

# **Prototipo web de un sistema de recomendación de videojuegos**

**David Alejandro Moreno Zapata**

**Universidad del Valle  
Facultad de Ingeniería  
Ingeniería de Sistemas  
Tuluá, Valle del Cauca**

**2023**

# **Prototipo de un sistema de recomendación de videojuegos**

**David Alejandro Moreno Zapata**

**Código 1765509**

[david.alejandro.moreno@correounalvalle.edu.co](mailto:david.alejandro.moreno@correounalvalle.edu.co)

Director

**Joshua David Triana Madrid, Msc**

Profesor de la Escuela de Ingeniería de Sistemas y Computación

[joshua.triana@correounalvalle.edu.co](mailto:joshua.triana@correounalvalle.edu.co)

**Universidad del Valle**

**Facultad de Ingeniería**

**Ingeniería de Sistemas**

**Tuluá**

**2023**

## **Agradecimientos**

A mis padres por ser mi pilar y apoyo incondicional.

A mis amigos Daniel, Isabel y Paola, por su compañía, amor y motivación a lo largo de este pregrado.

Al Magíster, Joshua Triana, por su excelente labor y su comprensión en mis momentos mas críticos.

# Tabla de Contenido

<b>1. Introducción</b>	<b>6</b>
<b>2. Planteamiento del Problema</b>	<b>7</b>
2.1. Descripción del problema . . . . .	7
2.2. Definición del problema . . . . .	8
<b>3. Objetivos</b>	<b>9</b>
3.1. Objetivo general . . . . .	9
3.2. Objetivos específicos . . . . .	9
<b>4. Marco Referencial</b>	<b>10</b>
4.1. Antecedentes . . . . .	10
4.1.1. Desarrollo de un recomendador de videojuegos . . . . .	10
4.1.2. Recomendación de videojuegos basado en análisis semántico y minería de opinión . . . . .	10
4.1.3. Online videogame recommendation system using content and collaborative filtering techniques . . . . .	10
4.1.4. Recommendation of videogames with fuzzy logic . . . . .	10
4.2. Marco teórico . . . . .	11
<b>5. Desarrollo de la aplicación</b>	<b>13</b>
5.1. Metodología de desarrollo . . . . .	13
5.2. Modelo de recomendación . . . . .	13
5.2.1. Dataset . . . . .	13
5.2.2. Técnica de recomendación . . . . .	14
5.2.3. Justificación del Modelo . . . . .	15
5.3. Tecnologías utilizadas . . . . .	20
5.4. Demostración del prototipo . . . . .	22
<b>6. Pruebas</b>	<b>27</b>
6.1. Pruebas de usabilidad . . . . .	27
6.1.1. Pregunta 1 . . . . .	28
6.1.2. Pregunta 2 . . . . .	29
6.1.3. Pregunta 3 . . . . .	29
6.1.4. Pregunta 4 . . . . .	30
6.1.5. Pregunta 5 . . . . .	31
6.1.6. Pregunta 6 . . . . .	32

6.1.7. Pregunta 7 . . . . .	33
6.1.8. Pregunta 8 . . . . .	34
6.1.9. Pregunta 9 . . . . .	35
6.2. Pruebas de software . . . . .	36
<b>7. Conclusiones y trabajos futuros</b>	<b>37</b>
7.1. Conclusiones del proyecto . . . . .	37
7.2. Desarrollos futuros . . . . .	39
<b>8. Referencias</b>	<b>40</b>
<b>9. Anexos</b>	<b>43</b>

# Índice de figuras

1.	Pruebas realizadas utilizando la librería SentEval.[1] . . . . .	15
2.	Pruebas realizadas utilizando la librería GLUE Benchmark.[2] . . . . .	15
3.	Resultados promedio de la calidad de los clusters en 60 pruebas por conjunto de datos, obtenidos con las métricas de semejanza y evaluados por su entropía.[3]	16
4.	Diagrama de flujo de la aplicación . . . . .	18
5.	Diagrama de flujo del modelo de recomendación . . . . .	19
6.	Arquitectura del prototipo . . . . .	21
7.	Registro de usuario . . . . .	22
8.	Inicio de sesión . . . . .	23
9.	Preguntas para determinar gustos del usuario (cold start) . . . . .	24
10.	Calificación por parte de los usuarios para crear su perfil de usuario (cold start)	25
11.	Pantalla de carga . . . . .	25
12.	Recomendaciones . . . . .	26
13.	Pregunta 1 . . . . .	28
14.	Pregunta 2 . . . . .	29
15.	Pregunta 3 . . . . .	30
16.	Pregunta 4 . . . . .	31
17.	Pregunta 5 . . . . .	32
18.	Pregunta 6 . . . . .	33
19.	Pregunta 7 . . . . .	34
20.	Prueba de carga primer endpoint . . . . .	36
21.	Prueba de carga segundo endpoint . . . . .	36

## **Índice de cuadros**

1.	Objetivos específicos y su ubicación en el documento. Fuente: elaboración propia.	9
2.	Comparación de fuentes de información. Fuente: elaboración propia . . . . .	14
3.	Calificaciones de usuarios. Fuente: elaboración propia. . . . . . . . . . .	16
4.	Aplicación de one hot encoding. Fuente: elaboración propia. . . . . . . . .	17
5.	Ponderación de géneros (Perfil de usuario). Fuente: elaboración propia. . . .	17

## 1. Introducción

El crecimiento de la industria de los videojuegos se ha incrementado considerablemente en los últimos años, nada más en Steam, plataforma autodefinida como “el mejor lugar para jugar, hablar sobre juegos y crearlos”, que, en 2006, según datos de SteamDB[4] lanzaba apenas 71 juegos, hasta llegar a la impresionante cifra de 10033 juegos en 2020, esto demuestra que el consumo de juegos es cada vez más grande, y que la industria es rentable. No obstante, el mercado está tan repleto de nuevos títulos que los jugadores muchas veces no saben que jugar o juegan aquellos títulos de estudios reconocidos. Esto a largo plazo sería poco rentable para estudios pequeños ya que sus juegos no serían ni siquiera contemplados por el público. Por ejemplo, en 2020, según datos de SteamDB[5], se lanzaron 6624 juegos indie, de los cuales el 50porciento vendieron menos de 640 unidades, según una investigación realizada por Thomas Altenburger, director del estudio independiente Flying Oak Games[6].

En ese orden de ideas, las personas necesitan recomendaciones, es un hecho que las recomendaciones son muy útiles para descubrir cosas desconocidas, y en nuestro contexto no es la excepción. Así, en aplicaciones como Facebook, donde se recomiendan amigos o Netflix, que recomienda películas, los jugadores o “gamers” necesitan recomendaciones de videojuegos. Buscar ese juego que va a generar entretenimiento para las próximas 20, 60 o 1000 horas, puede ser una tarea más difícil de lo que parece, entre tanta variedad y adicionalmente las personas muchas veces no saben exactamente lo que quieren, solo tienen una pista, una idea vaga. Esto genera una incógnita “¿Cuál será el próximo juego que me va a entretenecer?”, que, aunque pueda parecer una cuestión muy personal, es un tema bastante profundo e interesante.

## **2. Planteamiento del Problema**

### **2.1. Descripción del problema**

En la actualidad el ámbito de los jugadores ha explotado su popularidad exponencialmente, tanto que ya lo gamer dejó de ser un mero hobbie y se volvió una profesión. Este “boom” aumentó el consumo de videojuegos, pero con un atenuante, solo los juegos más populares o los que son publicitados por los gamers con más seguidores son jugados; o simplemente los jugadores ya han completado sus bibliotecas y no encuentran que jugar. Esto no supondría un problema para los gamers hardcore, sin embargo, esta es una pequeña porción del total. Solo en Colombia se tienen registro de 2.1 millones de gamers, de los cuales el 79 % juegan de manera casual, el 44 % juegan todos los días, y solo el 26 % buscan nuevos juegos[7]. Ahora ampliando esto a la región de Latinoamérica, se tiene que la población online, es decir, latinoamericanos que se encuentran conectados a internet, son aproximadamente 453 millones de personas, y la mitad se identifican como gamers; bien, de ese grupo de personas, solo el 11porciento son considerados gamers hardcore[8], [9]. Esto deja un amplio grupo de personas que juegan casual y posiblemente solo sean juegos de las grandes compañías, estas personas probablemente nunca descubran cosas nuevas.

