

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA  
CURSO DE CIÊNCIAS DA COMPUTAÇÃO  
DISCIPLINA: Engenharia De Software - INE5417  
PROFESSORA: Patrícia Vilain

**PROJETO PÔQUER: SEGUNDA ITERAÇÃO**

Chrystian Guth  
Diego Gonçalves  
Gabriel Gava  
Lucas Pereira

Florianópolis  
Novembro de 2011

## **1. VISÃO**

### **1.1 Visão Geral Do Problema**

### **1.2 Requisitos Funcionais**

### **1.3 Requisitos Não Funcionais**

### **1.4 Glossário**

### **1.5 Iterações**

## **2. REQUISITOS DA 2ª ITERAÇÃO**

### **2.1 Casos De Uso**

### **2.2 Diagramas De Casos De Uso**

## **3. ANÁLISE DA 2ª ITERAÇÃO**

### **3.1 Diagrama De Classes Conceituais**

## **4. PROJETO DA 2ª ITERAÇÃO**

### **4.1 Diagramas De Iteração**

### **4.2 Diagrama De Classes De Projeto**

### **4.3 Diagrama De Classes De Projeto**

### **4.4 Diagramas De Atividades**

### **4.5 Diagramas De Statecharts**

# 1. VISÃO

## 1.1 Visão Geral Do Problema

Visamos a criação de um sistema online para o jogo de pôquer. O sistema contará com controle de usuários, torneios e mantenedores (responsáveis por criar torneios). Teremos também a implantação do próprio jogo de pôquer na modalidade No-Limit **Texas Hold'em**, já que esta se trata da variante de pôquer mais jogada atualmente.

Os sistema será baseado apenas em dinheiro fictício. Assim, ao se cadastrar o usuário ganhará uma quantia x de dinheiro fictício e poderá se inscrever nos torneios com este dinheiro. Além disso, os melhores colocados nos torneios ganham premiações (em dinheiro fictício).

## 1.2 Requisitos Funcionais

- CRUD de **Jogadores**;
- CRUD de **Mantenedores** do sistema, que terão privilégios para criar torneios. O privilégio de cadastrar, editar e deletar mantenedores será de um usuário **Administrador**. O Usuário administrador já está pré-definido no sistema e contém uma senha para acessar a área administrativa;
- CRUD de **Torneios**. Os torneios poderão ter entrada gratuita ou serem pagos (com dinheiro fictício) e deverão ser criados pelos mantenedores;
- Jogo de pôquer, baseado nas regras do Texas Hold'em No-Limit;
- Caixa do jogador que conterà informações sobre o mesmo, bem como a quantidade dinheiro fictício que ele possui;
- Área com instruções básicas sobre a modalidade Texas Hold'em.

## 1.3 Requisitos Não Funcionais

- Usabilidade: Deve ficar explicito visualmente quando for a vez de determinado jogador realizar sua jogada;
- Usabilidade: A interface deve ser simples e intuitiva, contendo apenas as

opções necessárias para o uso do sistema, sem que haja demasiada informação na tela;

- Segurança: Qualquer uso do sistema, tanto por parte dos jogadores quanto por parte dos mantenedores requer autenticação com usuário e senha;
- Segurança: Somente o administrador pré-definido no sistema pode cadastrar novos mantenedores. A senha deste mantenedor será gerada automaticamente e enviada para seu e-mail cadastrado;
- Segurança: O jogador não pode acessar o sistema em mais de um computador ao mesmo tempo;
- Performance: O tempo de resposta de alguma jogada deve ser mínimo e não pode ultrapassar de 1 segundo, contando a partir do momento em que a jogada feita pelo jogador é recebida pelo sistema;
- Suporte: O administrador deve ter uma interface gráfica onde pode configurar todas as opções de funcionamento do sistema conforme desejar. Opções como: quantia inicial de dinheiro fictício que o jogador ganhará, número máximo de torneios em que um jogador pode estar ao mesmo tempo, tempo máximo de uma jogada, entre outros;
- Implementação: Utilização da tecnologia Java no lado do servidor e JavaScript no lado do cliente, sendo suportado ao menos pelos seguintes navegadores: Mozilla Firefox 6, Google Chrome 13, Opera 11 e Safari 5 (ou versões superiores).

## **1.4 Glossário**

O glossário é o mesmo da primeira iteração.

## **1.5 Iterações**

Na primeira iteração buscamos fazer a implantação do jogo de pôquer. Assim, para podermos descrever o funcionamento de um torneio de pôquer também implantamos a parte de cadastro e acesso de jogadores, pois, o jogador é o ator principal em um jogo de pôquer online. Já a parte de edição e remoção de jogadores será feita nessa segunda iteração.

Também será feita nessa segunda iteração os requisitos que envolvem o administrador, os mantenedores, os torneios, o caixa do jogador e a área com instruções básicas.

Casos de uso da primeira iteração:

- Cadastrar jogador;
- Autenticar jogador;
- Entrar em um torneio;
- Distribuir as cartas;
- Jogar o pré-flop;
- Mostrar o flop;
- Jogar depois do flop, depois do river ou depois do turn;
- Mostrar o turn;
- Mostrar o river;
- Mostrar showdown;
- Ir all in;
- Ir all in sem conseguir pagar a aposta.

Casos de uso da segunda iteração:

- Editar jogador;
- Excluir jogador;
- Autenticar administrador;
- Cadastrar mantenedor;
- Excluir mantenedor;
- Autenticar mantenedor;
- Criar torneio;
- Excluir torneio.

Uma observação interessante a ser feita é que na primeira iteração os escopo de todos casos de uso apresentados foram Sistema Jogador. Já na segunda iteração serão discutidos além dos escopo Sistema Jogador, o escopo Sistema Administrador e Sistema Mantenedor.

## 2. REQUISITOS DA 2ª ITERAÇÃO

### 2.1 Casos De Uso

Caso de uso: Editar jogador:

**Escopo:** Sistema Jogador.

**Ator primário:** Jogador.

**Stakeholders e seus interesses:**

- Jogador: Deseja alterar alguns de seus dados no sistema.

**Pré-condições:** O jogador já está autenticado no sistema (caso de uso: Autenticar jogador da primeira iteração).

**Pós-condições:** O cadastro do jogador é atualizado no banco de dados.

**Fluxo principal:**

1. O jogador acessa o seu Caixa.
2. O jogador edita os dados de sua conta, podendo alterar nome, e-mail e senha.
3. O jogador confirma a edição dos dados e os mesmos são atualizados.

**Fluxo Alternativo:**

- 2a. O jogador cancela a edição:
  1. A edição é cancelada e o jogador é retornado ao Caixa.
- 3a. O jogador informou um algum dado inválido:
  1. É mostrada uma mensagem de erro ao jogador informando qual foi o erro cometido.
  2. É retornado ao passo 2 do fluxo principal.

Caso de uso: Excluir conta do jogador:

**Escopo:** Sistema Jogador.

**Ator primário:** Jogador.

**Stakeholders e seus interesses:**

- Jogador: Deseja excluir sua conta do sistema.

**Pré-condições:** O jogador já está autenticado no sistema (caso de uso: Autenticar jogador da primeira iteração).

**Pós-condições:** O cadastro do jogador e todos seus dados são removidos do banco de dados e o jogador não possui mais acesso ao sistema.

**Fluxo principal:**

1. O jogador acessa o seu Caixa.
2. O jogador exclui sua conta.
3. É pedida a senha do jogador para confirmar a exclusão.
4. O sistema é redirecionado para a página inicial.

