

Fecha: 03/05/2019
Universidad del Valle de Guatemala
Sección 30
Algoritmos y Estructuras de Datos



GameBell

Sistema de recomendación de videojuegos

Grupo 5
Integrantes: Mario Perdomo 18029
Josue Sagastume 18173
Andrés Quan 17652

Índice

Empatía
Definición
Ideación
Prototipos
Pruebas

Design Thinking:

Empatía

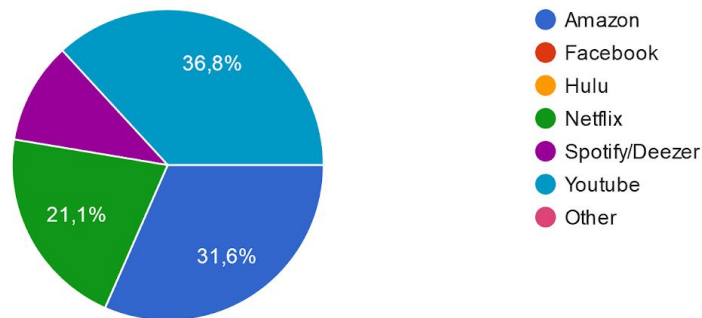
Existen muchos problemas generales cuando se trata con la población comercial. Entre ellos, existen los problemas causados por la ambigüedad necesaria para poder mantener un perfil general, y así vender productos a una etnia o población más grande. Esto se debe a que los productos, por su naturaleza, se tratan de bienes físicos, tocables, que solo son producidos una cantidad limitada de veces. Cuando se tratan con estos productos, se debe de tener cuidado con el enfoque, ya que un enfoque muy pequeño puede no atraer a la suficiente clientela para la venta; mientras que un enfoque demasiado amplio puede ser poco conveniente para apelar a alguien. Estos problemas suelen ser causados por muchas cosas, pero lo más importante se presenta a continuación:

Primero que todo, existe el problema que enfrenta la sociedad moderna. Por la naturaleza de ésta, todo está en constante cambio, y es difícil para el mercado adaptarse a estos en muchas ocasiones. Esto se debe al nacimiento del *yo* moderno, el cual solo es amplificado por las redes sociales, entre otros factores. Este *yo*, resumido perfectamente por la frase de YouTube de “*Broadcast Yourself*” es una razón por la cual los cambios son tan comunes. Las ya famosas “tendencias” son comunes en tiempos modernos, mientras que antes solían durar por años de años. Este problema del *yo* moderno también puede ser tomado como una necesidad humana, ya que el *boom* poblacional ha ayudado a que la cantidad de personas no solo aumenta, sino que sus tendencias también cambian rápidamente gracias a este aumento. Poniéndolo de manera simple, mientras más personas hayan en un mercado, más fácil es que aparezcan más tendencias, ya que existen más lugares de donde puedan darse.

Pero, a todo esto ¿por qué utilizar sistemas de recomendación si solo es utilizar más dinero para personalizar algo que ya funcionaba bien? La respuesta es muy simple. Un usuario que encuentra lo que quiere es mucho más propenso a regresar a dicha tienda. Al final de cuentas, las interacciones entre el usuario y el programa no deberían de ser inconsecuentes. Cuando se tiene un programa que se siente más *vivo*, también se tiene un usuario que quiere interactuar con él por mucho más tiempo. Para aclarar estos puntos, hicimos una encuesta a varias personas con este formato:

Out of every service listed here, which one do you believe is the best one with the whole "Recommended because you watched/saw..." system?