Ahora bien, existen distintas páginas web o revistas como Eurogamer, Levelup o Metacritic, que tienen revisiones, calificaciones y críticas sobre videojuegos, pueden ayudar en esta problemática en cierta medida, ya que las personas pueden consultarlas, y tener un punto de referencia, sin embargo, salvo por Metacritic, todas las revistas cumplen con un patrón, y es que todos los comentarios y calificaciones de los usuarios convergen en las revisiones o reseñas creadas por los empleados de estas, y si bien, no es algo necesariamente malo, cuando se observan las respuestas que dan los usuarios, se encuentra que la mayoría son comentarios sobre la reseña y no sobre el juego reseñado. Al final, toda esta información es una sobrecargar a los usuarios, por ende, se hace precisa la necesidad de centralizar los datos, converger las distintas fuentes y volverlas metadatos y así obtener meta-recomendaciones, o bien se podría ver como recomendaciones de las recomendaciones.

Por otra parte, Steam, en los ultimos años, ha venido diseñando un sistema de recomendación, que según su web, con base a los videojuegos que se tengan en la biblioteca y sus patrones de juego, arroja recomendaciones personalizadas para cada usuario[10], sin embargo, este recomendador aún se encuentra en fase experimental y por este motivo las recomendaciones no son lo bastante precisas, por ejemplo, en el caso del autor, se incluyen recomendaciones de videojuegos de fútbol, con los que se tiene poca interacción previa, además, se nota la baja inclusión de videojuegos de conducción, cuando estos son los más jugados por el autor.

## **2.2. Definición del problema**

Más del 50 % de la población gamer de Latinoamérica, juega los mismos juegos siempre, sin la oportunidad de conocer nuevos títulos, con base a esto y lo visto anteriormente, se plantea la siguiente pregunta problematizadora:

*¿Cómo implementar un sistema de recomendación de videojuegos basado en distintas fuentes de información, los gustos de los jugadores y que complemente al recomendador interactivo de Steam?*

### **3. Objetivos**

#### **3.1. Objetivo general**

Crear un prototipo funcional de un sistema de recomendación para videojuegos extrayendo información de diferentes fuentes.

#### **3.2. Objetivos específicos**

<b>Objetivo específico</b>	<b>Sección del documento</b>
Recolección y caracterización de información sobre los videojuegos del género escogido	5.2.2
Recolección de información de los usuarios para determinar su personalidad (gustos del usuario).	5.2.3
Selección de una técnica de recomendación para la solución del problema	5.2.3
Desarrollo del prototipo del sistema de recomendación.	5.3
Análisis de resultados obtenidos.	6

Tabla 1: Objetivos específicos y su ubicación en el documento. Fuente: elaboración propia.

## **4. Marco Referencial**

### **4.1. Antecedentes**

#### **4.1.1. Desarrollo de un recomendador de videojuegos**

Se trata de un prototipo de sistema de recomendación de videojuegos basado en filtro colaborativo, que utiliza la red social Twitter como su proveedor de información, este sistema extrae los tweets que tengan que ver con X juego, previamente escogido por el usuario, hace un análisis de sentimiento de los tweets en cuestión y retorna una calificación. También tiene un componente para empresas, el cual hace el mismo análisis de sentimiento de los tweets sobre un juego X, pero retorna una lista de partidarios y opositores, y tener mejores datos para un estudio de mercado.[11]

#### **4.1.2. Recomendación de videojuegos basado en análisis semántico y minería de opinión**

Es un prototipo que recopila técnicas de extracción e interpretación semántica de datos, en un dominio informal, este sistema extrae las opiniones de usuarios de la plataforma Steam, los filtra para obtener los que aporten valor semántico y con una descripción proporcionada por el usuario, es capaz de devolver un juego o un grupo de juegos.[12]

#### **4.1.3. Online videogame recommendation system using content and collaborative filtering techniques**

Se introduce un método novedoso para recomendar juegos en línea utilizando el filtrado colaborativo. Con el objetivo de mejorar las recomendaciones, se ha desarrollado un enfoque innovador para estimar las puntuaciones implícitas, considerando las horas de juego y generando recomendaciones más precisas. El dispositivo de recomendación propuesto mejora los resultados de las diversas estrategias existentes en el ámbito de la técnica de recomendación en este campo.[13]

#### **4.1.4. Recommendation of videogames with fuzzy logic**

Esta investigación propone una arquitectura de recomendación de videojuegos y un sistema basado en Lógica Difusa. Se utilizaron 16 reglas y conjuntos difusos para desarrollar el sistema, utilizando datos de una amplia base de datos de juegos. Los resultados obtenidos muestran una alta precisión en las recomendaciones generadas por la arquitectura propuesta.[14]

## 4.2. Marco teórico

Este capítulo aborda los conceptos necesarios para comprender el desarrollo de un sistema de recomendación híbrido o meta-recomendador.

- **API:** Según un artículo de IBM, en el nivel más básico, una API es un mecanismo que permite a una aplicación o servicio acceder a un recurso dentro de otra aplicación o servicio. La aplicación o servicio que realiza el acceso se denomina cliente y la aplicación o servicio que contiene el recurso se denomina servidor[15].
- **REST:** Según un artículo de Mdn web, el término "Transferencia de Estado Representacional"(REST) representa un conjunto de características de diseño de arquitecturas software que aportan confiabilidad, eficiencia y escalabilidad a los sistemas distribuidos. Un sistema es llamado RESTful cuando se ajusta a estas características[16].
- **API REST:** Según un artículo de IBM, una API REST es una API que se ajusta a los principios de diseño de REST, un estilo de arquitectura también denominado transferencia de estado representacional. Por este motivo, las API REST son a veces denominadas API RESTful[15].
- **Sistemas de Recomendación:** Según González, un sistema de recomendaciones es mucho más que un algoritmo o un filtro que selecciona productos con más o menos acierto. Podemos dividir un recomendador en 4 partes: la base de conocimiento (la información, los datos), el procesamiento de la base de conocimientos (tecnología, algoritmos, filtros), la analítica y control de negocio (medir todo, estrategia de negocio) y finalmente la interface del usuario[17].
- **Support vector machine:** Es uno de los algoritmos de machine learning supervisado usado para la clasificación de datos. Este se entrena con un conjunto de datos estructurado para que el algoritmo después de ejecutar sus operaciones y alcanzar un nivel de aprendizaje adecuado, pueda predecir o definir nuevos ejemplos con el conjunto de datos de prueba. [18]
- **Filtrado basado en contenido:** Según Kanetkar, es una técnica de machine learning utilizada en los sistemas de recomendación y que utiliza los datos o atributos específicos de la información que se le suministre para hacer el filtro y clasificación de las recomendaciones. Puede realizar las recomendaciones basándose en la información de los productos o en la información de los usuarios. Esta técnica aprende de las preferencias e intereses de los usuarios para realizar las recomendaciones, por esto requiere de métodos de aprendizaje como support vector machine (SVM) y redes neuronales. [19]
- **Filtrado colaborativo:** Según Xin, otra técnica popular de machine learning y los

sistemas de recomendaciones que se enfoca más en la calidad de los elementos para hacer las recomendaciones. Es útil cuando no se cuentan con mucha información sobre los usuarios o productos, pero requiere de la retroalimentación de los usuarios respecto a los productos que se manejan en el sistema de recomendación, como calificaciones o comentarios sobre ellos. Utiliza matriz de factorización entre elementos o usuarios para hacer las predicciones sobre los intereses de los usuarios. [20]

- **Similitud de Coseno:** Es una función trigonométrica utilizada para medir el ángulo entre dos vectores en el espacio vectorial. El ángulo entre dos vectores indica que tan cercano está uno del otro en términos de dimensionalidad. El resultado del cálculo de coseno entre dos vectores varía entre cero y uno, donde cero indica la mayor diferencia entre los vectores y uno indica que no hay diferencia entre ellos en el espacio dimensional. [3]

## **5. Desarrollo de la aplicación**

### **5.1. Metodología de desarrollo**

El desarrollo del prototipo se implementó bajo una metodología ágil muy reconocida, Scrum, esta es ideal para mantener una supervisión constante del estado y avances del proyecto. De esta metodología se planearon sprints semanales (de plazo corto), donde se definían tareas que no se modificaban a lo largo del sprint, las reuniones diarias fueron descartadas por temas de disponibilidad con el director de este trabajo y la retrospectiva del sprint se realizaba justo antes de la planificación de un nuevo sprint.

### **5.2. Modelo de recomendación**

#### **5.2.1. Dataset**

En aras de tener información siempre actualizada, se decide por la utilización de APIS enfocadas a los videojuegos, como fuentes de información, estas tienen una base de datos muy amplias. Se obtienen cuatro fuentes de información que son, SteamSpy, GameSpot, IGDB y RAWGames, de los cuales se seleccionan dos, sobre estas se desarrolló el prototipo.

