

Casino Online



Base de Dados

Grupo 504

Ana Raquel Rodrigues dos Reis up201905529
João Tomás Mata Cardoso up202004587
Linda Inês de Pina Marques Rodrigues up202005545

Novembro, 2021

Índice

Índice	2
1.Contexto	3
2.Diagrama UML (revisto)	4
3.Esquema Relacional (revisto)	5
4. Análise dependências funcionais e formas normais (revisto)	6
5.Restrições(revisto)	8
6.Lista de Interrogações	11
7.Gatilhos	12
8.AUTOAVALIAÇÃO	12

1.Contexto

Um casino online pretende armazenar os serviços que a sua plataforma digital oferece numa base de dados. Tem ao seu dispor um conjunto de funcionários que irá prestar o serviço de apoio ao cliente, através da linha telefónica, correspondência electrónica ou a partir de um *live chat*. O casino necessita de recolher informação relativamente às pessoas que contribuem para o funcionamento (funcionários e jogadores), entre as quais, o nome, data de nascimento, país, morada e telefone. Além disso, sobre os jogadores é necessário obter o seu email, nome de utilizador, palavra-passe e tipo de dispositivo. É de salientar que, um funcionário não poderá ser um jogador no casino online.

A empresa necessita de se financiar recorrendo a diferentes patrocinadores, com os quais deve acordar as formalidades de patrocínio, nomeadamente, o montante. Cada patrocínio pode oferecer aos jogadores descontos para usarem no casino. Outro modo de aceder às promoções é o desempenho do apostador na plataforma.

O jogador ao aceder pela primeira vez ao casino online efetua um depósito, obtendo a equivalência na moeda oficial da plataforma. Pode ainda, ao longo do tempo, realizar vários depósitos para reforçar o saldo da sua conta. Este montante registado na conta, assim como, o investimento feito pela pessoa influenciam também o saldo. O último mostra, igualmente, o balanço do dinheiro ganho/perdido de todas as apostas realizadas. Quando na necessidade de levantar o dinheiro, o utilizador tem a possibilidade de fazer a conversão da moeda de *gambling* para outra à sua disposição.

As apostas dependem do tipo de jogo selecionado, mediante a última restrição, estas podem ou não possuir um valor mínimo ou máximo. Os jogos são fornecidos por múltiplos estúdios; tendo uma multiplicidade de níveis e ainda de versões de jogabilidade, isto é, se aceita o modo multiplayer em tempo real, ou se é contra um computador.

2.Diagrama UML (revisto)