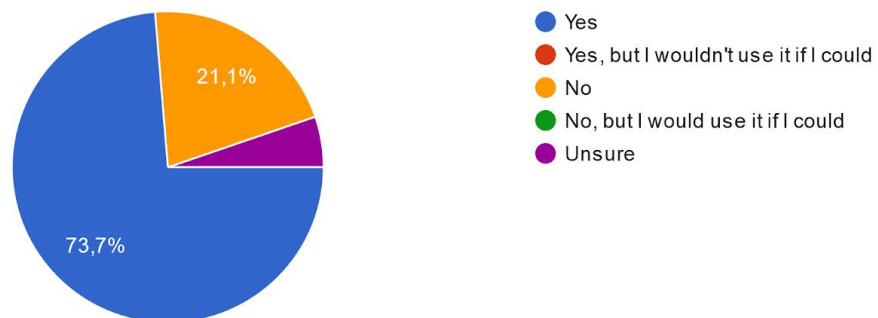
19 respuestas



Gráfica #1: Porcentaje de personas utilizando diferentes sistema de recomendación

Do you end up using the recommended system a lot?

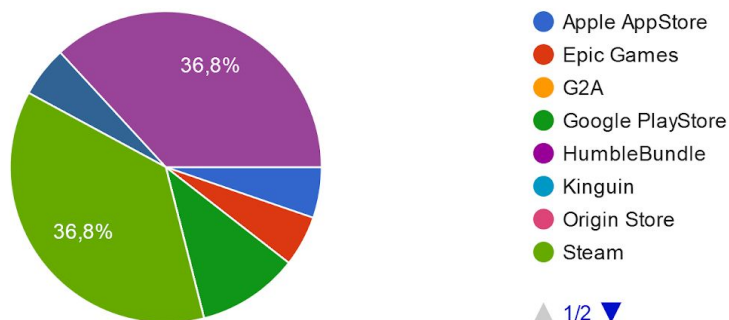
19 respuestas



Gráfica #2: Porcentaje de personas utilizando el sistema de recomendación.

If you had to choose between the following, which one do you believe has the best recommendation system?

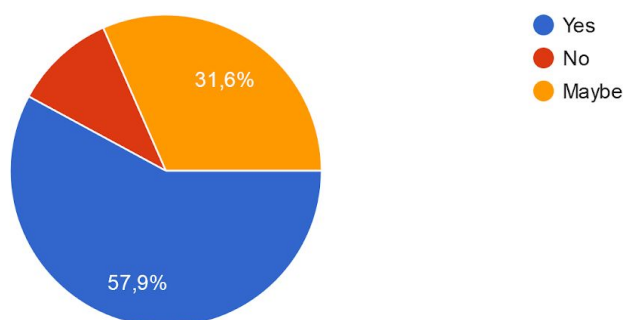
19 respuestas



Gráfica #3: Porcentaje de diferentes preferencia en sistema de recomendaciones.

Do you consider that a program with recommendations is better than a program without recommendations?

19 respuestas



Gráfica #4: Porcentaje en la funcionalidad de un sistema de recomendaciones.

Según las encuestas previamente vistas, existen un porcentaje mucho más alto de personas que buscan y utilizan el sistema de recomendación, mientras que un porcentaje diminuto no lo utilizan. ¿Qué nos dice eso? Nos dice que el sistema, aunque no sea muy intrusivo, es importante para la experiencia de las personas.

Por último, se puede concluir lo siguiente: el sistema de recomendación es un bien necesario en la industria de los videojuegos, ya que sin él, los problemas serían multiplicados, y las ventas serían menores. Asimismo, la experiencia del usuario se vería disminuida, y la falta de recomendaciones no le permitiría disfrutar al máximo de todos los bienes en la tienda. La experiencia individual es uno de los bienes más preciados en la sociedad moderna, y la satisfacción del cliente suele depender de esto.

