

UNIVERSIDADE FEDERAL DE VIÇOSA
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
DEPARTAMENTO DE INFORMÁTICA

PROJETO PRÁTICO DA DISCIPLINA INF311

POKÉMON GO CLONE

Lucas Francisco da Matta Vegi
(Professor e Coordenador da disciplina)

VIÇOSA - MINAS GERAIS
2017/1

ÍNDICE

LISTA DE ILUSTRAÇÕES	iii
LISTA DE TABELAS	iv
1 – VISÃO GERAL DO SISTEMA	1
2 – REQUISITOS DO SISTEMA	1
2.1 – Requisitos Funcionais e Não-Funcionais	1
2.2 – Requisitos Suplementares	17
3 – PROJETO DE INTERFACE	18
3.1 – Telas do Sistema.....	18
3.2 – Fluxo de Navegação das Telas (Storyboard)	21
4 – PROJETO DE PERSISTÊNCIA	22
4.1 – Banco de Dados Local.....	22
4.2 – Banco de Dados do Servidor.....	22
5 – PROJETO DO SISTEMA.....	23
5.1 – Diagrama de Casos de Uso	23
5.1.1 – Listagem de casos de uso de alto nível	23
5.2 – Diagrama de Pacotes	23
5.3 – Diagramas de Classes de Projeto	24
5.4 – Diagramas de Sequência (Casos de Uso Expandidos).....	26
5.5 – Diagramas de Comunicação (Contratos de Operação e Consulta)	28
5.5.1 – Classe “ControladoraFachadaSingleton”	28
5.5.2 – Classe “Usuario”	32
5.5.3 – Classe “Pokemon”	34
APÊNDICES e ANEXOS.....	35

LISTA DE ILUSTRAÇÕES

Figura 1 - Tela de início (Splash).....	18
Figura 2 - Tela de Login	18
Figura 3 - Tela de cadastro.....	19
Figura 4 - Tela do Mapa Principal	19
Figura 5 - Tela de captura	19
Figura 6 - Tela do Pokédex	19
Figura 7 - Tela de detalhes do pokémon	20
Figura 8 - Tela do Mapa de captura de um pokémon	20
Figura 9 - Tela do Mapa de todas as capturas	20
Figura 10 - Tela do Perfil do usuário	20
Figura 11 - Fluxo de navegação entre as telas	21
Figura 12 - Modelo Entidade Relacionamento do banco de dados local.....	22
Figura 13 - Modelo Entidade Relacionamento do banco de dados do servidor	22
Figura 14 - Diagrama de casos de uso	23
Figura 15 - Diagrama de Pacotes do projeto.....	23
Figura 16 - Diagrama de classes de projeto do pacote Model	24
Figura 17 - Diagrama de classes de projeto do pacote Controller	25
Figura 18 - Diagrama de classes de projeto do pacote Util	25
Figura 19 - Diagrama de classes de projeto do pacote View	25
Figura 20 - Diagrama de sequência do caso de uso “Fazer login”	26
Figura 21 - Diagrama de sequência do caso de uso “Caçar pokémon”	27
Figura 22 - Diagrama de sequência do caso de uso “Capturar pokémon”	28
Figura 23 - Diagrama de comunicação do construtor da classe ControladoraFachadaSingleton	28
Figura 24 - Diagrama de comunicação do método daoTipo()	29
Figura 25 - Diagrama de comunicação do método daoPokemons()	29
Figura 26 - Diagrama de comunicação do método daoUsuario()	29
Figura 27 - Diagrama de comunicação do método sorteiaAparecimentos().....	30
Figura 28 - Diagrama de comunicação do método loginUser().....	31
Figura 29 - Diagrama de comunicação do método logoutUser().....	31
Figura 30 - Diagrama de comunicação do método cadastrarUser().....	31
Figura 31 - Diagrama de comunicação do método temSessao().....	32
Figura 32 - Diagrama de comunicação do método construtor da classe Usuario	32
Figura 33 - Diagrama de comunicação do método preencherCapturas().....	33
Figura 34 - Diagrama de comunicação do método capturar()	34
Figura 35 - Diagrama de comunicação do método preencherTipos().....	34
Figura 36 - Diagrama de comunicação do método construtor da classe Pokemon.....	35

LISTA DE TABELAS

Tabela 1 - F1: Apresentar Splash screen na abertura do aplicativo	1
Tabela 2 - F2: Verificar a existência de conexão com a Internet na abertura do jogo.....	2
Tabela 3 - F3: Verificar a existência de sessão ativa de usuário.....	2
Tabela 4 - F4: Login de acesso	3
Tabela 5 - F5: Verificar a existência de conexão com a Internet no Login	4
Tabela 6 - F6: Cadastrar usuário	4
Tabela 7 - F7: Verificar a existência de conexão com a Internet no Cadastro de usuário	5
Tabela 8 - F8: Encerrar de sessão usuário.....	5
Tabela 9 - F9: Obter a geolocalização do usuário.....	6
Tabela 10 - F10: Personalizar o marcador e o ícone de perfil do usuário.....	7
Tabela 11 - F11: Tocar música tema da jornada de caçada pokémon	7
Tabela 12 - F12: Sortear pokémons para o mapa.....	7
Tabela 13 - F13: Chamar Pokémon selvagem para batalhar	9
Tabela 14 - F14: Verificar a existência de conexão com a Internet no Início de uma batalha	9
Tabela 15 - F15: Tocar música da batalha pokémon	10
Tabela 16 - F16: Posicionar um pokémon na tela de captura	10
Tabela 17 - F17: Realidade Aumentada.....	11
Tabela 18 - F18: Arremessar Pokébola.....	11
Tabela 19 - F19: Verificar se Pokémon da batalha é inédito	12
Tabela 20 - F20: Capturar pokémon	12
Tabela 21 - F21: Persistir Pokémon capturado	13
Tabela 22 - F22: Exibir relatório do perfil do usuário	14
Tabela 23 - F23: Exibir mapa de capturas	14
Tabela 24 - F24: Exibir PokéAgenda.....	15
Tabela 25 - F25: Tocar música da PokéAgenda	15
Tabela 26 - F26: Calcular métricas da PokéAgenda.....	16
Tabela 27 - F27: Exibir detalhes de uma espécie na PokéAgenda	16
Tabela 28 - Requisitos Suplementares	17
Tabela 29 - Casos de uso de alto nível.....	23

1 – VISÃO GERAL DO SISTEMA

O aplicativo que será desenvolvido neste projeto chamará Pokémon GO Clone, e em muito se assemelha ao jogo original desenvolvido pela Niantic Inc. O mesmo deverá permitir ao usuário capturar pokémons que estejam próximos a sua geolocalização e fornecer detalhes sobre as capturas realizadas.

O usuário poderá ver o seu próprio deslocamento em um mapa, obtido por meio do GPS, e estará cercado por pokémons aleatórios. Se estiver próximo o suficiente de um pokémon, poderá tentar capturá-lo. Será utilizada realidade aumentada durante as capturas, sendo assim necessário o uso da câmera e do giroscópio dos dispositivos móveis.

Ao movimentar o dispositivo móvel durante uma tentativa de captura, o usuário verá a imagem do Pokémon se movendo pela tela, como se o Pokémon estivesse fixo em um local do ambiente real. Para capturá-lo será necessário acertar o mesmo com uma pokébola, movida por meio do toque na tela do aparelho.

Para acompanhar o progresso da jornada Pokémon, o aplicativo deverá possuir uma PokéAgenda. Esta mostrará todos os pokémons já capturados pelo treinador, assim como detalhes sobre cada uma dessas capturas.

Para começar a jornada Pokémon, um treinador deverá se cadastrar e logar. A implementação poderá ser toda off-line ou possuir também comunicação com um servidor Web. Caso contenha comunicação com um servidor, será possível para um usuário continuar a sua jornada de onde parou, mesmo que o troque de dispositivo móvel.

Caso a implementação opte por comunicar-se com um servidor, será necessário sincronizar os dados locais e globais em momentos específicos do jogo. No caso da ausência de Internet no momento do uso do aplicativo, avisos deverão ser emitidos para o usuário e algumas funcionalidades do jogo ficarão limitadas.

2 – REQUISITOS DO SISTEMA

2.1 – Requisitos Funcionais e Não-Funcionais

Tabela 1 - F1: Apresentar Splash screen na abertura do aplicativo

F1 - Apresentar Splash screen na abertura do aplicativo	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Abrir a tela de splash screen automaticamente quando o aplicativo for iniciado e mantê-la ativa enquanto os recursos são carregados.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF1.1 - Tocar o tema de abertura	Tocar o tema de abertura do jogo até o final enquanto a tela estiver aberta.	Interface	()	(X)

Tabela 2 - F2: Verificar a existência de conexão com a Internet na abertura do jogo

F2 - Verificar a existência de conexão com a Internet na abertura do jogo.	Oculto (X)	Desejável (X)	Permanente ()	
Descrição: Verificar a existência de conexão com Internet na splash screen e só permitir a entrada no jogo caso exista uma conexão ativa, seja por wi-fi ou rede de dados.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF2.1 - Avisar ao usuário que não existe conexão com a Internet.	Caso não exista conexão, exibir uma mensagem de alerta para o usuário e fechar o aplicativo.	Confiabilidade	(x)	()

Tabela 3 - F3: Verificar a existência de sessão ativa de usuário.

F3 - Verificar a existência de sessão ativa de usuário.	Oculto (X)	Desejável ()	Permanente (X)	
Descrição: Sempre que o aplicativo for aberto, verificar automaticamente se existe uma sessão ativa de usuário antes de exibir a tela que solicita login e senha.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF3.1 - Começar o jogo.	Caso exista sessão ativa, iniciar o jogo com o usuário autenticado.	Segurança	()	(x)
NF3.2 - Sincronizar dados com o servidor.	Verificar se o número de pokémons capturados pelo usuário é igual localmente e no servidor. Caso seja diferente, apagar a base local de pokémons capturados e substituí-la pelos dados dos pokémons capturados presentes no servidor.	Confiabilidade	(x)	(x)

Tabela 4 - F4: Login de acesso

F4 - Login de acesso.	Oculto ()	Desejável ()	Permanente (X)	
Descrição: O usuário deve fazer Login no aplicativo para poder utilizá-lo caso não exista uma sessão aberta.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF4.1 - Verificar a reentrada de usuário.	A cada novo login, verificar se o usuário é o mesmo que fez logout por último. Se não for, apagar todos os dados de usuário e pokémons capturados presentes na base local, substituí-los por dados vindos do servidor e abrir uma sessão para o usuário se ele existir. Caso seja o mesmo usuário que está reentrando, apenas reativar a sessão localmente.	Segurança	(X)	()
NF4.2 - Sincronizar dados com o servidor.	Verificar se o número de pokémons capturados pelo usuário é igual localmente e no servidor. Caso seja diferente, apagar a base local de pokémons capturados e substituí-la pelos dados dos pokémons capturados presentes no servidor.	Confiabilidade	(x)	(x)
NF4.3 - Validar formato de login e senha.	Permitir apenas que valores alfanuméricos de login e senha sejam submetidos à autenticação de usuário.	Segurança	()	()
NF4.4 - Começar o jogo.	Após abrir ou reativar a sessão, iniciar o jogo com o usuário autenticado.	Segurança	()	(x)

Tabela 5 - F5: Verificar a existência de conexão com a Internet no Login

F5 - Verificar a existência de conexão com a Internet no Login.	Oculto (X)	Desejável (X)	Permanente ()	
Descrição: Verificar a existência de conexão com Internet na tela de Login antes de submeter dados ao servidor.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF5.1 - Avisar ao usuário que não existe conexão com a Internet.	Caso não exista conexão, exibir uma mensagem de alerta para o usuário e manter o aplicativo aberto.	Confiabilidade	(x)	()

Tabela 6 - F6: Cadastrar usuário

F6 - Cadastrar usuário.	Oculto ()	Desejável ()	Permanente (X)	
Descrição: O usuário deve ser capaz de se cadastrar no sistema, informando Nome, login, senha, confirmação de senha e sexo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF6.1 - Verifica o preenchimento de dados obrigatórios.	Verificar se nome, login, senha e sexo foram preenchidos antes de persistir no banco e abrir sessão de usuário. Avisar ao usuário caso algum dado tenha faltado ou se a confirmação de senha não for igual a senha escolhida.	Usabilidade	()	()
NF6.2 - Verifica a disponibilidade do login.	Verificar no servidor se o login escolhido pelo usuário	Usabilidade e Confiabilidade	(X)	(X)

	está disponível. Avisá-lo caso o mesmo esteja indisponível.			
NF6.3 - Persistir dados do usuário no servidor.	Caso o usuário escolha um login disponível, persistir os dados do mesmo no servidor.	Segurança	(X)	(X)
NF6.4 - Validar formato de login e senha.	Permitir apenas que valores alfanuméricos de login e senha sejam submetidos à autenticação de usuário.	Segurança	()	()
NF6.5 - Começar o jogo.	Caso exista sessão ativa, iniciar o jogo com o usuário autenticado.	Segurança	()	(x)

