



UNIVERSIDADE FEDERAL DE SANTA MARIA
ENGENHARIA DA COMPUTAÇÃO

Alex Mortari, André Palharini, Gabriel Fronza Schuster, João Pedro Souto

TG1
Ciência de Dados

Santa Maria
2024.

1. INTRODUÇÃO

O relatório apresentado tem como objetivo descrever o processo de desenvolvimento de um modelo de banco de dados relacional para o sistema de análise de partidas focado em campeonatos anuais. A problemática é criar um sistema que permite armazenar informações e estatísticas das diferentes partidas jogadas em um campeonato de League of Legends que ocorra de forma anual, como os universitários, podendo ser utilizado amplamente pela comunidade atendendo a necessidade de que campeonatos não oficiais possam haver formas de analisar o desempenho dos jogadores e equipes para que seja possível utilizar a estratégia também, fora do jogo para alcançar melhores resultados.

2. DESCRIÇÃO DO PROBLEMA DE MODELAGEM

O problema de modelagem proposto envolve o desenvolvimento de um sistema relacional capaz de armazenar, organizar e fornecer acesso eficiente a informações e estatísticas de partidas em campeonatos de League of Legends. Esses torneios podem ser aplicados em diferentes contextos, como competições universitárias, ligas amadoras ou até eventos oficiais maiores, envolvendo múltiplas equipes e jogadores. Com a crescente popularidade dos esportes eletrônicos (e-sports), torna-se essencial o uso de sistemas que possam registrar e analisar os dados gerados por cada partida de maneira eficaz.

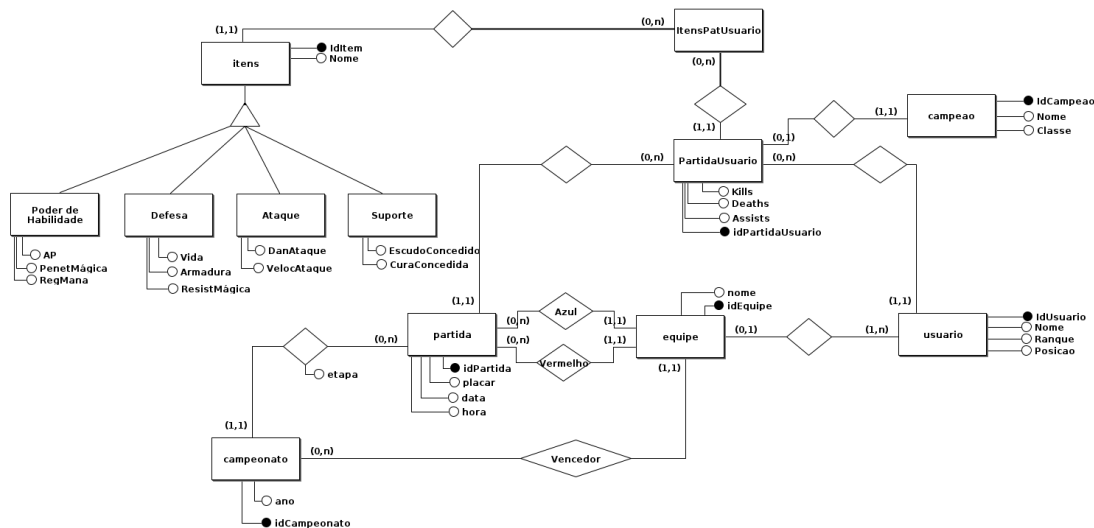
Os campeonatos de League of Legends geralmente consistem em uma série de partidas, organizadas em etapas como fase de grupos, eliminatórias e finais, envolvendo diversas equipes, com um grande volume de dados gerados a cada jogo. Esses dados incluem informações como número de abates, assistências, mortes, estruturas destruídas, tempo de jogo, itens adquiridos, entre outros, tanto no nível individual (jogadores) quanto no coletivo (equipes). Além disso, informações de contexto, como data, também são cruciais para uma análise mais completa.

A necessidade de uma modelagem dos dados surge da complexidade e do volume de informações que precisam ser armazenadas de forma organizada, para que possam ser consultadas e analisadas posteriormente. O objetivo da modelagem é garantir que o sistema seja eficiente no registro dessas estatísticas e que possibilite consultas rápidas e precisas, seja para análise de desempenho de equipes e jogadores, monitoramento de estatísticas ao longo do campeonato, ou até mesmo para fornecer relatórios completos de desempenho ao fim de cada evento.

Diante disso, a criação de um modelo relacional é a solução proposta para lidar com essa complexidade. Um banco de dados bem estruturado permitirá o armazenamento de dados de forma normalizada, garantindo integridade e consistência das informações. Esse modelo também será capaz de lidar com o relacionamento entre as várias entidades envolvidas, como equipes, jogadores, partidas e campeonatos, facilitando o cruzamento de informações e a criação de estatísticas detalhadas a partir dos dados coletados.