**Fluxo Alternativo:**

- 3a. O jogador cancela a exclusão:
  1. A exclusão é cancelada e o jogador é retornado ao Caixa.
- 3b. O jogador informou uma senha inválida:
  1. É mostrada uma mensagem de erro ao jogador informando que a senha fornecida está incorreta.
  2. É retornado ao passo 3 do fluxo principal.

Caso de uso: Autenticar administrador:

**Escopo:** Sistema Administrador.

**Ator primário:** Administrador.

**Stakeholders e seus interesses:**

- Administrador: Deseja ter acesso ao sistema.

**Pré-condições:** O administrador possui uma senha pré-definida no sistema e é através dessa senha que o mesmo realiza a autenticação.

**Pós-condições:** Acesso ao sistema.

**Fluxo principal:**

1. O administrador acessa a área administrativa do sistema.
2. O administrador preenche a senha.
3. O administrador envia a senha para verificação.
4. O administrador recebe acesso ao sistema.

**Fluxo Alternativo:**

- 3a. O administrador informou uma senha inválida:
  1. É mostrada uma mensagem de erro ao administrador informando que a senha está incorreta.
  2. O administrador pode voltar ao ponto 2 do fluxo principal e preencher a senha novamente ou pode abandonar o sistema.

Caso de uso: Cadastrar mantenedor:

**Escopo:** Sistema Administrador.

**Ator primário:** Administrador.

**Stakeholders e seus interesses:**

- Administrador: Deseja cadastrar um mantenedor para que este possa auxiliar no gerenciamento de torneios.
- Mantenedor: Deseja receber um e-mail com as informações de sua conta.

**Pré-condições:** O administrador está autenticado no sistema (definido em: Autenticar administrador).

**Pós-condições:** Cadastro do mantenedor é salvo no banco de dados e este pode ter acesso ao sistema.

**Fluxo principal:**

1. O administrador inicia o cadastro de um mantenedor.
2. O administrador preenche as seguintes informações: nome de mantenedor e e-mail.
3. Uma senha é gerada automaticamente.
4. O administrador confirma o cadastro do mantenedor.
5. É enviada uma mensagem para o e-mail cadastrado do mantenedor com os dados do cadastro.

**Fluxo Alternativo:**

- 2a. O administrador cancela o cadastro:
  1. O administrador cancela o cadastro do mantenedor.
- 4a. O administrador informou algum campo inválido:
  1. É mostrada uma mensagem de erro ao administrador informando o problema.
  2. O administrador volta ao ponto 2 do fluxo principal para corrigir os campos inválidos.

Caso de uso: Excluir mantenedor:

**Escopo:** Sistema Administrador.

**Ator primário:** Administrador.

**Stakeholders e seus interesses:**

- Administrador: Deseja excluir um mantenedor cadastrado, impossibilitando este de ter acesso ao sistema.
- Mantenedor: Deseja receber um e-mail avisando que sua conta foi excluída.

**Pré-condições:** O administrador está autenticado no sistema (definido em: Autenticar administrador).



**Pós-condições:** Cadastro do mantenedor é removido no banco de dados e este não pode mais ter acesso ao sistema.

**Fluxo principal:**

1. O administrador seleciona o mantenedor que deseja remover em uma lista de mantenedores.
2. O administrador confirma a exclusão do mantenedor do sistema.
3. É enviada uma mensagem para o e-mail cadastrado do mantenedor informando sobre a exclusão.

**Fluxo Alternativo:**

- 1a. O administrador cancela a exclusão:
  1. O administrador cancela a exclusão do mantenedor.

Caso de uso: Autenticar mantenedor:

**Escopo:** Sistema mantenedor.

**Ator primário:** Mantenedor.

**Stakeholders e seus interesses:**

- Mantenedor: Deseja ter acesso ao sistema.

**Pré-condições:** O mantenedor já foi cadastrado no sistema pelo administrador (caso de uso: Cadastrar mantenedor).

**Pós-condições:** Acesso ao sistema.

**Fluxo principal:**

1. O mantenedor acessa a área de mantenedores do sistema.
2. O mantenedor preenche o nome de mantenedor e a senha.
3. O mantenedor envia os dados para verificação.
4. O mantenedor recebe acesso ao sistema.

**Fluxo Alternativo:**

- 3a. O mantenedor informou nome de mantenedor ou senha inválidos:
  1. É mostrada uma mensagem de erro ao mantenedor informando o problema.
  2. O mantenedor pode voltar ao ponto 2 do fluxo principal e preencher os campos novamente ou pode abandonar o sistema.

Caso de uso: Criar torneio:

**Escopo:** Sistema Mantenedor.

**Ator primário:** Mantenedor.

**Stakeholders e seus interesses:**

- Mantenedor: Deseja criar um novo torneio.
- Jogador: Deseja entrar em torneios e para isso é necessário que estes sejam criados pelos mantenedores.

**Pré-condições:** O mantenedor está autenticado no sistema (definido em: Autenticar mantenedor).

**Pós-condições:** Torneio é criado e se torna disponível para receber inscrições de jogadores.

**Fluxo principal:**

1. O mantenedor inicia a criação de um torneio.
2. O mantenedor preenche os campos: descrição do torneio, número de vagas, valor da entrada, quantidade de fichas inicial para cada jogador e valores iniciais do small blind e big blind.
3. O mantenedor confirma a criação do torneio.
4. O torneio está disponível para receber inscrições do jogadores.

**Fluxo Alternativo:**

- 2a. O mantenedor cancela a criação do torneio:
  1. A criação do torneio é cancelada.
- 3a. O mantenedor preencheu algum campo inválido:
  1. O mantenedor é informado do erro.
  2. O mantenedor é retornando ao passo 2 do fluxo principal.

Caso de uso: Excluir torneio:

**Escopo:** Sistema Mantenedor.

**Ator primário:** Mantenedor.

**Stakeholders e seus interesses:**

- Mantenedor: Deseja excluir um torneio.

**Pré-condições:** O mantenedor está autenticado no sistema (definido em: Autenticar mantenedor) e o torneio a ser excluído não está em andamento, nem possui jogadores inscritos.

