



Universidad de Concepción

Ingeniería Civil Informática



Docente a cargo: **Marcela Varas**

Proyecto: Hito 2

Modelado Conceptual

Grupo 13

- ❖ Joaquín Sebastián Ávalos Panes
(2021413898)
- ❖ Matías López Jara
(2021770259)
- ❖ Matías Medina De La Peña
(2021409637)
- ❖ líder: Raúl Zapata Villalobos
(2019900803)

- ❖ Horario de reunión:
martes 15:00 – 16:00

índice

Introducción	2
Datos	3
Requerimientos	4
Características del Sistema	5
Interfaz del Sistema	6
Modelo Conceptual	9
Documentación del Modelo Conceptual	10
Tipos de entidad.....	10
Tipos de Interrelación	14
Estructuras de Generalización.....	16
Restricciones Estáticas no modeladas.....	16
Suposiciones.....	16
Esquema Relacional.....	17
Link a código de la base de datos.....	19
Plan de Trabajo.....	20
Decisiones de Diseño.....	21
Restricciones de Integridad	22
Consultas	23
Conclusión	25
Bibliografía.....	26

Introducción

En el apasionante mundo de los deportes electrónicos (e-sports), la gestión de datos se ha convertido en un componente fundamental para el éxito de cualquier torneo. Los e-sports han experimentado un crecimiento meteórico en popularidad en los últimos años, atrayendo a jugadores y fanáticos de todo el mundo. Con la expansión de este mercado, la necesidad de una base de datos eficiente y bien estructurada se vuelve cada vez más crítica. La base de datos se diseñará con el propósito de optimizar la gestión de información, desde el registro de jugadores y equipos hasta el seguimiento de resultados y estadísticas en tiempo real.

La organización de un torneo de e-sports implica una gran cantidad de datos que deben ser gestionados de manera eficiente y precisa. Estos datos incluyen, entre otros:

- Registro de Jugadores y Equipos: Es crucial contar con información completa y actualizada de todos los participantes, incluyendo sus nombres, identificaciones, equipos, y contactos.
- Programación de Partidas: El diseño de un calendario de partidas equitativo y coherente requiere un sistema que pueda manejar fechas, horarios y ubicaciones de manera eficiente.
- Resultados en Tiempo Real: Para mantener a los espectadores y participantes informados, es esencial registrar los resultados de las partidas en tiempo real y actualizar las clasificaciones.
- Estadísticas de Juego: Los e-sports a menudo se basan en datos estadísticos detallados, como asesinatos, muertes, asistencias y objetivos. Estos datos son esenciales para la narrativa del torneo y el análisis posterior.
- Gestión de Premios: La distribución de premios y recompensas a los ganadores debe registrarse y gestionarse adecuadamente.

Datos

Los datos se van a obtener en dos tandas, periodo de inscripciones y periodo de torneo. En el periodo de inscripciones los datos a ingresar serán, los datos de los jugadores: nombre completo, matricula, IGN, carrera; a su vez se podrán inscribir los equipos que jugaran que tendrán los siguientes datos: nombre, carrera que representa, los jugadores y el juego en que participaran.

Terminada la fecha de inscripción se da inicio al torneo, durante la duración de este se guardarán las partidas de cada equipo, el ganador de cada una, la fecha en que se jugó y el tipo de partida que fue (fase de grupos, playoff, final).

En un periodo normal, la mayor cantidad de datos ingresados será en la fase de inscripciones ya que de ciencia cierta no se sabe cuántos equipos competirán y cuantos jugadores se inscribirán, pero la media está en 30 jugadores.

La fase del torneo la mayor cantidad de datos depende de los equipos, al ser un torneo universitario, la mayor cantidad de partidos se jugarán en fines de semana, en la noche usualmente, por lo que la precisión de la fecha deberá ser de minutos o segundos.

Los partidos tendrán un árbitro que verificara la identidad de los jugadores y sus cuentas para evitar suplantación de identidad, a su vez los árbitros ganan una comisión dependiendo de la cantidad de partidos que hayan arbitrado, a su vez se debe saber el primer y segundo lugar del torneo para entregar los respectivos premios.