Api	Resultado
GameSpot	Sus endpoints retornan información en formato XML, documentación confusa y desactualizada. <b>Descartada.</b>
IGDB	Cuenta con numerosos endpoints, la información se puede filtrar de distintas formas y cuenta con un lenguaje propio llamado APICalypse, usado para tener consultas más precisas. Aunque la obtención de su Api key llega a ser complicada. <b>Seleccionada.</b>
SteamSpy	Es de uso libre no-comercial, no se necesita Api Key, la información está en formato JSON, sin embargo, no es muy robusta en cuanto a filtrado de información. <b>Descartada.</b>
RAWG	Tiene una versión gratuita de uso no comercial, se necesita Api key, pero esta es muy sencilla de obtener, cuenta con más de 500.000 juegos (incluidos juegos que no se encuentran en Steam), y para hacer contraste, su atributo de valoración es tomado de la revista Metacritic. <b>Seleccionada.</b>

Tabla 2: Comparación de fuentes de información. Fuente: elaboración propia

Siguiendo la linea del alcance de este trabajo, se seleccionaron dos fuentes de información, IGDB y RAWG, estas serán mencionadas como Fuente 1 y Fuente 2, respectivamente.

### 5.2.2. Técnica de recomendación

Una de las limitaciones de las fuentes de información seleccionadas es la falta de calificaciones de los usuarios, lo que dificulta el uso de técnicas como Clustering o vecinos cercanos, para lograr un filtrado colaborativo. Por esta razón, se optó por utilizar un enfoque de filtrado basado en contenido para ambas fuentes de información.

En el caso de la Fuente 1, se aprovechó el atributo de resumen de la trama para recomendar juegos con tramas similares. Para lograr esto, se realizó la transformación de lenguaje natural a vectores para calcular la similitud entre los juegos. Para llevar a cabo esta transformación, se recomiendan algoritmos reconocidos como BERT o regresión logística. Para calcular la similitud entre vectores, se recomiendan técnicas como la similitud de coseno, distancias euclidianas o kernel por pares (pairwise kernel).

Este enfoque basado en contenido permitió superar la limitación de la falta de calificaciones

de los usuarios y ofrecer recomendaciones precisas y relevantes a los usuarios del sistema.

### 5.2.3. Justificación del Modelo

La elección del algoritmo que trasforma texto a vector, se hizo con base a estudios realizados en [2], [21], [1], donde hacen una comparativa de la precisión de modelos como BERT, multinomial NB, OpenAIGPT, etc. En estos BERT resulta como la mejor opción para estas transformaciones, ya que tiene una alta precisión, resolviendo tareas de GLUE Benchmark[22] o SentEval[23].

<b>Model</b>	<b>MR</b>	<b>CR</b>	<b>SUBJ</b>	<b>MPQA</b>	<b>SST</b>	<b>TREC</b>	<b>MRPC</b>	<b>Avg.</b>
Avg. GloVe embeddings	77.25	78.30	91.17	87.85	80.18	83.0	72.87	81.52
Avg. fast-text embeddings	77.96	79.23	91.68	87.81	82.15	83.6	74.49	82.42
Avg. BERT embeddings	78.66	86.25	94.37	88.66	84.40	92.8	69.45	84.94
BERT CLS-vector	78.68	84.85	94.21	88.23	84.13	91.4	71.13	84.66
InferSent - GloVe	81.57	86.54	92.50	<b>90.38</b>	84.18	88.2	75.77	85.59
Universal Sentence Encoder	80.09	85.19	93.98	86.70	86.38	<b>93.2</b>	70.14	85.10
<b>SBERT-NLI-base</b>	83.64	89.43	94.39	89.86	88.96	89.6	<b>76.00</b>	87.41
<b>SBERT-NLI-large</b>	<b>84.88</b>	<b>90.07</b>	<b>94.52</b>	90.33	<b>90.66</b>	87.4	75.94	<b>87.69</b>

Figura 1: Pruebas realizadas utilizando la librería SentEval.[1]

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	Average
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
BERT <sub>BASE</sub>	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
BERT <sub>LARGE</sub>	<b>86.7/85.9</b>	<b>72.1</b>	<b>92.7</b>	<b>94.9</b>	<b>60.5</b>	<b>86.5</b>	<b>89.3</b>	<b>70.1</b>	<b>82.1</b>

Figura 2: Pruebas realizadas utilizando la librería GLUE Benchmark.[2]

Para el algoritmo que calcula la similitud de dos vectores, se utilizó la Similitud de Coseno, basado en las conclusiones dadas en [3], donde tras realizar distintas pruebas de semejanza y evaluándolos por su entropía, llegan a la conclusión que la Similitud del Coseno es la técnica que obtiene los mejores resultados.

BD \ Métrica	Jaccard	Coseno	Euclíadiana
ArtC	0.497497	<b>0.402200</b>	0.947375
SW	0.490805	<b>0.419818</b>	1.030132
7 Sectors	<b>1.801457</b>	1.805953	1.807561
webKB	1.403985	<b>1.386061</b>	1.474647
Co-training	0.522883	<b>0.512858</b>	0.522899

Figura 3: Resultados promedio de la calidad de los clusters en 60 pruebas por conjunto de datos, obtenidos con las métricas de semejanza y evaluados por su entropía.[3]

En el caso de la Fuente 2, la elección de un enfoque basado en contenido para este sistema de recomendación se basa en su capacidad para ofrecer recomendaciones precisas y personalizadas, incluso en sistemas con pocos usuarios o con información limitada sobre ellos. Además, este enfoque también permite una mayor flexibilidad en cuanto a la información utilizada para generar recomendaciones, lo que lo convierte en una opción adecuada para el objetivo principal de este trabajo de grado. En este, el usuario califica una serie de juegos presentados aleatoriamente por el prototipo, con base en estos, se construye un perfil de usuario, ponderando los géneros de los juegos calificados.

Esto se logró mediante un producto punto entre el género y la calificación del usuario, para esto, se generó una tabla de géneros mediante la codificación one hot encoding[24], en la que se obtienen todos los géneros del dataset y se asigna un 1 al género que tenga el juego y un 0, en el caso contrario.

Juego	Género	Calificación
Splatoon 3	Shooter, Action	1
Resident Evil 2	Shooter, Adventure, Action	1
Ori and the Will of the Wisps	Platformer, Adventure, Action	5
Hades	Indie, Adventure, Action, RPG	5
Sayonara Wild Hearts	Casual, Indie, Action	4

Tabla 3: Calificaciones de usuarios. Fuente: elaboración propia.

Action	Adventure	RPG	Platformer	Shooter	Indie	Casual	Racing
1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0
1.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0
1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

Tabla 4: Aplicación de one hot encoding. Fuente: elaboración propia.

Género	Ponderación
Action	16
Advcenture	14
RPG	5
Platformer	4
Shooter	6
Indie	6
Casual	1
Racing	0
Simulation	0
Sports	0
Fighting	0

Tabla 5: Ponderación de géneros (Perfil de usuario). Fuente: elaboración propia.

Con el perfil de usuario creado, se recomendaron los juegos que más se ajustaron a dicho perfil.