**Pós-condições:** Torneio é excluído e não poderá mais receber inscrições.

**Fluxo principal:**

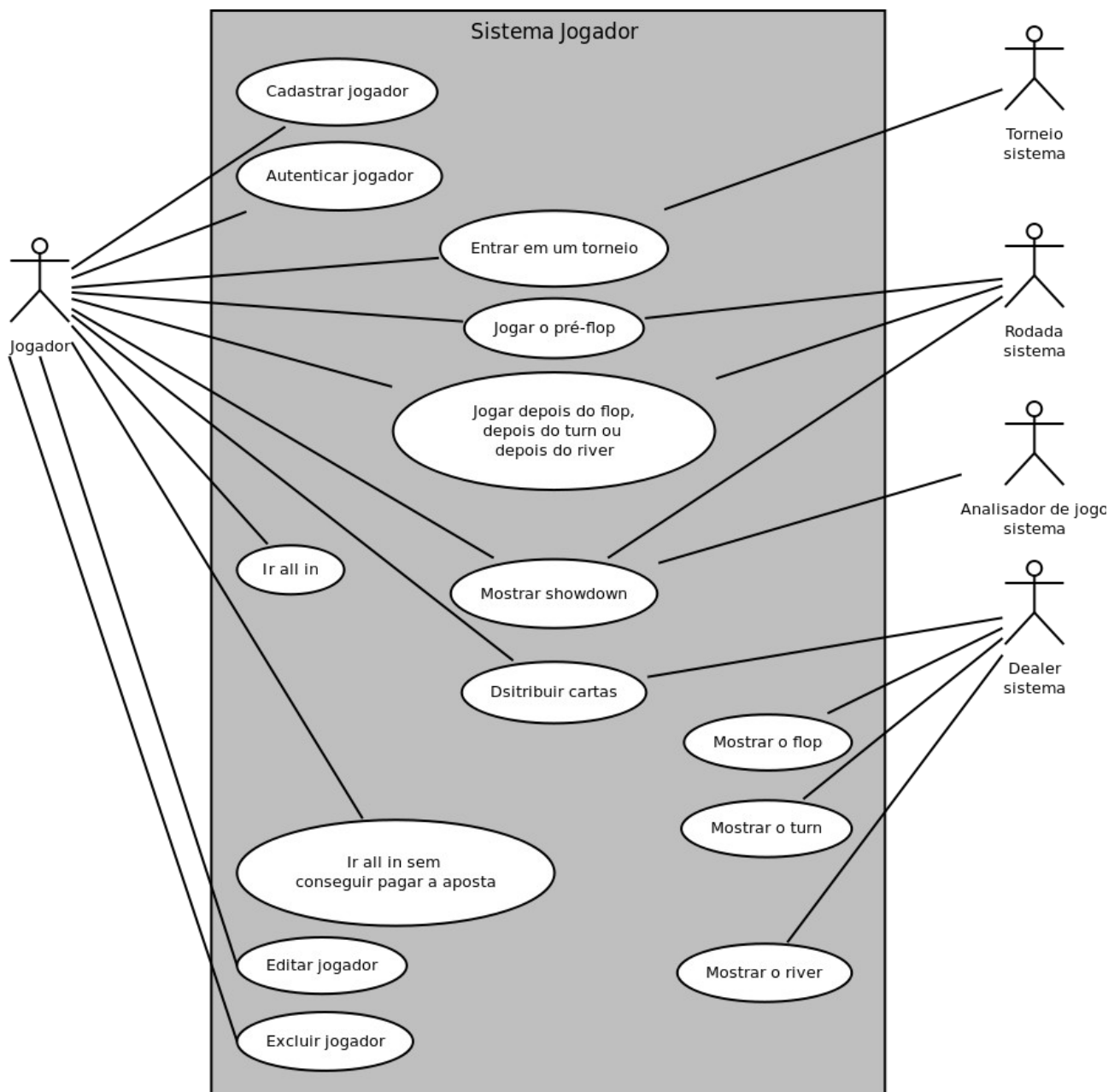
1. O mantenedor seleciona o torneio que deseja remover em uma lista de torneios.
2. O mantenedor confirma a exclusão do torneio.

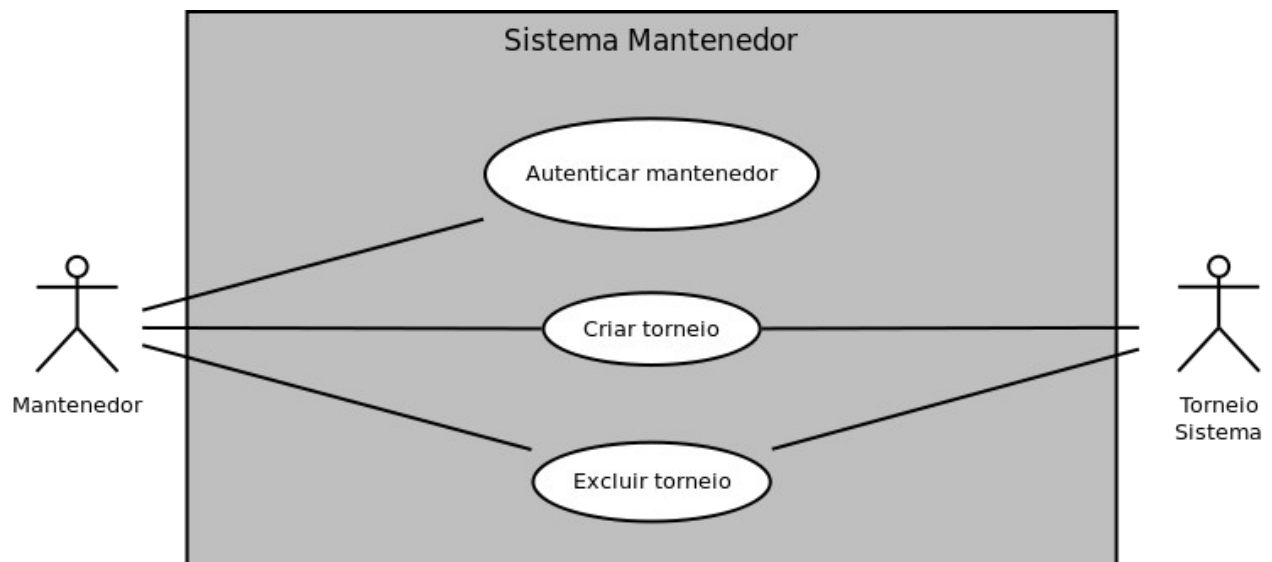
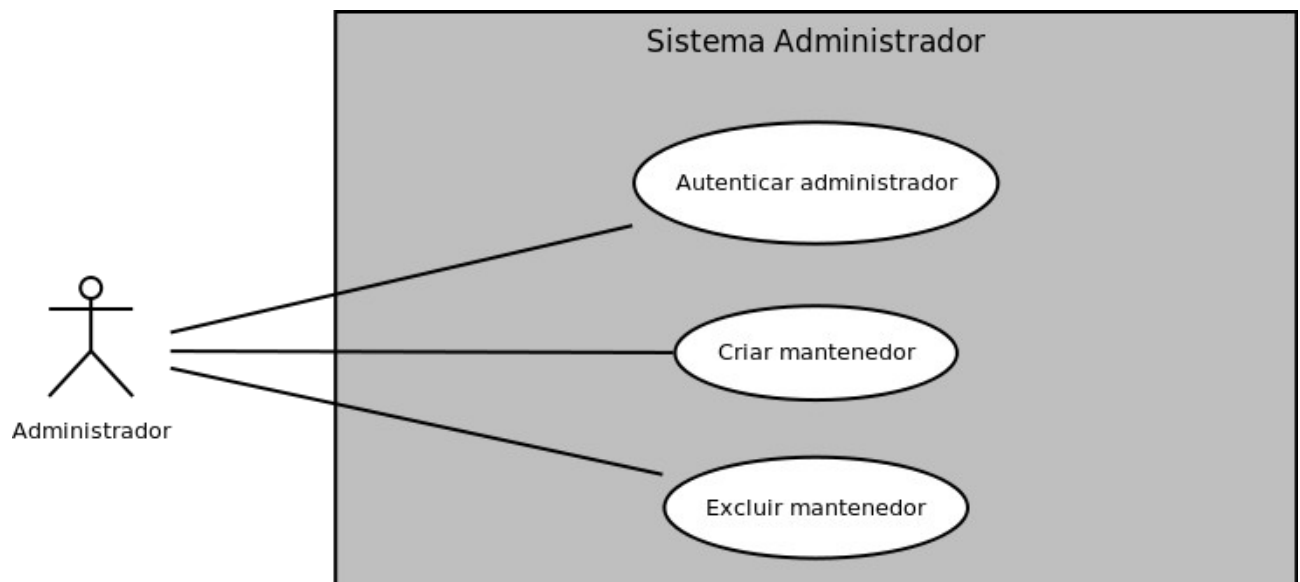
3. O torneio é excluído e não pode mais receber inscrições.

**Fluxo Alternativo:**

- 1a. O mantenedor cancela a exclusão:
  1. O mantenedor cancela a exclusão do torneio.
- 2a. O mantenedor tentou excluir um torneio em andamento ou um torneio com jogadores inscritos:
  1. O mantenedor é retornado ao ponto 1 do fluxo principal.

