Trabalho Prático

Projeto Detalhado de Software 2022.1 - UFC Quixadá

API LocaGames 4all

Univesidade Federal do Ceará - Campus Quixadá
Disciplina: Projeto Detalhado de Software
Professor: Camilo Camilo

Repositório: https://github.com/DiFeitoza/locagames4all

Entrega 01 - 20/06/2022

Equipe:

- Ayrton Sousa - mat. 408377
- Diego Feitoza - mat. 412981
- Guilherme Willian - mat. 475980
- Stherfany Alves - mat. 417205

Sumário

Descrição do sistema

A API consiste em um sistema de locação de jogos de mesa (e seus diversos tipos) para clientes cadastrados. Um estabelecimento manipula o cadastro de jogos e de clientes e pode realizar operações de locação. Cada cliente possui uma carteira de saldo e pode locar diversos jogos. Em caso de atrasos ou devolução antecipada o saldo é contabilizado na carteira, podendo ser consultado a qualquer tempo. O sistema também permite consultas diversas sobre cada locação, cliente ou jogo.

Requisitos funcionais

Listagem

[RF01] – Cadastrar cliente
[RF02] – Excluir cadastro de cliente
[RF03] – Atualizar cadastro de cliente
[RF04] – Buscar cliente por CPF
[RF05] – Cadastrar jogo
[RF06] – Excluir cadastro de jogo
[RF07] – Atualizar cadastro de jogo
[RF08] – Buscar jogo por nome
[RF09] – Buscar jogo por status
[RF10] – Buscar jogos por quantidade de jogadores
[RF11] – Buscar jogos por valor de diária
[RF12] – Buscar jogos por tipo
[RF13] – Favoritar um jogo
[RF14] – Consultar jogos favoritos
[RF15] – Desfavoritar um jogo
[RF16] – Efetuar locação
[RF17] – Buscar locações por jogo
[RF18] – Buscar histórico de locações por cliente
[RF19] – Buscar locações em aberto por cliente
[RF20] – Buscar locação por cliente e jogo
[RF21] – Consultar saldo de locação
[RF22] – Efetuar devolução
[RF23] – Excluir locação

Requisitos detalhados

Identificação:	**[RF01] – Cadastrar cliente**		
Descrição:	O sistema deve permitir que um novo cliente seja cadastrado.		
Detalhamento:	O operador do sistema deve informar nome, CPF, endereço, telefone e email do cliente. O sistema irá gerar um id único para cada novo cliente.		
Endpoint:			
Retorno:	Retorna o cliente cadastrado.		
Regra de negócio:	Só é permitido um único cadastro por CPF. O CPF deve ser um número válido. O telefone deve permitir o DDD e o 9º dígito do celular. O email deve ser válido. Quando um cliente for cadastrado e seu CPF existir no banco de dados em um cadastro com atributo "excluído" como “true”, então o cadastro existente é reaproveitado no mesmo identificador e os demais dados são atualizados.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF02] – Excluir cadastro de cliente**		
Descrição:	O sistema deve permitir que um cliente cadastrado seja excluído do sistema.		
Detalhamento:	O operador do sistema deve informar um CPF válido de um cliente cadastrado para que seja excluído.		
Endpoint:			
Retorno:	Retorna a confirmação de que o usuário foi excluído do sistema.		
Regra de negócio:	Um cadastro de cliente não pode ser excluído se ele tiver alguma locação em aberto. Ao excluir um cadastro ele se torna inativo: o atributo "excluído" recebe o valor “true”, os dados dos atributos são eliminados, exceto o atributo clienteId e o CPF, pois são necessários para manter os dados de locação. Quando um cadastro de cliente é excluído seus dados de favoritos devem ser eliminados do banco de dados. Em caso de recadastramento de um usuário excluído, o cadastro se torna ativo, o atributo "excluído" recebe o valor “false” e é atualizado com os novos dados.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF03] – Atualizar cadastro de cliente**