Los datos serán dinámicos ya que se podrán agregar jugadores, cambiar la alineación de un equipo mientras juegan el torneo o si hubo un error en los ganadores de una partida y se deba rehacer.

Requerimientos

El sistema de gestión de base de datos para el campeonato de League of Legends debe cumplir con una serie de requerimientos funcionales para garantizar su eficiencia y utilidad.

Primero, el sistema debe ser capaz de permitir un acceso rápido y en tiempo real de los datos, lo que requerirá un sistema en línea. Esto facilitará la toma de decisiones y la generación de informes estadísticos basado en información actualizada.

Segundo, el sistema debe admitir el acceso simultáneo por varios usuarios. Esto puede incluir:

- **Administradores:** Quienes tendrán acceso completo a la base de datos y pueden realizar tareas de gestión, como agregar, modificar, o eliminar registros.
- **Analistas:** Quienes pueden realizar consultas avanzadas y generar informes basados en los datos del sistema.
- **Usuarios:** Tienen acceso a consultas predefinidas o informes generados por analistas, pero no pueden realizar modificaciones en la base de datos.

Entre los ejemplos de consultas que podremos realizar en la base de datos podemos considerar,

1. Consulta de Información de un Equipo:
Datos de entrada: Nombre de Equipo.
Output: Detalles del equipo, incluyendo información de los jugadores.
2. Consulta de Estadísticas de un Jugador:
Datos de entrada: Nombre de Equipo.
Output: Estadísticas detalladas del jugador, como promedio de asesinatos, campeones más utilizados.
3. Análisis de Desempeño del Campeonato:
Datos de Entrada: Periodo de tiempo (temporada o año).
Output: Informe con estadísticas globales del campeonato, como promedio de duración de partidos, equipo con más victorias y jugador con más asesinatos.
4. Búsqueda de Partidos por Fecha:
Datos de Entrada: Fecha o rango de fechas.
Output: Lista de partidos que tuvieron lugar en la fecha especificada.
5. Generación de Reportes de Temporadas:
Datos de Entrada: Temporada
Resultado: Resumen de desempeño de los equipos en esa temporada, incluyendo posición final en la clasificación y estadísticas clave.

Características del Sistema

El sistema modela el torneo mechón realizado el primer semestre del año 2023, el sistema se podría a escalar al posible torneo interfacultades realizado por Udec e-sport, la entidad encargada de la realización de los torneos de juegos electrónicos en la universidad. Por lo que es posible agregar más juegos, con distintos sistemas de torneo (torneo suizo, round-robin, etc.).

La actualización y mantenimiento del sistema se realizará cada semestre, que es la duración real de un torneo interfacultades (tomar como ejemplo torneo interfacultades de futbol o basquetbol). Las actualizaciones pueden ser dependiendo de los juegos de moda en el momento y que tengan alguna escena competitiva coherente con un torneo universitario.

A mediano plazo podría existir nuevas entidades tales como staff técnico, como entrenadores, analistas o personal médico (kinesiólogos, psicólogos) tal como un equipo profesional trabajaría. En temas de interfaz se podría crear un sistema de creación de perfiles personalizado y que el mismo sistema al finalizar la fase de inscripciones, genere los partidos a jugar en el modo deseado y se pueda ver gráficamente cada partida.

El sistema puede ser pensado como una aplicación web que ayudara a la logística de futuros torneos que formen parte de la universidad o de más universidades.

Interfaz del Sistema



Logo

Buscar (Jugador, equipo)

Usuario

Partidos

Equipos

Estadísticas

Jugador

Equipos

Logo

Buscar (Jugador, equipo)