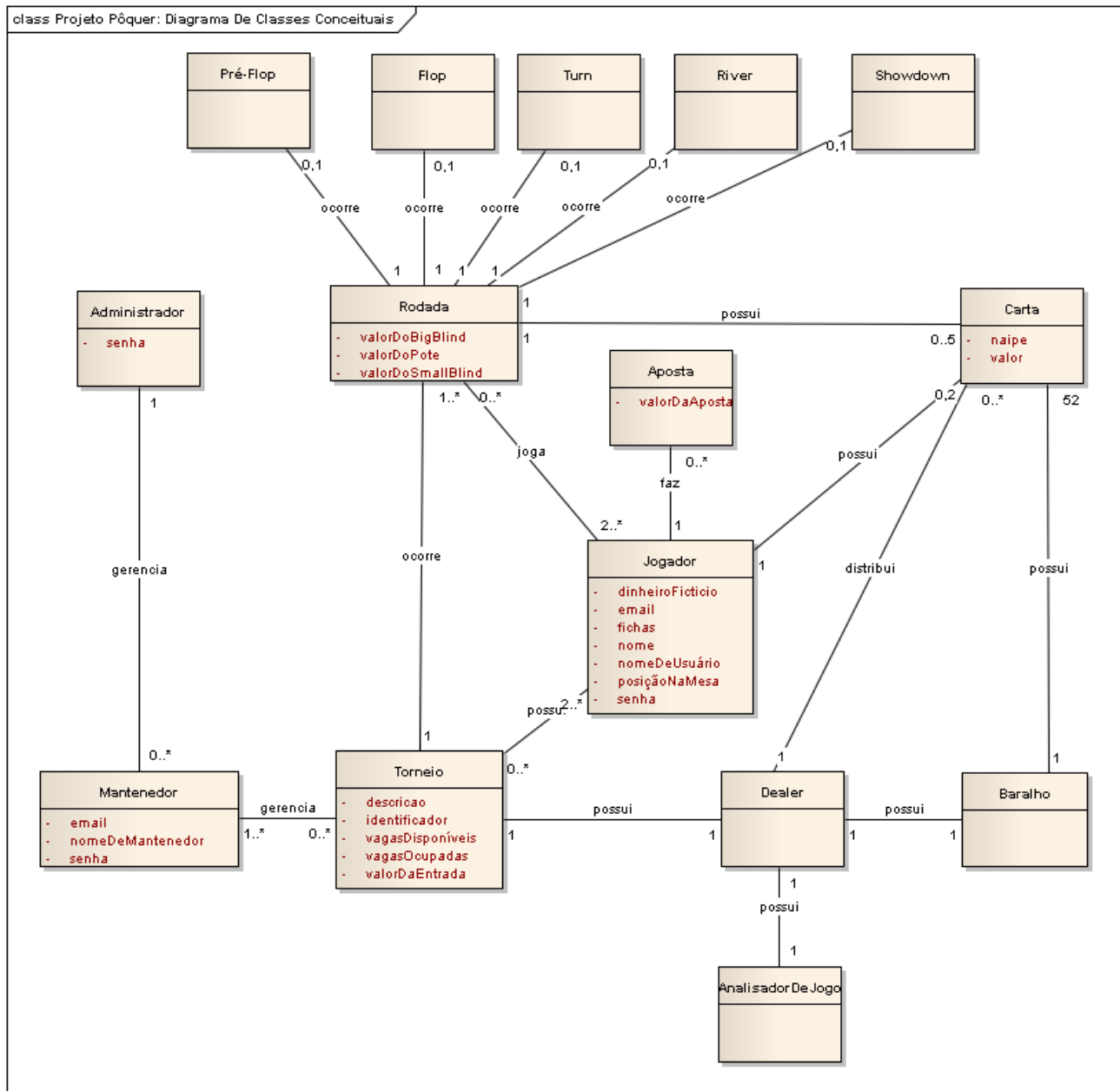
## 2.2 Diagramas De Casos De Uso





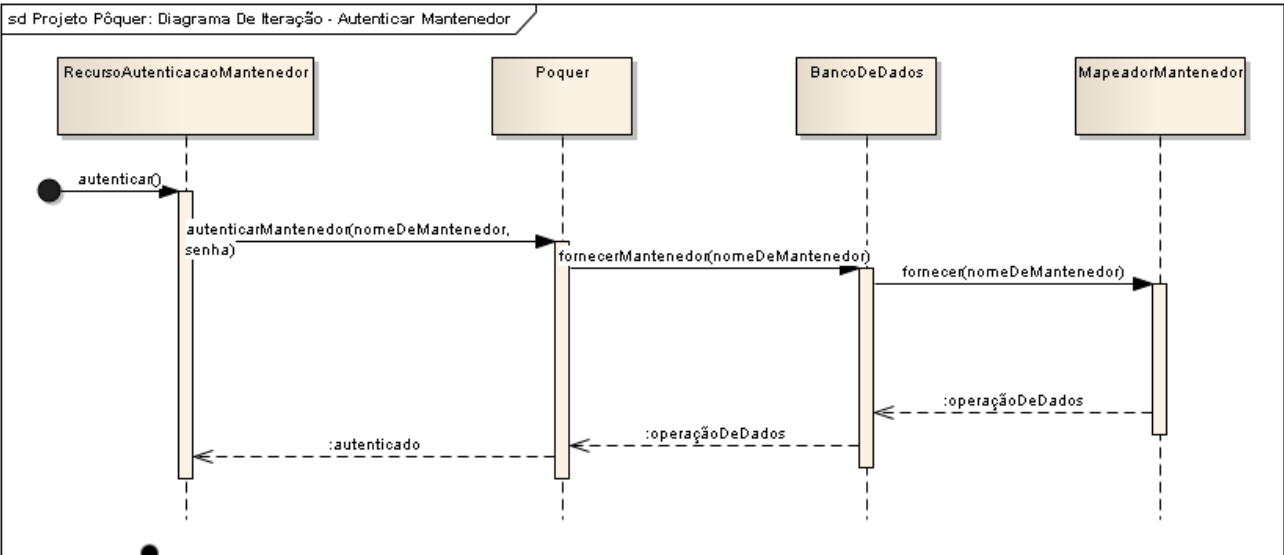
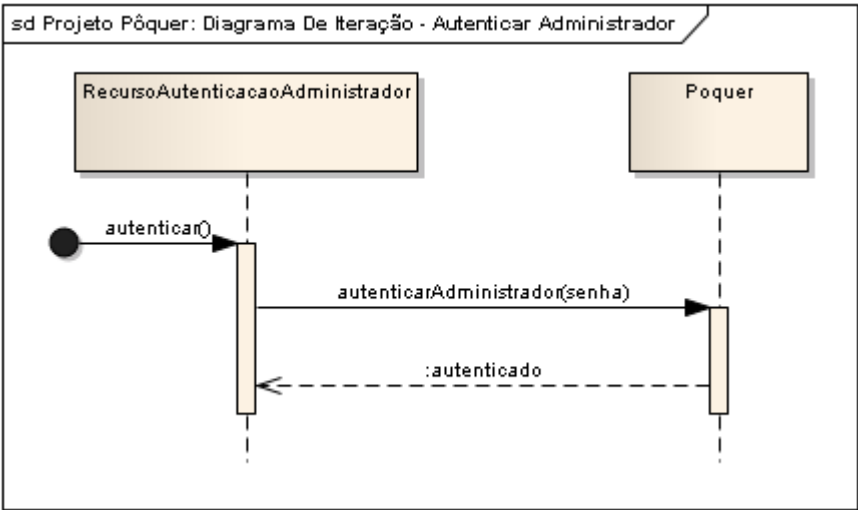
### 3. ANÁLISE DA 2ª ITERAÇÃO