Descrição:	O sistema deve permitir que os dados do cliente cadastrado possam ser atualizados.		
Detalhamento:	O operador do sistema deve informar o CPF do cliente, seguido dos novos valores dos atributos que pretende atualizar.		
Endpoint			
Retorno:	Retorna os dados do cliente, com atualização.		
Regra de negócio:	O CPF deve ser válido e o cliente deve estar cadastrado. O telefone deve permitir o DDD e o 9º dígito do celular. O email deve ser válido.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF04] – Buscar cliente por CPF**		
Descrição:	O sistema deve permitir a consulta de cliente pelo CPF cadastrado.		
Detalhamento:	O operador do sistema deve informar um CPF válido para busca de um cliente cadastrado.		
Endpoint			
Retorno:	O sistema retorna os dados do cliente encontrado.		
Regra de negócio:	O CPF deve ser válido e estar cadastrado no sistema.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF05] – Cadastrar jogo**
Descrição:	O sistema deve permitir que um novo jogo seja cadastrado.
Detalhamento:	O operador do sistema deve informar nome, tipo, status, descrição, valorDiaria, qtdMinJogadores e qtdMaxJogadores. O sistema irá gerar um identificador único para o novo jogo cadastrado e irá preencher o atributo “status” como “disponível”.
Endpoint	
Retorno:	Retorna os dados do jogo cadastrado.
Regra de negócio:	Cada jogo possui identificador único, os demais dados podem ser iguais. Logo, um mesmo jogo pode ser cadastrado mais vezes com identificadores diferentes caso haja mais de uma unidade. O tipo de um jogo pode assumir apenas valores pré-definidos, como p.e. “tabuleiro”, “cartas”, “RPG”. Além do status “disponível”, um jogo pode assumir os valores “alugado” ou “indisponível”.

	O status indisponível é a possibilidade de impedir momentaneamente o aluguel de um jogo, sem que precise excluí-lo. O valor da diária é expresso em reais e centavos e não pode ser negativo. A quantidade mínima de jogadores é um jogador. O jogo também possui o atributo booleano “excluido”, para quando um jogo for excluído ele receber o valor “true” em vez de apagá-lo do banco de dados.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF06] – Excluir cadastro de jogo**		
Descrição:	O sistema deve permitir que um jogo seja excluído.		
Detalhamento:	O operador do sistema deve informar o identificador de um jogo para que seja excluído.		
Endpoint:			
Retorno:	Retorna a confirmação de que o jogo foi excluído do sistema.		
Regra de negócio:	O jogo se torna inativo, o atributo "excluído" recebe o valor “true”, porém seus dados continuam no banco de dados para fins de registro das transações de locação. Dados acessórios podem ser excluídos, ficando apenas os essenciais para fim de registro das transações. Quando um jogo for excluído ele deve sair dos favoritos de clientes cadastrados.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF07] – Atualizar cadastro de jogo**		
Descrição:	O sistema deve permitir que os dados do jogo cadastrado possam ser atualizados.		
Detalhamento:	O operador do sistema deve informar o identificador do jogo, seguido dos novos valores dos atributos que pretende atualizar.		
Endpoint:			
Retorno:	Retorna o jogo com os dados atualizados		
Regra de negócio:	O jogo deve ter sido cadastrado previamente, o identificador deve ser válido. O atributo “status” deve assumir um dos possíveis valores (disponível, alugado, indisponível)		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF08] – Buscar jogo por nome**

Descrição:	O sistema deve permitir a busca de jogo por nome.		
Detalhamento:	O operador do sistema deve informar um nome válido e o sistema retorna todos os jogos que possuem este nome cadastrado.		
Endpoint:			
Retorno:	Retorna a relação de jogos encontrados ou o erro, caso não seja encontrado.		
Regra de negócio:			
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF09] – Buscar jogo por status**		
Descrição:	O sistema deve permitir a busca de jogo pelo status.		
Detalhamento:	O operador do sistema deve informar o status de busca de um jogo (disponível, indisponível ou alugado) e o sistema exibe a lista de jogos com o respectivo status.		
Endpoint:			
Retorno:	Retorna a relação de jogos encontrados ou o erro, caso não seja encontrado.		
Regra de negócio:			
Prioridade:	Essencial	[x] Importante	Desejável