Tabela 7 - F7: Verificar a existência de conexão com a Internet no Cadastro de usuário

F7 - Verificar a existência de conexão com a Internet no Cadastro de usuário.	Oculto (X)	Desejável (X)	Permanente ()	
Descrição: Verificar a existência de conexão com Internet na tela de Cadastro antes de submeter dados ao servidor.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF7.1 - Avisar ao usuário que não existe conexão com a Internet.	Caso não exista conexão, exibir uma mensagem de alerta para o usuário e manter o aplicativo aberto.	Confiabilidade	(x)	()

Tabela 8 - F8: Encerrar de sessão usuário

F8 - Encerrar de sessão usuário.	Oculto ()	Desejável ()	Permanente (X)	
Descrição: O usuário deve ser capaz fazer logout do aplicativo.				
Requisitos Não-Funcionais				

Nome	Restrição	Categoria	Desejável	Permanente
NF8.1 - Manter dados locais.	Ao fazer logout, apenas fechar a sessão do usuário vigente. Os dados locais referentes ao mesmo e às capturas realizadas deverão ser mantidos no banco de dados.	Usabilidade	()	(X)

Tabela 9 - F9: Obter a geolocalização do usuário

F9 - Obter a geolocalização do usuário.	Oculto (x)	Desejável ()	Permanente (X)	
Descrição: Ao entrar no jogo, obter automaticamente a localização do usuário em latitude e longitude, preferencialmente por meio do GPS.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF9.1 - Intervalo de tempo para atualização da posição do usuário.	A cada 5 segundos, atualizar a posição do usuário.	Desempenho	()	(X)
NF9.2 - Atualizar o marcador do treinador no mapa.	Sempre que uma nova posição para o treinador for obtida, atualizar o marcador do mesmo no mapa.	Interface	()	(X)
NF9.3 - Posicionar o mapa e configurar variáveis de visualização do mesmo.	Após obter a posição do usuário pela primeira vez, centralizar o mapa no local onde se encontra o usuário, dar um zoom de 18 e inclinar o ângulo de visualização do mesmo em 60. Ao obter novas posições, esse requisito não deve se repetir.	Interface	()	(X)

Tabela 10 - F10: Personalizar o marcador e o ícone de perfil do usuário

F10 - Personalizar o marcador e o ícone de perfil do usuário.	Oculto (x)	Desejável ()	Permanente (X)
Descrição: Ao entrar no jogo, verificar automaticamente o sexo do usuário e personalizar o marcador do mesmo no mapa e o ícone do perfil de usuário no menu.			

Tabela 11 - F11: Tocar música tema da jornada de caçada pokémon

F11 - Tocar música tema da jornada de caçada pokémon.	Oculto (x)	Desejável ()	Permanente (X)	
Descrição: Após obter a posição do usuário pela primeira vez, iniciar o tema da rota 1.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF11.1 - Configurar Looping.	A música deverá ficar repetindo em loop enquanto o jogo estiver aberto.	Interface	()	(X)
NF11.2 - Dar pause na música.	Sempre que o usuário entrar em uma opção do menu, começar a batalhar com um Pokémon ou o celular dormir, a música deverá ser pausada, continuando do mesmo ponto quando o usuário voltar para o mapa principal.	Interface	()	(X)

Tabela 12 - F12: Sortear pokémons para o mapa

F12 - Sortear pokémons para o mapa.	Oculto (x)	Desejável ()	Permanente (X)
Descrição: Após obter a posição do usuário pela primeira vez, sortear pokémons automaticamente de forma assíncrona periodicamente.			
Requisitos Não-Funcionais			

Nome	Restrição	Categoria	Desejável	Permanente
NF12.1 - Raio de distância para o sorteio.	O pokémons sorteados deverão estar em posições aleatórias em um raio de 300 metros do treinador.	Desempenho	()	(X)
NF12.2 - Quantidade de pokémons sorteados.	A cada sorteio, dez pokémons aleatórios devem ser colocados no mapa.	Desempenho	()	(X)
NF12.3 - Critério para escolha de pokémons conforma raridade.	A cada sorteio, dentre os dez pokémons aleatórios sorteados: - 5 ou 6 deverão ser “COMUNS”; - 3 deverão ser “INCOMUNS”; - 1 deverá ser “RARO”; - e 0 ou 1 deverá ser “LENDÁRIO”. Caso um Pokémon lendário seja sorteado, 5 pokémons comuns serão sorteados, senão serão 6.	Desempenho	()	(X)
NF12.4 - Critério para o aparecimento de Pokémon lendário.	A cada sorteio, um Pokémon lendário aleatório será sorteado se: - no sorteio anterior pokémons lendários não tenham sido sorteados; - e dois números randômicos entre 1 e 100 sorteados automaticamente pelo aplicativo sejam pares; - e a soma dos minutos e segundos do relógio do sistema no momento do sorteio seja ímpar.	Desempenho	()	()
NF12.5 - Periodicidade dos sorteios.	Pokémons deverão ser sorteados a cada 3 minutos. A cada novo sorteio, os pokémons sorteados anteriormente saem do mapa, sendo substituídos pelos 10 novos.	Desempenho	()	(X)

Tabela 13 - F13: Chamar Pokémon selvagem para batalhar

F13 - Chamar Pokémon selvagem para batalhar	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Ao clicar em um Pokémon sorteado presente no mapa principal, uma batalha poderá ser iniciada.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF13.1 - Proximidade para iniciar a batalha.	Uma batalha será iniciada caso o treinador esteja a uma proximidade de pelo menos 40 metros do pokémon.	Desempenho	()	(X)
NF13.2 - Aviso de aproximação mínima.	Quando o treinador tentar batalhar com um Pokémon e ele estiver longe, o aplicativo deverá avisar quantos metros o treinador precisa se aproximar do pokémon.	Usabilidade	()	(X)

Tabela 14 - F14: Verificar a existência de conexão com a Internet no Início de uma batalha

F14 - Verificar a existência de conexão com a Internet no Início de uma batalha.	Oculto (X)	Desejável (X)	Permanente ()	
Descrição: Verificar a existência de conexão com Internet ao clicar em um Pokémon no mapa principal antes de iniciar a batalha.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF14.1 - Avisar ao usuário que não existe conexão com a Internet.	Caso não exista conexão, exibir uma mensagem de alerta para o usuário e manter o aplicativo aberto.	Confiabilidade	(x)	()
NF14.2 - Fuga de Pokémon selvagem da batalha.	Caso não exista conexão, o marcador do Pokémon deverá ser removido do mapa após o click e o som	Confiabilidade	(x)	()

	de fuga do Pokémon deverá ser tocado.			
--	---------------------------------------	--	--	--

Tabela 15 - F15: Tocar música da batalha pokémon

F15 - Tocar música da batalha pokémon.	Oculto (X)	Desejável ()	Permanente (X)	
Descrição: Ao iniciar uma batalha com um Pokémon, tocar o tema de batalha.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF15.1 - Configurar Looping.	A música deverá ficar repetindo em loop enquanto o jogo estiver aberto.	Interface	()	(X)
NF15.2 - Parar a música.	A música da batalha deverá ser parada se o usuário sair da batalha ou capturar o Pokémon.	Interface	()	(X)

Tabela 16 - F16: Posicionar um pokémon na tela de captura

F16 - Posicionar um pokémon na tela de captura	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Ao iniciar uma batalha com um Pokémon, o mesmo deverá ser posicionado na tela de acordo com as medições do Giroscópio.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF16.1 - Posição inicial e dimensão do pokémon.	Todo Pokémon estará inicialmente localizado no centro da tela, ocupando uma largura de 50% da mesma e uma altura proporcional.	Interface	()	(X)
NF16.2 - Deslocamento do Pokémon na tela.	Conforme o usuário girar o celular em seus três eixos (X, Y e Z), a posição e rotação do	Usabilidade	()	(X)

	pokémon na tela serão atualizadas.			
NF16.3 - Definir escala para conversão de graus para pixel e pixel para graus.	Utilizar de uma escala proporcional para mover uma imagem em pixels, quando a medição do Giroscópio se dão em ângulos.	Implementação	()	()
NF16.4 - Campo de Visão	A largura e a altura da tela do celular (resolução em pixel) deverão representar 72°	Implementação	()	()
NF16.5 - Retornar Pokémon para a tela.	A imagem do Pokémon deverá retornar gradativamente para a tela quando os giros acumulados nos eixos X ou Y se aproximarem de 360°	Implementação e Interface	()	()

Tabela 17 - F17: Realidade Aumentada

F17 - Realidade Aumentada	Oculto (X)	Desejável ()	Permanente (X)
Descrição: Ao iniciar uma batalha com um Pokémon, a imagem do mesmo deverá aparecer sobreposta às imagens feitas pela câmera do dispositivo móvel em tempo real (Camera Preview).			

Tabela 18 - F18: Arremessar Pokébola

F18 - Arremessar Pokébola	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Permitir ao treinador mover a pokébola pela tela por meio do touch screen.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF18.1 - Som de arremesso da pokebola.	Sempre que a pokébola for arremessada, tocar o som de arremesso da mesma.	Interface	()	(X)
NF18.2 - Retornar a pokébola para a posição inicial.	Caso o usuário erre o arremesso da pokébola, jogando-a para fora da	Usabilidade	()	(X)

	tela, a mesma deverá ser reposicionada no centro inferior da tela. Um aviso deverá ser dado ao usuário solicitando que ele tente novamente.			
NF18.3 - Movimentação automática da pokébola.	Além de poder ser movida livremente pelo usuário por meio do touch screen, a pokébola deverá se mover automaticamente na tela após o usuário retirar o dedo da mesma. Para isso deverá ser calculada a velocidade da pokébola em pixel/segundo nos planos X e Y da tela.	Implementação	()	()

Tabela 19 - F19: Verificar se Pokémon da batalha é inédito

F19 - Verificar se Pokémon da batalha é inédito	Oculto (X)	Desejável ()	Permanente (X)	
Descrição: Ao iniciar uma batalha, o aplicativo deverá informar ao usuário se o Pokémon em questão já foi capturado anteriormente.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF19.1 - Label informando se Pokémon é novo ou conhecido.	No canto superior esquerdo da tela de captura, deverá ser exibido para o usuário se o Pokémon é novo ou conhecido.	Interface	()	(X)
NF19.2 - Label com o nome do pokémon.	No canto superior direito da tela de captura, deverá ser exibido para o usuário o nome do Pokémon que está na batalha.	Interface	()	(X)

Tabela 20 - F20: Capturar pokémon

F20 - Capturar pokémon	Oculto (X)	Desejável ()	Permanente (X)
-------------------------------	------------	---------------	----------------

Descrição: Um Pokémon deverá ser capturado quando houver interseção entre o espaço ocupado pelo mesmo na tela e o espaço ocupado pela pokébola após a mesma ser movida pelo usuário.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF20.1 - Som de quique da pokebola.	Quando houver interseção entre o Pokémon e a pokébola, o som de quique da pokébola deverá ser tocado 3 vezes em loop.	Interface	()	(X)
NF20.2 - Entrada do Pokémon na pokébola.	Quando houver interseção entre o Pokémon e a pokébola, a imagem do Pokémon deverá ser substituída pela imagem de uma explosão, permanecendo assim por 350 milissegundos. Após esse tempo a imagem da explosão deverá ficar invisível.	Interface	()	(X)
NF20.3 - Som de sucesso na captura.	Após a pokébola balançar três vezes, o som de sucesso da captura deve ser tocado.	Interface	()	(X)
NF20.4 - Balançar a pokébola.	Quando houver interseção entre o Pokémon e a pokébola, a imagem da Pokébola deverá girar para a esquerda e direita três vezes, em um ângulo de 20° e -20°	Interface	()	(X)
NF20.5 - Avisar que Pokémon foi capturado	Exibir uma mensagem de alerta após o som de sucesso, avisando ao usuário que o pokémon foi capturado.	Interface	()	()

Tabela 21 - F21: Persistir Pokémon capturado

F21 - Persistir Pokémon capturado	Oculto (X)	Desejável ()	Permanente (X)	
Descrição: Após a finalização da batalha com um Pokémon capturado, essa captura deverá ser persistida no banco de dados.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente

NF21.1 - Persistência no banco de dados local.	Sempre que um Pokémon for capturado, os dados sobre a captura deverão ser persistidos localmente.	Implementação	()	(X)
NF21.2 - Persistência no banco de dados do servidor.	Sempre que um Pokémon for capturado, os dados sobre a captura deverão ser persistidos também no servidor.	Implementação	(X)	(X)

Tabela 22 - F22: Exibir relatório do perfil do usuário

F22 – Exibir relatório do perfil do usuário	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Após o usuário escolher a opção do menu, deverá ser exibido um relatório contendo o login do usuário, data e horário que o mesmo começou a se aventurar pelo jogo e o total de capturas realizadas desde o início.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF22.1 – Imagem do treinador	A imagem do treinador exibida no perfil deverá corresponder ao sexo do usuário.	Interface	()	(X)

Tabela 23 - F23: Exibir mapa de capturas

F23 – Exibir mapa de capturas.	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Após o usuário escolher a opção do menu, deverá ser exibido um mapa contendo marcadores personalizados nos locais aonde o treinador já capturou um pokémon.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF23.1 – Pokémons exibidos no mapa	Este relatório terá duas variações: - Exibir todos os Pokémon capturados de todas as espécies. - Exibir todos os Pokémon capturados de uma espécie específica.	Interface	()	(X)

NF23.2 – Imagem do marcador	Cada marcador deverá apresentar a imagem do ícone do Pokémon que ele representa.	Interface	()	(X)
NF23.3 – Detalhes da captura	Quando um marcador for tocado pelo usuário, deverão ser exibidos dados com detalhes da captura, como o nome do Pokémon, a data e hora que o mesmo foi capturado.	Interface	()	(X)

Tabela 24 - F24: Exibir PokéAgenda

F24 – Exibir PokéAgenda	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Após o usuário escolher a opção do menu, deverá ser exibida uma lista contendo todas as espécies de pokémon existentes no jogo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF24.1 – Informações sobre os pokémon	Cada linha da lista deverá corresponder à uma espécie de Pokémon, contendo um ícone, o nome e o número da espécie. Se a linha representar uma espécie que nunca foi capturada pelo treinador, o ícone e o nome do Pokémon deverão denotar algo desconhecido.	Interface	()	(X)

Tabela 25 - F25: Tocar música da PokéAgenda

F25 - Tocar música da PokéAgenda	Oculto (X)	Desejável ()	Permanente (X)	
Descrição: Ao abrir a PokéAgenda, tocar a música tema do menu.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF25.1 - Configurar Looping.	A música deverá ficar repetindo em loop enquanto a PokéAgenda estiver aberta.	Interface	()	(X)

NF25.2 - Parar a música.	A música da agenda deverá ser parada se o usuário sair do aplicativo ou voltar para o mapa principal.	Interface	()	(X)
---------------------------------	---	-----------	-----	-----

Tabela 26 - F26: Calcular métricas da PokéAgenda

F26 – Calcular métricas da PokéAgenda.	Oculto (X)	Desejável ()	Permanente (X)
Descrição: Ao exibir a PokéAgenda, deverá ser calculado e informado para o usuário o total de espécies pokémon que ele já capturou e quantas ainda faltam para completar a agenda.			

Tabela 27 - F27: Exibir detalhes de uma espécie na PokéAgenda

F27 – Exibir detalhes de uma espécie na PokéAgenda	Oculto ()	Desejável ()	Permanente (X)	
Descrição: Ao visualizar a PokéAgenda, o usuário deverá ser capaz de clicar sobre as espécies pokémon já capturadas para conhecer maiores detalhes a respeito das mesmas.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF27.1 – Informações detalhadas sobre uma espécie Pokémon.	Os detalhes de cada espécie Pokémon já capturada contemplam foto, número identificador, nome, tipos e quantidade de espécimes capturadas da espécie em questão.	Interface	()	(X)
NF27.2 – Cores utilizadas na identificação de Tipos de pokémon.	Cada um dos 18 possíveis tipos de Pokémon deverá ter uma cor de fundo específica a ser utilizada quando detalhes do Pokémon forem exibidos. <u>Sugestões de cores:</u> Normal: #a8a878 Fire: #f08030 Fighting: #c03028 Water: #6890f0 Flying: #a890f0 Grass: #78c850	Interface	(X)	()

	Poison: #a040a0 Electric: #f8d030 Ground: #e0c068 Psychic: #f85888 Rock: #b8a038 Ice: #98d8d8 Bug: #a8b820 Dragon: #7038f8 Ghost: #705898 Dark: #705848 Steel: #b8b8d0 Fairy: #ee99ac			
--	--	--	--	--

2.2 – Requisitos Suplementares

Tabela 28 - Requisitos Suplementares

Nome	Restrição	Categoria	Desejável	Permanente
S1 - Compatibilidade do aplicativo	Um dispositivo móvel deverá ter no mínimo Android 4.2 Jelly Bean instalado para rodar o aplicativo. (API Level 17)	Compatibilidade	()	(X)
S2 - Orientação das telas	Todas as telas do aplicativo deverão ser travadas na orientação retrato (portrait).	Interface	()	(X)
S3 – Banco de dados pré-populado	Tanto o banco de dados local, quanto o banco de dados do servidor deverão conter previamente ao primeiro uso do aplicativo os dados de todas as espécies de Pokémon e de seus possíveis tipos cadastrados.	Implementação	()	(X)
S4 - Comunicação com Servidor	O Aplicativo irá se comunicar com o servidor por meio de requisições HTTP e JSON.	Interoperabilidade	()	(X)
S5 – Dados locais sobre usuários e pokémons capturados	O banco de dados local deverá persistir apenas os dados de um usuário por sessão. Isso sobrecarregará menos o dispositivo utilizado para jogar, evitando grandes volumes de dados persistidos simultaneamente.	Desempenho	()	(X)

S6 – Multimídia local	Todas as imagens e áudios utilizados no jogo deverão estar contidos na APK do mesmo. Embora isso aumente o tamanho final da APK, trará grande ganho de desempenho.	Desempenho	()	(X)
-----------------------	--	------------	-----	-----

3 – PROJETO DE INTERFACE

3.1 – Telas do Sistema



Figura 1 - Tela de início (Splash)

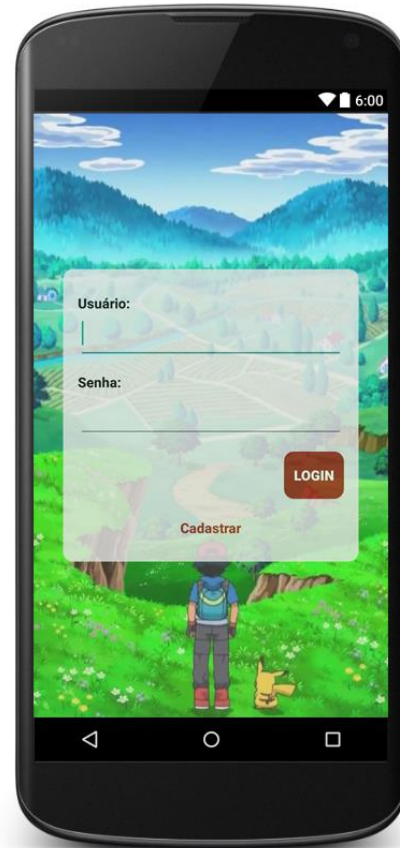


Figura 2 - Tela de Login



Figura 3 - Tela de cadastro

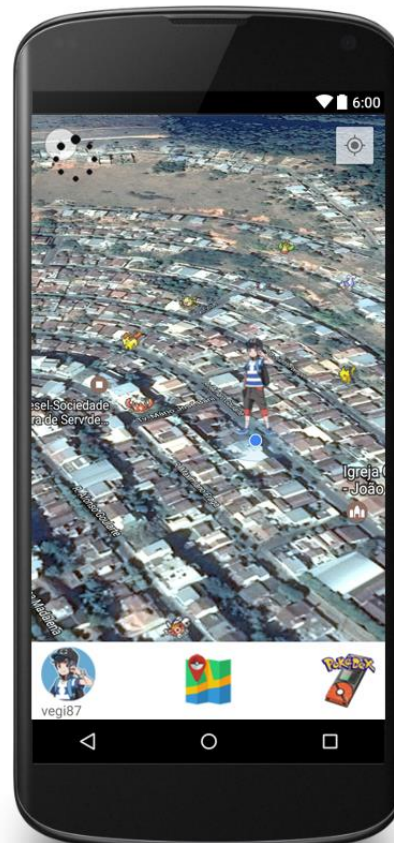


Figura 4 - Tela do Mapa Principal



Figura 5 - Tela de captura

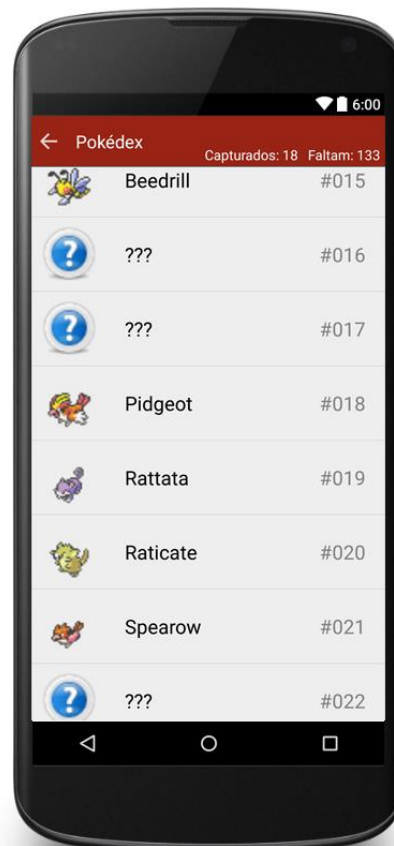


Figura 6 - Tela do Pokédex

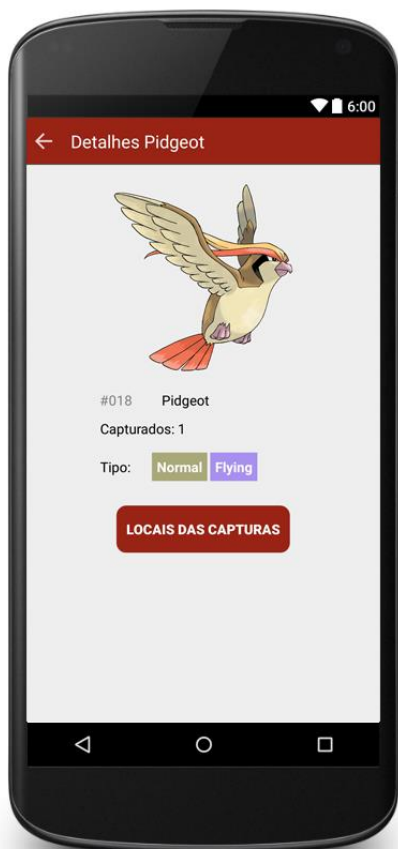


Figura 7 - Tela de detalhes do pokémon

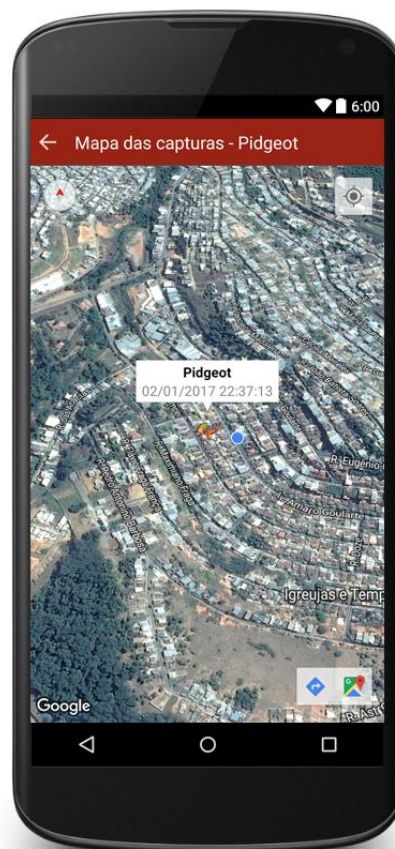


Figura 8 - Tela do Mapa de captura de um pokémon

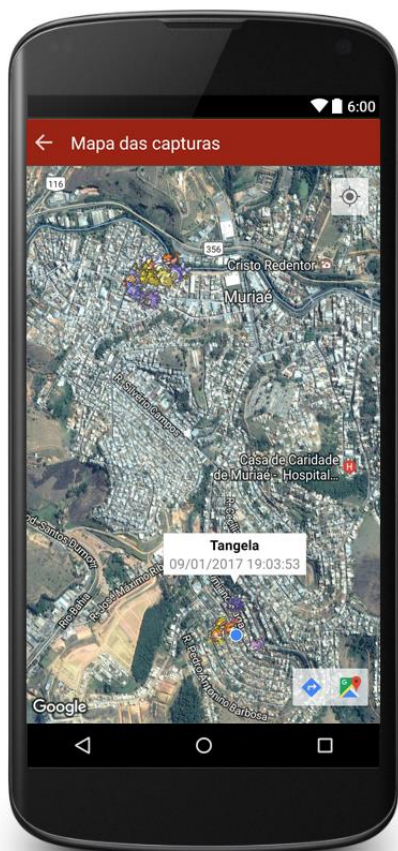


Figura 9 - Tela do Mapa de todas as capturas

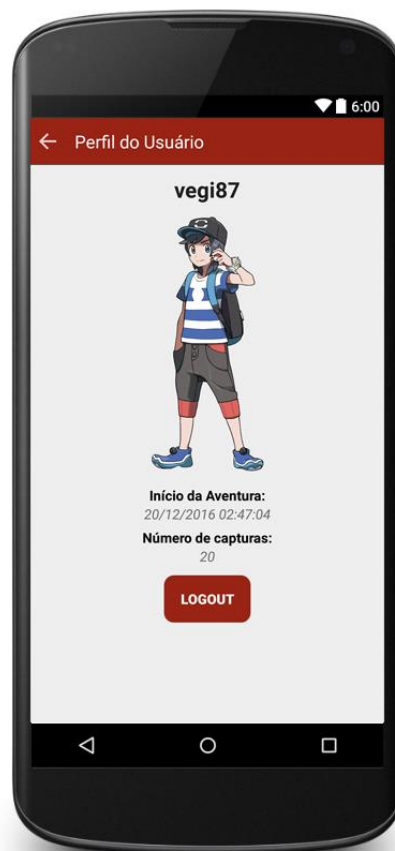


Figura 10 - Tela do Perfil do usuário

3.2 – Fluxo de Navegação das Telas (Storyboard)