Definición

El problema más imponente cuando se trata con este tipo de realidades, es que el usuario se encuentra limitado a una vista general de una tienda virtual. ¿Cómo tratan de resolver esto las tiendas? con sistemas de recomendaciones generales. Por ejemplo, cuando uno hace clic en “*I’m Interested*” en la tienda de Steam a algún juego, comienzan a aparecer recomendaciones personalizadas bajo el título de “*Recommended For You.*”

Las grandes empresas tratan de resolver este problema de una forma muy interesante. En tiempos ya olvidados, se trataban de resolver los problemas mediante la generalización poblacional, la cual daba paso a los estereotipos. Conforme fue avanzando el tiempo, se inventaron sistemas modernos con grandes capacidades de procesamiento, lo que a su vez permitía que se pudiera personalizar más efectivamente la experiencia del usuario. La segunda solución más reciente, utilizada más que todo en las tiendas de reventa, es aquella que divide a los juegos por su género (*es decir, por acción, shooter, horror, etc*) ya que es una manera sencilla de organizar y acudir a una población más generalizada. Aunque anticuada, era efectiva, y es una de las maneras más populares de organizar los videojuegos. Gracias al salto virtual, se han encontrado maneras más efectivas de enseñarle al usuario lo que quiere ver directamente.

Estos sistemas son los de recomendación. Los sistemas de recomendación permiten que el usuario vea y escoja lo que quiere a base de una selección basada en lo que ya había visto con anterioridad. Esto, asimismo, ha aumentado la longevidad de los videojuegos, ya que se basan solamente en lo que prefiere el usuario, no en la edad del producto (esto no es cierto para Steam, ya que ésta plataforma sí incluye filtros temporales.)

Ideación

Basándonos en la escasez de sistemas de recomendación para videojuegos específicos y en la necesidad de las tiendas virtuales para que el usuario siga explorando dicha plataforma, se decidió realizar un sistema de recomendación específico para videojuegos. El propósito de este sistema es que el usuario no se encuentre limitado a descargar solamente juegos basándose en su género, sino también por lo que la gente que comparte gustos con este usuario recomiendan, para poder brindarle la mejor experiencia.

Para realizar este sistema se utilizarán dos conjuntos de datos, los cuales nos serán necesarios para la creación de una base de datos basada en grafos. El primer conjunto de datos consta de los usuarios, los juegos y los distintos géneros de juegos que existen en la actualidad. Este conjunto se utilizaría para crear los nodos de nuestro grafo. El segundo

conjunto de datos consta de los likes que los usuarios otorgan a los videojuegos, las recomendaciones de los usuarios y otros juegos que los usuarios han descargado. Estos datos serían las aristas o conexiones de nuestros nodos.

De esta manera, el usuario será capaz de utilizar una plataforma que implemente esta base de datos, en la que el jugador tendrá una cuenta en la que podrá calificar y recomendar juegos y también podrá obtener recomendaciones de juegos de acuerdo a sus gustos y a sus últimas descargas.