Identificação:	**[RF10] – Buscar jogo por quantidade de jogadores**		
Descrição:	O sistema deve permitir a busca de jogo pela quantidade de jogadores permitida.		
Detalhamento:	O operador do sistema realiza uma busca no sistema passando um intervalo possível de jogadores (mínimo e máximo) e o sistema exibe uma lista de jogos que correspondem ao resultado da busca.		
Endpoint:			
Retorno:	Retorna a relação dos jogos que estão no intervalo de valor.		
Regra de negócio:			
Prioridade:	Essencial	[x] Importante	Desejável

Identificação:	**[RF11] – Buscar jogo por valor de diária**

Descrição:	O sistema deve permitir a busca de jogo pelo valor da diária de aluguel.		
Detalhamento:	O operador do sistema realiza a busca no sistema passando o valor da diária em reais e centavos e o sistema retorna uma relação de jogos que possuem valor menor ou igual ao informado.		
Endpoint:			
Retorno:	Retorna a relação de jogos em ordem decrescente de valor, exibindo todos com valor igual ou menor ao informado.		
Regra de negócio:			
Prioridade:	Essencial	Importante	[x] Desejável

Identificação:	**[RF12] – Buscar jogo por tipo**		
Descrição:	O sistema deve permitir a busca de jogos pelo tipo.		
Detalhamento:	O operador do sistema realiza uma busca no sistema passando o tipo de jogo procurado e o sistema exibe uma lista de jogos que correspondam ao tipo informado. Por exemplo “cartas”, “tabuleiro”, “RPG”.		
Endpoint:			
Retorno:	Retorna a relação de jogos encontrados ou erro, caso não seja encontrado.		
Regra de negócio:			
Prioridade:	Essencial	[x] Importante	Desejável

Identificação:	**[RF13] – Favoritar um jogo**		
Descrição:	O sistema deve permitir que um cliente cadastrado possa favoritar jogos.		
Detalhamento:	O operador do sistema deve informar um CPF, válido e cadastrado, e o identificador de um jogo para que este par seja registrado como um favorito na tabela de JogosFavoritos do banco de dados.		
Endpoint:			
Retorno:	Retorna uma confirmação de que o jogo foi favoritado, trazendo o identificador do cliente e o identificador do jogo favoritado.		
Regra de negócio:	Se o jogo já estiver como favorito, retorna uma mensagem de erro informando que o jogo é um favorito daquele cliente.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF14] – Consultar jogos favoritos**		
Descrição:	O sistema deve permitir a listagem de todos os jogos favoritos de um cliente cadastrado.		
Detalhamento:	O operador do sistema deve informar um CPF, válido e cadastrado, e o sistema retorna uma relação dos jogos favoritos do cliente.		
Endpoint:			
Retorno:	Retorna a relação de jogos encontrados ou erro, caso não seja encontrado.		
Regra de negócio:			
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF15] – Desfavoritar um jogo**		
Descrição:	O sistema deve permitir que um cliente cadastrado possa desfavoritar um jogo da sua lista de favoritos.		
Detalhamento:	O operador do sistema deve informar um CPF, válido e cadastrado, e o identificador de um jogo. O sistema verifica se o jogo está como favorito do cliente informado e retira dos favoritos deste cliente no banco de dados. Caso não seja um favorito do cliente informado, retorna erro.		
Endpoint:			
Retorno:	Retorna a confirmação de que o jogo foi removido dos favoritos do cliente informado. Caso não seja um favorito do cliente, retorna a mensagem de que o jogo não era um favorito do cliente informado.		
Regra de negócio:			
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF16] – Efetuar locação**
Descrição:	O sistema deve permitir que um cliente cadastrado possa efetuar a locação de um jogo cadastrado.
Detalhamento:	O operador do sistema deve informar um CPF, válido e cadastrado, o identificador de um jogo cadastrado e a quantidade de dias da locação. O sistema verifica se o jogo está com situação “disponível” e efetua a locação. Caso a quantidade de dias seja menor do que dois, o sistema retorna erro, informando que a quantidade mínima é de dois dias.
Endpoint:	

