Trabajo Final: Bases de Datos 1  
Participantes: Thomas Uriel Fernández Sierra y Mariano Vaz

Tema elegido: League of Legends

Problemática:

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, nos pidieron hacer 6 tablas y sus inserts, además de darnos una breve descripción la información que habrá en cada columna y 20 ejercicios para probar si los datos son correctos.

Partida:

* Id\_Entrada (Clave Primaria): Sirve para identificar individualmente cada entrada de datos.
* Id\_Partida: Es un identificador único para cada partida que se juega, cada partida tiene 10 entradas, ya que tiene 5 jugadores por equipo.
* Id\_Jugador (Clave Foránea => Jugador.Id\_Jugador): Es un identificador único para cada jugador de la partida, hay uno por entrada.
* Id\_Banned\_Champ (Clave Foránea => Campeon. Id\_Campeon): Representa el campeón bloqueado por un jugador, significa que en toda la partida ya ningún jugador podrá elegir dicho campeón.
* Id\_Campeon (Clave Foránea => Campeon. Id\_Campeon): Representa a un campeón elegido, un campeón no puede ser elegido 2 veces en la misma partida.
* Id\_Build (Clave Foránea => Build.Id\_Build): Representa una build que elige el jugador y puede elegirse la misma build varias veces en una partida, pero una por jugador.
* Resultado: Esto es un campo simple que define el resultado, ‘Victoria’ o ‘Derrota’.

Campeon:

* Id\_Campeon (Clave Primaria): identifica a cada campeón por separado.
* Nombre\_Campeon: Un campo para el nombre del campeón.
* Id\_Linea: Un campo para la línea recomendada.
* Rol: Un campo para el rol del campeón.

Jugador:

* Id\_Jugador (Clave Primaria): Identificador único para cada jugador, clave primaria de la tabla.
* Nombre\_Jugador: Es un campo para el nombre del jugador.
* Nivel\_Jugador: Un campo para saber cuanta experiencia tiene la cuenta del jugador.

Build:

* Id\_Build (Clave Primaria): Identificador único de cada build, las build se componen de 4 objetos y 6 runas.
* Id\_Runa\_Clave (Clave Foránea => Runa.Id\_Runa): El Jugador puede elegir 4 runas principales para su build, estas runas se dividen por según 2 parámetros, Nivel de runa y Rama de runa, hay 4 niveles, Clave, Superior, Intermedia e Inferior, el jugador debe usar una runa de cada nivel. Sobre las ramas, son Valor, Precisión, Brujería, Dominación e Inspiración, cada build tiene 2 ramas, una para las runas principales y otra para las runas secundarias.
* Id\_Runa\_Superior (Clave Foránea => Runa.Id\_Runa)
* Id\_Runa\_Intermedia (Clave Foránea => Runa.Id\_Runa)
* Id\_Runa\_Inferior (Clave Foránea => Runa.Id\_Runa)
* Id\_Primera\_Runa\_Secundaria (Clave Foránea => Runa.Id\_Runa): Las Runas Secundarias son 2 runas extra que el jugador puede usar, no pueden ser runas clave, ni runas de una rama que el jugador haya usado en sus runas principales, además no se puede repetir nivel de runa, por ejemplo 2 runas superiores.
* Id\_Segunda\_Runa\_Secundaria (Clave Foránea => Runa.Id\_Runa)
* Id\_Primer\_Itemcore (Clave Foránea => Item.Id\_Item): Todos los items core referencian a un objeto comprado por el jugador dentro de la partida, su única restricción es que no se puede comprar el mismo objeto más de una vez.
* Id\_Segundo\_Itemcore (Clave Foránea => Item.Id\_Item)
* Id\_Tercer\_Itemcore (Clave Foránea => Item.Id\_Item)
* Id\_Cuarto\_Itemcore (Clave Foránea => Item.Id\_Item)

Runa:

* Id\_Runa (Clave Primaria): Identificador único de cada runa.
* Nombre\_Runa: Nombre de dicha runa.
* Nivel\_Runa: Nivel de dicha runa.
* Rama: Rama de dicha runa.

Item:

* Id\_Item (Clave Primaria): Identificador único de cada item.
* Nombre\_Item: Nombre de dicho item.
* Precio: Precio en Oro de cada item.
* Categoria: Categoría de cada objeto según el rol recomendado.