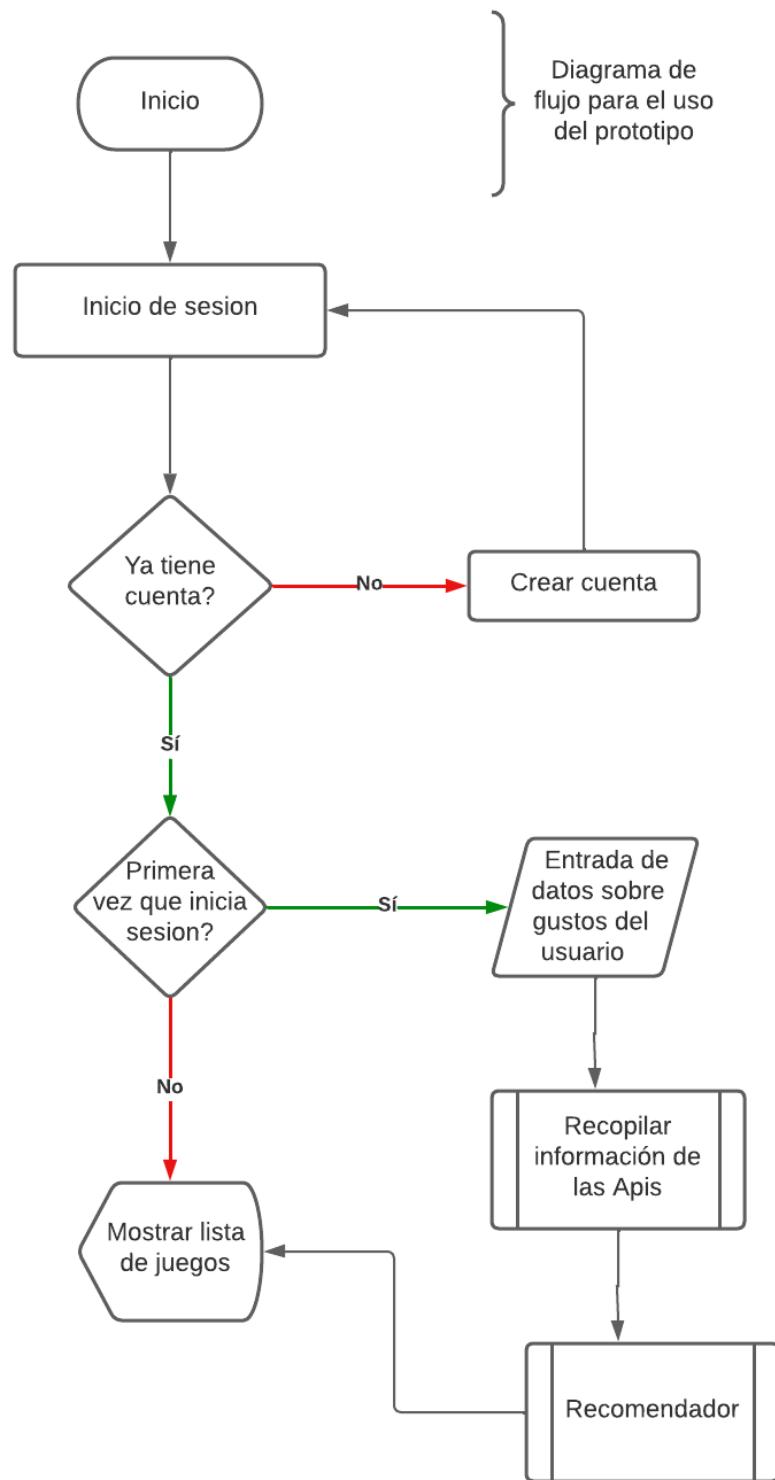


Figura 4: Diagrama de flujo de la aplicación

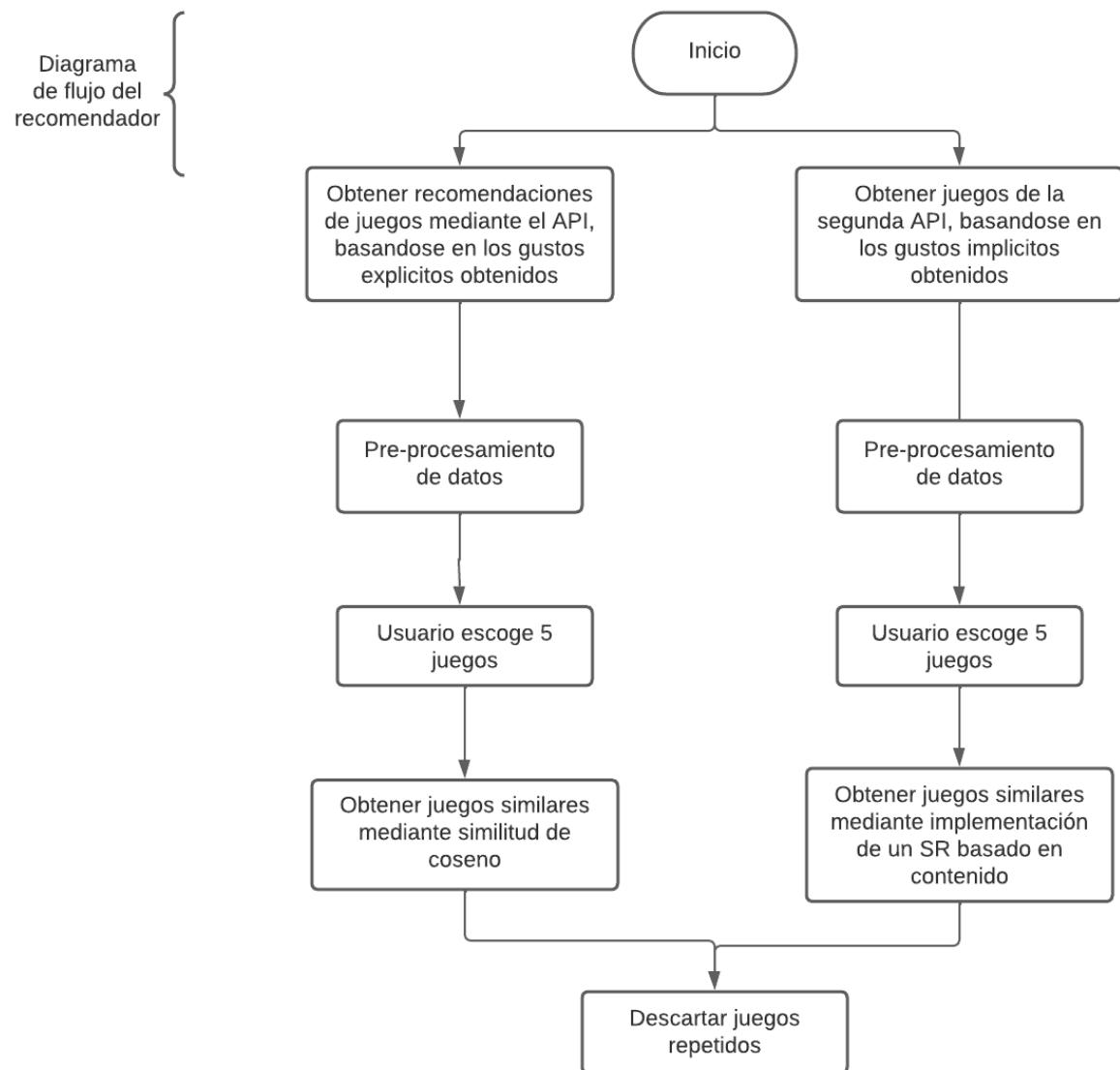


Figura 5: Diagrama de flujo del modelo de recomendación

### **5.3. Tecnologías utilizadas**

Para este trabajo se utilizaron distintos frameworks y tecnologías:

- **Python:** En su versión 3.10, es uno de los lenguajes más populares y versátiles de la actualidad, ideal para crear modelos de inteligencia artificial.
- **Django:** Framework utilizado para crear aplicaciones web robustas y escalables. De fácil uso y con una comunidad muy extensa.
- **Django rest framework:** Framework que añade funcionalidades a Django para la creación de web services o APIs Web.
- **PostgreSQL:** Gestor de base de datos gratuito y potente, que se integra fácilmente con Django.
- **React:** Es una biblioteca de JavaScript que se utiliza para construir interfaces de usuario altamente interactivas y dinámicas para aplicaciones web.
- **Railway:** Un servicio de host basado en la nube, que entre otras cosas, permite despliegues automáticos desde Git.