Prototipo

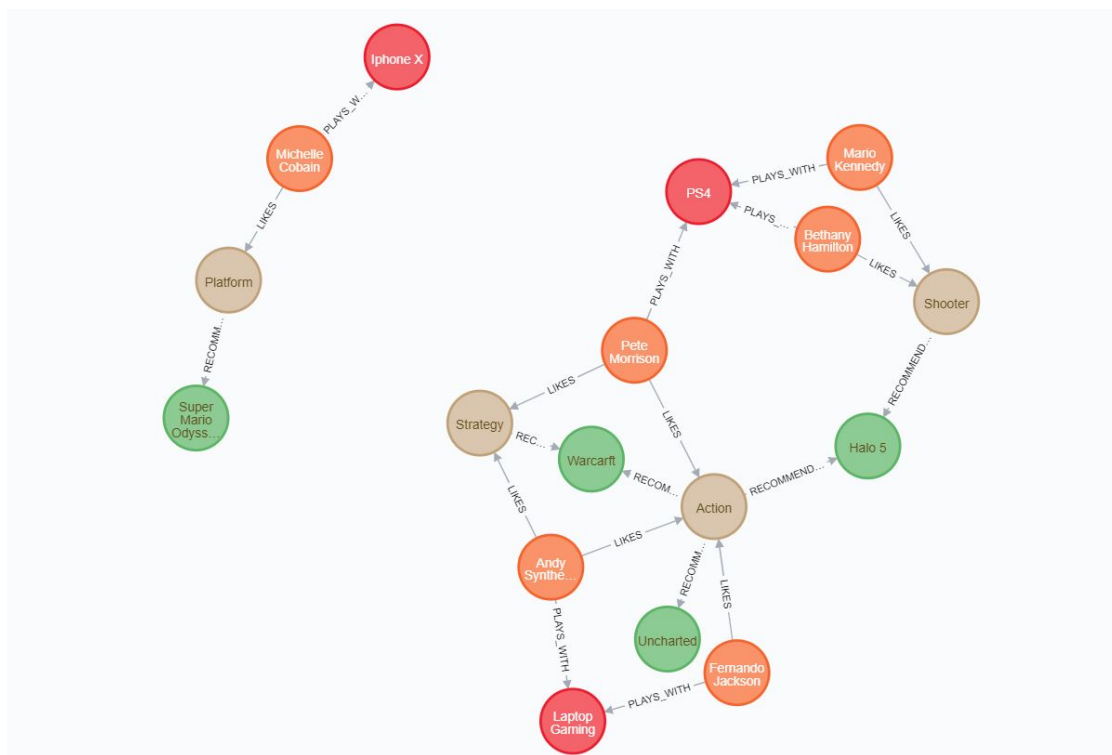


Figura #1: Grafos relacionados el tipo de género y posibles recomendaciones de videojuegos.

Nodos utilizados como información:

- Persona: Este nodo contiene los atributos de nombre y edad.
- Juego: Este nodo contiene los atributos del título y edad.
- Género: Este nodo contiene el título de un género

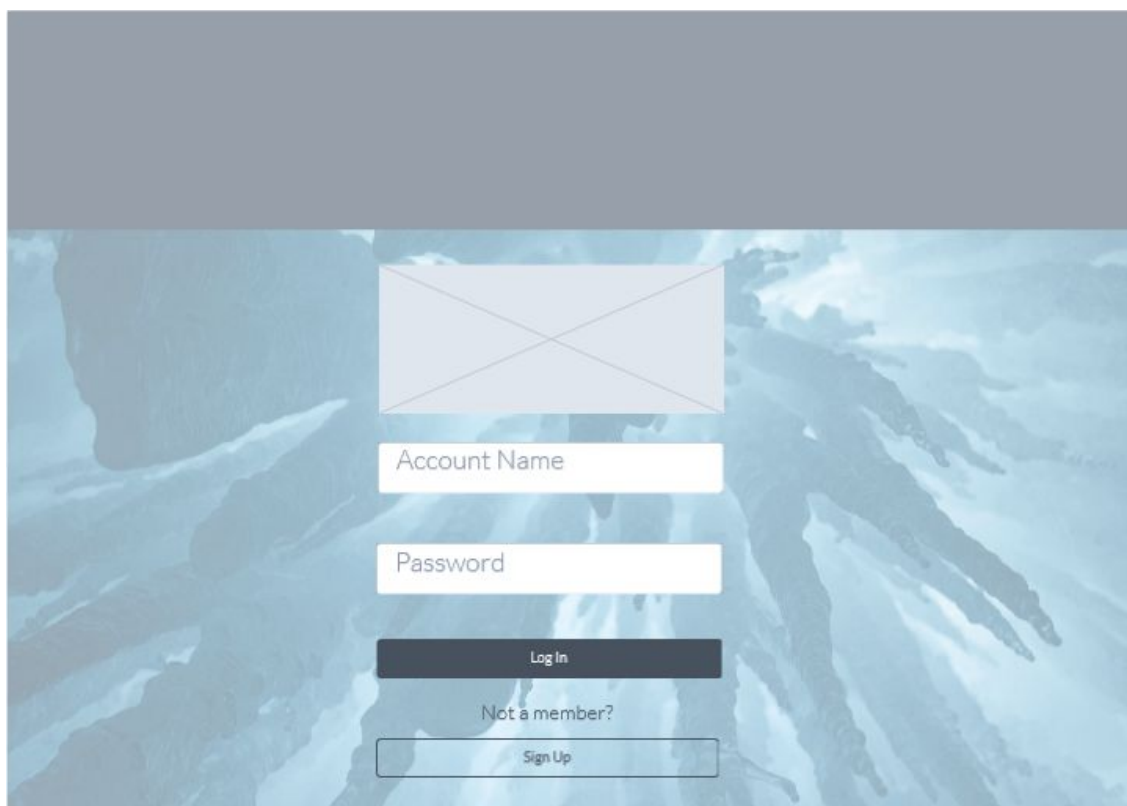
Aristas:

- Likes: Conexión de que le gusta este tipo de género en específico
- Recommendation: Conexión del juego recomendado para el usuario.
- Plays_with: Conexión que prefiere utilizar tal sistema para sus videojuegos.

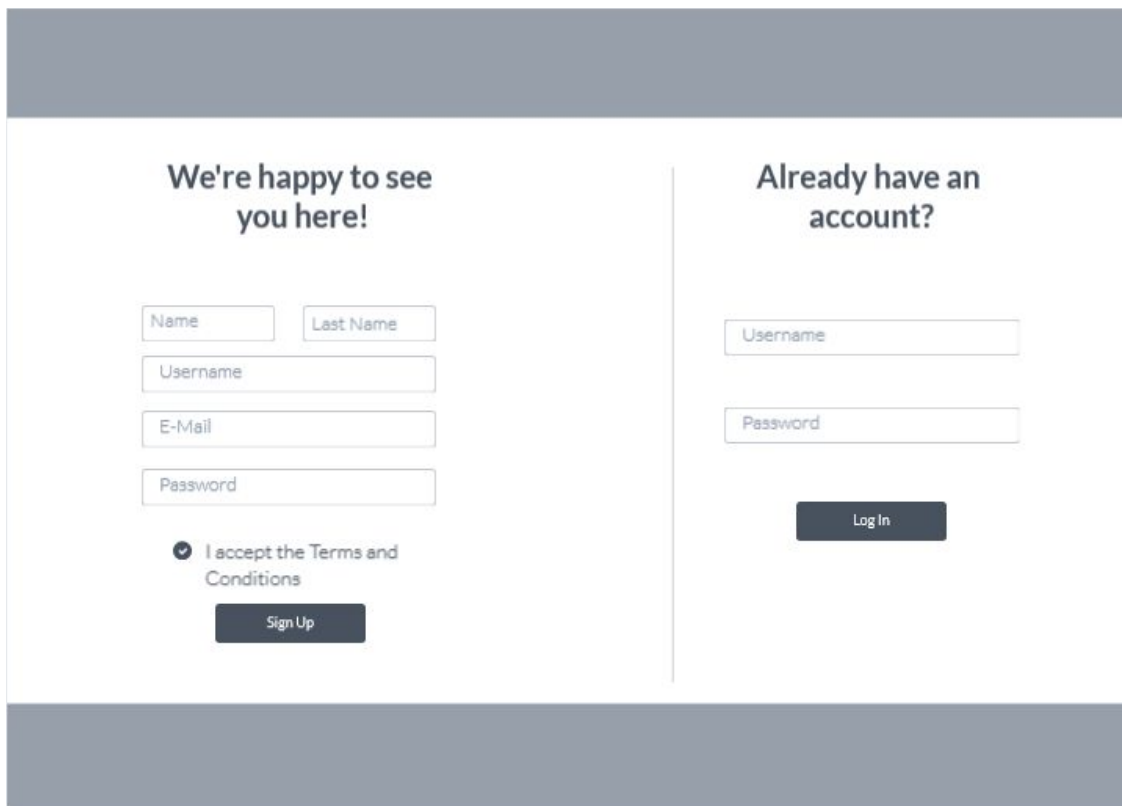
Este prototipo genera recomendaciones para posible juegos que le interesa basados a los gustos del usuario. No será tan sencillo el algoritmo, ya que se utilizará un sistema dinámico binario, en el cual se utilizan dos parámetros para generar las recomendaciones. El primer parámetro, dado por el usuario de manera directa, se trata del género del juego. Gracias a que los juegos caen dentro de varios géneros, se utilizará el más prominente. El género viene directamente señalado por el usuario, el cual lo verá en una tab de “Recommended for You.” Esta tab es influida en un principio por el usuario, pero se modifica a sí misma con el tiempo, gracias a las interacciones usuario-computador (aquí es donde se encuentran los pesos de las aristas, entre otros factores.)