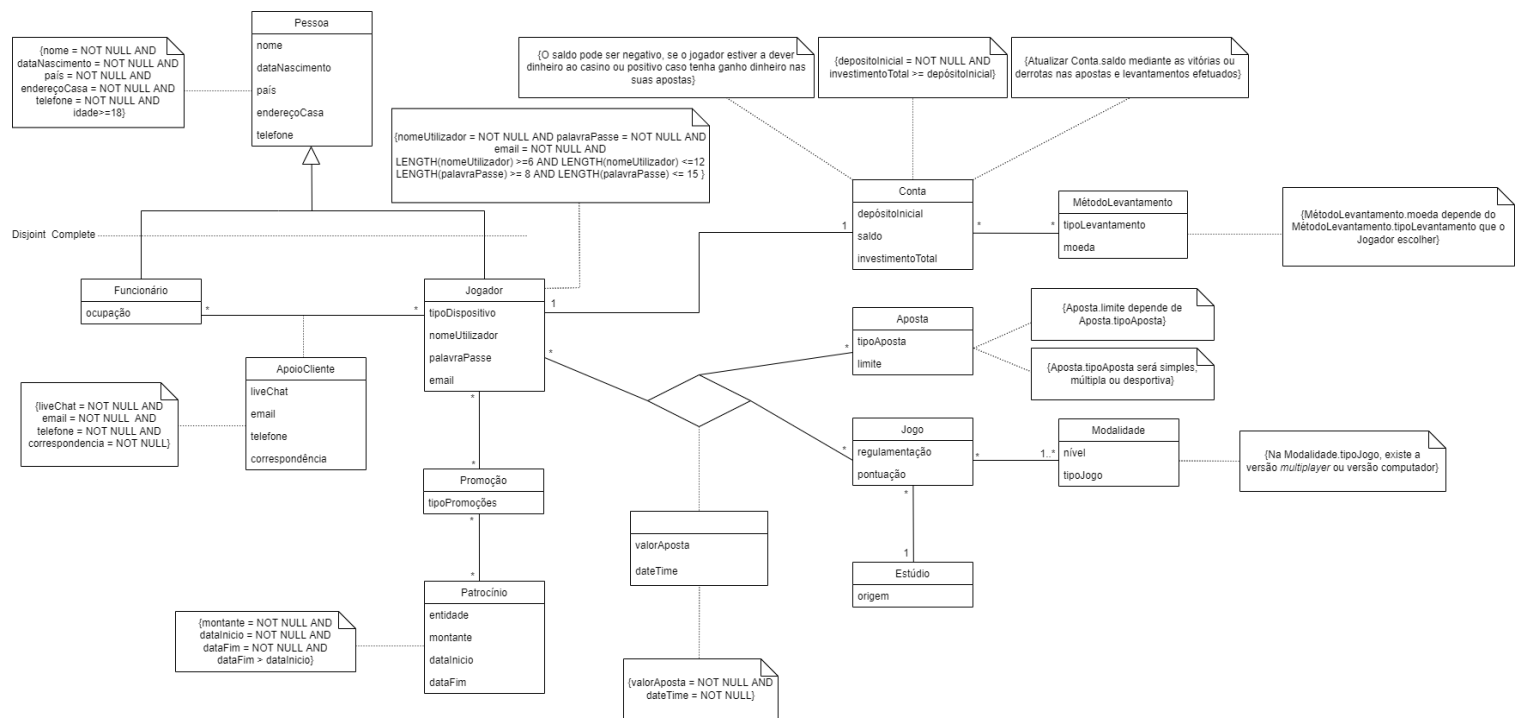


Figura 1: Diagrama UML disponível em <https://url.gratis/cYaepA>

3. Esquema Relacional (revisto)

- Pessoa(idPessoa, nome, dataNascimento, país, endereçoCasa, telefone)
- Funcionário(idPessoa->Pessoa, ocupação)
- Jogador(idPessoa->Pessoa, tipoDispositivo, nomeUtilizador, palavraPasse, email)
- Promoção(idPromoção, tipoPromoções)
- PromoçãoJogador(idJogador -> Jogador , idPromoção-> Promoção)
- Patrocínio(idPatrocínio, entidade, montante, dataInício, dataFim)
- PromoçãoPatrocínio(idPromoção-> Promoção, idPatrocínio-> Patrocínio)
- Conta(idConta, depósitoInicial, saldo, investimentoTotal, idJogador-> Jogador)
- MétodoLevantamento(idLevantamento, tipoLevantamento, moeda)
- ContaMétodoLevantamento(idConta-> Conta , idLevantamento-> MétodoLevantamento)
- Aposta(idAposta, tipoAposta, limite)
- Jogo(idJogo, regulamentação, pontuação, idEstúdio-> Estúdio)
- JogoModalidade(idJogo -> Jogo, idModalidade -> Modalidade)
- Modalidade(idModalidade, tipoJogo, nível)
- Estúdio(idEstúdio, origem)
- ApoioCliente(idFuncionário -> Funcionário, idJogador -> Jogador, liveChat, email, telefone, correspondência)
- JogadorApostaJogo(idJogador -> Jogador, idAposta-> Aposta, idJogo-> Jogo, valorAposta, dateTime)