Retorno:	Retorna os dados da locação gerada: identificador da locação, identificador do cliente, identificador do jogo, data de locação (atual), data de prevista para devolução, data de devolução (vazia) e o valor da diária.		
Regra de negócio:	O cliente deve estar cadastrado e não pode possuir locações em débito. Só serão realizadas as locações de jogos com situação “disponível”. O prazo mínimo previsto para devolução deve ser de dois dias. ~~As diárias são contabilizadas após o meio-dia, ou seja, se o jogo for entregue após este horário, é devida uma nova diária, considerando a devolução como o dia seguinte.~~ ~~A locação mínima é por uma diária, ainda que seja devolvido antes das 12h do dia seguinte.~~		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF17] – Buscar locações por jogo**		
Descrição:	O sistema deve permitir a busca da relação de todas as locações de um jogo.		
Detalhamento:	O operador do sistema deve informar o identificador do jogo e o sistema retorna uma relação das locações, em aberto ou concluídas, vinculadas ao respectivo identificador.		
Endpoint			
Retorno:	Retorna uma relação das locações que atendem ao critério de busca. O identificador deve ser válido.		
Regra de negócio:	Entende-se por locação em aberto aquela em que o jogo não foi devolvido.		
Prioridade:	Essencial	Importante	[x] Desejável

Identificação:	**[RF18] – Buscar histórico de locações por cliente**
Descrição:	O sistema deve permitir a busca do histórico de locações por cliente.
Detalhamento:	O operador do sistema deve informar o CPF do cliente cadastrado e o sistema exibe uma relação de todas as locações vinculadas ao CPF, sejam em aberto ou concluídas.
Endpoint	
Retorno:	Retorna a relação de todas as locações vinculadas ao CPF, sejam em aberto ou concluídas.
Regra de negócio:	A relação exibida inclui todas as locações vinculadas ao respectivo CPF, incluindo locações em aberto ou concluídas. Entende-se por locação em aberto aquela em que o jogo não foi devolvido. O CPF deve ser válido e estar cadastrado no sistema.

Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF19] – Buscar locações em aberto por cliente.**		
Descrição:	O sistema deve permitir a busca de locações em aberto de um cliente cadastrado.		
Detalhamento:	O operador do sistema deve informar o CPF, válido e cadastrado, e o sistema exibe uma relação das locações vinculadas ao referido que ainda não tiveram a devolução registrada no sistema.		
Endpoint:			
Retorno:	Retorna uma relação das locações vinculadas ao respectivo CPF que ainda não tiveram a devolução registrada no sistema.		
Regra de negócio:	O CPF deve ser válido e estar cadastrado no sistema. Entende-se por locação em aberto aquela em que o jogo não foi devolvido.		
Prioridade:	Essencial	[x] Importante	Desejável

Identificação:	**[RF20] – Buscar locações por cliente e jogo**		
Descrição:	O sistema deve permitir a consulta de locações de um jogo por um cliente específico.		
Detalhamento:	O operador do sistema deve informar o CPF do cliente e o identificador do jogo para o sistema verificar todas as locações deste jogo pelo cliente informado.		
Endpoint			
Retorno:	Retorna uma relação de todas as locações vinculadas ao respectivo CPF e jogo, incluindo locações em aberto e concluídas.		
Regra de negócio:	O CPF deve ser válido e estar cadastrado no sistema. Entende-se por locação em aberto aquela em que o jogo não foi devolvido..		
Prioridade:	Essencial	[x] Importante	Desejável

Identificação:	**[RF21] – Consultar saldo de locação**
Descrição:	O sistema deve permitir a consulta de saldo de um locação.
Detalhamento:	O operador do sistema informa o CPF de um cliente cadastrado, o identificador do jogo locado e o sistema retorna os dados da locação em aberto para esse par de informações, incluindo o saldo (positivo ou negativo) relativo a esta locação.
Endpoint:	

