

FORKIF

Allan Kardec Medeiros de Oliveira, Renato de Carvalho Lima

allankomedeiros@gmail.com, renatolimarn@hotmail.com

IFRN, Natal, Rio Grande do Norte, 2011

Esse relatório faz referência ao registro formal do início do desenvolvimento do software ForkIF. Nesse primeiro documento será explorada a parcela do desenvolvimento referente à elicitação de requisitos e a modelagem via diagramação UML.

1. INTRODUÇÃO

Uma das funções mais ativas no ambiente das redes de computadores, principalmente quando se considera a usabilidade por parte dos usuários comuns, faz referência ao acesso a páginas de internet. Para que o usuário tenha acesso a essas páginas faz-se necessário a criação de um ambiente de rede que dê suporte ao armazenamento dos arquivos do site, bem como ao gerenciamento das requisições de acesso.

Dentro desse contexto, mas com foco no desenvolvimento de software, o ForkIF será lançado. O ForkIF será um software em formato de jogo da forca baseado nos princípios de cliente-servidor, que fará uso dos principais recursos das redes de computadores, mais especificamente o recurso de sockets.

Nessa primeira parte, o foco será centralizado no levantamento de requisitos e em todos os principais processos que o envolve. Assim sendo, abordaremos a elicitação de requisitos e o desenvolvimento da diagramação desses requisitos via UML.

2. OBJETIVOS

Desenvolver o levantamento e a modelagem dos requisitos de um jogo da forca que interaja em um ambiente de rede com base nos princípios da programação orientada a objetos.

3. PROCEDIMENTO EXPERIMENTAL

3.1 Materiais

Como suporte ao registro do desenvolvimento da elicitação de requisitos será usado o software para modelagem de diagramas Astah community.

3.2 Métodos

A metodologia para o desenvolvimento da elicitação de requisitos deste projeto baseia-se nos princípios da análise orientada a objetos mesclados ao desenvolvimento de protótipos gráficos que servirão e servirão de referência visual e concreta para o levantamento das necessidades.

4. ANÁLISE DE REQUISITOS

Toda a análise de requisitos desenvolvida sobre o ForkIF foi totalmente baseada no desenvolvimento de protótipos gráficos, conforme o ilustrado a seguir:

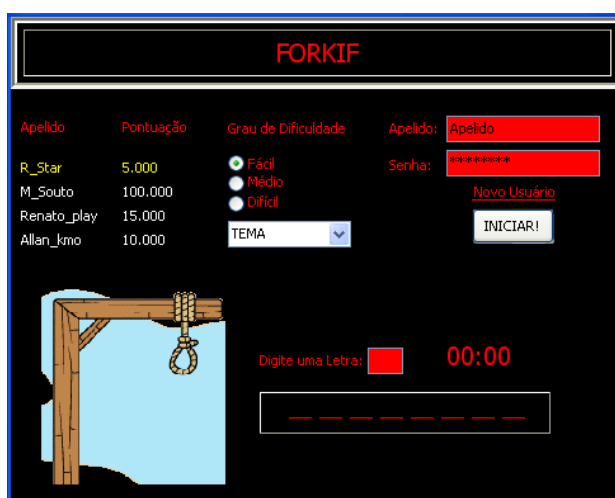


Fig. 1 Protótipo do ForkIF

Para facilitar a elaboração e o levantamento dos requisitos inerentes ao projeto, a prototipagem

foi pensada em módulos, conforme ilustração a seguir:

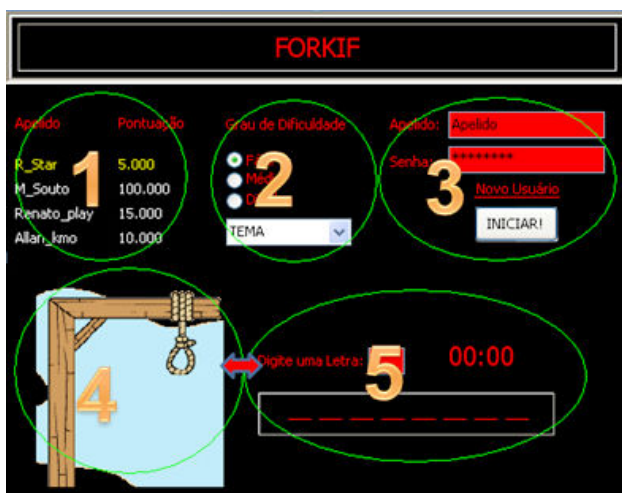


Fig. 2 Protótipo do ForkIF pensado em módulos

Os módulos em destaque são: Módulo para classificação do jogador; Módulo para filtragem da palavra; Módulo para acesso; Módulo para controle gráfico; Módulo para gerenciamento do fluxo do jogo.