Usuario

Partidos

Equipos

Estadísticas

Jugador

Seleccione equipo o jugador

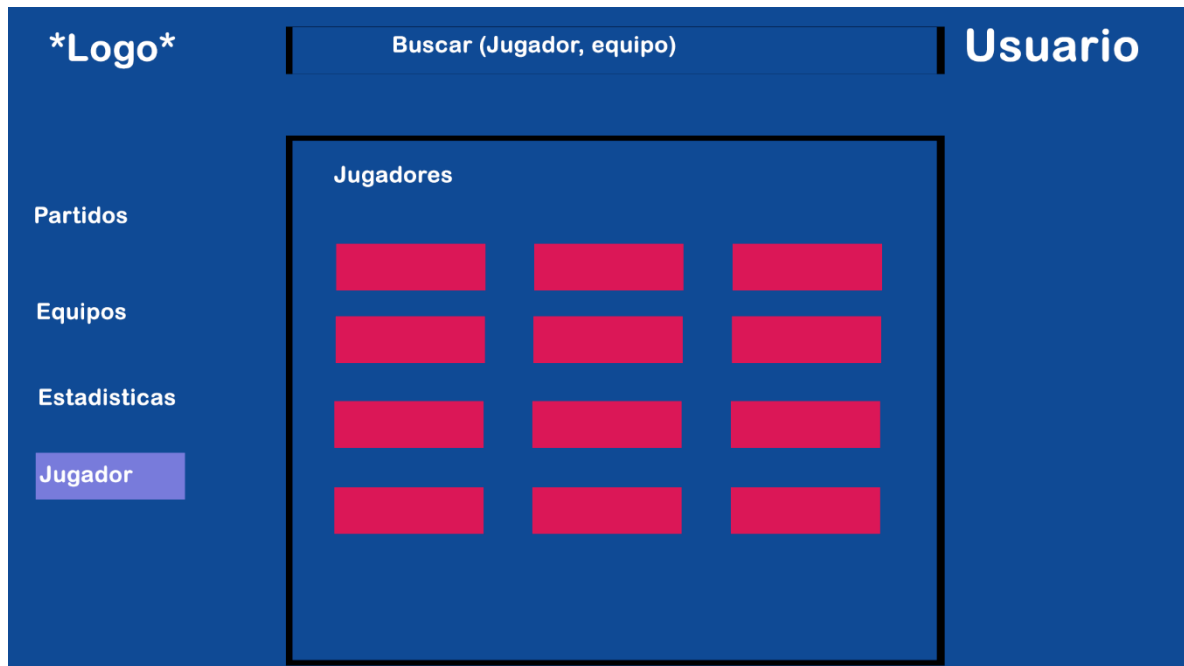
Equipo

v

Jugador

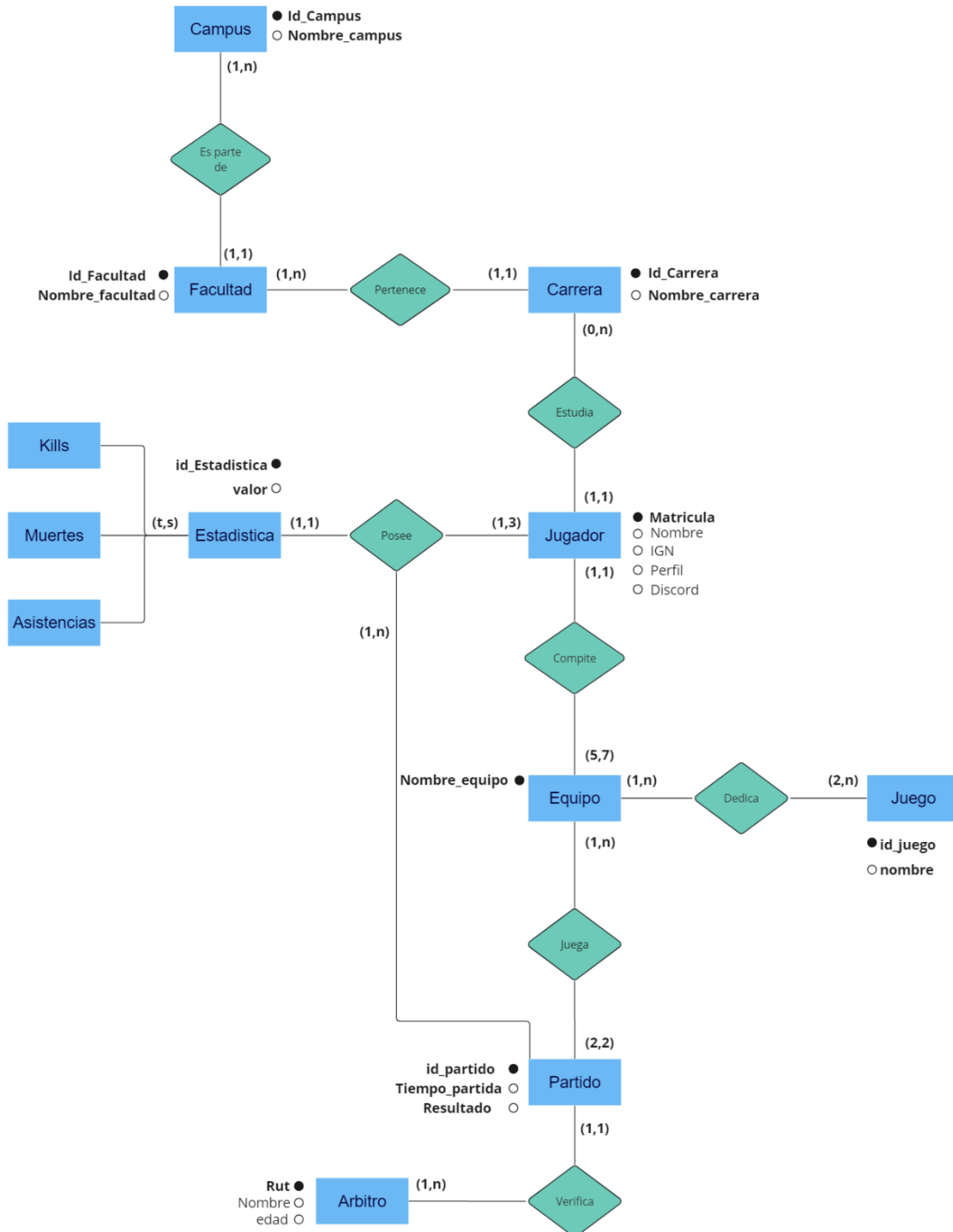
v

Buscar



Cabe recalcar que estas imágenes son referenciales/ilustrativas y no tiene por qué ser igual que el diseño final

Modelo Conceptual



Documentación del Modelo Conceptual

Tipos de entidad

Tipo de Entidad	Campus	
Descripción	Conjunto de terrenos y edificios en que se desarrollan las actividades de una universidad.	
Atributo	Dominio	Cardinalidad
Id_Campus	Nº identificador del campus	(1,n)
Nombre_campus	{Concepción, Los Ángeles, Chillán}	

Tipo de Entidad	Facultad	
Descripción	centro docente donde se imparten estudios superiores especializados en alguna materia o rama del saber.	
Atributo	Dominio	Cardinalidad
Id_Facultad	Nº identificador de la facultad	(1,1) Pertenece a un campus (1,n) posee distintas carreras