4. Análise dependências funcionais e formas normais (revisto)

- Pessoa(idPessoa, nome, dataNascimento, país, endereçoCasa, telefone)
FDs.: idPessoa -> nome, dataNascimento, país, endereçoCasa, telefone
Formas: BCNF: Sim
3NF: Sim
- Funcionário(idPessoa->Pessoa, ocupação)
FDs.: idPessoa -> ocupação
Formais: BCNF: Sim
3NF: Sim
- Jogador(idPessoa->Pessoa, tipoDispositivo, nomeUtilizador, palavraPasse, email)
FDs.: idPessoa -> tipoDispositivo, nomeUtilizador, palavraPasse, email
email -> nomeUtilizador, palavraPasse
Formas: BCNF: Sim
3NF: Sim
- ApoioCliente(idFuncionário -> Funcionário, idJogador -> Jogador, liveChat, email, telefone, correspondência)
FDs.: idFuncionario, Jogador -> liveChat, e-mail, telefone, correspondência
Formas: BCNF: Sim
3NF: Sim
- Promoção(idPromoção, tipoPromoções)
FDs.: idPromoção -> tipoPromoções
Formas: BCNF: Sim
3NF: Sim
- PromoçãoJogador(idJogador -> Jogador, idPromoção-> Promoção)
FDs.: -
Formas: BCNF: Sim
3NF: Sim
- Patrocínio(idPatrocínio, entidade, montante, dataInício, dataFim)
FDs.: idPatrocínio -> entidade, montante, dataInício, dataFim
Formas: BCNF: Sim
3NF: Sim
- PromoçãoPatrocínio(idPromoção-> Promoção, idPatrocínio-> Patrocínio)
FDs.: -
Formas: BCNF: Sim
3NF: Sim
- Conta(idConta, depósitoInicial, saldo, investimentoTotal, idJogador-> Jogador)
FDs.: idConta -> depósitoInicial, saldo, investimentoTotal, idJogador
idJogador-> idConta
investimentoTotal -> saldo
Formas: BCNF: Sim
3NF: Sim
- MétodoLevantamento(idLevantamento, tipoLevantamento, moeda)
FDs.: idLevantamento -> tipoLevantamento, moeda
tipoLevantamento -> moeda

- Formas: BCNF: Sim
3NF: Sim
- Aposta(idAposta, tipoAposta, limite)
FDs.: idAposta → tipoAposta, limite
tipoAposta → limite
Formas: BCNF: Sim
3NF: Sim
- Jogo(idJogo, regulamentação, pontuação, idEstúdio → Estúdio)
FDs.: idJogo → regulamentação, pontuação, idEstúdio
Formas: BCNF: Sim
3NF: Sim
- JogadorApostaJogo(idJogador → Jogador, idAposta → Aposta, idJogo → Jogo, valorAposta, dateTime)
FDs.: idJogador, idAposta, idJogo → valorAposta, dateTime
Formas: BCNF: Sim
3NF: Sim
- Modalidade(idModalidade, tipoJogo, nível)
FDs.: idModalidade → tipoJogo, nível
tipoJogo → nível
Formas: BCNF: Sim
3NF: Sim
- JogoModalidade(idJogo → Jogo, idModalidade → Modalidade)
FDs.: -
Formas: BCNF: Sim
3NF: Sim
- Estúdio(idEstúdio, origem)
FDs.: idEstúdio → origem
Formas: BCNF: Sim
3NF: Sim

NOTA : Na classe MétodoLevantamento é considerado que o tipo de moeda é selecionado consoante o tipo de levantamento escolhido pelo jogador.

Considerando as relações estabelecidas, todas seguem a Forma Normal Boyce Codd(BCNF) ,assim como, a 3ª Forma Normal (3NF).

- Uma relação está na BCNF se $X \rightarrow Y$, na qual X é uma (super)key;
- Uma relação está na 3NF caso $X \rightarrow Y$ seja uma (super)key ou Y consiste em atributos primos, isto é, atributos que pertencem a pelo menos uma chave da relação.