No módulo para classificação do jogador, a listagem classificatória se dará por referência ao apelido (unique) e a pontuação e respeitará a ordem decrescente das pontuações, exceto em casos em que o usuário já esteja logado, reservando-se a ele assim a primeira posição da listagem com destaque diferenciado. A pontuação será atribuída e acrescida a cada acerto, com peso proporcional ao nível de dificuldade e inversamente proporcional ao tempo decorrido.

A atribuição das pontuações deve seguir os seguintes critérios:

- **Nível de dificuldade Fácil:**
Existirão 300 pontos em jogo e 2 minutos de tempo de jogo. Deve-se dividir essa pontuação igualmente pelo tempo considerando-se etapas de 20 segundos. Exemplo: $300/6 = 50$ pontos. A cada 20 segundos passados a pontuação será atualizada e decrementada em 50 pontos. Nos últimos 20 segundos não haverá decrementação da pontuação, restando ao jogador, em caso de acerto, os 50 pontos restantes.
- **Nível de dificuldade Médio:**
Existirão 600 pontos em jogo e 1 minuto e 30 segundos de tempo de jogo. Deve-se dividir essa pontuação igualmente pelo tempo considerando-se etapas de 30 segundos. Exemplo: $600/3 = 200$ pontos. A cada 30 segundos passados a pontuação será atualizada e decrementada em 200 pontos. Nos últimos 30 segundos não haverá decrementação da pontuação, restando ao

jogador, em caso de acerto, os 200 pontos restantes.

- **Nível de dificuldade Difícil:**

Existirão 900 pontos em jogo e 1 minuto de tempo de jogo. Deve-se dividir essa pontuação igualmente pelo tempo considerando-se etapas de 20 segundos. Exemplo: $900/3 = 300$ pontos. A cada 20 segundos passados a pontuação será atualizada e decrementada em 300 pontos. Nos últimos 20 segundos não haverá decrementação da pontuação, restando ao jogador, em caso de acerto, os 300 pontos restantes.

O módulo para filtragem da palavra trabalhará diretamente com a base de dados principal.

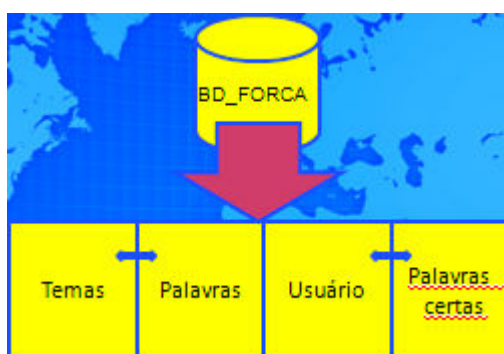


Fig. 3 Base de Dados principal do ForkIF

Ele selecionará randomicamente a palavra com base nos princípios da filtragem estabelecidos tais como: uma palavra será considerada fácil se tiver de seis a oito letras; uma palavra será considerada de dificuldade mediana se tiver de nove a onze letras; uma palavra será considerada difícil se possuir mais de onze letras ou possuir mais de seis letras e não possuir repetição de vogais. Tratará também da verificação na base de dados com fins de identificação da palavra que foi gerada randomicamente, caso já tenha sido acertada pelo usuário.