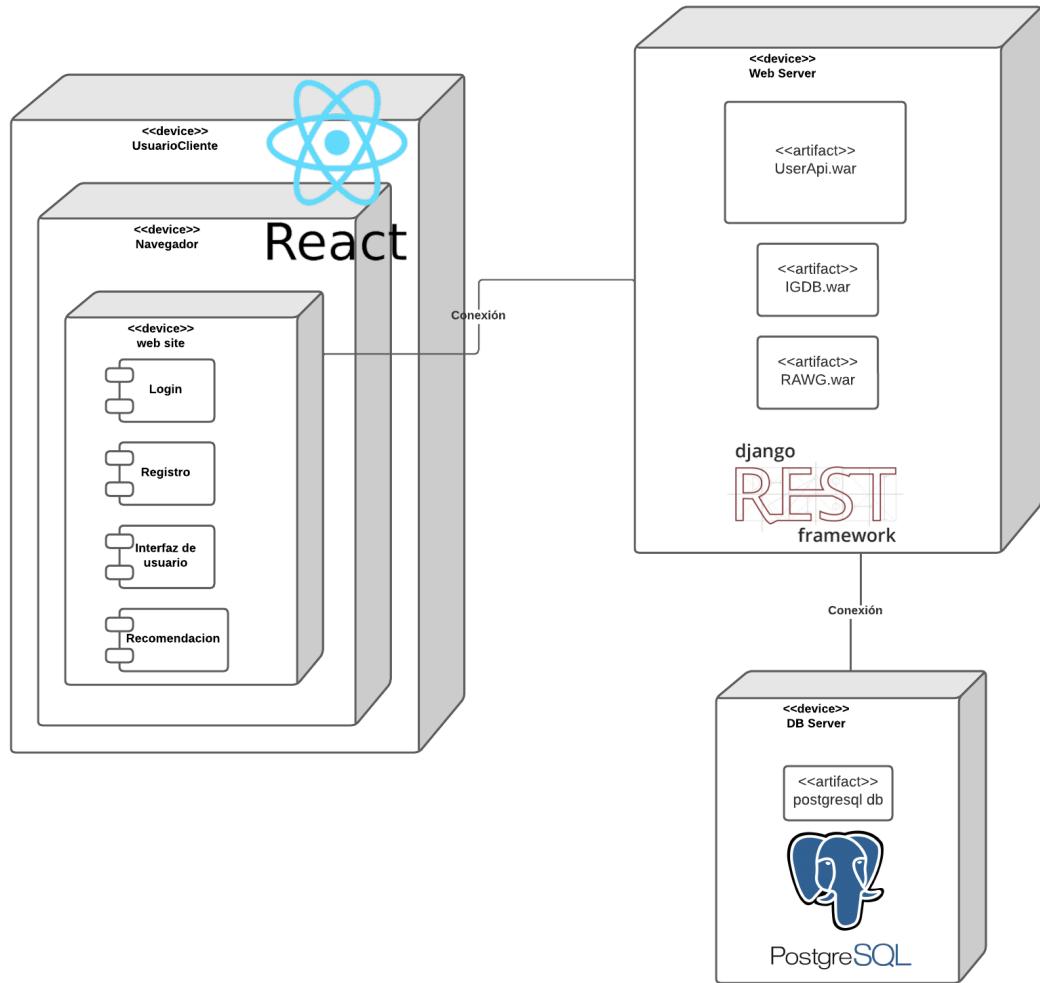


Figura 6: Arquitectura del prototipo

## 5.4. Demostración del prototipo

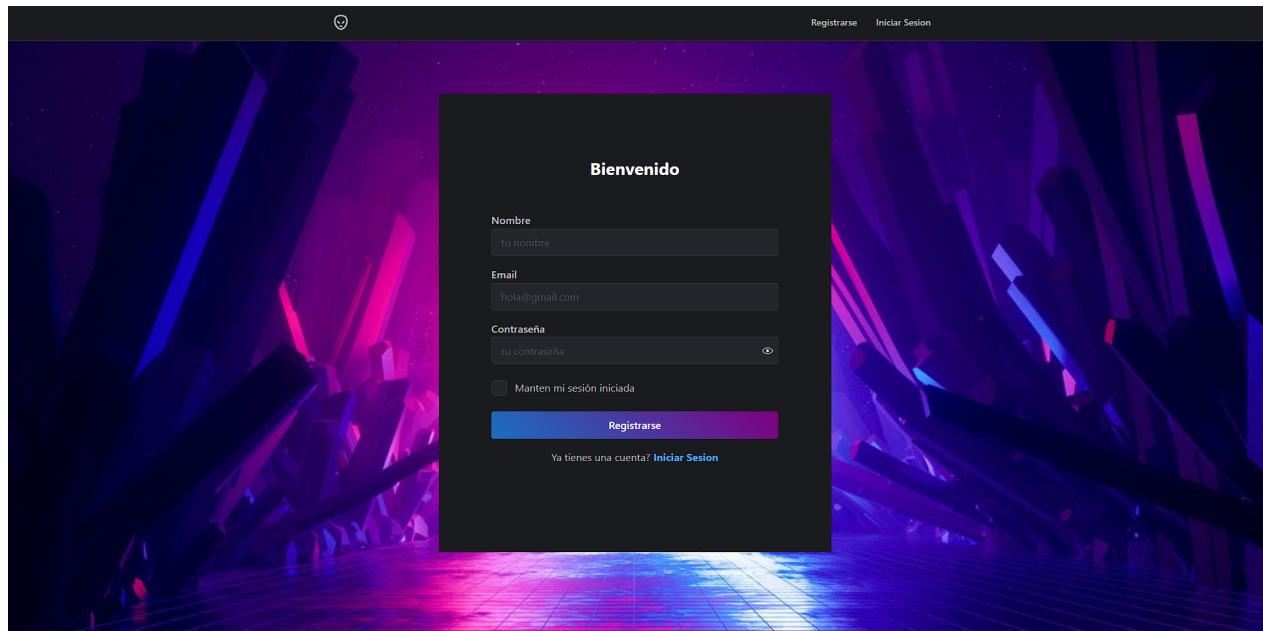


Figura 7: Registro de usuario

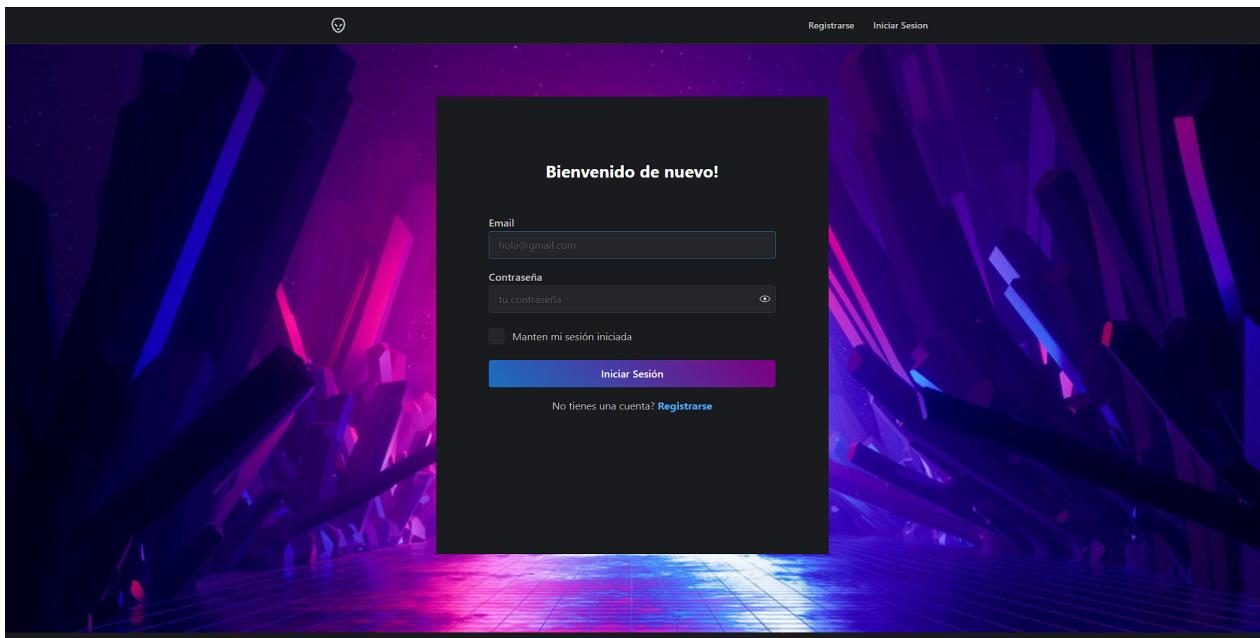
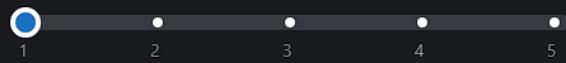


Figura 8: Inicio de sesión

## Test de perfil de videojuegos

Esta encuesta de gustos en videojuegos tiene como objetivo recolectar información sobre las preferencias y hábitos de juego de el encuestado. Los resultados de la encuesta se utilizarán para entender mejor las tendencias y patrones en los gustos de el jugador. Califica de 1 a 5 según el orden de cada pregunta.

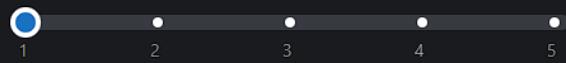
Eres más de juegos actuales (> 2020), no tan viejos (2019, 2015), viejos pero sustanciosos (2014, 2010), el prime de los videojuegos (2009, 2000) o clásicos (<2000)?



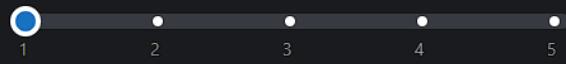
Te gustan juegos en primera persona, tercera persona, de vista isométrica, vista de lado, realidad virtual?



Te gustan juegos cooperativos, de un solo jugador, multijugador, multijugador masivo (MMO) O multijugador de arena de batalla (MOBA)?



Te gustan los juegos Indie?