Figura 11 - Fluxo de navegação entre as telas

4 – PROJETO DE PERSISTÊNCIA

4.1 – Banco de Dados Local

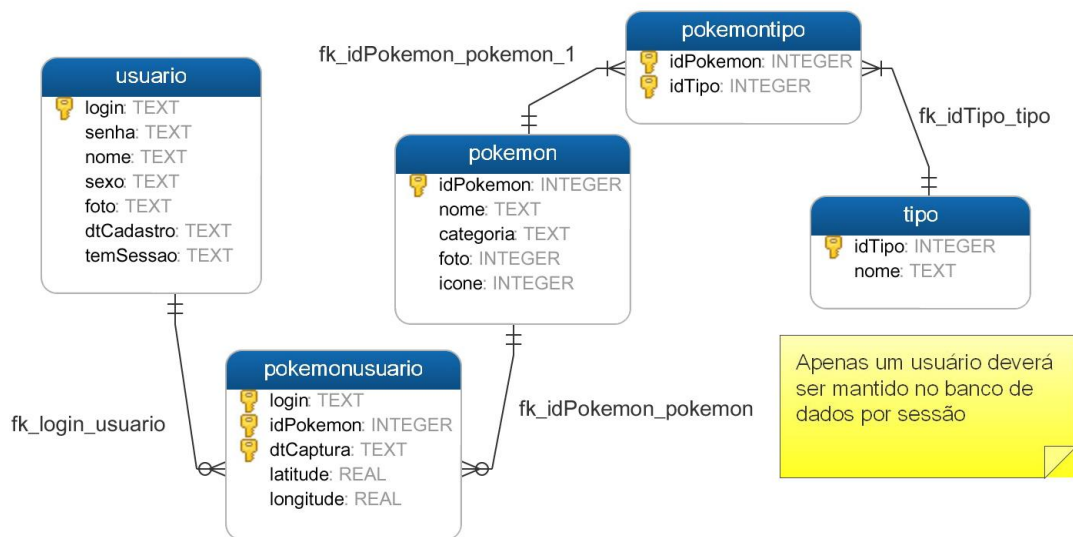


Figura 12 - Modelo Entidade Relacionamento do banco de dados local

4.2 – Banco de Dados do Servidor

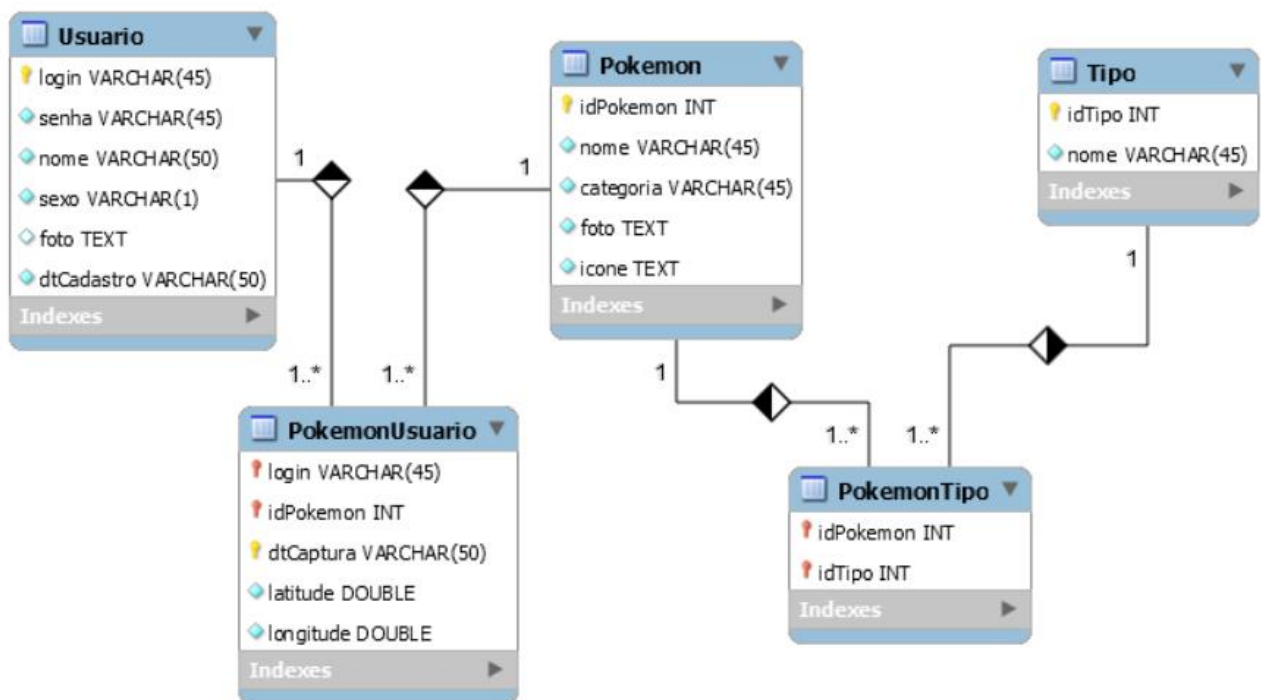


Figura 13 - Modelo Entidade Relacionamento do banco de dados do servidor

5 – PROJETO DO SISTEMA

5.1 – Diagrama de Casos de Uso

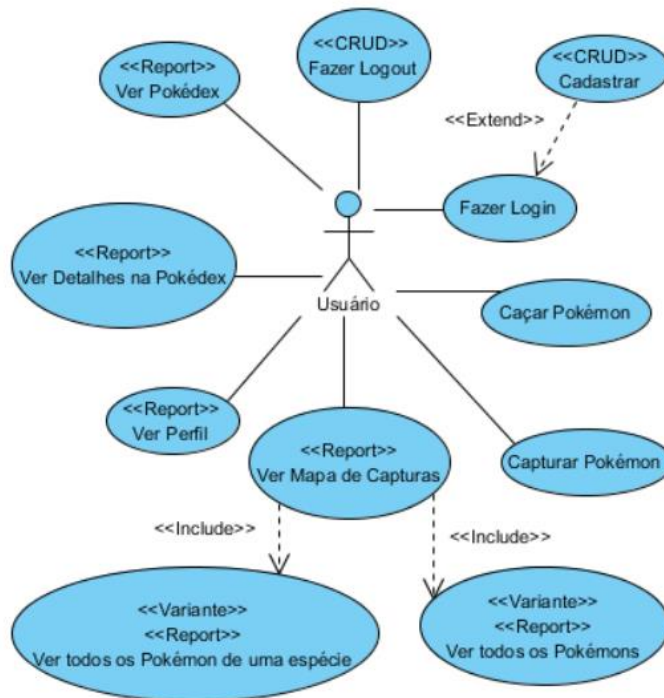


Figura 14 - Diagrama de casos de uso

5.1.1 – Listagem de casos de uso de alto nível

Tabela 29 - Casos de uso de alto nível

Nome	Referências Cruzadas
Fazer Login	F1, F2, F3, F4, F5
<<CRUD>> Cadastrar	F6, F7
<<CRUD>> Fazer Logout	F8
Caçar Pokémon	F9, F10, F11, F12, F13, F14
Capturar Pokémon	F15, F16, F17, F18, F19, F20, F21
<<Report>> Ver Perfil	F22
<<Report>> Ver Mapa de Capturas	F23
<<Report>> Ver todos os Pokémon de uma espécie	F23
<<Report>> Ver todos os Pokémons	F23
<<Report>> Ver Pokédex	F24, F25, F26, F27
<<Report>> Ver detalhes na Pokédex	F27