5. Restrições (revisto)

- Pessoa {nome = NOT NULL, dataNascimento = NOT NULL, país = NOT NULL, endereçoCasa = NOT NULL, telefone = NOT NULL}
 - A. IdPessoa - PRIMARY KEY: não pode haver duas pessoas com o mesmo ID.
 - B. NOT NULL, uma vez que estes atributos estão obrigatoriamente associados a cada pessoa.
 - C. idade ≥ 18 - para frequentar ou trabalhar no casino é, legalmente, necessário ter pelo menos 18 anos.
- Funcionário
 - A. idPessoa PRIMARY KEY REFERENCES Pessoa(idPessoa): não pode haver dois funcionários com o mesmo ID, sendo que este corresponde ao ID de uma pessoa que pertence à tabela Pessoa.
- Jogador {nomeUtilizador = NOT NULL, palavraPasse = NOT NULL, email = NOT NULL, $6 \leq \text{LENGTH}(\text{nomeUtilizador}) \leq 12$, $8 \leq \text{LENGTH}(\text{palavra Passe}) \leq 15$ }
 - A. idPessoa PRIMARY KEY REFERENCES Pessoa(idPessoa): não pode haver dois jogadores com o mesmo ID, sendo que este corresponde ao ID de uma pessoa que pertence à tabela Pessoa.
 - B. NOT NULL, uma vez que estes atributos estão necessariamente associados a cada Jogador.
 - C. O nomeUtilizador apenas aceita entre 6 a 12 caracteres.
 - D. A palavraPasse apenas aceita entre 8 a 15 caracteres.
 - E. tipoDispositivo CHECK (tipoDispositivo = 'telemovel' OR tipoDispositivo = 'computador' OR tipoDispositivo = 'tablet') - O tipo de dispositivo a ser utilizado pelo jogador pode ser telemóvel, computador ou tablet.
- Promoção
 - A. idPromocao - PRIMARY KEY: não pode haver duas promoções com o mesmo ID.
 - B. tipoPromocoes CHECK (tipoPromocoes = 'codigoPromocao' OR tipoPromocoes = 'boasVindas') - Os jogadores podem usufruir de um código promocional ou um bónus de boas vindas.
- PromoçãoJogador
 - A. (idPromocao, idJogador) - PRIMARY KEY : não pode haver duas instâncias desta classe iguais.
 - B. idJogador REFERENCES Jogador(idPessoa), o ID de jogador deve corresponder a um ID da tabela Jogador.
 - C. idPromocao REFERENCES Promocao(idPromocao), o ID de promoção deve corresponder a um ID da tabela Promoção
- Patrocínio {montante = NOT NULL, dataInicio = NOT NULL, dataFim = NOT NULL, dataFim > dataInicio}

- A. NOT NULL, uma vez que os atributos indicados estão necessariamente associados a cada patrocínio.
 - B. CHECK (dataFim > dataInicio) a data final do contrato entre o patrocínio e o casino tem que ser superior à data inicial.
- PromoçãoPatrocínio
 - A. (idPromocao, idPatrocinio) - PRIMARY KEY : não pode haver duas instâncias desta tabela iguais.
 - B. idPromocao REFERENCES Promocao(idPromocao), o ID de promoção deve corresponder a um ID da tabela Promoção.
 - C. idPatrocinio REFERENCES Patrocinio(idPatrocinio), o ID de patrocínio deve corresponder a um ID da tabela Patrocínio.
- Conta {depositoInicial = NOT NULL, investimentoTotal >= depósitoInicial}
 - A. idConta - PRIMARY KEY: não pode haver duas contas com o mesmo ID.
 - B. idJogador REFERENCES Jogador(idPessoa), o ID de jogador deve corresponder a um ID da tabela Jogador.
 - C. CHECK (investimentoTotal >= depositoInicial) o InvestimentoTotal tem que ser maior ou igual ao depósitoInicial.
- MétodoLevantamento
 - A. idLevantamento - PRIMARY KEY: não pode haver dois métodos de levantamento com o mesmo ID.
 - B. moeda CHECK (moeda = 'euro' OR moeda = 'bitcoin')
- ContaMétodoLevantamento
 - A. (idConta, idLevantamento) - PRIMARY KEY: não pode haver duas instâncias desta classe iguais.
 - B. idConta REFERENCES Conta(idConta), o ID de conta deve corresponder a um ID da tabela Conta.
 - C. idLevantamento REFERENCES MetodoLevantamento(idLevantamento), o ID de levantamento deve corresponder a um ID da tabela de MétodoLevantamento.
- Aposta {Aposta.limite depende da Aposta.tipoAposta
Aposta.tipoAposta será simples, múltipla ou desportiva}
 - A. idAposta - PRIMARY KEY: não pode haver duas apostas com o mesmo ID.
 - B. O limite de uma aposta vai sempre depender do tipoAposta que o jogador escolher.
 - C. tipoAposta CHECK (tipoAposta = 'simples' OR tipoAposta = 'multipla' OR tipoAposta = 'desportiva').
- Estúdio
 - A. idEstudio - PRIMARY KEY: não pode haver dois estúdios com o mesmo ID.
- Jogo