Retorno:	Retorna os dados da locação em aberto para esse par cliente e jogo informados, incluindo o saldo (positivo ou negativo) relativo a esta locação.		
Regra de negócio:	O saldo da locação é a diferença entre a data de devolução (DD) e a data prevista para devolução (DP) multiplicado pelo valor da diária da locação. Saldo = (-DD + DP) * VD.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF22] – Efetuar devolução**		
Descrição:	O sistema deve permitir a devolução de um jogo locado.		
Detalhamento:	O operador do sistema deve informar o CPF do cliente e o identificador do jogo a ser devolvido. O sistema verifica se os dados estão corretos e correspondem a uma locação em aberto entre os respectivos cliente e jogo, registra a data de entrega e retorna o resultado da devolução.		
Endpoint:			
Retorno:	O sistema retorna a locação com a data de devolução preenchida e o valor de saldo (positivo ou negativo).		
Regra de negócio:	O saldo da devolução de locação é a diferença entre a data de devolução (DD) e a data prevista para devolução (DP) multiplicado pelo valor da diária (VD) da locação. Saldo = (-DD + DP) * VD.		
Prioridade:	[x] Essencial	Importante	Desejável

Identificação:	**[RF23] – Excluir locação**		
Descrição:	O sistema deve permitir a exclusão de uma locação.		
Detalhamento:	O operador do sistema informa o identificador de uma locação e o sistema exclui a locação do banco de dados.		
Endpoint:			
Retorno:	Retorna a confirmação de exclusão da locação ou uma mensagem de erro.		
Regra de negócio:	Uma locação só pode ser excluída se o jogo referente a ela já tiver sido devolvido.		
Prioridade:	[x] Essencial	Importante	Desejável

Especificação de Banco de Dados

Entidades

Cliente

- **clienteId**: (long, chave primária) identificador único do cliente cadastrado.
- **nome**: (string) nome completo do cliente cadastrado.
- **cpf**: (string) cpf do cliente cadastrado.
- **endereço**: (string) endereço de residência do cliente cadastrado. Formato: (logradouro, número, bairro, cidade, estado).
- **telefone**: (string) telefone principal do cliente cadastrado. Formato (p.e 85 9 9999 0000, ddd seguido do dígito adicional para celular, seguido de oito dígitos)
- **e-mail**: (string) email do cliente cadastrado.
- **excluido:** (boleano) identifica se o cadastro do cliente foi excluído do sistema.

Jogo

- **jogoId**: (long, chave primária) identificador único do jogo cadastrado.
- **nome**: (string) nome do jogo cadastrado
- **tipo**: (enum) tipos de jogos cadastrados no sistema, p.e. tabuleiro, cartas, RPG.
- **status** (enum: disponível, indisponível, alugado) identifica a situação de um jogo quanto a disponibilidade para locação.
- **descrição**: (string) descrição geral do conceito e objetivos do jogo.
- **qtdMinJogadores**: (inteiro) indica a quantidade mínima de jogadores.
- **qtdMaxJogadores**: (inteiro) indica a quantidade máxima de jogadores.
- **valorDiaria**: (float) valor da locação por dia de um jogo cadastrado.
- **excluido:** (boleano) identifica se o cadastro do jogo foi excluído do sistema.