5.2 – Diagrama de Pacotes

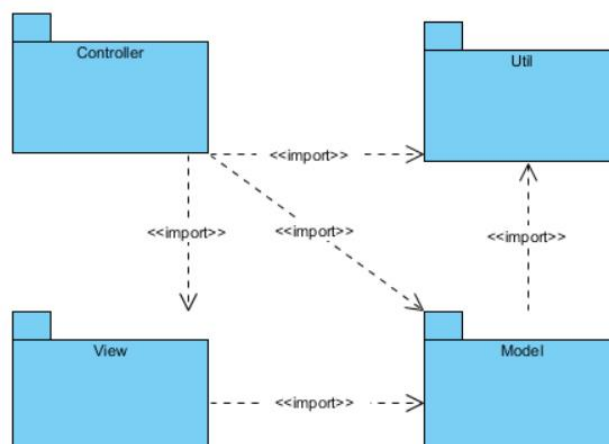


Figura 15 - Diagrama de Pacotes do projeto

5.3 – Diagramas de Classes de Projeto

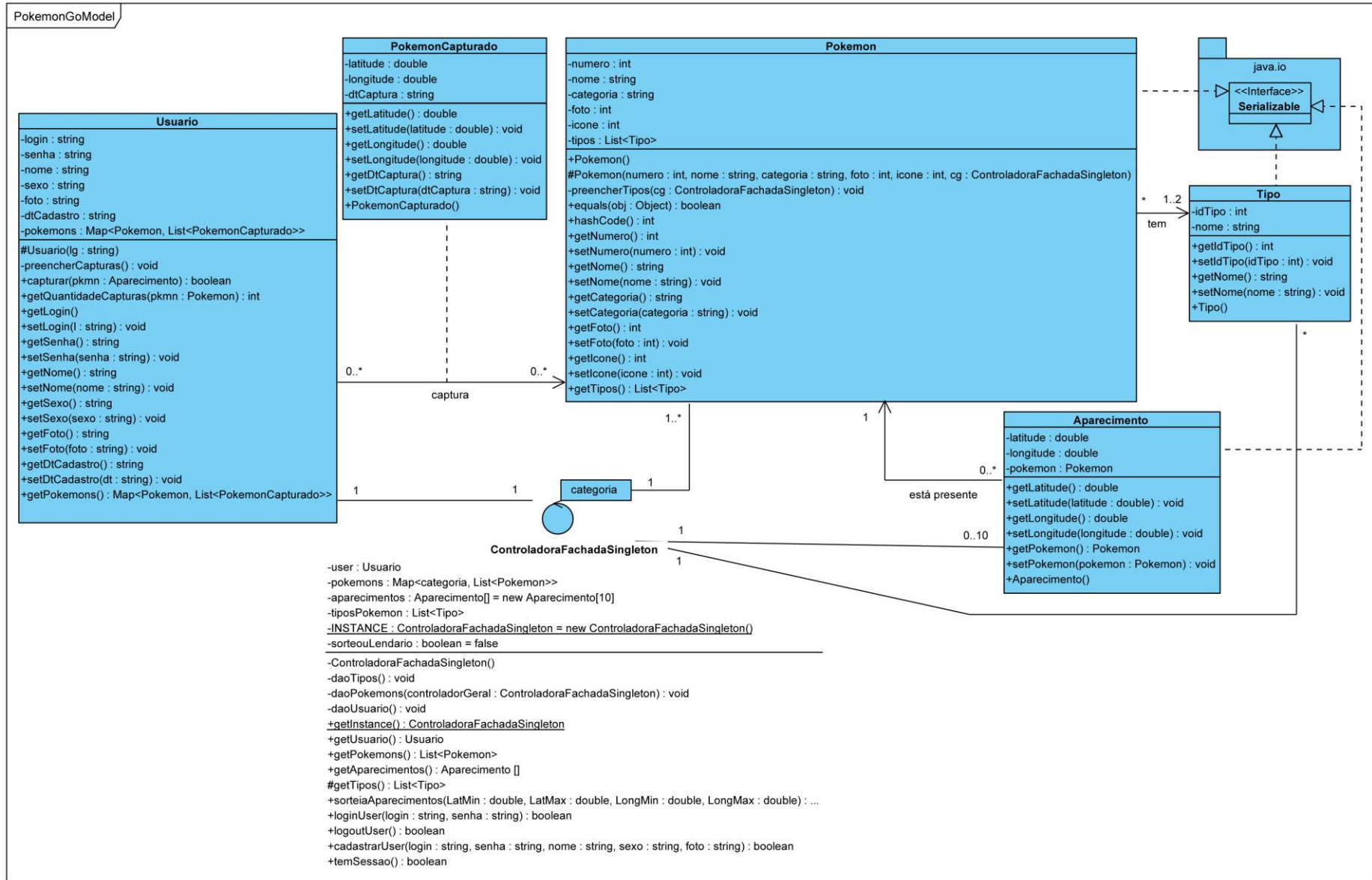


Figura 16 - Diagrama de classes de projeto do pacote Model

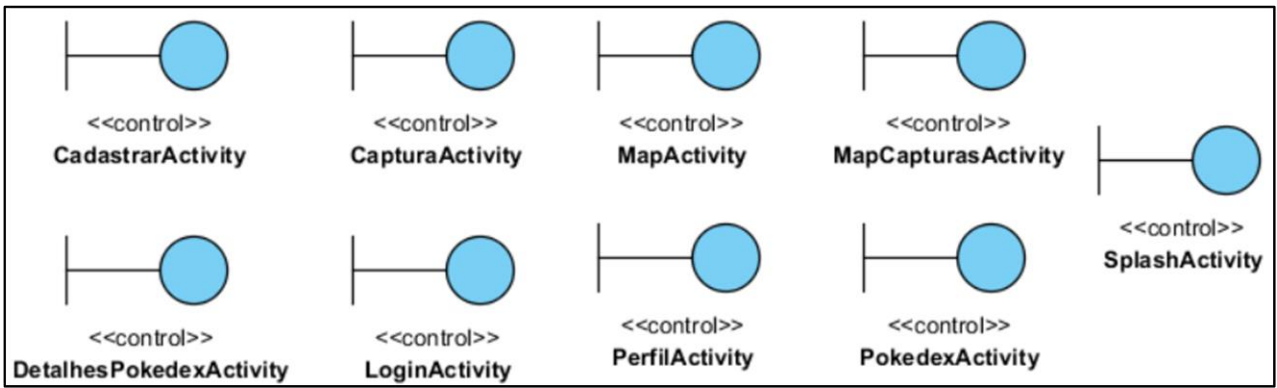


Figura 17 - Diagrama de classes de projeto do pacote Controller

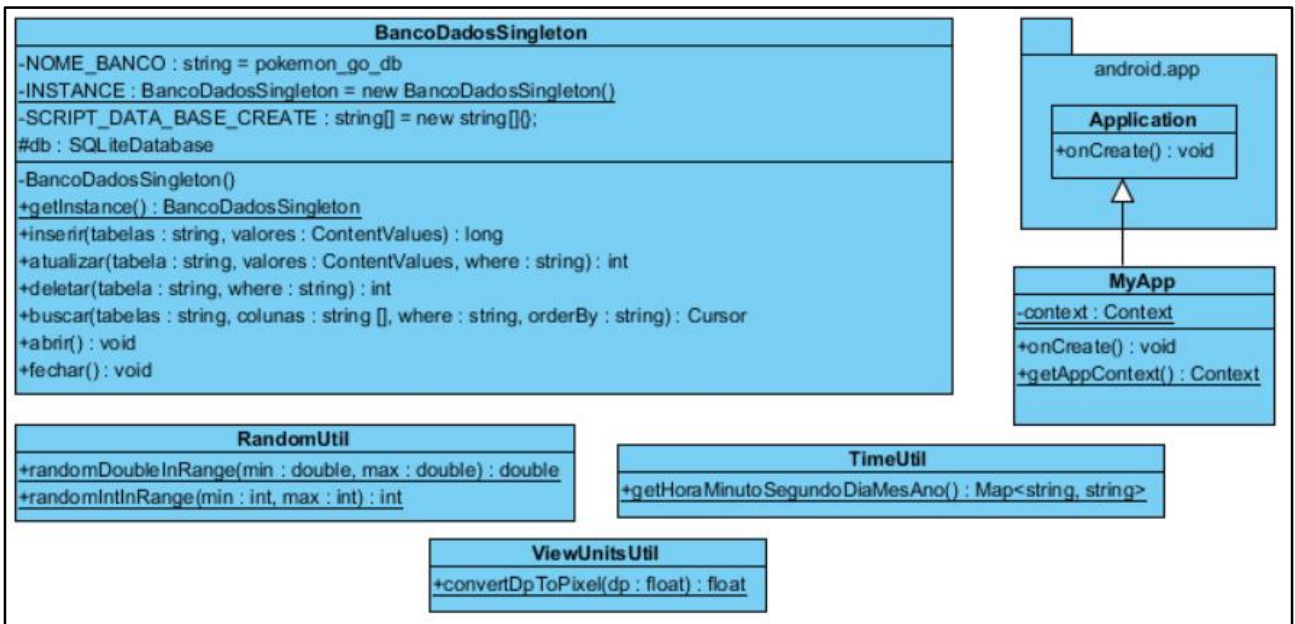


Figura 18 - Diagrama de classes de projeto do pacote Util



Figura 19 - Diagrama de classes de projeto do pacote View

5.4 – Diagramas de Sequência (Casos de Uso Expandidos)

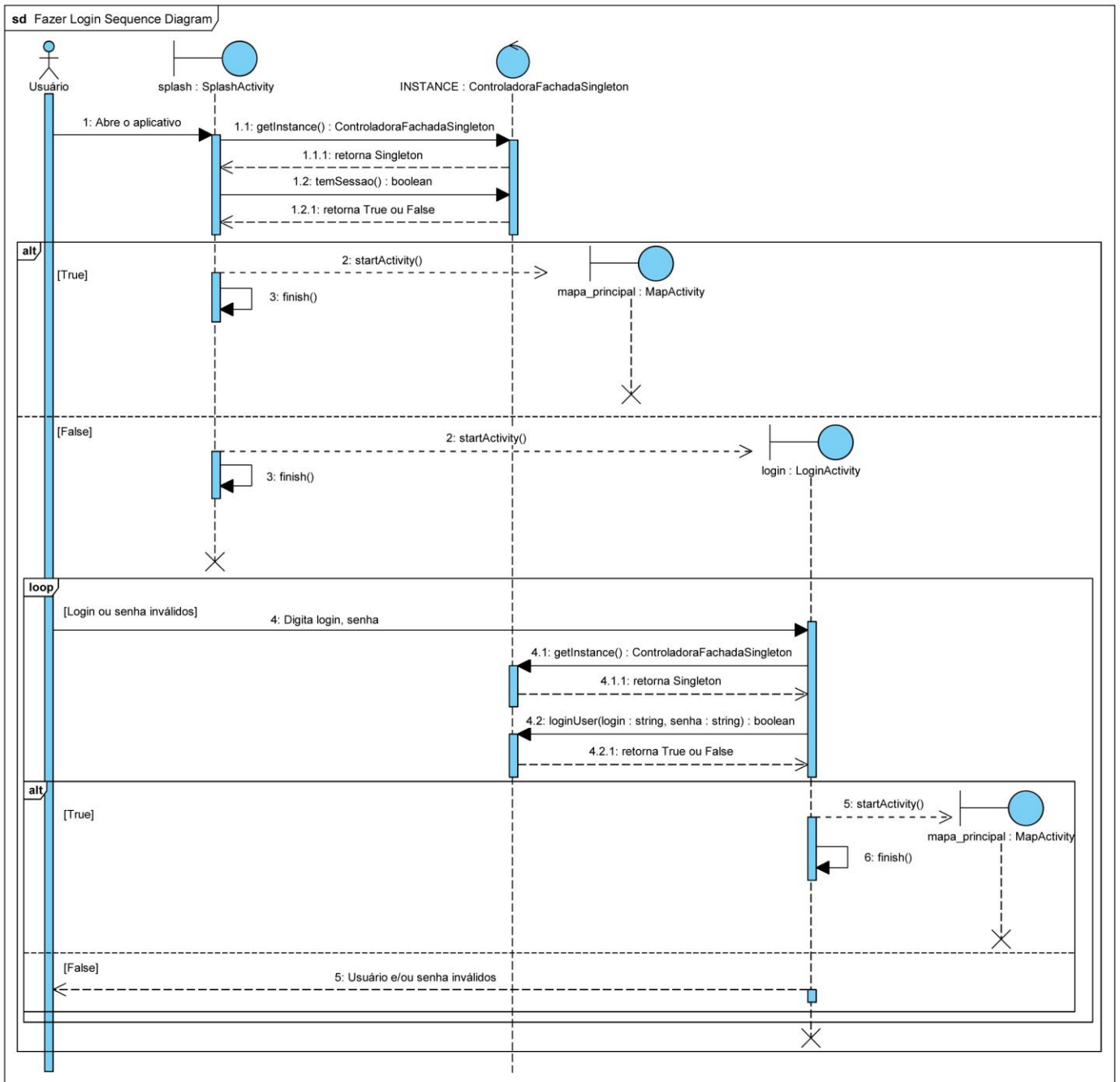


Figura 20 - Diagrama de sequência do caso de uso “Fazer login”

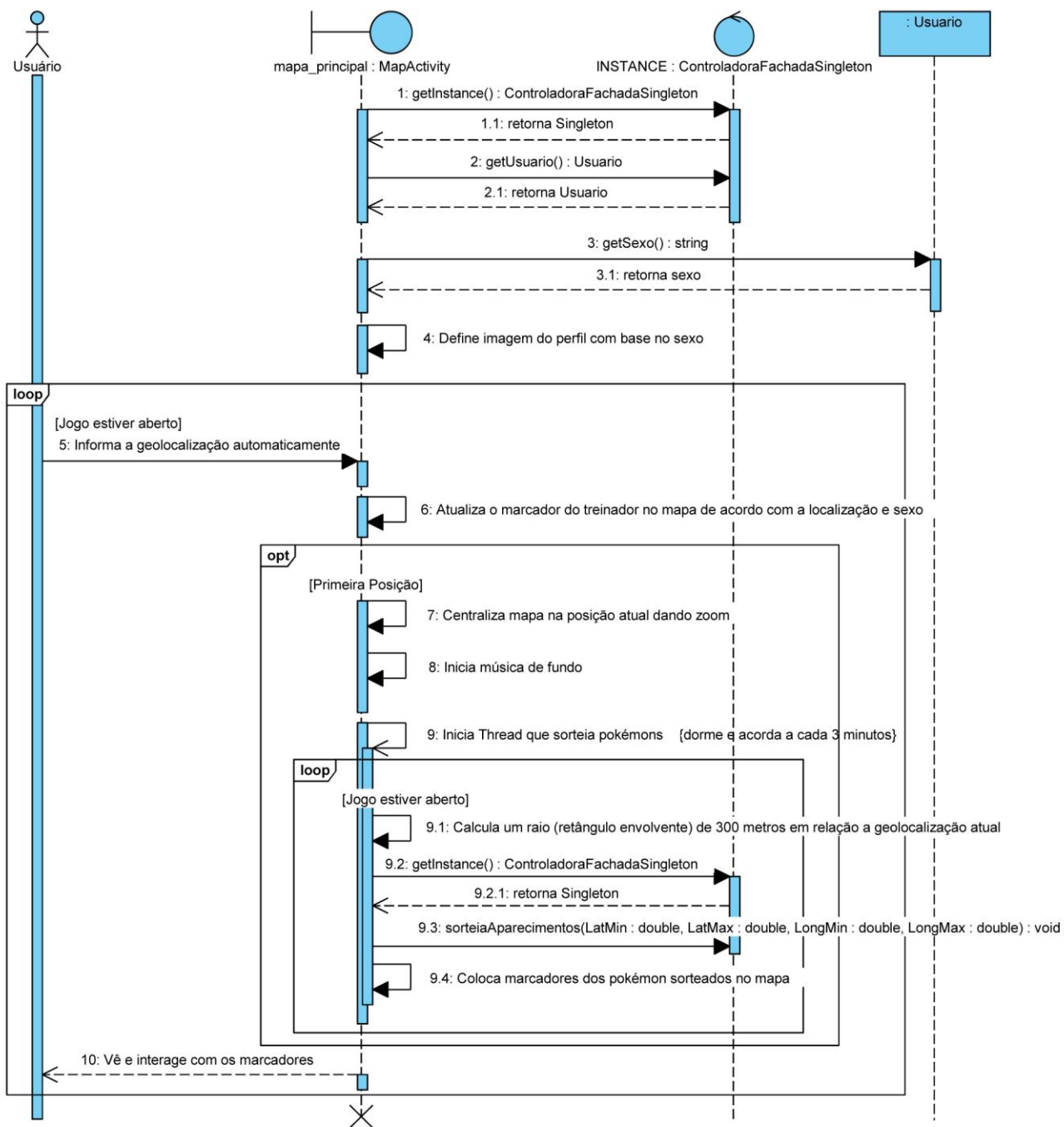


Figura 21 - Diagrama de seqüência do caso de uso “Caçar pokémon”

