Thomas Uriel Fernandez

Projecto integrador final

Caso de estudio:

League of Legends es uno de los juegos mas populares de los últimos tiempos, millones de personas juegan y sobre todo compiten en las partidas clasificatorias, donde lo único que los jugadores buscan es ganar la mayor cantidad de partidas para poder subir su rango. Los rangos sirven para emparejar a los jugadores según su habilidad en el juego, los factores que definen la habilidad de un jugador son muchos y todos son influyentes en el resultado final de la partida, cosas como la habilidad mecánica para usar el mouse y teclado y ejecutar difíciles combos con ventanas de micras de segundo entre cada input; control sobre las oleadas de súbditos (Si juegas en una línea) para denegarle oro y experiencia a tu oponente; limpieza de campamentos (Si eres jungla) para tener más oro, experiencia y tempo sobre el jungla rival; manejo de la visión para poder usar la información a tu favor a la hora de tomar decisiones en equipo o para denegarle dicha información al rival y que tenga que tomar decisiones arriesgadas. Pero todos estos son factores que tienen juego una vez que la partida comienza, el factor mas importante se decide antes de jugar, cosas como las runas que usaras, el campeón que elegirás para cada línea y rol, el campeón que bloquearas, que objetos planeas comprar y en cual orden, a toda esta información se la conoce como META (most effective tactic available), esto significa cuales runas son mas fuertes con que campeón, cual campeón pierde mas contra otro, que objetos suelen ganar mas por el menor oro, todo basado en las miles de partidas diarias, para acceder fácilmente a esa información los jugadores utilizan múltiples paginas y aplicaciones que recopilan la extensa base de datos de Riot Games, desarrolladora de League of Legends, y la muestra de forma simple pero eficiente, Ejemplos podrían ser OP.GG, Mobalytics, U.GG, MOBAfire, etc. Riot Games decidió que era hora de monopolizar esa herramienta con su propia aplicación, por lo que nos pidieron hacer una demo con una base de datos totalmente nueva, con sus tablas y todas las funciones y agregados necesarios para una base sólida, integra y útil. A continuación una explicación de las tablas y herramientas de la base de datos.

Tablas y esquemas:

Esquemas y Tablas:

La base cuenta con 2 esquemas para dividir el trabajo, información y funcionalidades de la base de datos en 2 equipos, BLNC\_CAMPEON que se encargará de revisar la relevancia de los campeones del juego y su rendimiento, y BLNC\_MISC que hará lo mismo pero orientado a runas y objetos.

BLNC\_CAMPEON: Tablas

CAMPEON:

ID\_CAMPEON: Representa la clave primaria de la tabla y es el identificador único de cada campeón.

NOMBRE\_CAMPEON: muestra solo el nombre de los campeones del juego y su identificador único.

RESUMEN:

ID\_RESUMEN: Tiene múltiples campos, ID\_RESUMEN sirve para identificar las entradas de RESUMEN, es además la clave primaria de la tabla.

ID\_BUILD: Hace referencia a una entrada de la tabla BUILD en el esquema BLNC\_MISC.

ID\_CAMPEON, ID\_RIVAL y ID\_BAN: Todos hacen referencia a un campeón de la tabla CAMPEON, Campeón siendo el campeón jugado, Rival siendo el campeón enemigo, y Ban siendo el campeón bloqueado por el jugador

KDA: Representa un número con coma que hace referencia a la puntuación de asesinatos, muertes y asistencias del jugador con dicho campeón. Asesinatos + Asistencias / Muertes.

TORRES\_DESTRUIDAS: Hace referencia a la cantidad de torres que el jugador derribo con dicho campeón durante la partida.

PRIMERA\_SANGRE: Es un bit que dice si el jugador obtuvo el primer asesinato de la partida.

RESULTADO: Dice si el jugador obtuvo o no la victoria con su campeón, build, y forma de jugar.

Resumen tiene además un trigger que verifica que los resultados de RESUMEN y BUILD sean correlativos, TRG\_CHECK\_RESULTADO\_RESUMEN.

BLNC\_MISC: Tablas

RUNA:

ID\_RUNA: Es el identificador único de cada runa y es la clave primaria de la tabla.

NOMBRE\_RUNA: Es el nombre de cada runa.

NIVEL: Representa el nivel de cada runa (Clave, Superior, Intermedio e Inferior).

Rama: Representa la rama a la que pertenece cada runa (Precisión, Valor, Dominación, Brujería e Inspiración).

BUILD\_RUNA:

ID\_BUILD\_RUNA: Es la clave primaria de la tabla, representa una combinación única y valida de runas.

ID\_RUNA\_CLAVE: Representa el id de una runa clave de la tabla RUNA, la runa clave define la rama que deben seguir las runas subsecuentes, y la rama a la cual las runas secundarias no pueden pertenecer.

ID\_RUNA\_SUP: Representa el id de una runa superior de la tabla RUNA.

ID\_RUNA\_INT: Representa el id de una runa intermedia de la tabla RUNA.

ID\_RUNA\_INF: Representa el id de una runa inferior de la tabla RUNA.

ID\_RUNA\_SEC: Representa el id de una runa de la tabla RUNA, las runas secundarias no pueden ser de nivel clave, deben ser de una rama distinta a las runas principales y deben compartir rama entre ellas, además no pueden compartir nivel.

ID\_RUNA\_SEC\_ALT: Representa el id de la segunda runa secundaria.

Además, BUILD\_RUNA cuenta con 2 triggers, uno para verificar que los niveles de las runas sean validos TRG\_CHECK\_RUNA\_NIVEL, y otro para verificar que las ramas de las runas sean validas TRG\_CHECK\_RUNA\_RAMA.

OBJETO:  
ID\_OBJETO: Es la clave primaria de la tabla y representa un identificador único para cada objeto.

NOMBRE\_OBJETO: Representa el nombre de cada objeto.

PRECIO: Representa el precio de cada objeto.

BUILD\_OBJETO

ID\_BUILD\_OBJETO: Es la clave primaria de la tabla y representa una combinación valida de objetos.

ID\_PRIMER\_OBJETO, ID\_PRIMER\_OBJETO, ID\_PRIMER\_OBJETO, ID\_PRIMER\_OBJETO, ID\_PRIMER\_OBJETO, ID\_PRIMER\_OBJETO: Hacen todos referencia a un id de un objeto de la tabla OBJETO, no puede haber combinaciones repetidas, ni objetos repetidos en una misma entrada.

BUILD

ID\_BUILD: Representa el identificador de una combinación de ID\_BUILD\_OBJETO y ID\_BUILD\_RUNA, además es la clave primaria de la tabla.

ID\_BUILD\_RUNA: Hace referencia a un ID\_BUILD\_RUNA de la tabla ID\_BUILD\_RUNA.  
ID\_BUILD\_OBJETO: Hace referencia a un ID\_BUILD\_OBJETO de la tabla ID\_BUILD\_OBJETO.

RESULTADO: Es un booleano que indica si esa combinación resulto o no victoriosa, además los resultados deben coincidir con los resultados de la tabla RESUMEN.