Nombre_facultad	{Facultad de Agronomía, Facultad de Arquitectura, Urbanismo y Geografía, Facultad de Cs. Ambientales, Facultad de Cs. Biológicas, Facultad de Cs. Económicas y Administrativas, Facultad de Cs. Físicas y Matemáticas, Facultad de Cs. Forestales, Facultad de Cs. Jurídicas y Sociales, Facultad de Cs. Naturales y Oceanográficas, Facultad de Cs. Químicas, Facultad de Cs. Sociales, Facultad de Cs. Veterinarias, Facultad de Educación, Facultad de Enfermería, Facultad de Farmacia, Facultad de Humanidades y Arte, Facultad de Ingeniería, Facultad de Ingeniería Agrícola, Facultad de Medicina, Facultad de Odontología, Campus Chillán, Campus Los Ángeles.}	

Tipo de Entidad	Carrera	
Descripción	Estudios pertenecientes a una entidad de educación superior en alguna materia especializada o rama del saber	
Atributo	Dominio	Cardinalidad
Id_Carrera	Nº identificador de la carrera	(1,1) Pertenece a una sola facultad (0,n) Pueden participar tantos estudiantes de la carrera como deseen, como también pueden no participar.
Nombre_carrera	Carreras pertenecientes a la universidad de concepción	