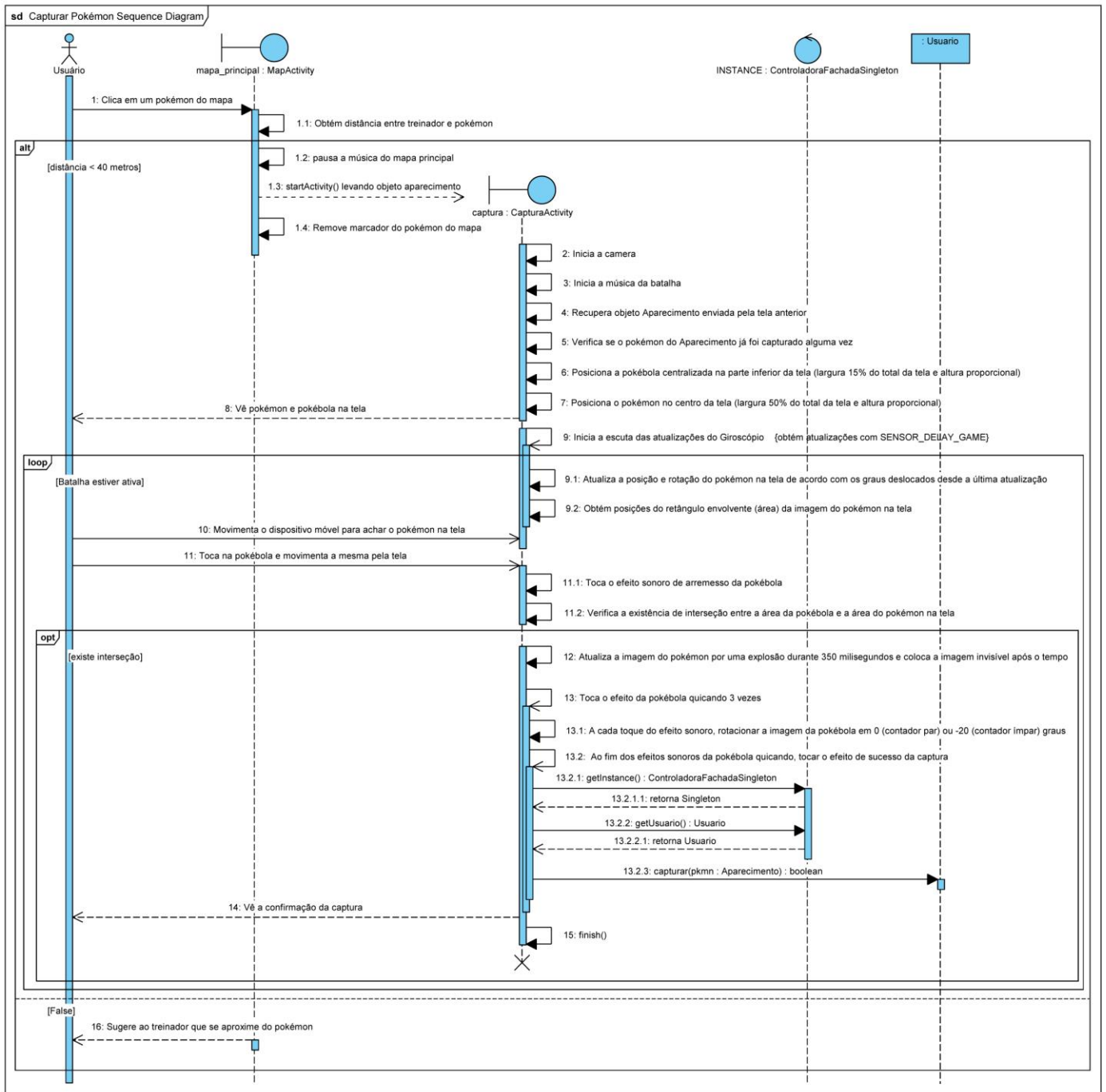


Figura 22 - Diagrama de sequência do caso de uso “Capturar pokémon”

5.5 – Diagramas de Comunicação (Contratos de Operação e Consulta)

5.5.1 – Classe “ControladoraFachadaSingleton”

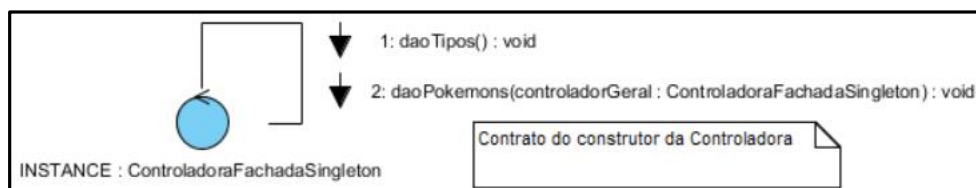


Figura 23 - Diagrama de comunicação do construtor da classe ControladoraFachadaSingleton

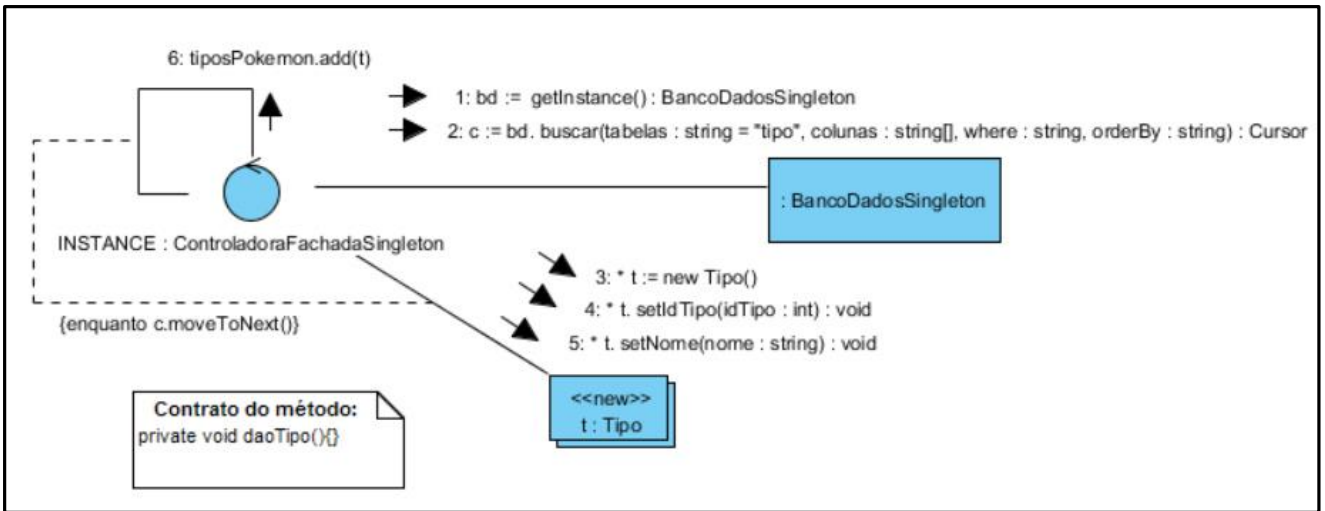


Figura 24 - Diagrama de comunicação do método daoTipo()

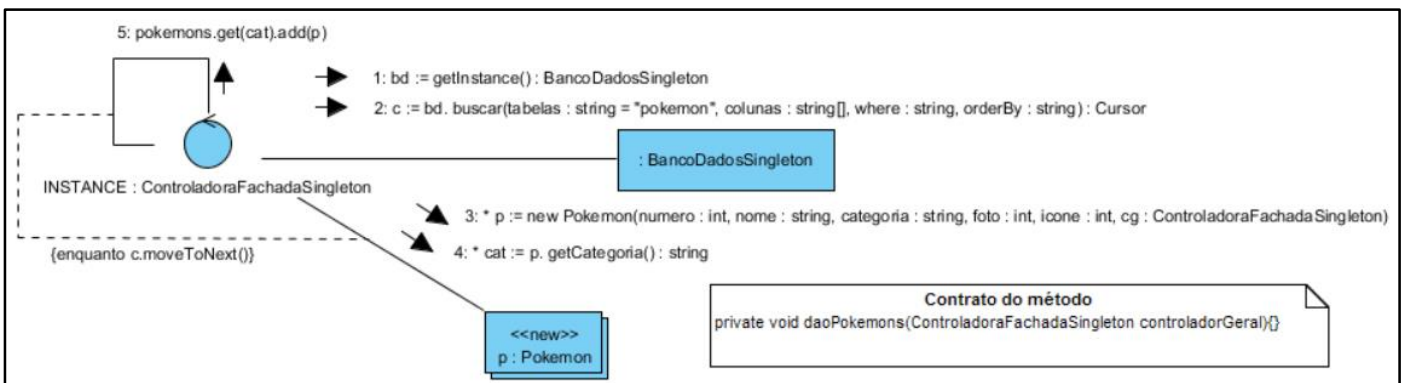


Figura 25 - Diagrama de comunicação do método daoPokemons()

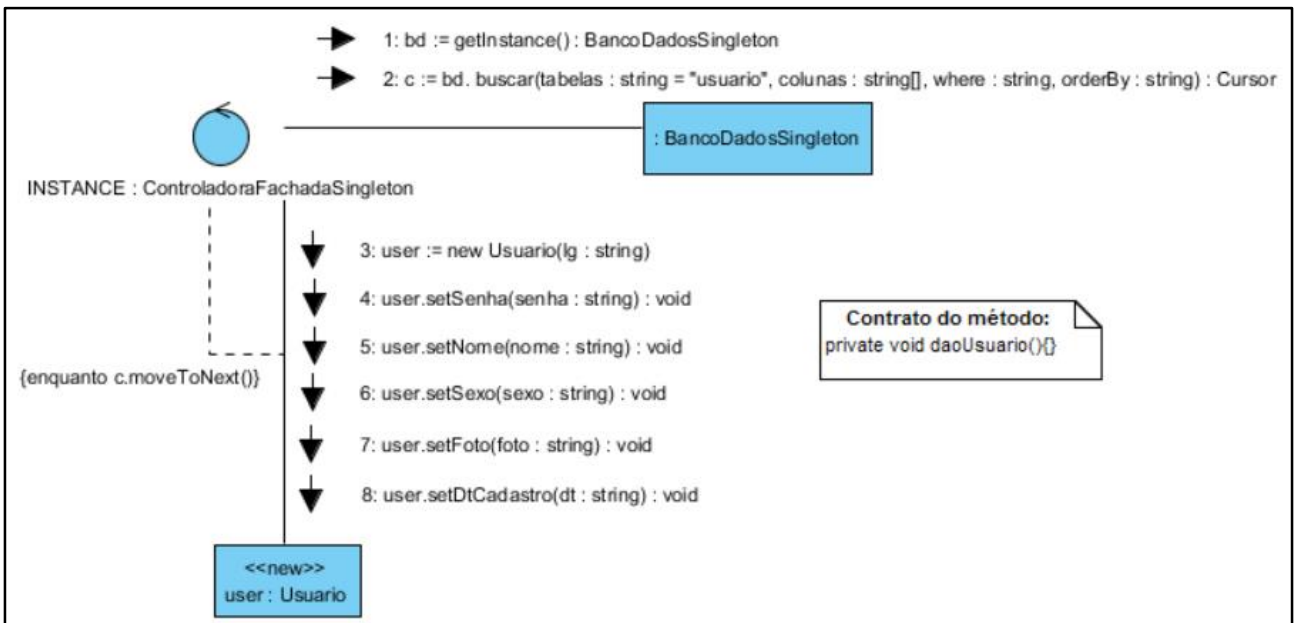


Figura 26 - Diagrama de comunicação do método daoUsuario()