Enviar

Figura 9: Preguntas para determinar gustos del usuario (cold start)

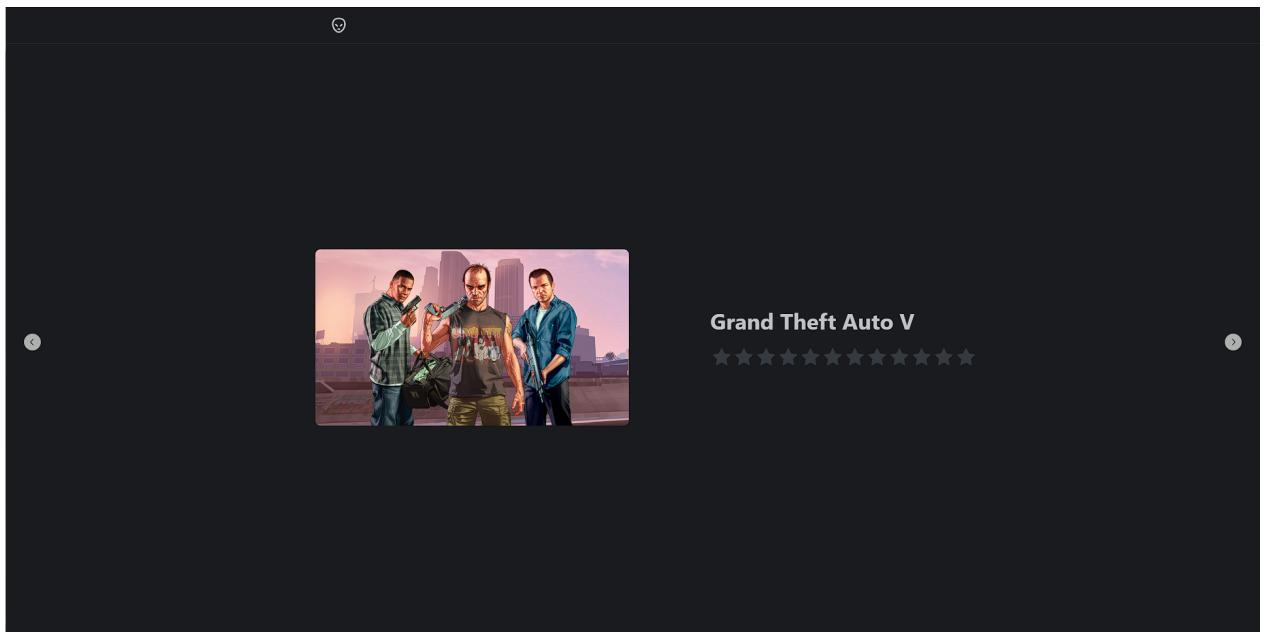


Figura 10: Calificación por parte de los usuarios para crear su perfil de usuario (cold start)



Figura 11: Pantalla de carga

# ¡Tus recomendaciones, querido Gamer!



Borderlands 2



God of War (2018)



Destiny 2



Horizon Zero Dawn



Cyberpunk 2077

Figura 12: Recomendaciones

## **6. Pruebas**

En esta sección se muestra el proceso de pruebas del prototipo y un análisis de los resultados obtenidos.

### **6.1. Pruebas de usabilidad**

Para medir la usabilidad del prototipo, este fue presentado a un grupo de usuarios para que pudieran explorar sus funciones. Para obtener una retroalimentación cuantificable, se optó por presentar una encuesta con nueve preguntas que se presentarán más adelante. Al final se obtuvieron 55 respuestas a la encuesta.

Las preguntas se basaron en la escala de Likert, es un tipo de instrumento de medición o de recolección de datos que se dispone en la investigación social para medir actitudes. Consiste en un conjunto de ítems bajo la forma de afirmaciones o juicios ante los cuales se solicita la reacción (favorable o desfavorable, positiva o negativa) de los individuos[25] y dos preguntas al final de tipo abiertas, para obtener la opinión de los encuestados.

### 6.1.1. Pregunta 1

Buscó medir la usabilidad del prototipo en cuanto a su facilidad de uso y lo intuitivo que pudo ser. Dando como resultado que mas de la mitad de los encuestados lo encontraron fácil de usar.

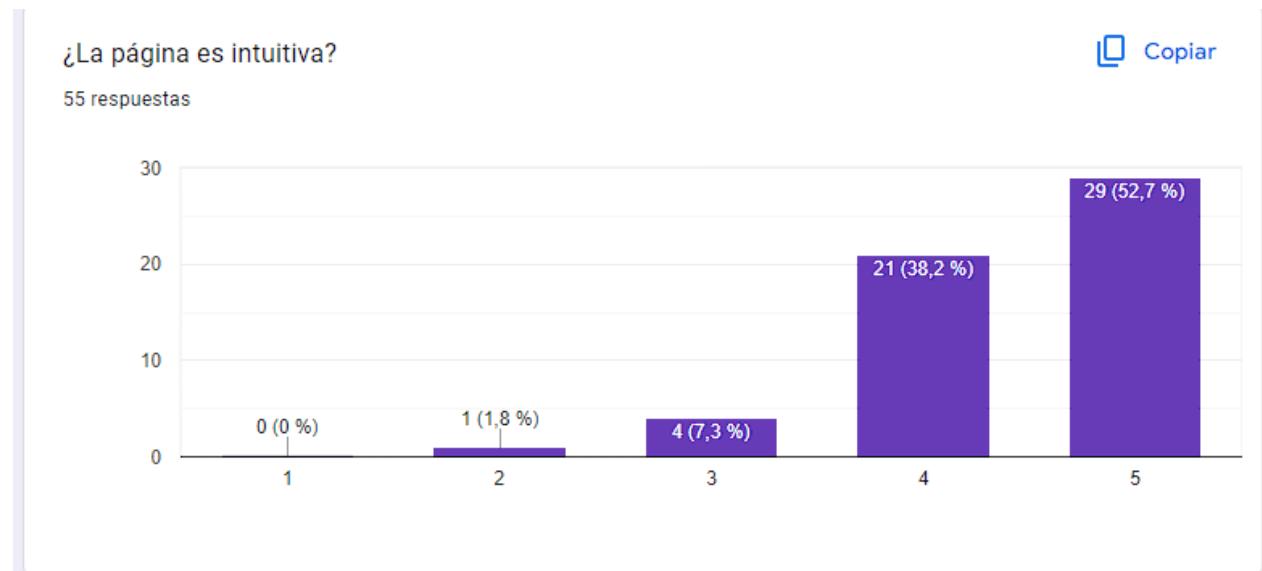


Figura 13: Pregunta 1

### 6.1.2. Pregunta 2

Buscó medir la satisfacción del usuario con las recomendaciones dadas. Dando como resultado, nuevamente, que mas de la mitad de los encuestados se encontraron de acuerdo con sus recomendaciones.



Figura 14: Pregunta 2

### 6.1.3. Pregunta 3

Esta pregunta buscaba medir el grado de familiaridad de los encuestados con las recomendaciones dadas, dado que conocer o no un juego, puede influir en el sentimiento de aceptación de la recomendación. Los resultados fueron muy divididos, sin embargo, el 32,7% indicó que conocía la mayoría de juegos, dando un indicio que el prototipo puede servir para jugadores entusiastas.

¿Conocías los juegos que te fueron recomendados?

55 respuestas

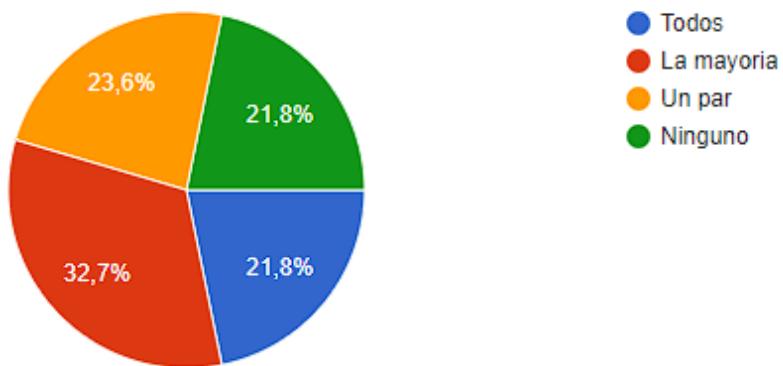


Figura 15: Pregunta 3

#### 6.1.4. Pregunta 4

Esta pregunta es un complemento de la anterior, ahora no es solo si el usuario conoce el juego, sino, si lo ha jugado, aquí hubo un empate entre las opciones "Un par" y "Ninguno" con 41,8 %, esto da un indicio de que las recomendaciones tienen valor, ya que son juegos que el usuario aun no ha probado.