Tipo de Entidad	Jugador	
Descripción	La persona que perteneciente a un equipo juega en el torneo a un juego/s en específico	
Atributo	Dominio	Cardinalidad
Matricula	NUM (10 cifras), entregado por la institución a cada estudiante (ej:2021456789)	(1,1) jugador estudia una carrera (1,3) posee como mínimo una estadística relacionada, y como máximo 3. (1,1) un jugador compite en un equipo
Nombre	Nombre completo de la persona definido como nombres y apellidos	
IGN	Alias de la persona en el juego	
Perfil	URL de una página web administradora de perfiles (ej: op.gg)	
Discord	Perfil de Discord de la persona definido por la aplicación (ej: UsuarioDC123)	

Tipo de Entidad	Estadística	
Descripción	Las estadísticas del jugador en una partida (ej: las veces que murió, asistió o mato)	
Atributo	Dominio	Cardinalidad
Nombre_estadistica	{Asesinatos, Muertes, Asistencias}	(1,1) La estadística pertenece a un jugador en específico.
valor	Valor entero ≥ 0	(1,n) Se obtienen estadísticas de cada partido jugado

Tipo de Entidad	Juego	
Descripción	Juego en equipo online de estilo 5v5 con características competitivas	
Atributo	Dominio	Cardinalidad
Id_juego		(2,n) deben dedicarse 2 o más equipos a un juego
nombre	Nombre del juego (valorant, league of legends)	

Tipo de Entidad	Partido	
Descripción	Encuentro entre dos equipos en un juego específico	
Atributo	Dominio	Cardinalidad
Id_partido		(1,3) De un partido se obtienen 1 o 3 estadísticas
Tiempo_partida	Hora de inicio.	(2,2) En un partido juegan solo 2 equipos
Resultado	{0,1,2}: 0: no se ha jugado aún 1: Ganador equipo 1 2: Ganador equipo 2	(1,1) Un partido lo verifica solo un arbitro

Tipo de Entidad	Equipo	
Descripción	Grupo de jugadores que participaran en el torneo	
Atributo	Dominio	Cardinalidad
Nombre_equipo	Nombre regido por el equipo {colo-colo, T1, KLG}	(1,1) un equipo tiene 1 nombre y un nombre pertenece a 1 equipo

Tipo de Entidad	Arbitro	
Descripción	Persona que verifica las partidas entre los equipos	
Atributo	Dominio	Cardinalidad
Rut	Rut del arbitro {15.666.777-k}	(1,n) un árbitro verifica mínimo 1 partido, y como máximo n
Nombre	Nombre de la persona {Elvis teck}	
Edad	Edad de la persona {33}	

Tipos de Interrelación

Tipo de Interrelación	Es parte de	
Descripción	Una facultad es parte del campus	
Tipos de entidad relacionados	Rol	Cardinalidad
Facultad	Es parte de un Campus	(1,1)
Campus	En cada Campus existen distintas facultades	(1,n)

Tipo de Interrelación	Pertenece	
Descripción	Una carrera tiene una facultad	
Tipos de entidad relacionados	Rol	Cardinalidad
Facultad	Una facultad esta formada por una o más carreras	(1,n)
Carrera	Cada carrera tiene su facultad	(1,1)

Tipo de Interrelación	Estudia	
Descripción	Los jugadores estudian una carrera	
Tipos de entidad relacionados	Rol	Cardinalidad
Jugador	Todos los jugadores estudian una carrera	(1,1)
Carrera	Una carrera esta formada por varios jugadores	(0,n)

Tipo de Interrelación	Posee	
Descripción	Cada jugador posee su propio historial estadístico	
Tipos de entidad relacionados	Rol	Cardinalidad
Jugador	Todos los jugadores tienen al menos una estadística asociada	(1,3)
Estadística	Existe un historial estadístico asociado a cada jugador	(1,1)
Partido	Las estadísticas de cada jugador se asociarán a un partido	(1,n)