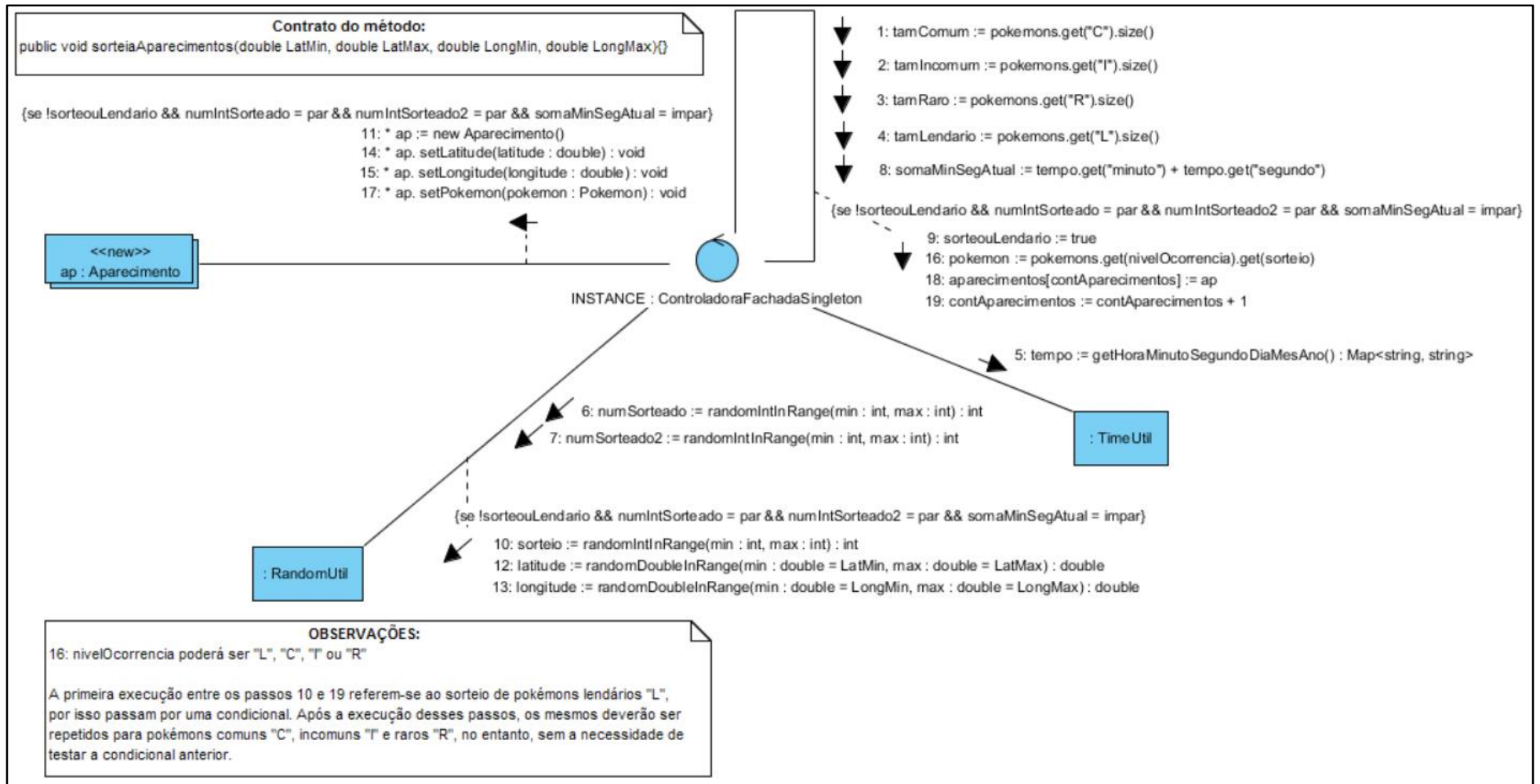


Figura 27 - Diagrama de comunicação do método sorteiaAparecimentos()

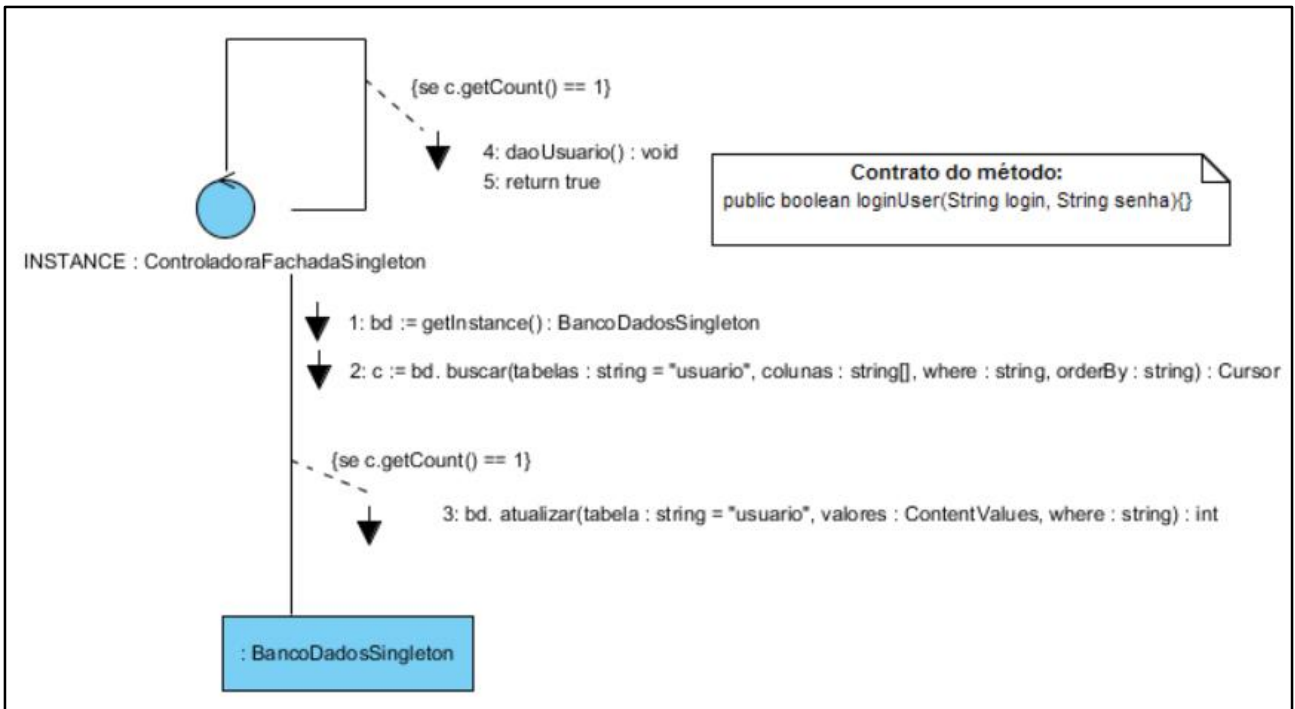


Figura 28 - Diagrama de comunicação do método `loginUser()`

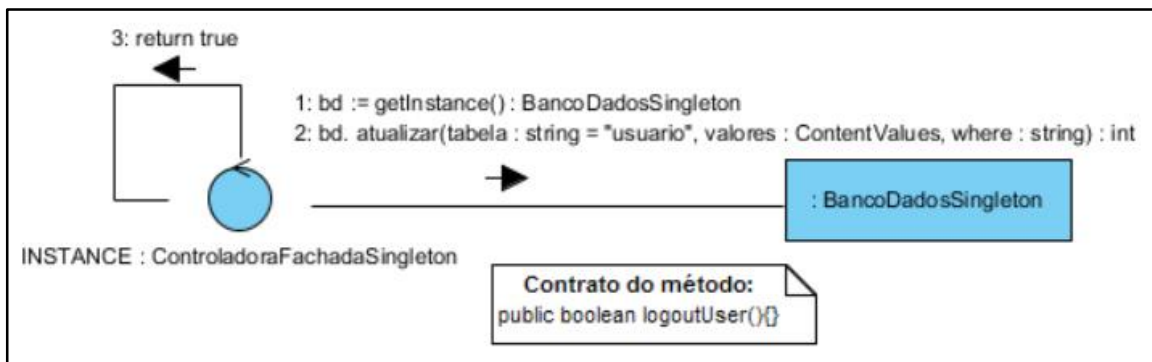


Figura 29 - Diagrama de comunicação do método `logoutUser()`

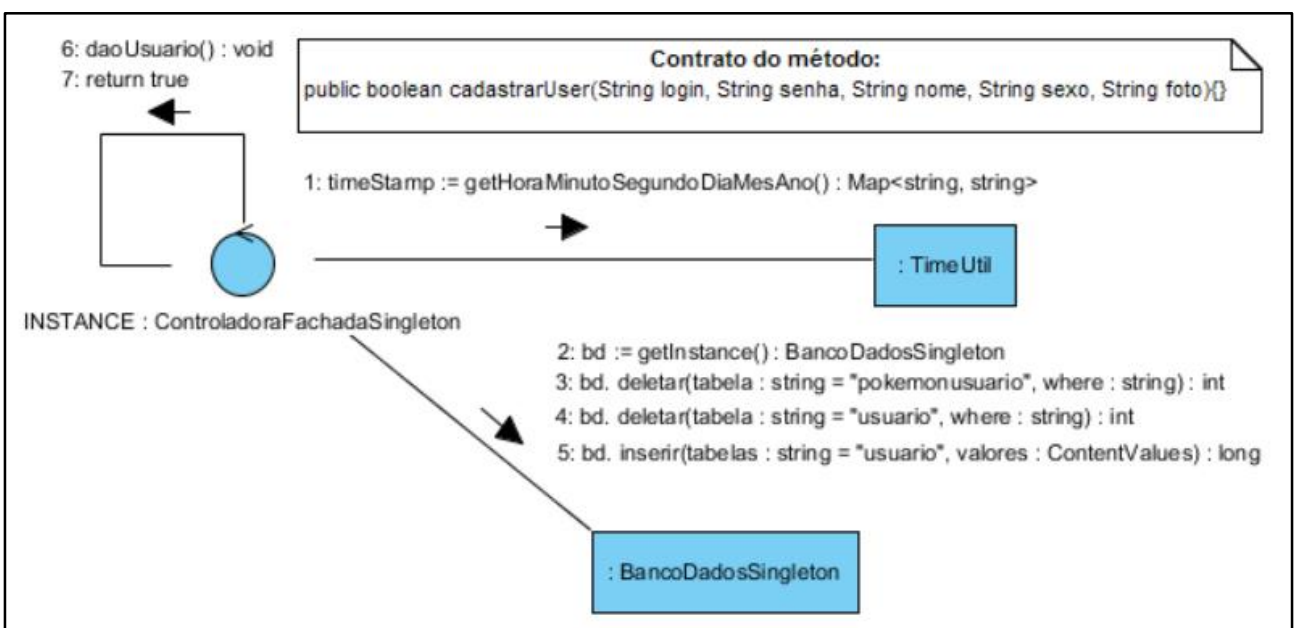


Figura 30 - Diagrama de comunicação do método `cadastroUser()`

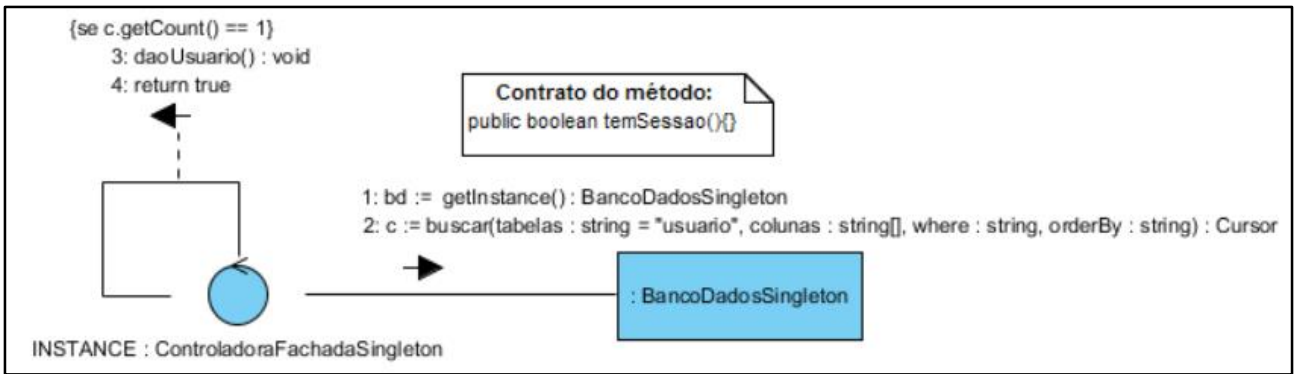


Figura 31 - Diagrama de comunicação do método temSessao()