#### 3.1 Diagrama De Classes Conceituais

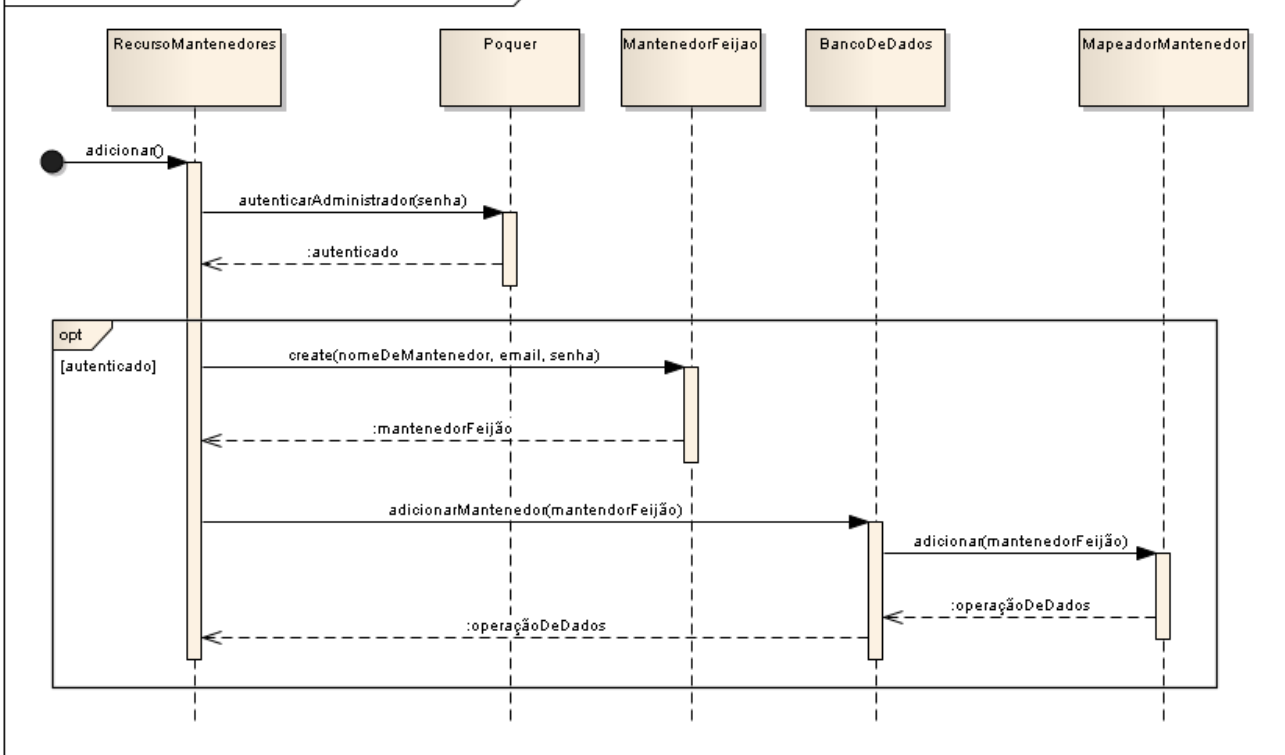


4. PROJETO DA 2ª ITERAÇÃO

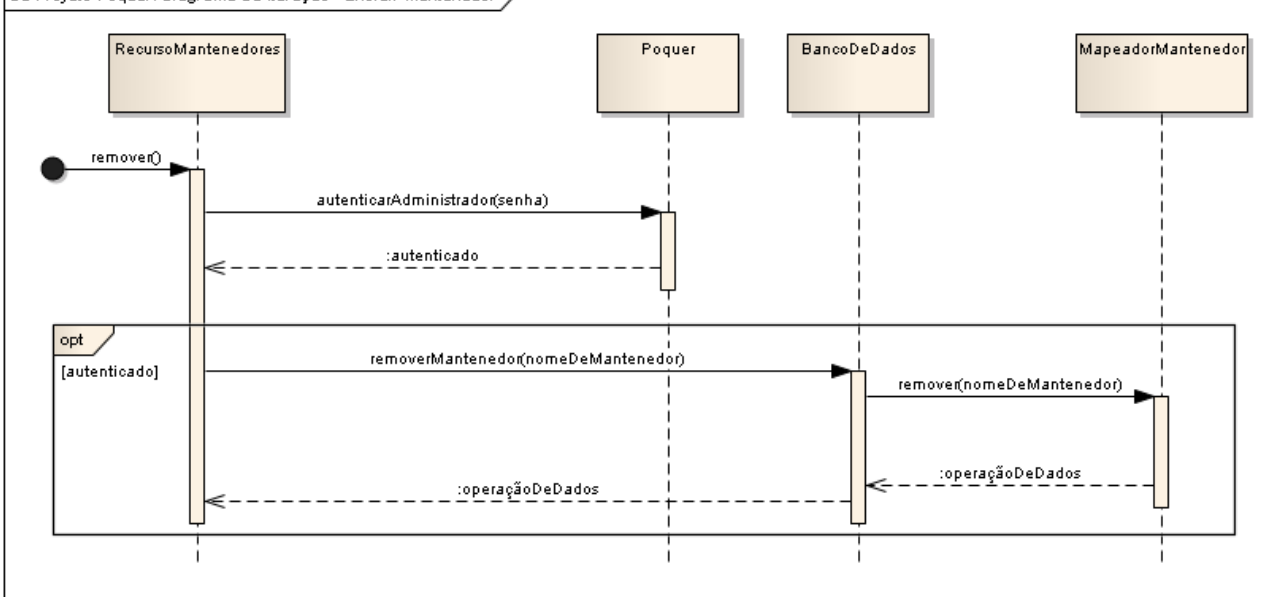
4.1 Diagramas De Iteração



sd Projeto Pôquer: Diagrama De Iteração - Cadastrar Mantenedor