---

**¿Has jugado los juegos que te fueron recomendados?**

55 respuestas

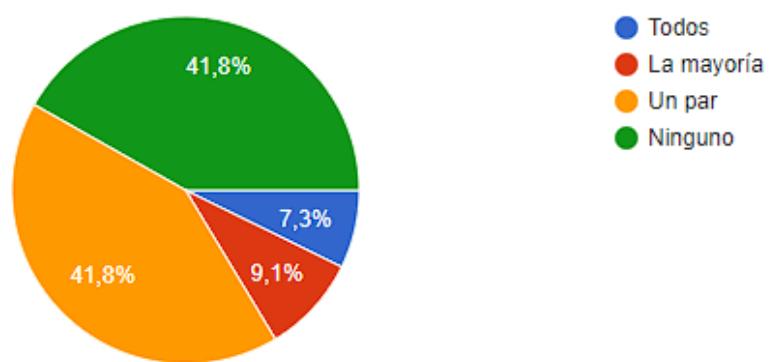


Figura 16: Pregunta 4

#### 6.1.5. Pregunta 5

Buscó medir el grado de satisfacción con el funcionamiento del prototipo de manera general, desde el registro de usuario hasta la entrega de recomendaciones, dejó como resultado que mas de la mitad de encuestados tuvieron una buena experiencia usando el prototipo.

¿Cómo calificarías tu experiencia usando la pagina?

 Copiar

55 respuestas

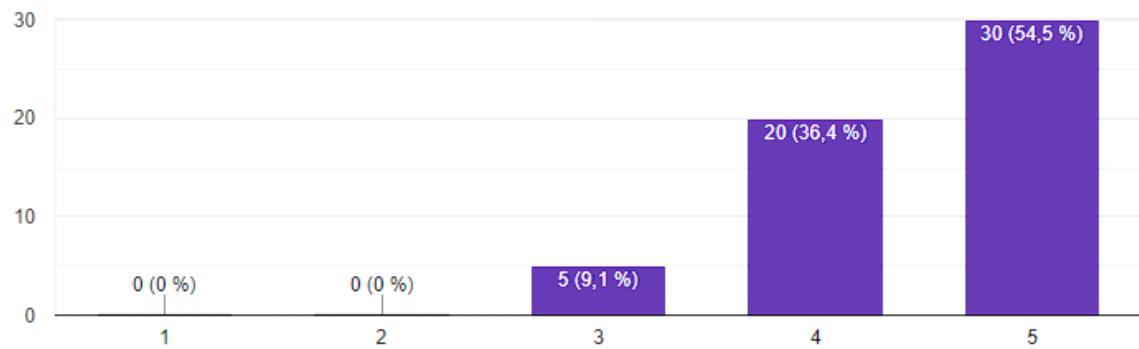


Figura 17: Pregunta 5

#### 6.1.6. Pregunta 6

Buscó medir el grado de interés de los encuestados por los juegos recomendados, dando como resultado que el 80% probaría los juegos recomendados, un indicador que demuestra que las recomendaciones son del interés de los usuarios.

¿Consideraría probar los juegos recomendados?

55 respuestas

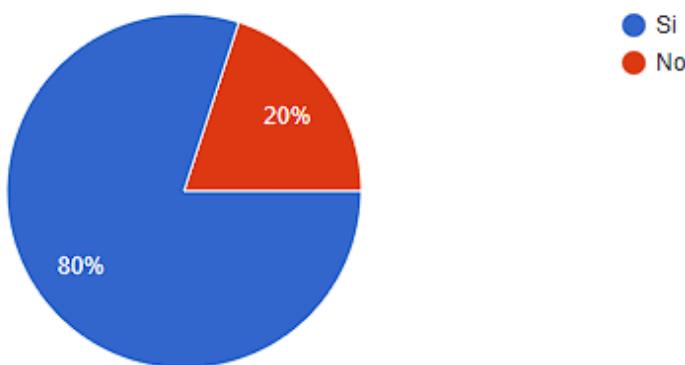


Figura 18: Pregunta 6

#### 6.1.7. Pregunta 7

Buscó medir el grado de conocimiento por otras alternativas, en este caso el recomendador de Steam, ya que esta plataforma es la que tiene la mayoría de usuarios a nivel mundial. Dio como resultado que el 69,1 % no conoce esta herramienta, esto se puede deber a que Steam no le da la suficiente visibilidad.

¿Has utilizado el recomendador interactivo de Steam?

55 respuestas

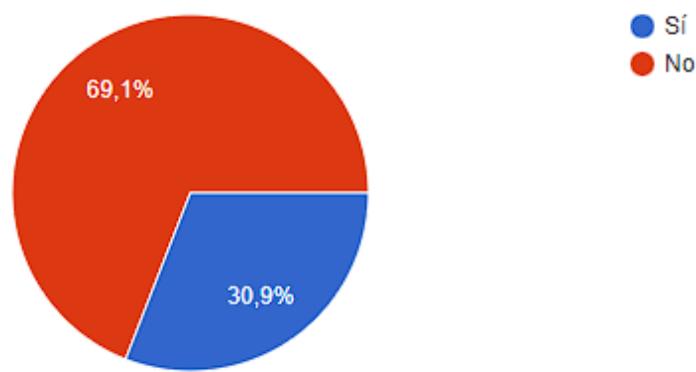


Figura 19: Pregunta 7

#### 6.1.8. Pregunta 8

Esta pregunta es de tipo abierta y complementa a la anterior, aquí se pregunta si en caso de conocer el recomendador de Steam, consideraría a este prototipo como un complemento o alternativa adecuada. Algunas respuestas fueron:

- Siento que dan recomendaciones similares.
- Siento que si, porque el de steam a veces da recomendaciones "fuera de contexto".
- Siento que el prototipo puede mejorar.
- Siento que puede ser un buen complemento al tener mas fuentes de información.

Se puede evidenciar que los usuarios ven a este prototipo como una buena alternativa a Steam.

### **6.1.9. Pregunta 9**

Esta pregunta también fue de tipo abierta, y buscó recibir comentarios u opiniones de los usuarios sobre su experiencia al usar el prototipo, algunas de las respuestas fueron:

- Me gusta el estilo de la pagina y las recomendaciones fueron adecuadas.
- Aunque es fácil de utilizar, no es muy intuitiva.
- Recomendó juegos que tenia en mente para probar.
- Me gusta que recomiende juegos indie.
- Sugiero mejorar las preguntas del inicio.
- Me parece que da recomendaciones muy acertadas.

Según las pruebas de usuario realizadas en la aplicación de recomendación de videojuegos, donde más del 50 % de los participantes respondió que la aplicación era bastante usable, amigable e intuitiva, y que las recomendaciones les parecieron acertadas, podemos obtener varias conclusiones positivas:

- La aplicación da una buena experiencia de usuario: El hecho de que más del 50 % de los usuarios tuvieran una buena experiencia de uso, indica que está diseñada de manera efectiva y que cumple con las expectativas de los usuarios en términos de funcionalidad y facilidad de uso.
- La aplicación es amigable e intuitiva: La opinión positiva sobre la amabilidad e intuición de la aplicación sugiere que los usuarios pueden navegar y utilizarla sin dificultades significativas. Esto indica que la interfaz de usuario y la estructura de la aplicación son intuitivas y se ajustan a las expectativas de los usuarios.
- Las recomendaciones son acertadas: El hecho de que los usuarios consideren que las recomendaciones de la aplicación son acertadas es un indicativo de que el algoritmo de recomendación está funcionando correctamente. Esto implica que la aplicación está ofreciendo sugerencias de videojuegos relevantes y que se adaptan a los gustos y preferencias de los usuarios.

## 6.2. Pruebas de software

Durante el desarrollo de la API, se buscaba garantizar la calidad y el rendimiento del software antes de ponerlo en producción. Para lograr esto, se decidió realizar pruebas de carga para evaluar el comportamiento de la API bajo diferentes niveles de tráfico. Para estas pruebas de carga, se seleccionó Locust, un marco de pruebas de carga de código abierto basado en Python[26]. Locust fue elegido debido a su facilidad de uso, escalabilidad y la capacidad de escribir scripts de prueba en Python. Con Locust, se pudo simular fácilmente un gran número de usuarios concurrentes que realizaron solicitudes a dos endpoints de la API, que son los encargados de retornar las recomendaciones, y medir su rendimiento y capacidad de respuesta bajo diferentes condiciones de carga. Los resultados de las pruebas de carga ayudaron a identificar y solucionar problemas de rendimiento y escalabilidad antes de que la API fuera puesta en producción, lo que garantizó una mayor calidad del software y una mejor experiencia para el usuario final.



Figura 20: Prueba de carga primer endpoint



Figura 21: Prueba de carga segundo endpoint

Para estas pruebas se simularon 20 usuarios concurrentes, que hacían peticiones constantemente a los endpoints de la API. Como se evidencia en las figuras 19 y 20, ninguna petición falló y sus tiempos nunca llegaron a los 2 segundos de espera, esto da indicios positivos sobre la posible escalabilidad de este prototipo.

## **7. Conclusiones y trabajos futuros**

### **7.1. Conclusiones del proyecto**

El enfoque basado en contenido se ha demostrado como una opción adecuada para el prototípico. Este enfoque utiliza características específicas de los juegos, como el género y la temática para encontrar juegos similares y ofrecer recomendaciones personalizadas y precisas.