Tipo de Interrelación	Compite	
Descripción	Los jugadores compiten en un equipo	
Tipos de entidad relacionados	Rol	Cardinalidad
Jugador	Todos los jugadores tienen su equipo	(1,1)
Equipo	Un equipo este compuesto de jugadores	(5,7)

Tipo de Interrelación	Dedica	
Descripción	El equipo se dedica a jugar el juego	
Tipos de entidad relacionados	Rol	Cardinalidad
Juego	El juego es jugado por un equipo	(2,n)
Equipo	Un equipo juega el juego	(1,n)

Tipo de Interrelación	Juega	
Descripción	El equipo juega un partido	
Tipos de entidad relacionados	Rol	Cardinalidad
Partido	El partido es jugado por al menos 2 equipos	(2,2)
Equipo	Un equipo juega en el partido	(1,n)

Tipo de Interrelación	Verifica	
Descripción	Se verifica el partido por un árbitro para regular el juego	
Tipos de entidad relacionados	Rol	Cardinalidad
Partido	El partido es verificado por un arbitro	(1,1)
Arbitro	El árbitro verifica el partido	(1,n)

Estructuras de Generalización

Generalización	Tipo de entidad Genérica	Estadística
Cobertura	(t,s)	
Tipos de entidad subconjunto	Kills, Muertes, Asistencias	

Restricciones Estáticas no modeladas

Restricciones Estáticas		
Id Restricción	Objetos involucrados	Restricción
1	Equipo – Juego - Partido	En la fase de grupos pasa el mejor de 1 partido
2	Equipo – Juego - Partido	En la fase de play-off pasa el mejor de 3 partidos
3	Equipo – Juego - Partido	En la fase de final pasa el mejor de 5 partidos

Suposiciones

Suposiciones		
Id Suposición	Objetos involucrados	Suposición
1	Equipo – Jugador - Partido	Los jugadores no tienen una posición designada

Esquema Relacional

- **Arbitro (rut, nombre, edad)**
PK: rut
DF: Rut → nombre
Edad → edad
Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Asistencias (id_estadistica, n_asistencias)**
FK: id_estadistica → Estadistica(id_estadistica)
PK: id_estadistica
DF: id_estadistica → n_asistencias
Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Campus (id_campus, nombre_campus)**
PK: id_campus
DF: id_campus → nombre_campus
Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Carrera (id_carrera, nombre_carrera, id_facultad)**
PK: id_carrera
FK: id_facultad → Facultad(id_facultad)
DF: id_carrera → nombre_carrera
Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Equipo(nombre_equipo)**
PK: nombre_equipo
Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Equipo_Juego (nombre_equipo, id_juego)**
PK: nombre_equipo, id_juego
FK: nombre_equipo → Equipo(nombre_equipo)
id_juego → Juego(id_juego)
DF: nombre_equipo → id_juego
Id_juego → nombre_equipo
Esta en 4FN ya que sus atributos son parte de la clave compuesta.
- **Estadistica (id_estadistica, valor, matricula)**
PK: id_estadistica
DF: id_estadistica → valor
Esta en 4FN ya que su PK se relaciona solo con 1 atributo.

- **Facultad (id_facultad, nombre_facultad, id_campus)**
 PK: id_facultad
 FK: id_campus → Campus(id_campus)
 DF: id_facultad → nombre_facultad
 Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Juego (id_juego, nombre_juego)**
 PK: id_juego
 DF: id_juego → nombre_juego
 Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Jugador (matricula, nombre, ign, perfil, discord, id_carrera, equipo)**
 PK: matricula
 FK: equipo → Equipo(nombre_equipo)
 id_carrera → Carrera(id_carrera)
 DF: matricula → nombre
 matricula → ign
 matricula → perfil
 matricula → discord
 Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Kills (id_estadistica, n_kills)**
 FK: id_estadistica → Estadistica(id_estadistica)
 PK: id_estadistica
 DF: id_estadistica → n_kills
 Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Muertes (id_estadistica, n_muertes)**
 FK: id_estadistica → Estadistica(id_estadistica)
 PK: id_estadistica
 DF: id_estadistica → n_muertes
 Esta en 4FN ya que su PK se relaciona solo con 1 atributo.
- **Partido (id_partido, equipo1, equipo2, tiempo_partida, resultado, rut_arbitro, id_estadistica)**
 PK: id_partido
 FK: rut_arbitro → Arbitro(rut)
 equipo1 → Equipo(nombre_equipo)
 equipo2 → Equipo(nombre_equipo)
 DF: id_partido → tiempo_partida
 id_partido → resultado
 Esta en 4FN ya que su PK se relaciona solo con 1 atributo.