- A. idJogo - PRIMARY KEY: não pode haver dois jogos com o mesmo ID.
 - B. idEstudio REFERENCES Estudio(idEstudio) o ID de estúdio deve corresponder a um ID da tabela Estúdio.
- Modalidade {Na Modalidade.tipoJogo existe a versão *multiplayer* ou versão computador}
 - A. idModalidade - PRIMARY KEY: não pode haver duas modalidades com o mesmo ID.
 - B. tipoJogo CHECK (tipoJogo = 'multiplayer' OR tipoJogo = 'computador').
- JogoModalidade
 - A. (idJogo,idModalidade) - PRIMARY KEY: não pode haver duas instâncias desta tabela com o mesmo par.
 - B. idJogo REFERENCES Jogo(idJogo) o ID de jogo deve corresponder a um ID da tabela Jogo.
 - C. idModalidade REFERENCES Modalidade(idModalidade) o ID de modalidade deve corresponder a um ID da tabela Modalidade.
 - D. Cada jogo tem que ter pelo menos uma Modalidade.
- ApoioCliente {liveChat = NOT NULL, email = NOT NULL, telefone = NOT NULL, correspondencia = NOT NULL}
 - A. (idFuncionario, idJogador) - PRIMARY KEY: não pode haver duas instâncias com o mesmo par.
 - B. idFuncionario REFERENCES Funcionario(idPessoa) o ID de funcionário deve corresponder a um ID da tabela Funcionário.
 - C. idJogador REFERENCES Jogador(idPessoa) o ID jogador deve corresponder a um ID da tabela Jogador.
 - D. NOT NULL, uma vez que os atributos indicados estão necessariamente associados à tabela ApoioCliente.
- JogadorApostaJogo {valorAposta = NOT NULL, dateTime = NOT NULL}
 - A. (idJogador,idAposta,idJogo) - PRIMARY KEY: não pode haver duas instâncias formadas pela n-ary iguais.
 - B. NOT NULL, uma vez que os atributos indicados estão necessariamente associados à tabela JogadorApostaJogo.

6. Lista de Interrogações

1. Tipos de levantamentos de moeda mais populares na plataforma online de apostas;

```
SELECT MetodoLevantamento.tipoLevantamento
FROM ContaMetodoLevantamento, MetodoLevantamento
WHERE ContaMetodoLevantamento.idLevantamento =
MetodoLevantamento.idLevantamento
GROUP BY MetodoLevantamento.tipoLevantamento
HAVING count(*) >= 2;
```

2. Listar nomes dos jogadores, por ordem alfabética, com apostas abaixo da média ;

```
SELECT DISTINCT Pessoa.nome
FROM JogadorApostaJogo, Pessoa
WHERE (JogadorApostaJogo.valorAposta) > (Select
avg(JogadorApostaJogo.valorAposta) FROM JogadorApostaJogo) AND
JogadorApostaJogo.idJogador = Pessoa.idPessoa
ORDER BY Pessoa.nome ASC;
```

3. Nomes dos utilizadores com acesso à promoção dos 'Bons vícios' e que não recorrem ao apoio ao cliente ;

```
SELECT distinct Jogador.nomeUtilizador
FROM Jogador, PromocaoJogador , Promocao
WHERE PromocaoJogador.idPromocao = 80 AND PromocaoJogador.idJogador =
Jogador.idPessoa
EXCEPT
SELECT Jogador.nomeUtilizador
FROM ApoioCliente, Jogador
WHERE ApoioCliente.idJogador = Jogador.idPessoa;
```

4. ID do utilizador com maior número de apostas e o respectivo número de apostas que seja de Espanha;

```
SELECT JogadorApostaJogo.idJogador , count(*) as nr_apostas
FROM JogadorApostaJogo
GROUP BY JogadorApostaJogo.idJogador
ORDER BY nr_apostas DESC limit 1;
```