El otro parámetro, un poco menos complicado, se utiliza más que todo cuando viene de empresas. El usuario verá recomendaciones basadas en los juegos que seleccionará, pero no basadas en sus géneros, sino en las empresas que los hicieron. Más que todo, se obtiene qué empresa hizo el juego, y luego se recomienda bajo una tab de “*Company* also has...” o algo con la misma idea.

Código: <https://github.com/Darlight/GameBell.git>



Pantalla #1: Pantalla de Inicio - Log In



The image shows a registration and login form with a light gray background and a dark gray header and footer. The form is divided into two columns by a vertical line. The left column is for registration, titled "We're happy to see you here!". It contains input fields for Name, Last Name, Username, E-Mail, and Password. Below these fields is a checkbox labeled "I accept the Terms and Conditions" and a "Sign Up" button. The right column is for login, titled "Already have an account?". It contains input fields for Username and Password, and a "Log In" button.

We're happy to see you here!

Name Last Name

Username

E-Mail

Password

☒ I accept the Terms and Conditions

Sign Up

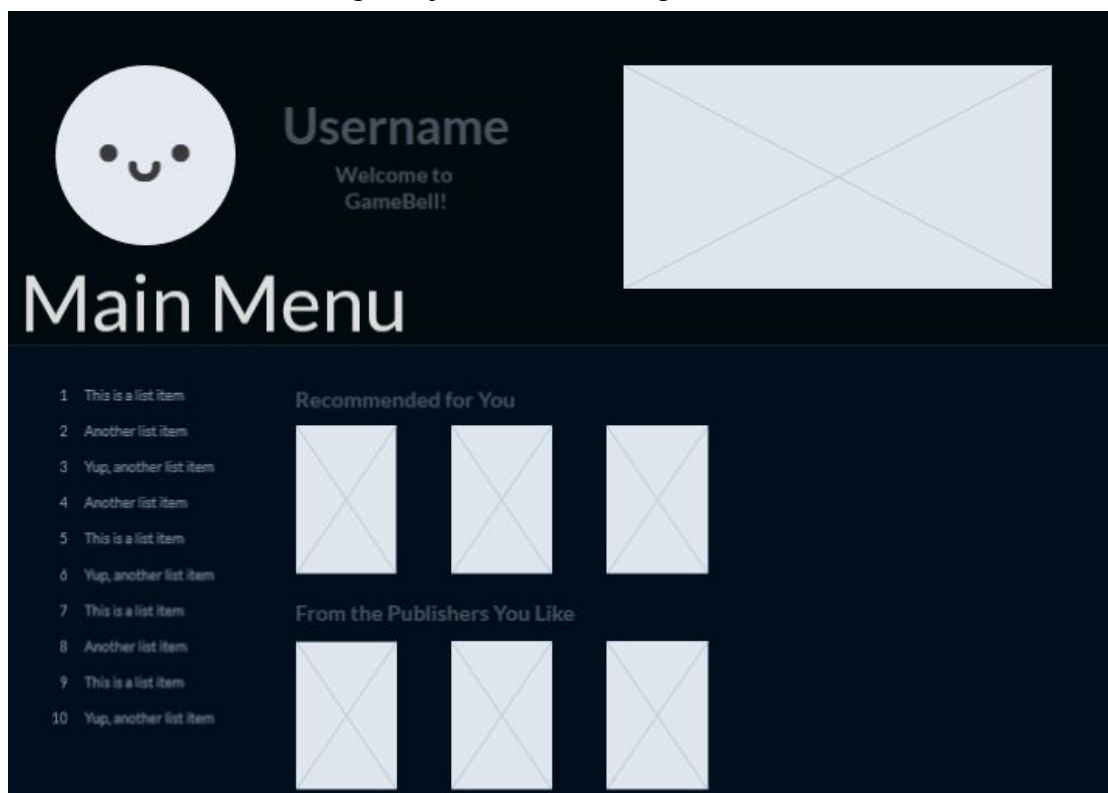
Already have an account?

