20Medio%20did%C3%A1ctico%20es%20cualquier%20material,hacer%20pr%C3%A1cticas%20de%20formulaci%C3%B3n%20qu%C3%ADmica>. Online; recuperado 08 de agosto de 2022.

MARYLAND, U. O. (s.f). International children's digital library. <http://es.childrenslibrary.org/library/lang18.html>. Online; recuperado 08 de agosto de 2022.

Mercedes, M. (2009). *Base de Datos*. D - Universitat Jaume I. Servei de Comunicació i Publicacions.

Monerris, L. C. (2015). Bases de datos orientadas a grafos aplicadas al estudio de informes radiológicos: utilizando entornos de computación en la nube para abordar estudios de gran dimensión. Master's thesis, Universidad Politécnica de Valencia.

Monte, J. (2016). *Implantar scrum con éxito*. Editorial UOC. Online; recuperado 08 de agosto de 2022.

Mortera, F. and Ramírez, M. (2013). Conexión de repositorios educativos digitales: Educonector.info. *Researchgates*.

Neo4j (2022). Guía de inicio. <https://neo4j.com/docs/getting-started/current/>. Online; recuperado 08 de Agosto de 2022.

Oriols, M. B. and Gutiérrez, J. A. G. (2018). *El gran libro de Angular*. Marcombo.

Palacio, J. and Ruata, C. (2009). *Scrum Manager: Proyectos – apuntes de formación*. Safe Creative.

- Park, D., Kim, H., Choi, I., and Kim, J. (2012). A literature review and classification of recommender systems research. *Expert Systems with Applications*.
- Quanam (s.f). Modelos de recomendación. <https://quanam.com/modelos-de-recomendacion-recomendando-que-recomendar/>. Online; recuperado 08 de agosto de 2022.
- Romero, P., Domínguez, E. E., Óscar Núñez Pérez, and Ángel, J. (2019). *Base de Datos*. Grupo Editorial Patria.
- Saens, J. and Serna, L. (2019). *Manuel de estrategias educativas inclusivas para niños con parálisis cerebral espástica*.
- Schwaber, K. and Sutherland, J. (2017). La guía de scrum. *Creative Commons*.
- Tariq, M. and Francesco, R. (2009). Improving recommender systems with adaptive conversational strategies. pages 73–82.
- Vargas, M. G. (2017). Recursos educativos didácticos en el proceso enseñanza aprendizaje. *Unidad de Educación Virtual Facultad de Medicina (U.M.S.A.)*.
- Velásquez, W. (2014). Bases de datos orientadas a grafos y su enfoque en el mundo real. *Escuela Técnica Superior de Ingenieros en Telecomunicaciones (ETSIT - UPM)*.
- Viedma, H., Porcel, C., and Hidalgo, L. (2004). Sistemas de recomendaciones: herramientas para el filtrado de información en internet. *Hipertext.net*.

Anexos

Anexos A

Formato de la encuesta

Generar un repositorio de materiales didácticos para la recomendación de recursos educativos utilizando bases de datos orientadas a grafos a niños de 5- 7 años con y sin discapacidad.

Con la presente encuesta se pretende validar el funcionamiento y la utilidad del sistema.

1.1 Información de la aplicación

1. ¿Cuál es su posición frente al uso de nuevas tecnologías como internet y smartphones para el uso en la educación?

- Muy importante
- Importante
- Moderadamente importante
- Poco importante
- Nada importante

2. ¿Estás de acuerdo en que un sistema ofrezca sugerencias?

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

3. ¿Estás de acuerdo con las recomendaciones de actividades a realizar?

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

4. ¿Estás de acuerdo con las recomendaciones de recursos para las actividades a realizar?

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo ni en desacuerdo
- En desacuerdo
- Totalmente en desacuerdo

5. ¿Cual es tu nivel de satisfacción con el uso de la aplicación.?

- Altamente satisfecho
- Muy satisfecho
- Satisfecho

- Poco satisfecho
- Completamente insatisfecho