En comparación con otros enfoques, como los basados en la popularidad o en la colaboración entre usuarios, el enfoque basado en contenido no depende de la información de otros usuarios para generar recomendaciones. Esto lo hace especialmente útil en sistemas con pocos usuarios o con información limitada sobre ellos. Además, permite una mayor flexibilidad en cuanto a la información utilizada para generar recomendaciones.

El prototipo se evaluó con éxito a través de pruebas de usuario, lo que demuestra su efectividad en proporcionar recomendaciones personalizadas y precisas de videojuegos. En general, el enfoque basado en contenido fue la elección correcta para la realización del prototípico de sistema de recomendación de videojuegos, ofreciendo una solución efectiva y adaptada a las necesidades y preferencias de los usuarios.

En cuanto a cuestiones mas técnicas, se puede destacar lo siguiente:

- Centralizar la información en el contexto de los videojuegos, es una buena opción para agilizar el acceso a esta, y evitar múltiples búsquedas en las distintas fuentes de información para encontrar un nuevo juego para disfrutar.
- Durante el proceso de desarrollo, uno de los desafíos que se presentó fue el arranque en frío, que se debió a la falta de datos sobre los gustos de los usuarios. Este arranque en frío pudo abordarse efectivamente mediante el uso de diferentes técnicas convencionales, como los rankings o calificaciones del usuario y las filtraciones por géneros, modos de juego, etc. Al hacerlo, se pudo evitar que el filtrado fracase en sus inicios.
- Las pruebas satisfactorias indican que la técnica de recomendación elegida cumplió con los requisitos y objetivos establecidos para la solución del problema planteado. Esto demuestra que se realizó una evaluación cuidadosa y se optó por una técnica que proporciona resultados positivos.
- El desarrollo del prototípico del sistema de recomendación ha sido exitoso, validando la viabilidad técnica del sistema y proporcionando una base sólida para el desarrollo futuro.
- La encuesta brindó una forma directa de recopilar la retroalimentación de los usuarios.

rios sobre el prototipo. Esto permitió obtener opiniones, comentarios y sugerencias que provienen de la experiencia real de los usuarios al interactuar con el sistema de recomendación. Esta retroalimentación fue fundamental para identificar puntos débiles y oportunidades de mejora.

- El prototipo tiene la capacidad de ser escalable, si se garantizan mejores recursos computacionales.

En lo personal, el desarrollo de este prototipo fue una dura experiencia, sin embargo, me permitió conocer aptitudes como la resiliencia y la perseverancia, además de poder aprender nuevas tecnologías y herramientas para el desarrollo de software.

## 7.2. Desarrollos futuros

De acuerdo al alcance de este trabajo de grado se definieron una serie de objetivos a cumplir, sin embargo existen componentes que podrían ser mejorados en trabajos futuros, algunos de estos son:

- **Integrar mas fuentes de información:** Ya que el objetivo general es la centralización de la información, aumentar el numero de estas fuentes permitirá un acceso preciso a la información y expandiría la comunidad de usuarios del prototipo.
- **Optimización del modelo:** Optimizar el modelo de recomendación, daría como resultado un sistema de recomendación mas complejo a costa de un menor consumo de recursos computacionales.
- **Integrar otro enfoque de recomendación:** Esto va de la mano con la primer viñeta, si se logran integrar fuentes de información que contengan datos sobre sus usuarios (como gustos o calificaciones a los videojuegos), daría cabida a otros enfoques como el filtrado colaborativo, por ejemplo.
- **Inclusión de un modelo de personalidad:** Implementar un algoritmo que permita predecir la personalidad del usuario hará que las recomendaciones sean mas precisas, y, que dicho usuario sienta que las recomendaciones son personalizadas.
- **Mejora continua:** Si bien la técnica seleccionada obtuvo buenos resultados en las pruebas de usuario, siempre existe margen para mejorar y perfeccionar el algoritmo de recomendación. Se podría concluir que se requiere un enfoque continuo de mejora y refinamiento de la técnica para seguir aumentando la precisión de las recomendaciones y adaptarse a los cambios en los gustos y preferencias de los usuarios.

## 8. Referencias

- [1] K. Kathrani, “Medium.com.” <https://medium.com/@kashyapkathrani/all-about-embeddings-829c8ff0bf5b>, 2020.
- [2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” 2019.
- [3] P. A. A. Carrillo, E. R. F. González, and I. V. López, “Análisis comparativo de las medidas de semejanza aplicadas al contenido de documentos web,” 2007.
- [4] P. Djundik, “Steam game release summary by year.” <https://steamdb.info/stats/releases/>. Accedido en: 2021-10-13.
- [5] P. Djundik, “Steam game releases summary for indie.” <https://steamdb.info/stats/releases/?tagid=492>. Accedido en: 2021-10-13.
- [6] T. Altenburger, “Hilo de twitter.” <https://twitter.com/mrhelmut/status/1412736809017217030>, 7 2021. Accedido en: 2021-10-13.
- [7] L. Vargas, “Desde el inicio de la pandemia el número de jugadores virtuales en colombia incrementó 16%.” <https://www.larepublica.co/internet-economy/desde-el-inicio-de-la-pandemia-el-numero-de-jugadores-virtuales-en-colombia-incre> 10 2020. Accedido en: 2021-10-13.
- [8] C. Bernal, “En latinoamérica, el 49,7% de los gamers son mujeres.” <https://forbes.co/2020/02/18/forbes-women/en-latinoamerica-el-497-de-los-gamers-son-mujeres/>, 2020. Accedido en: 2021-10-19.
- [9] J. Schulkin, “América Latina representa el 10% de los gamers a nivel global.” <https://www.infobae.com/gaming/2020/12/30/america-latina-representa-el-10-de-los-gamers-a-nivel-global/>, 12 2020. Accedido en: 2021-10-15.
- [10] SteamBlog, “Presentamos el recomendador interactivo de steam.” <https://store.steampowered.com/news/app/593110/view/1716373422378712840?l=spanish>, 2020. Accedido en: 2022-02-28.
- [11] A. Bermejo, “desarrollo de un recomendador de videojuegos”, 9 2017.
- [12] D. Y. Pérez, “Recomendación de videojuegos basado en análisis semántico y minería de opinión,” 6 2016.

- [13] C. BharathiPriya, A. Sreenivasu, and S. Kumar, “Online videogame recommendation system using content and collaborative filtering techniques,” 2021.
- [14] H. Calderon-Vilca, N. M. Chavez, and J. M. R. Guimare, “Recommendation of video-games with fuzzy logic,” 2019.
- [15] IBMCloudEducation, “¿qué es una api rest? - colombia.” <https://www.ibm.com/co-es/cloud/learn/rest-apis>, 4 2021. Accedido en: 2021-05-17.
- [16] M. Contributors, “Rest - glosario — mdn.” <https://developer.mozilla.org/es/docs/Glossary/REST>, 2021. Accedido en: 2021-06-17.
- [17] A. González, “Sistemas de recomendación de contenido con machine learning.” <https://cleverdata.io/sistemas-recomendacion-machine-learning>, 2014. Accedido en: 2021-06-17.
- [18] Docs.opencv.org, “Introduction to support vector machines — opencv 2.4.13.2 documentation.” [https://docs.opencv.org/2.4/doc/tutorials/ml/introduction\\_to\\_svm/introduction\\_to\\_svm.html](https://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html), 2017. Accedido en: 2021-05-17.
- [19] S. Kanetkar, A. Nayak, S. Swamy, and G. Bhatia, “Web-based personalized hybrid book recommendation system,” *International Conference On Advances In Engineering Technology Research*, 2014.
- [20] L. Xin, E. Haihong, S. Junde, S. Meina, and T. Junjie, “Collaborative book recommendation based on readers’ borrowing records.,” *International Conference On Advanced Cloud And Big Data*, 2013.
- [21] S. González-Carvajal and E. Garrido-Merchán, “Comparing bert against traditional machine learning text classification,” 2005.
- [22] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, “Glue: A multi-task benchmark and analysis platform for natural language understanding,” 2019.
- [23] A. Conneau and D. Kiela, “Senteval: An evaluation toolkit for universal sentence representations,” 2018.
- [24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [25] N. Morales, N. Sequeira, T. Prendas, and K. Zúñiga, “Escala de likert una herramienta econÓmica,” 2016.

- [26] J. Heyman, H. Heyman, C. Byström, and J. Hamré, “Locust documentation.” <https://docs.locust.io/en/stable/>, 2014. Accedido en: 2023-05-17.

## **9. Anexos**

En esta sección se relaciona el enlace al repositorio dónde se encuentra alojado el código fuente de este prototipo.

- Repositorio en GitHub