Username

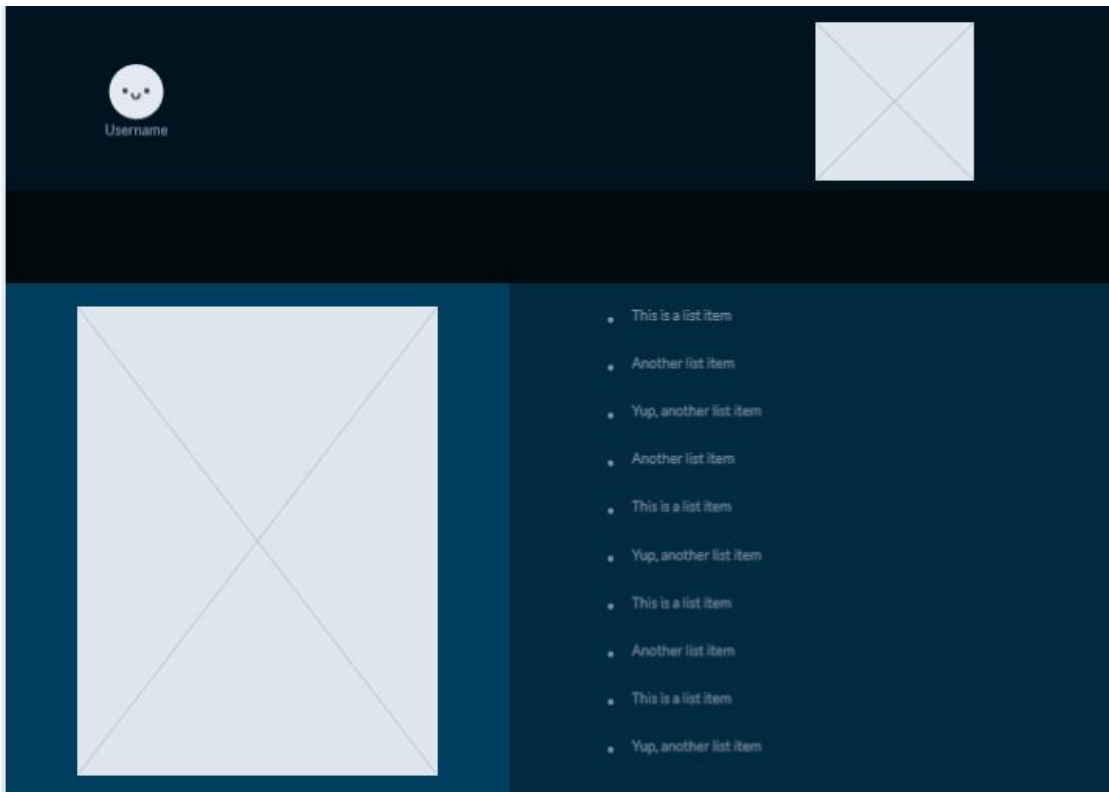
Password

Log In

Pantalla #2: Pantalla de registro junto a una de Log In



Pantalla #3: Pantalla de inicio después de hacer Log In. Cuadros blancos representan espacios para portadas de juegos y del logo (esquina superior derecha)



Pantalla #4: Demostración del juego, junto con su información y el logo.