O módulo para acesso é o responsável por validar o login e a senha. Ele se comunica com o módulo de filtragem para saber se o tema foi selecionado, pois é requisito obrigatório para o acesso. Se o módulo de filtragem responder como true, o módulo de acesso solicita que uma palavra seja buscada pelo módulo de filtragem e, por conseguinte, estabelece-se a comunicação com o módulo de controle gráfico e de gerenciamento do fluxo do jogo preparando-os para o início do jogo. É nele que encontramos o canal de intermediação para o cadastro de novos jogadores. Os dados para cadastro de novos jogadores serão o apelido que servirá com identificador único, a data de nascimento, o nome completo, o e-mail e a cidade de origem. Para cada cadastro será realizado uma pesquisa pelo módulo de acesso sobre a existência

de um mesmo nome, com a mesma data de nascimento, com o mesmo e-mail e com a mesma cidade de origem. Se a consulta retornar true, o cadastro não será permitido, mesmo que o identificador único (apelido) não exista. Para cada cadastro realizado com êxito será enviado uma mensagem de confirmação para o e-mail cadastrado contendo o usuário e a senha cadastrados. Conterá ainda um link para renovação de senha. A renovação de senha será um formulário simples que solicitará o apelido do usuário, nome completo, data de nascimento e a nova senha.

O módulo para controle gráfico terá basicamente a função de gerar partes gráficas do boneco da força a cada requisição do módulo do fluxo do jogo, que a cada erro estabelecerá essa requisição. Comunicará, ainda, ao módulo de gerenciamento do fluxo do jogo sobre a completude da ilustração gráfica, o que sinaliza o Game Over.

O módulo para gerenciamento do fluxo do jogo receberá do módulo de filtragem os parâmetros da palavra em jogo e fará o controle com base nas regras para as pontuações, se comunicará com o contador de tempo e realizará as decrementações, estabelecerá contato com a Base de Dados para atualização da pontuação e registro de palavras acertadas e se comunicará com o módulo de controle gráfico para atualização da figura da força.

O ForkIF terá um conjunto de mensagens padrões, as quais identificaram situações específicas referentes ao transcorrer do jogo. Elas são: “Base de dados não possui palavras para o nível desejado. Mude de nível ou aguarde até que atualizemos a base de dados”; “Parabéns, você acertou a palavra e ganhou x pontos”; “Fim do Jogo, Você perdeu!” e “Login e/ou senha Incorretos!”.

O módulo extra, cujo propósito é a da realização de atividades básicas administrativas, será uma interface limitada por login e senha já previamente cadastrados, cuja utilidade será a de alimentação da base de dados. A alimentação do banco será feita via manipulação de arquivo. Após o usuário administrativo logar no módulo extra, será disponibilizado uma opção para se carregar um arquivo local, obrigatoriamente .txt. O arquivo .txt deve conter apenas uma palavra por linha. O processamento do arquivo .txt se dará via análise de cada palavra. Caso a palavra já exista na base de dados, tal palavra não será inserida e o processamento averiguará a próxima linha. Existirá um procedimento para que todas as palavras inseridas na base de dados estejam em letra maiúscula, mesmo que no .txt elas não estejam. Para as demais manipulações da base de dados (tais como atualizações ou correções de palavras) tais ações se darão por acesso direto ao banco.

5. DASHBOARD

No presente momento, já com a elicitação desenvolvida e com a diagramação estruturada, daremos início à codificação propriamente dita. Vale ressaltar ainda, que apesar de estarmos iniciando a parte de codificação, a análise de requisitos não será considerada uma atividade estática, mas sim contínua. Ao longo do projeto, por motivos de adaptações, melhorias e impossibilidades, os requisitos podem se alterar, o que mostra o seu caráter altamente mutável.

Nesse momento, trataremos também da modelagem e implementação da base dados, fator imprescindível para a continuidade do projeto.

6. CONCLUSÃO

Na atmosfera do desenvolvimento de software, a modelagem de requisitos é sem sombra de dúvidas um dos artefatos mais importantes para o mapeamento do programa e oficialização dos acordos firmados.

Uma modelagem bem elaborada e elicitada, além de agilizar o processo de desenvolvimento, minimiza por antecipação o impacto nos erros durante o desenvolvimento.

Todavia, não basta apenas haver uma modelagem bem trabalhada. O desenvolvedor precisa conhecê-la, em especial os padrões de diagramação, para não interpretar incorretamente o que o modelo pretende passar.