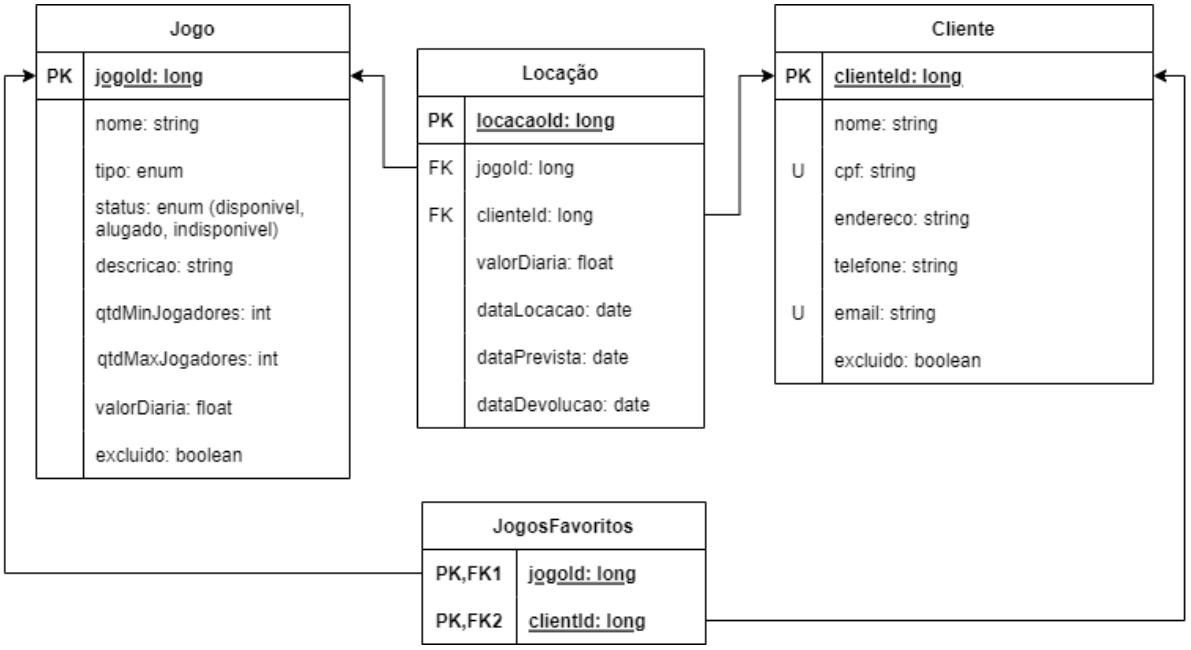
Locação

- **locacaoId**: (long, chave primária) identificador único de uma locação.
- **jogoId**: (long, chave estrangeira) identificador do jogo locado.
- **clienteId**: (long, chave estrangeira) identificador do cliente locatário.
- **valorDiaria**:(float) valor da locação por dia de um jogo cadastrado.
- **dataLocacao**: (date) identifica a data de locação do jogo.
- **dataPrevista**: (date) identifica a data prevista de devolução de um jogo locado.
- **dataDevolucao**: (date) identifica a data em que um jogo locado foi devolvido.