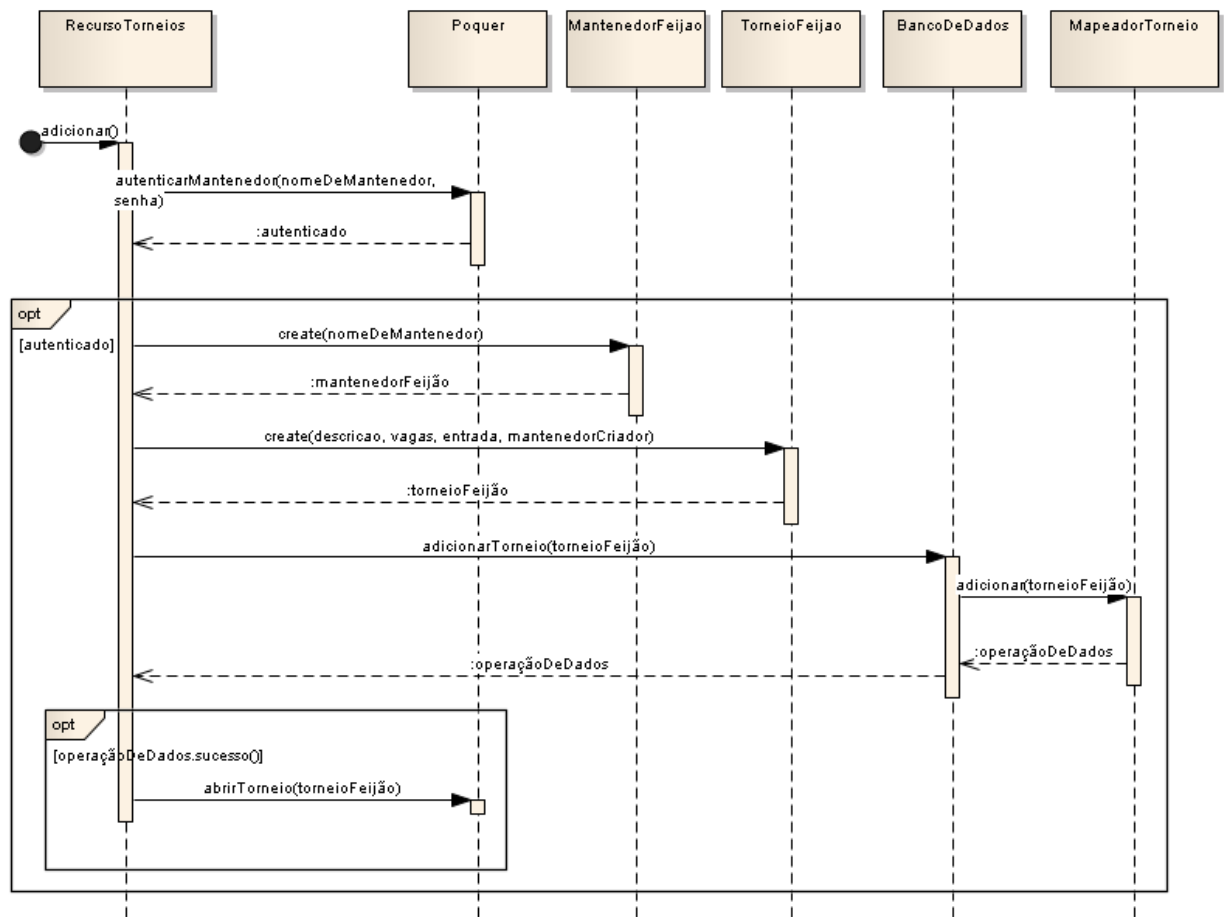


sd Projeto Pôquer: Diagrama De Iteração - Excluir Mantenedor

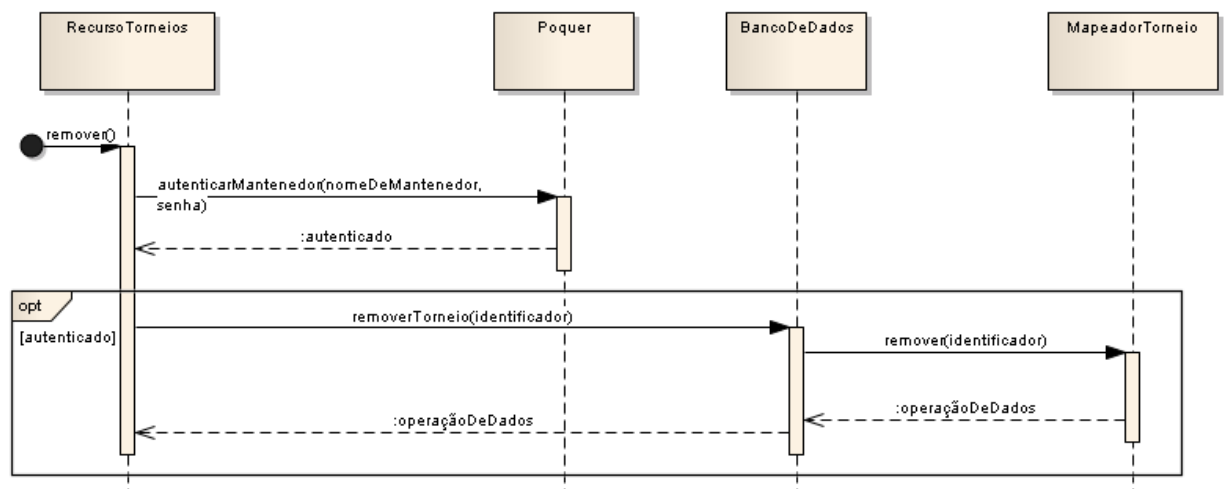




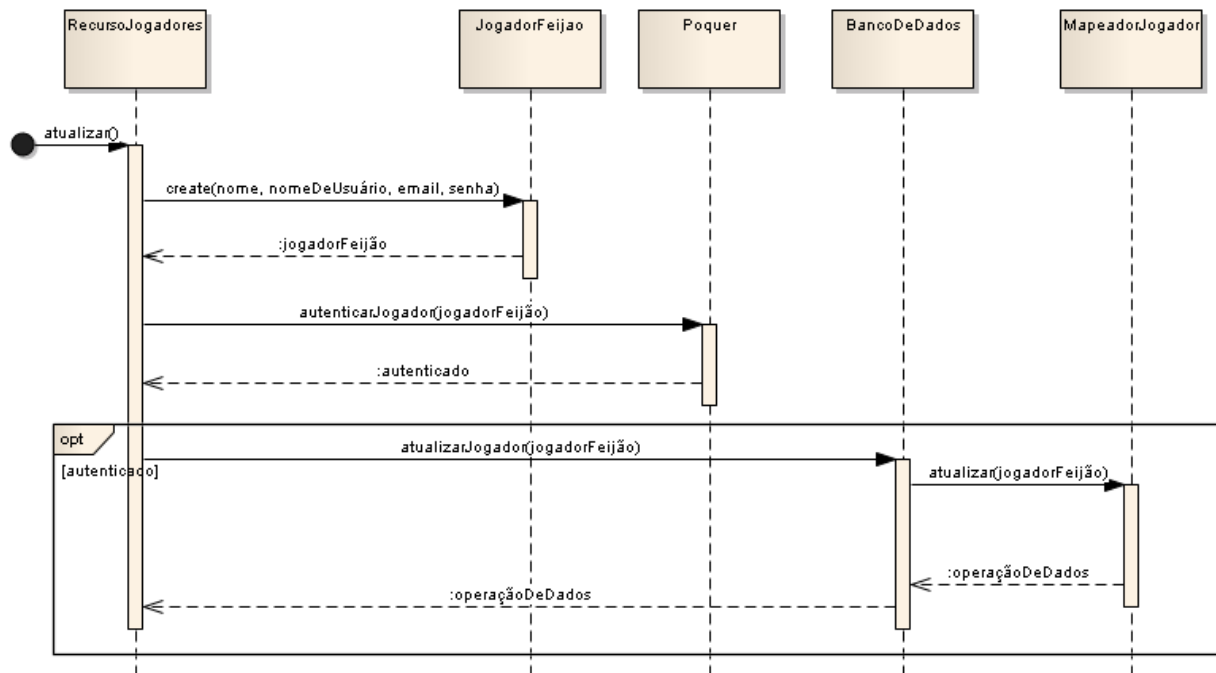
sd Projeto Pôquer: Diagrama De Iteração - Criar Torneio