- **Posee (matricula, id_partido, id_estadistica)**
FK: id_estadistica → Estadistica (id_estadistica)
id_partido → Partido (id_partido)
matricula → Jugador (matricula)
DF: matricula → id_partido
matricula → id_estadistica
id_partido → matricula
id_partido → id_estadistica
id_estadistica → matricula
id_estadistica → id_partido

Link a código de la base de datos
BD_HITO2

Plan de Trabajo

ID	Semana	descripción
Hito 2	9	Diseño Lógico
Avance	10	Estudio servidores
Avance	11	Implementación, Correcciones
Avance	12	Pruebas Consultas
Avance	13	Correcciones
Avance	14	Revisión final
Hito 3	15-17	Demostración implementación e interrogación

Decisiones de Diseño

A lo largo del desarrollo uno de los cambios importantes que se han realizado, es la eliminación de la relación directa entre Partido y Estadística, para generar una relación triple entre Jugador, estadística y Partido, de esta forma es factible la consulta de todas las estadísticas de los distintos jugadores participantes en el partido en cuestión, o a su vez es posible buscar directamente el partido en el cual el jugador tuvo unas estadísticas en específico.

Debido a las observaciones realizadas por el ayudante en el “hito_1”, se realizaron cambios en los atributos de: Campus (Nombre campus) → Campus (id Campus, Nombre_campus), Facultad(Nombre facultad) → Campus (id_Facultad, Nombre_facultad), Carrera (Nombre Carrera) → Campus (id Carrera, Nombre_carrera). Dejando de esta forma un “Id” (numérico) asociado para la correcta identificación. Otro cambio realizado, es la eliminación de la entidad tiempo que tenía relación con Partido, para dejar el tiempo como un atributo en la entidad Partido (Al igual que el cambio anterior, este fue realizado debido a la observación dada por el ayudante), a la vez se agregó un atributo Resultado en la misma entidad. También se corrigió el error notificado en la cardinalidad Partido → Equipo, la cual era (1,n) y paso a ser (2,2), lo cual permite un mínimo y máximo de dos equipos enfrentados en un partido en cuestión.

Restricciones de Integridad

A continuación, describiremos brevemente algunas restricciones de integridad dispuestas en nuestra base de datos.

- En la creación de la tabla Posee, añadimos una restricción de clave primaria llamada “PK_POSEE”.

```
CONSTRAINT PK_POSEE PRIMARY KEY (ID_ESTADISTICA,  
ID_PARTIDO, MATRICULA)
```

Esta restricción nos garantiza que los valores de las columnas ID_ESTADISTICA, ID_PARTIDO y MATRICULA sean únicos en cada fila de la tabla.

- En la creación de la tabla Equipo_Juego, añadimos también una restricción de clave primaria llamada “PK_Equipo_Juego”.

```
CONSTRAINT PK_EQUIPO_JUEGO PRIMARY KEY(NOMBRE_EQUIPO,  
ID_JUEGO)
```

Esta restricción nos garantiza que los valores de las columnas NOMBRE_EQUIPO y ID_JUEGO sean únicos en cada fila de la tabla.