3. DIAGRAMA DE ENTIDADE-RELACIONAMENTO

Figura 1: Modelo DER



Sobre as entidades do diagrama, podemos ressaltar:

- **Usuário (usuario):** contém os atributos *IdUsuario*, *Nome*, *Ranque*, e *Posicao*. Relaciona-se com a entidade **campeao** (1,n), pois um usuário pode escolher vários campeões.
- **Campeão (campeao):** possui os atributos *IdCampeao*, *Nome*, e *Classe*.
- **Partida (partida):** armazena informações como *idPartida*, *placar*, *data*, e *hora*. Está ligada a duas equipes: **Azul** e **Vermelho**. A entidade **Vencedor** é um relacionamento que indica qual equipe venceu.
- **Equipe (equipe):** contém o *idEquipe* e o *nome*. Relaciona-se com usuários (1,n), pois uma equipe pode ter vários usuários.
- **Itens (itens):** contém o *IdItem* e *Nome* e está relacionada com subclasses: **Poder de Habilidade**, **Defesa**, **Ataque** e **Suporte**. Cada uma dessas subclasses contém atributos específicos referentes a origem do item os quais são: Itens de Ataque (*DanAtaque*, *VelocAtaque*), itens de Poder de Habilidade (*AP*, *PenetMágica*, *RegMana*), itens de Defesa (*Vida*, *Armadura*, *ResistMágica*) e itens de Suporte (*EscudoConcedido*, *CuraConcedida*)
- **Itens do usuário na partida (ItensPatUsuario):** representa um relacionamento entre itens e usuários, indicando quais itens pertencem a quais usuários.

- Partida do Usuário (PartidaUsuario): relaciona-se com a partida e o usuário, contendo atributos como Kills, Deaths, Assists, que são estatísticas de desempenho do jogador durante uma partida.
- Campeonato (campeonato): armazena o idCampeonato e o ano. Relaciona-se com partidas (1,n), sugerindo que um campeonato pode incluir várias partidas.

4. MODELO LÓGICO

A transformação para o modelo lógico é essencial para garantir que o banco de dados seja eficiente, funcional e fácil de manter durante a implementação do sistema. Abaixo temos o modelo lógico:

- partida (idPartida, placar, data, hora, etapa, idEquipeAzul, idEquipeVermelha, idCampeonato)
 - idEquipeAzul referencia equipe
 - idEquipeVermelha referencia equipe
 - idCampeonato referencia campeonato
- campeonato (idCampeonato, ano, idEquipeVencedora)
 - idEquipeVencedora referencia equipe
- equipe (idEquipe, nome)
- usuario (IdUsuario, Nome, Ranque, Posicao, idEquipe)
 - idEquipe referencia equipe
- PartidaUsuario (idPartidaUsuario, idUsuario, idPartida, Kill, Deaths, Assists, idCampeao, NomeCampeao, ClasseCampeao)
 - idUsuario referencia usuario
- idPartida referencia partida itens(idItem, Nome, AP, PenetMagica, RegMana, Vida, Armadura, ResistMagica, DanAtaque, VelocAtaque, EscudoConcedido, CuraConcedida)
 - ItensPatUsuario(idPartidaUsuario, idItem)
 - idPartidaUsuario referencia PartidaUsuario
 - idItem referencia Itens

As regras de transformação vistas em aula foram aplicadas conforme as cardinalidades:

Para cardinalidades (1,1) e (1,0) ou (1,1) e (1,1), utilizamos a fusão de tabelas. Isso significa que as duas tabelas envolvidas na relação podem ser combinadas em uma única tabela, já que a correspondência entre as entidades é de um para um.

Para cardinalidades 1-n e (1,0) - (1,0), aplicamos a adição de colunas. Cada entidade pode ter uma referência direta para outra por meio de uma chave

estrangeira (ou coluna que armazena a referência à outra tabela), sem a necessidade de criar uma tabela intermediária.

Para cardinalidades (n-n), criamos uma tabela própria para representar a relação. Essa tabela contém duas colunas principais, cada uma armazenando a chave estrangeira de uma das entidades envolvidas.

5. CONCLUSÃO

O desenvolvimento do modelo de banco de dados relacional para análise de partidas de campeonatos de League of Legends tem como proposta uma solução eficiente para organizar e armazenar as informações desses eventos. A modelagem proposta garante que dados como desempenho dos jogadores, estatísticas das partidas, itens utilizados, e o andamento dos campeonatos sejam registrados de maneira estruturada e de fácil acesso. Através da normalização das entidades e seus relacionamentos, o sistema oferece flexibilidade para diferentes tipos de consultas, permitindo a análise detalhada de equipes e jogadores ao longo do torneio.

6. BIBLIOGRAFIA

Heuser. C.A. *Projeto de banco de dados*. 2009. Editora Bookman.