Pruebas

Al presentar la idea del algoritmo y del programa a distintos individuos, se lograron encontrar varias opiniones. Primero que todo, como es de esperarse, se comentó en el diseño del programa. Por alguna razón, muchas personas comentaron en los colores y cómo representaban muy bien a los “geeks.” De por sí, la idea del programa le agradó a varias personas, la mayoría hombres. Esto es porque a los hombres se les apega más la cultura de los videojuegos, y la idea de tener otra plataforma, disyuntiva con Netflix, les causaba placer. Más que todo, se quedaban atónitos cuando se les mencionaba la idea y cómo se iba a ejecutar, pero el diseño les agradaba bastante.



Foto #1: Una entrevistada observa y “califica” el diseño del programa

Algoritmos de recomendación:

Algoritmo de recomendación de productos de Amazon:

Este algoritmo utiliza dos conjuntos de datos. El primer conjunto de datos contiene reseñas de los clientes sobre los productos vendidos en Amazon. Cada revisión tiene una descripción textual de la experiencia del usuario; una calificación del producto en cuestión; y un voto de utilidad que representa qué tan útil fue ésta revisión para otros clientes.

El segundo conjunto de datos son los metadatos del producto, la cual incluye: la descripción del producto, el precio, el rango de ventas, la información de la marca y los enlaces de compra conjunta.

Utilizando estos conjuntos de datos, se crea un grafo donde cada nodo es un producto y cada arista es una relación entre dos productos. Cada nodo en el gráfico tiene un peso que se utilizará para calcular el peso de la arista del grafo.

Para calcular la puntuación del producto se usa el primer conjunto de datos. Para un producto objetivo, se obtiene todas las revisiones para éste y para cada revisión se extrae el voto de utilidad y la puntuación general. El proceso de pensamiento es el siguiente: cuanto más alto es el puntaje general, mejor es el producto. Además, cuanto mayor sea la utilidad, más personas considerarán ésta revisión al comprar el producto. Por lo tanto, se necesita crear una puntuación de producto que sea directamente proporcional a la puntuación general y al voto de utilidad.

Utilizando el segundo conjunto de datos, se crean los bordes entre los productos. Usando los enlaces de compra conjunta, se construyen aristas entre el producto objetivo y cada producto de compra conjunta de la siguiente manera:

- Arista con peso = 1 para productos comprados en conjunto
- Arista con peso = 2 para productos también comprados
- Arista con peso = 3 para productos también vistos

Después de crear los pesos de arista iniciales, se calculan los nuevos pesos en función a la puntuación del producto de cada nodo y el peso de la arista inicial. La nueva puntuación de una arista E entre el producto P_1 y el producto P_2 es la siguiente:

$$E(P_1, P_2) = \text{Peso Inicial Arista} * (1 - \text{puntuación del Producto } P_1) * (1 - \text{puntuación del producto } P_2) \quad (\text{I})$$

De esta manera, los productos con mayor puntaje de producto y mejor interacción inicial están más cerca en el grafo.

Para encontrar recomendaciones para cada producto objetivo, se busca los K productos más cercanos y se devuelven al usuario.

Comienza agregando todas las conexiones directas del producto A a la lista de candidatos. Para cada ítem en la lista de candidatos:

1. Se agrega si la lista de candidatos no tiene K ítems
2. O se agrega solo si la distancia al producto A es más pequeña que la distancia máxima de todos los productos en la lista.
3. Ordenar candidatos
4. Devuelve K primeros como recomendación

Algoritmo:

<https://github.com/Darlight/GameBell.git>

Bibliografía:

Gong, S. (2010). A collaborative filtering recommendation algorithm based on user clustering and item clustering. *JSW*, 5(7), 745-752.

Zhao, L., Hu, N. J., & Zhang, S. Z. (2002). Algorithm design for personalization recommendation systems. *Journal of computer research and development*, 39(8), 986-991.

Zhou, T., Ren, J., Medo, M., & Zhang, Y. C. (2007). Bipartite network projection and personal recommendation. *Physical Review E*, 76(4), 046115.