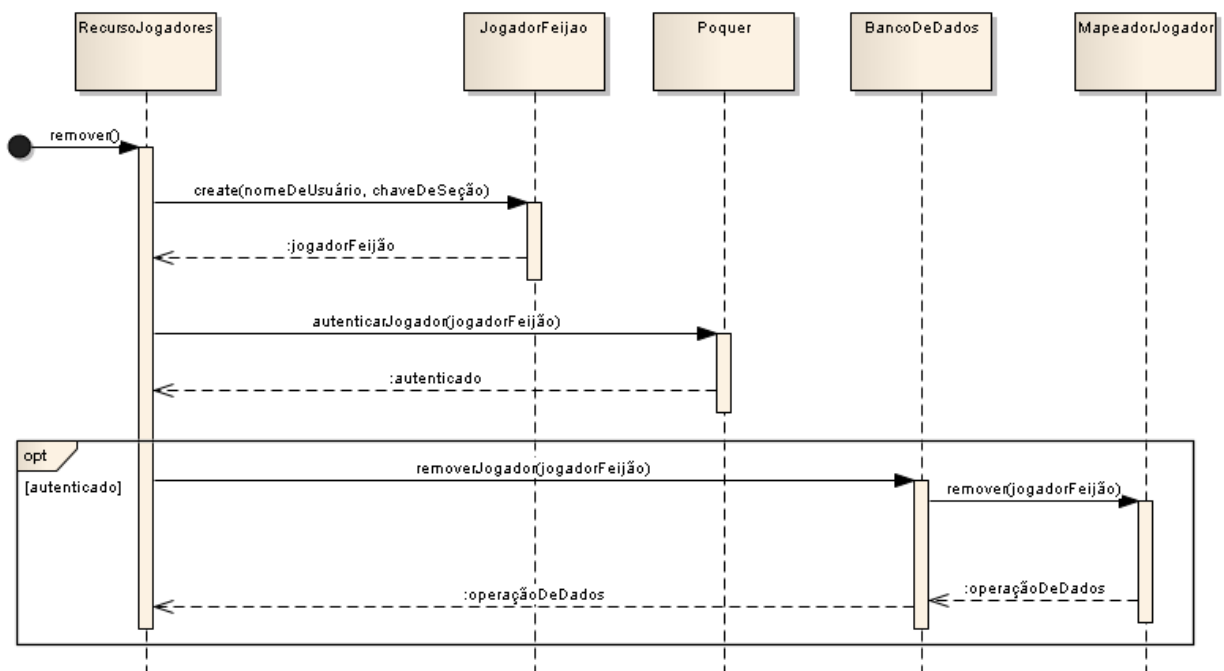
sd Projeto Pôquer: Diagrama De Iteração - Excluir Torneio



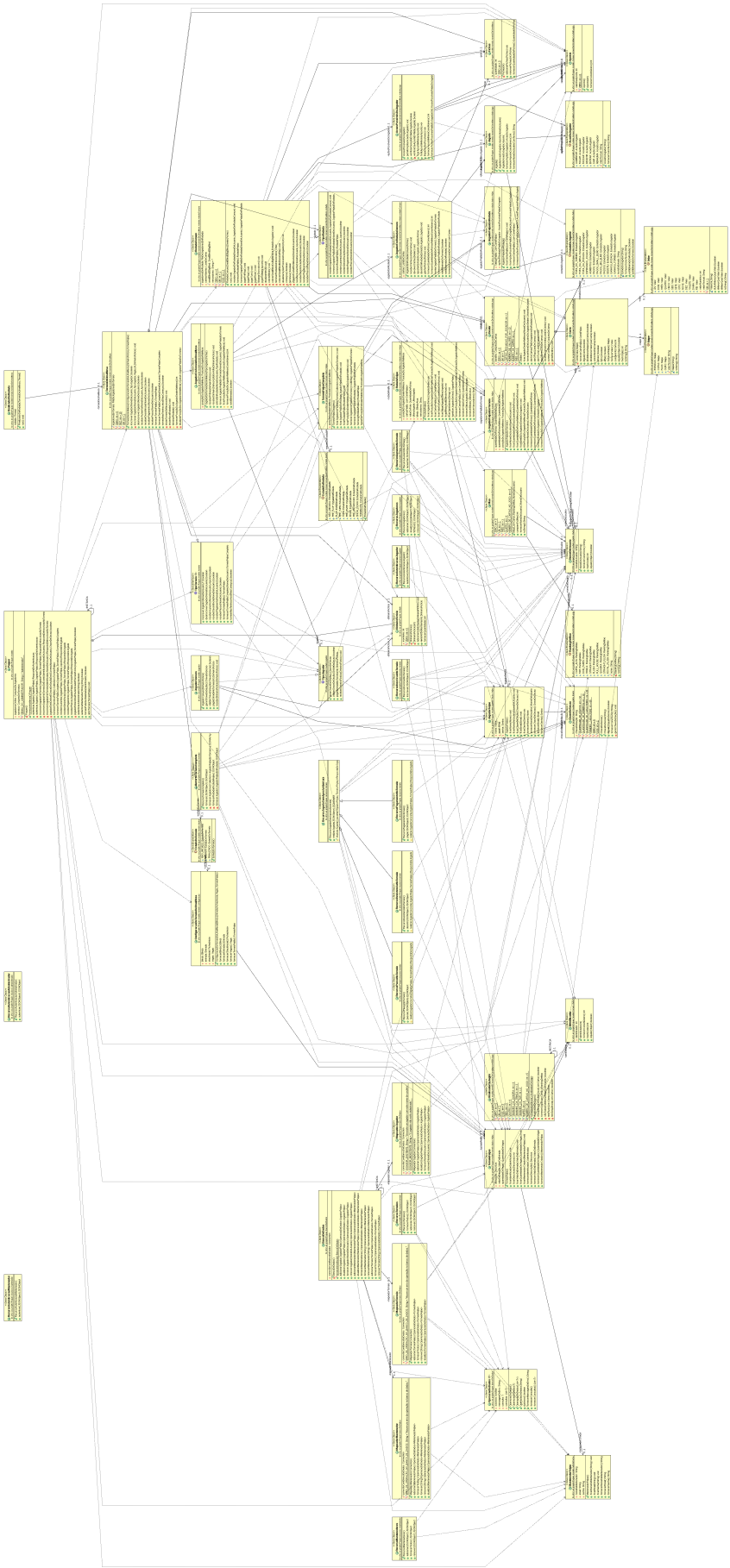
sd Projeto Pôquer: Diagrama De Iteração - Editar Jogador



sd Projeto Pôquer: Diagrama De Iteração - Excluir Jogador



4.2 Diagrama De Classes De Projeto



### 4.3 Projeto De Persistência

Nos mapeamentos abaixo os campos da tabela marcados com um fundo **amarelo** representam chaves primárias e campos marcados com **verde** representam uma chave estrangeira.