Consultas

- Seleccionar los jugadores que tienen más de 5 kills por partida

```
SELECT J.NOMBRE
FROM JUGADOR J
INNER JOIN POSEE P ON P.MATRICULA = J.MATRICULA
INNER JOIN KILLS K ON P.ID_ESTADISTICA = K.ID_ESTADISTICA
WHERE K.N_KILLS>5;
```

```
SELECT J.NOMBRE FROM JUGADOR J INNER JOIN POSEE P ON P.MATRICULA = J.MATRICULA INNER JOIN KILLS K ON P.ID_ESTADISTICA = K.ID_ESTADISTICA WHERE K.N_KILLS>5
```

NOMBRE
Josias Alejandro Lavin Opazo
Jaime Alfonso Jara Fuentes
Carlos Martín Concha Fuentes

- Seleccionar los equipos inscritos en el juego League of Legends

```
SELECT COUNT(E.NOMBRE_EQUIPO)
FROM EQUIPO E
INNER JOIN EQUIPO_JUEGO EP ON EP.NOMBRE_EQUIPO=E.NOMBRE_EQUIPO
WHERE EP.ID_JUEGO =1;
```

```
SELECT COUNT(E.NOMBRE_EQUIPO) FROM EQUIPO E INNER JOIN EQUIPO_JUEGO EP ON EP.NOMBRE_EQUIPO=E.NOMBRE_EQUIPO WHERE EP.ID_JUEGO =1
```

COUNT(E.NOMBRE_EQUIPO)
12

Statement processed. 0.00 seconds

- Seleccionar cuantos partidos has dirigido cada arbitro

```
SELECT A.NOMBRE, COUNT(A.RUT)
FROM ARBITRO A
```



```
LEFT JOIN PARTIDO P ON A.RUT = P.RUT_ARBITRO
GROUP BY A.NOMBRE;
```

```
SELECT A.NOMBRE, COUNT(A.RUT) FROM ARBITRO A LEFT JOIN PARTIDO P ON A.RUT = P.RUT_ARBITRO GROUP BY A.NOMBRE
```

NOMBRE	COUNT(A.RUT)
Joaquin avalos	3
Valeria ramos	2
Elvis teck	5

3 rows selected. 0.00 seconds

NOMBRE	COUNT(A.RUT)
Joaquin avalos	3
Valeria ramos	2
Elvis teck	5

3 rows selected. 0.00 seconds

Conclusión

En nuestro proyecto de base de datos para un campeonato de League of Legends en los campus de la Universidad de Concepción, hemos realizado un análisis exhaustivo de los requisitos y desafíos clave. Hemos definido una estructura sólida de la base de datos que incluye tablas importantes como Equipos, Jugadores, Partidos, Resultados, Estadísticas y Campeonatos, utilizando SQL con claves primarias y foráneas para garantizar un diseño eficiente.

Hemos identificado los requisitos de los usuarios finales, lo que nos permite comprender las necesidades de administradores, analistas y usuarios estándar, así como las operaciones y consultas necesarias para satisfacer esas necesidades. También hemos considerado los desafíos de control de concurrencia, seguridad, optimización de consultas y diseño de la interfaz de usuario, que serán cruciales en la implementación final.

Entre los aspectos desafiantes que nos vimos enfrentados a esta fase del proyecto fue corregir el MER que teníamos del hito 1, como la relación Posee entre Partido y Jugador. Además de crear la base de datos como tal, creando todas las tablas necesarias y agregando los datos correspondientes a cada tabla. También, desarrollamos expertiz en la normalización del MER, comprendiendo de mejor manera su correcta interpretación e importancia de normalizar el MER.

Por otro lado, lo que nos quedaría por hacer sería agregar mayor volumen de datos para lograr otros tipos de consultas y crear la interfaz correspondiente a nuestra base de datos.

En resumen, esta fase de planificación nos ha proporcionado una base sólida y una visión clara de cómo abordar los desafíos para desarrollar una base de datos efectiva que atienda las demandas de usuarios y aficionados de los campeonatos de League of Legends. A medida que avanzamos en la implementación, estamos comprometidos a superar estos desafíos y entregar un sistema que cumpla con las expectativas y necesidades de todos los involucrados.

Bibliografía

- LoL Esports.* (s/f). Lolesports.com. Recuperado el 12 de septiembre de 2023, de <https://lolesports.com/schedule?leagues=worlds,wqs,lec>
- Funciones.* (s/f). Oracle.com. Recuperado el 9 de septiembre de 2023, de <https://docs.oracle.com/cloud/help/es/analytics-cloud/ACSMD/GUID-4CBCE8D4-CF17-43BD-AAEF-C5D614A8040A.htm>