5. TOP 5 de dias com menor número de apostas e o respectivo tipo;

```
SELECT dateTime, count(*) as nr_apostas, tipoAposta
FROM JogadorApostaJogo
INNER JOIN Aposta
ON JogadorApostaJogo.idAposta = Aposta.idAposta
GROUP BY JogadorApostaJogo.dateTime
ORDER BY jogadas ASC limit 5;
```

6. TOP 3 de Jogadores com o maior investimento total e o respectivo saldo;

```
SELECT Pessoa.nome, Conta.investimentoTotal, Conta.saldo
FROM Pessoa, Conta
WHERE Conta.idJogador = Pessoa.idPessoa
ORDER BY Conta.investimentoTotal desc limit 3;
```

7. Tipos de jogos com maior número de apostadores;

```
SELECT (Modalidade.tipoJogo) ,count(*) as nr_jogadas
FROM Modalidade
LEFT JOIN JogoModalidade on Modalidade.idModalidade =
JogoModalidade.idModalidade
LEFT JOIN JogadorApostaJogo ON JogoModalidade.idJogo =
JogadorApostaJogo.idJogo
```

```
GROUP BY Modalidade.tipoJogo
ORDER BY nr_jogadas desc limit 1;
```

8. Jogadores que tentaram ultrapassar o limite de apostas imposta pela regulamentação do tipo de jogo;

```
SELECT DISTINCT Pessoa.nome, JogadorApostaJogo.valorAposta
FROM Pessoa JOIN JogadorApostaJogo
ON Pessoa.idPessoa = JogadorApostaJogo.idJogador
WHERE JogadorApostaJogo.idJogo = 13 AND JogadorApostaJogo.valorAposta >
200;
```

9. Qual o segundo patrocinador que oferece menos dinheiro ao casino online e quanto tempo falta para acabar o contrato que assinou;

```
SELECT entidade, montante, julianday(dataFim)-julianday(dataInicio)
'Days'
FROM Patrocinio
WHERE montante = (SELECT min(montante) FROM PATROCINIO WHERE montante NOT
IN (SELECT min(montante) FROM Patrocinio));
```

10. ID's de jogadores que jogam Barbie Blackjack.

```
SELECT DISTINCT idJogador
FROM JogadorApostaJogo
WHERE JogadorApostaJogo.idJogo = '12'
ORDER BY JogadorApostaJogo.idJogo ASC;
```

7. Gatilhos

Gatilho (1)

- Emissão de mensagem de erro ,sinalizando o jogador que tentou ultrapassar o valor máximo imposto pela regulamentação do jogo com id= '13';

Gatilho (2)

- Ao inserir um jogo tem que se inserir também uma modalidade, da mesma forma, eliminando qualquer dado assimilado na tabela Jogo e na tabela JogoModalidade;

Gatilho (3)

- Caso o jogador tenha saldo com uma quantia inferior a -300 euros ou mais, a sua conta é suspensa, podendo regressar assim que pagar a quantia em falta.

Nota: (1) É de salientar que os dados presentes na base de dados foram ligeiramente alterados em relação à entrega nº2 para otimização dos resultados;

(2) Relativamente à 3ª Entrega ,era pedido que fossem estabelecidos 3 triggers,no entanto, ao implementá-los para certas funcionalidades, gatilho nº2 , não foi possível satisfazer a condição estabelecida num só único trigger, pelo que se teve de recorrer a mais que um tipo, que se encontram enumerados de seguida:

- Caso o id da Modalidade fosse menor que 14 , a modalidade adquirida era o modo 'multiplayer';
- Por outro lado se fosse maior , a modalidade escolhida seria o modo ' computador'.
- Por fim, recorria-se a um *DELETE* para restaurar a coesão dos dados, ou seja, para que todos os dados estivessem em sintonia com os regulamentos de cada jogo.

8.AUTOAVALIAÇÃO

Ana Raquel Reis 47.5 %

João Mata 5 %

Linda Inês Rodrigues 47.5 %