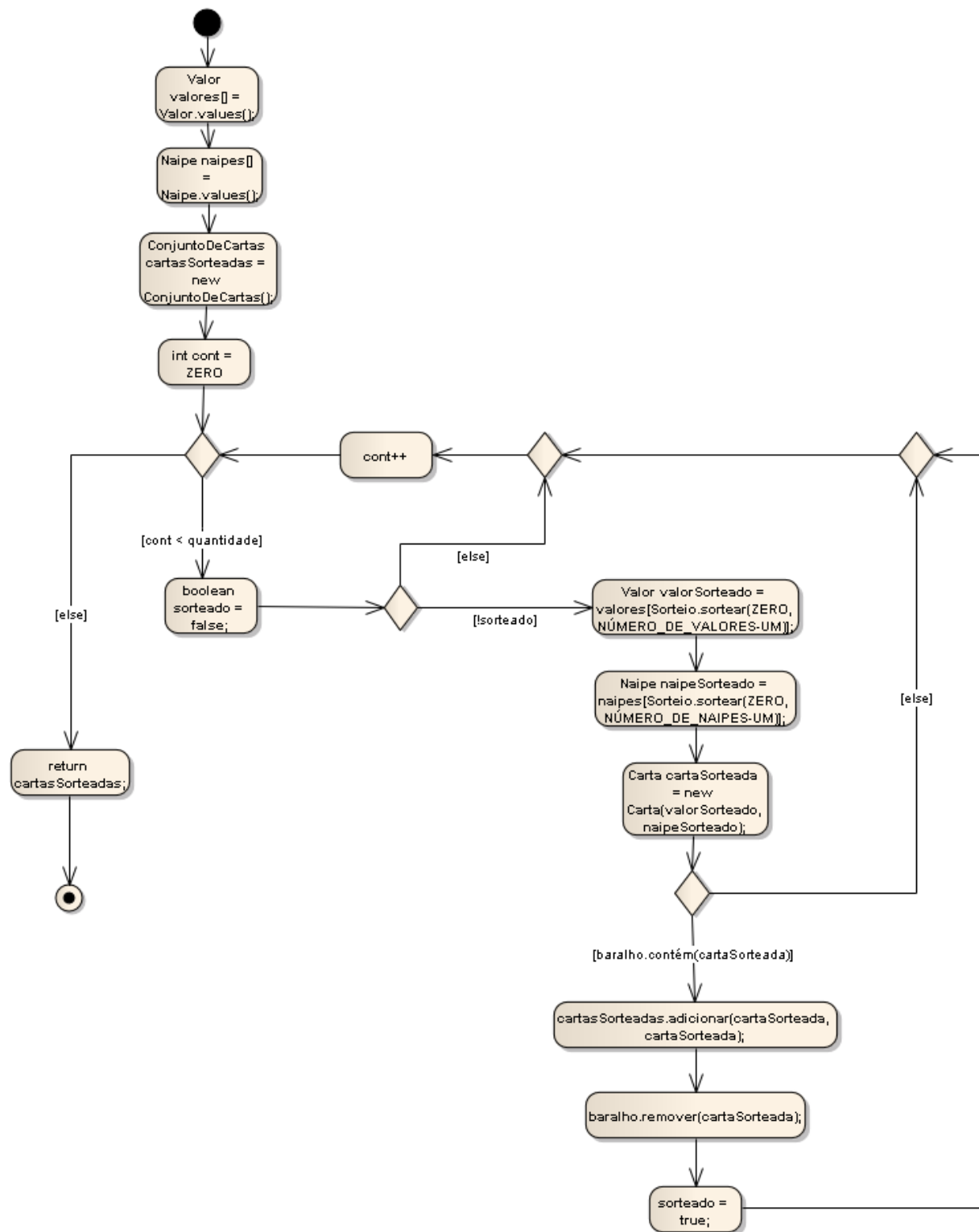
Classe: JogadorFeijão		Tabela: jogador	
nome	String	nome	VARCHAR(64)
nomeDeUsuario	String	nomeDeUsuario	VARCHAR(16)
email	String	email	VARCHAR(64)
senha	String	senha	VARCHAR(16)

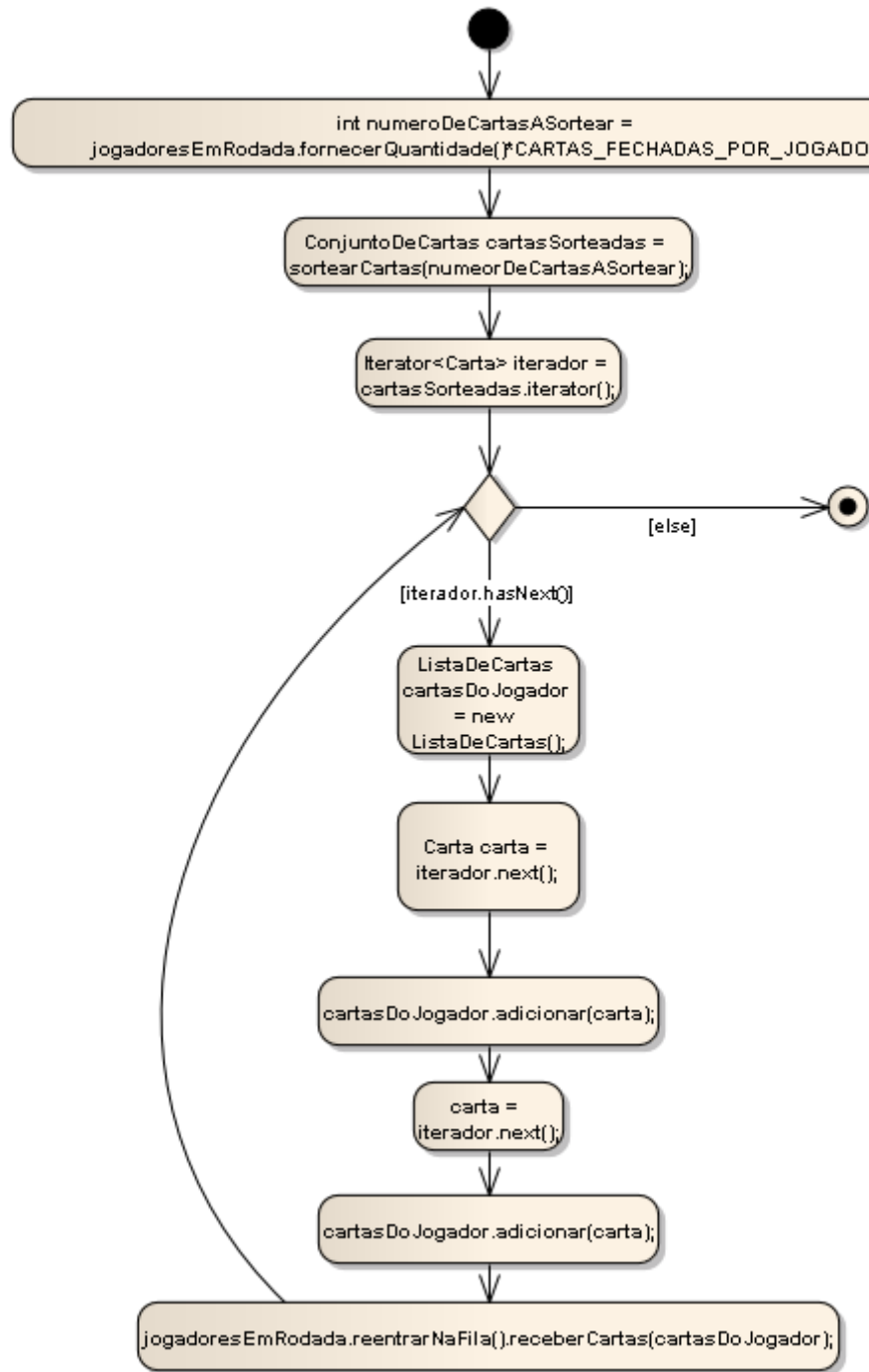
Classe: MantenedorFeijão		Tabela: mantenedor	
nomeDeMantenedor	String	NomeDeMantenedor	VARCHAR(16)
email	String	email	VARCHAR(64)
senha	String	senha	VARCHAR(16)

Classe: TorneioFeijão		Tabela: torneio	
identificador	int	identificador	INTEGER AUTO_INCREMENT
descricao	String	descricao	VARCHAR(128)
entrada	int	valorDaEntrada	INTEGER
vagasOcupadas	Int	vagasOcupadas	INTEGER
vagasDisponiveis	int	vagasDisponiveis	INTEGER
mantenedorCriador	MantenedorFeijão	mantenedorCriador REFERENCES mantenedor(nomeDe Mantenedor)	VARCHAR(16)

## 4.4 Diagramas De Atividades

act Projeto Pôquer: Diagrama De Atividades - Dealer.sortearCartas()





4.5 Diagramas De Statecharts