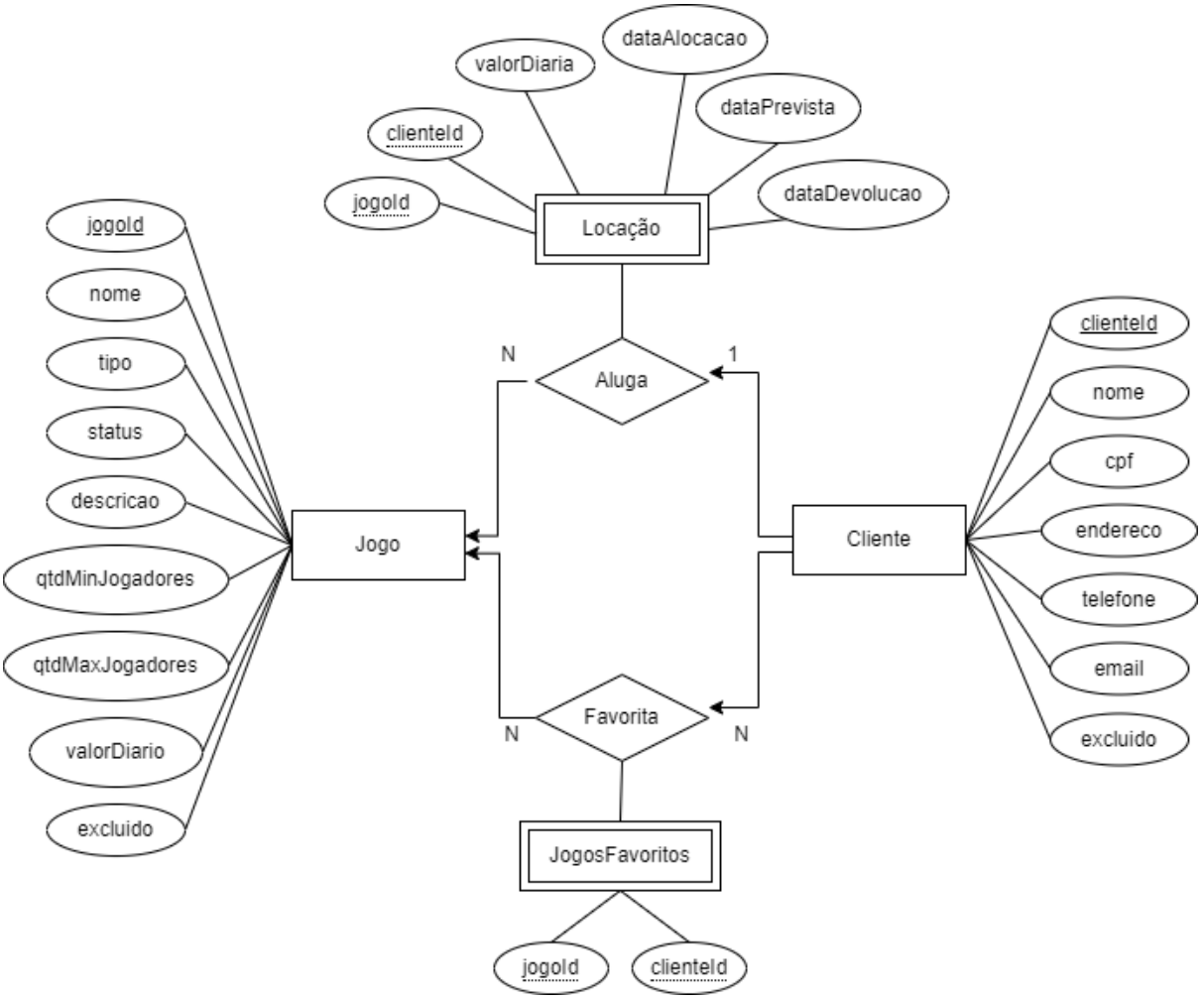
JogosFavoritos

- **jogoID**: (long, chave primária, chave estrangeira) identificador do jogo favoritado.
- **clienteID**: (long, chave primária, chave estrangeira) identificador do cliente.

Esquema Relacional do Banco de Dados



Modelo Entidade Relacionamento



Tarefas

stherfanyalves@alu.ufc.br

- ☑ ~~Elaboração dos requisitos~~ ~~7 de jun. de 2022~~
- ☑ ~~Revisão dos requisitos~~ ~~18 de jun. de 2022~~

Diego Feitoza

- ☑ ~~Elaboração da documentação e formatação~~ ~~7 de jun. de 2022~~
- ☑ ~~Elaboração do diagrama do BD~~ ~~7 de jun. de 2022~~
- ☑ ~~Revisão dos requisitos.~~ ~~16 de jun. de 2022~~
- ☑ ~~Decisões sobre as regras de negócio e mais revisão dos requisitos~~ ~~17 de jun. de 2022~~
- ☑ ~~configuração inicial do projeto (código)~~ ~~17 de jun. de 2022~~
- ☑ ~~ajustes diagramas de bd~~ ~~18 de jun. de 2022~~
- ☑ ~~Revisão dos requisitos.~~ ~~18 de jun. de 2022~~

Guilherme Willian Saraiva da Hora

- [ ] Verificar requisitos e propor alterações nos textos, conteúdo e tipo de prioridade
- [x] ~~Elaboração dos diagramas de BD~~ ~~17 de jun. de 2022~~

ec.ayrton@gmail.com

- [x] ~~decisões sobre as regras de negócio e revisão dos requisitos~~ ~~17 de jun. de 2022~~
- [x] ~~Configuração inicial do projeto (código)~~ ~~17 de jun. de 2022~~

Decisões

ec.ayrton@gmail.com

~~Como manter o log de locações após um jogo ou usuário ser excluído?~~

~~deveríamos excluir de uma base, mas manter em outra? ou apenas colocar como inativo, para não vir nas buscas? o recadastro de um cpf excluído, iria reativar o cadastro anterior com novos dados?~~

~~que dados devem ser mantidos? excluiremos os acessórios e manteríamos os principais?~~