5.5.2 – Classe “Usuario”

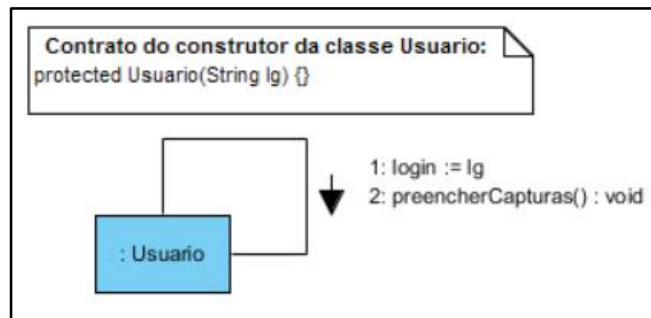


Figura 32 - Diagrama de comunicação do método construtor da classe Usuario

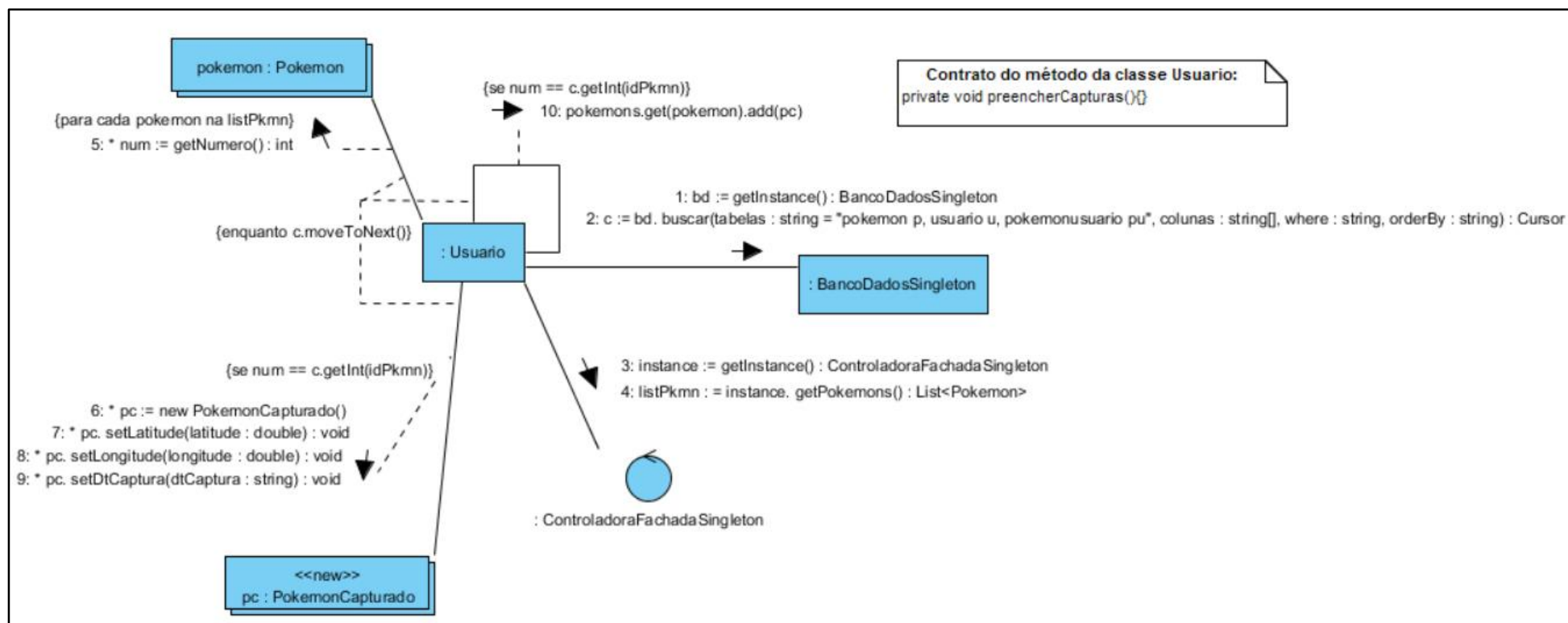


Figura 33 - Diagrama de comunicação do método preencherCapturas()

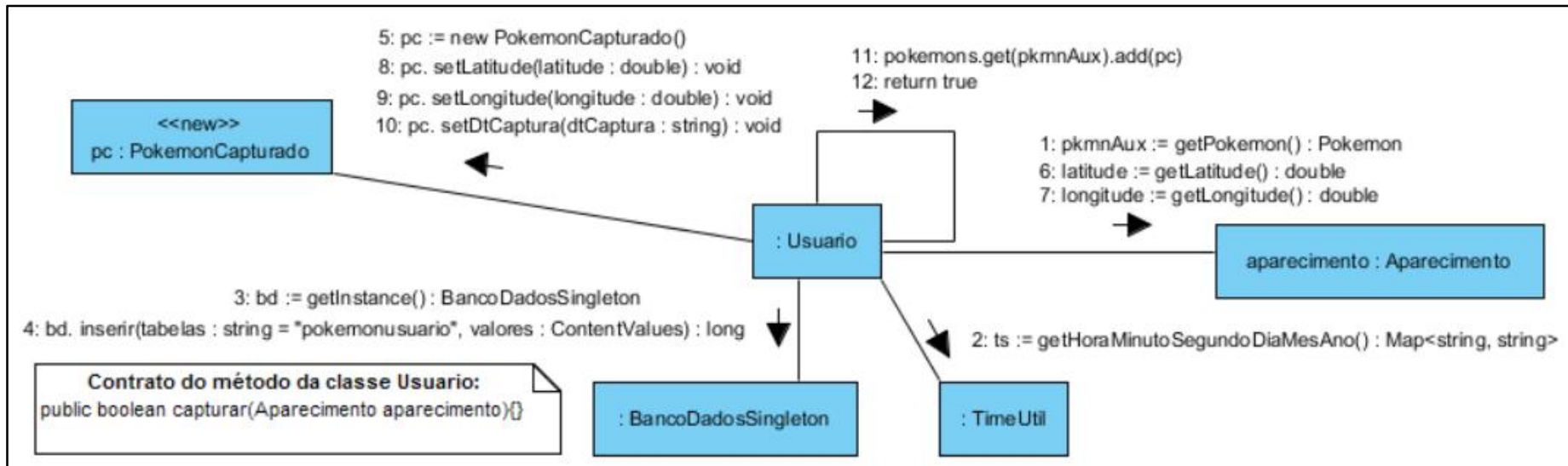


Figura 34 - Diagrama de comunicação do método capturar()

5.5.3 – Classe “Pokemon”

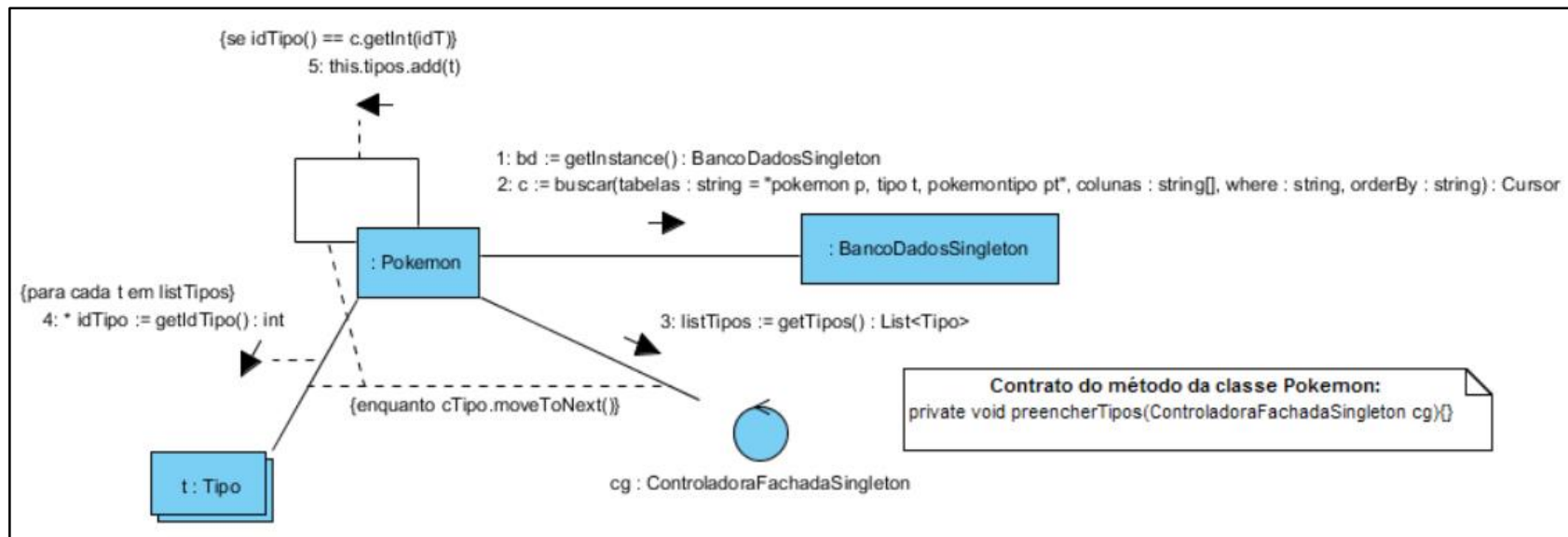


Figura 35 - Diagrama de comunicação do método preencherTipos()

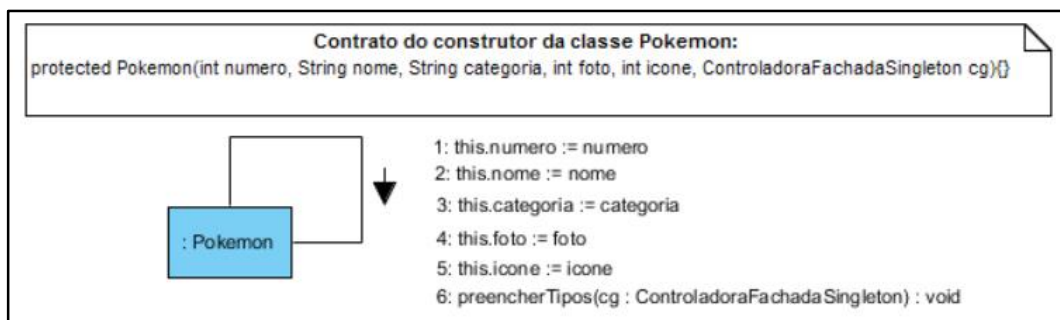


Figura 36 - Diagrama de comunicação do método construtor da classe Pokemon

APÊNDICES e ANEXOS

Encontram-se disponíveis no PVAnet uma série de arquivos complementares que ajudarão na implementação do projeto proposto.

- **ANEXO A - Imagens:** As imagens de todos os pokémon, seus ícones, assim como todas as imagens utilizadas para compor as telas e menus do aplicativo encontram-se compactadas no PVAnet (*ImagensPokemonGoClone.zip*)
- **ANEXO B - Áudios:** Todas as músicas e efeitos sonoros utilizados no aplicativo encontram-se compactadas no PVAnet (*AudiosPokemonGoClone.zip*)
- **APÊNDICE A – Scripts de criação do banco de dados:** Os scripts *.sql* para a criação do banco de dados local do aplicativo (SQLite), assim como para a criação do banco de dados do servidor (MySQL), encontram-se compactados no PVAnet. Esses scripts populam as tabelas Pokémon e Tipos, além de criar todas as tabelas e seus relacionamentos. (*ScriptsSQL.zip*)
- **APÊNDICE B - APKs:** As APKs de instalação do aplicativo Pokémon GO Clone encontram-se disponíveis para download no PVAnet. As mesmas poderão ser instaladas para realização de testes livremente em qualquer dispositivo Android que possua giroscópio. (*pokemonGoClone-v0.4.apk* e *pokemonGoClone-v0.6.apk*).
 - **Observação:** A versão 0.4 não possui nenhum tipo de comunicação com o servidor, enquanto a versão 0.6 exige conexão com a Internet para sincronizar com o servidor e funcionar plenamente.
- **APÊNDICE C - UML:** O projeto do sistema em UML encontra-se disponível no PVAnet em formato compatível com a ferramenta Visual Paradigm¹. (*pokemonGoOffline.vpp*)

¹ <https://www.visual-paradigm.com/download/community.jsp>