No mais, agradecemos pela oportunidade em desenvolver esse trabalho, o qual está sendo de grande valia para o meu crescimento intelectual e profissional.

6. REFERÊNCIAS BIBLIOGRÁFICAS

BEZERRA, Eduardo. Princípios de Análise e projeto de Sistemas com UML .1.ed. Rio de Janeiro: Campus,2004.

ANEXOS

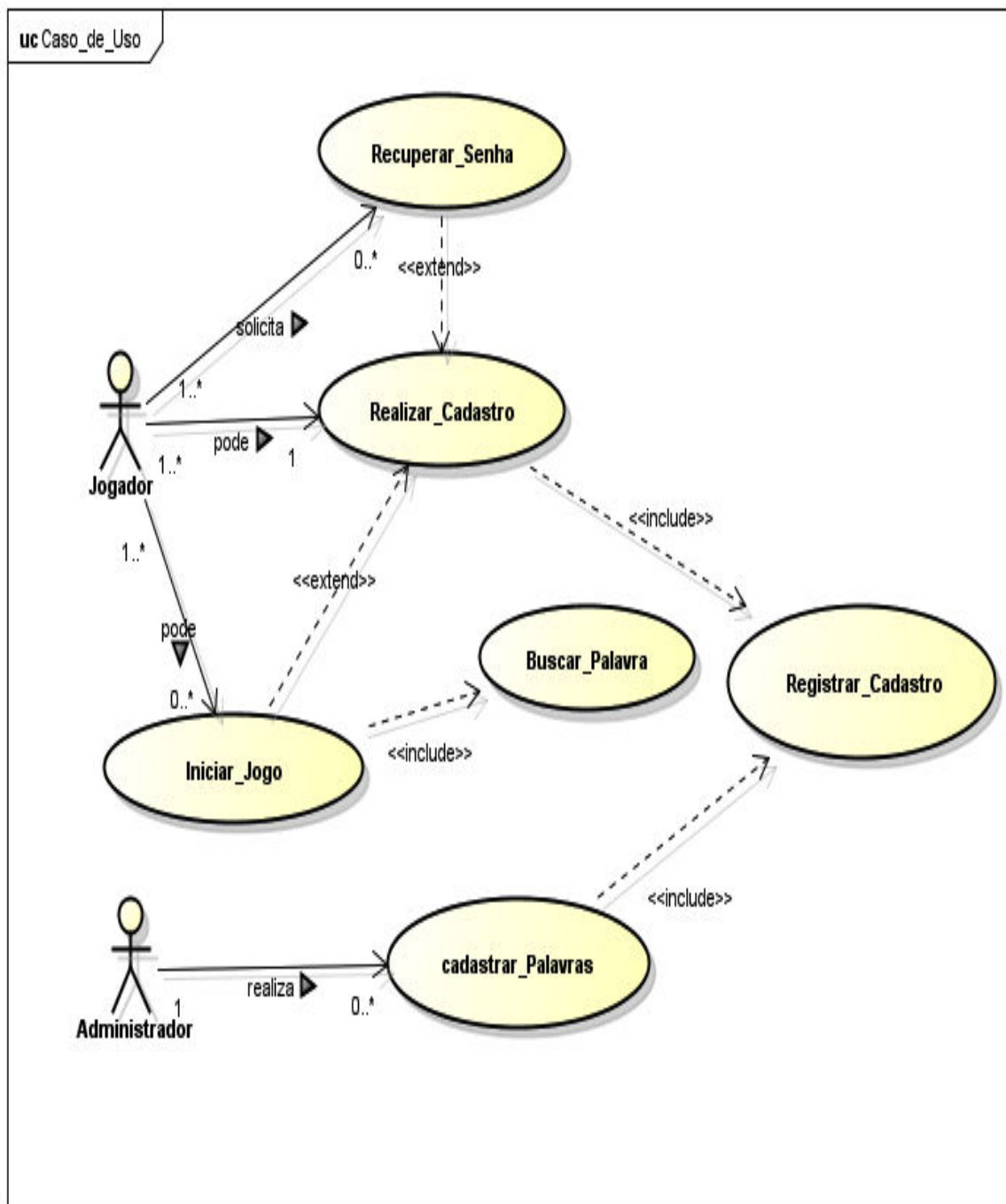


Fig. 4 Casos de Uso do ForkIF

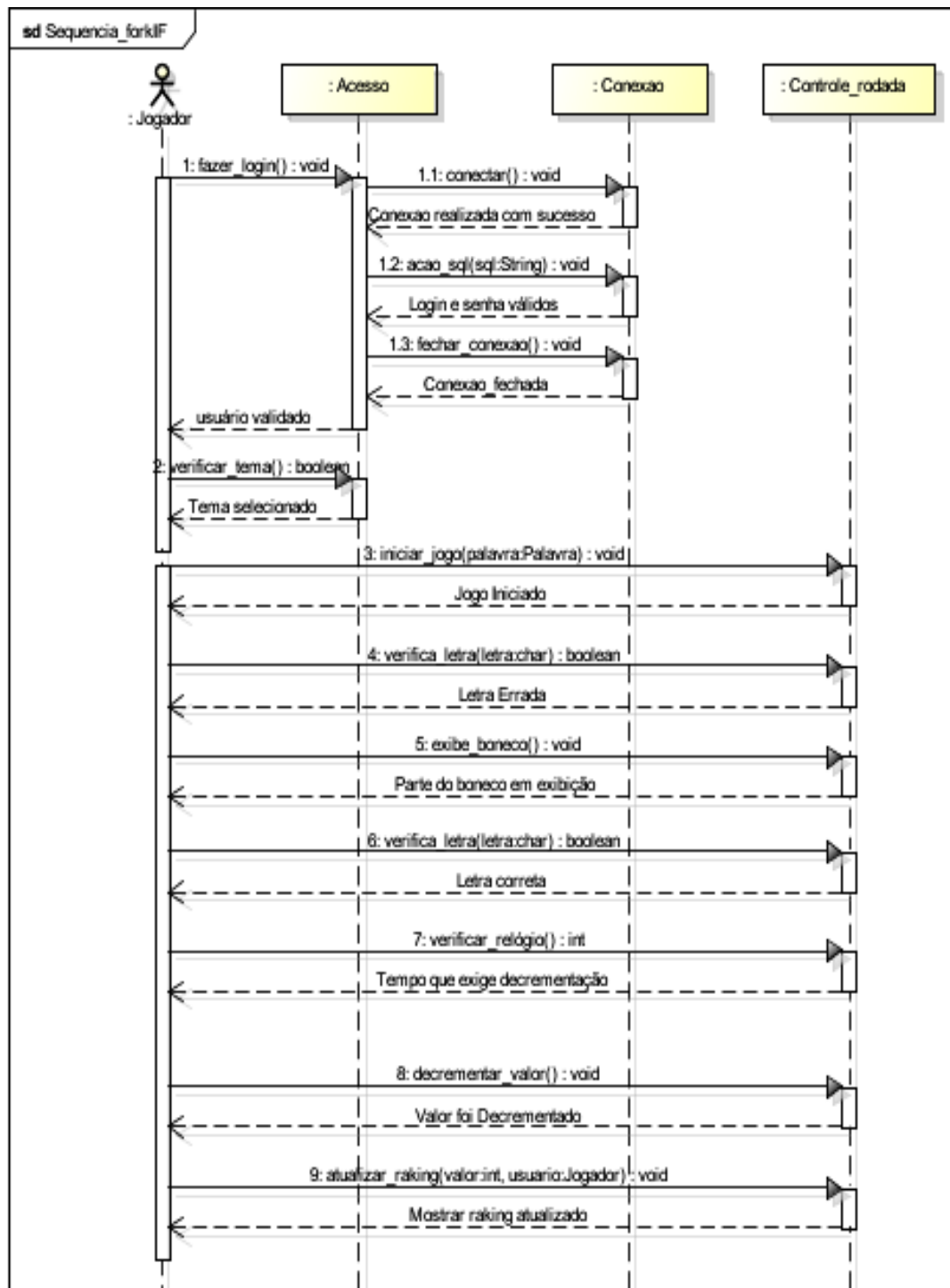


Fig. 5 Diagrama de sequência do ForkIF

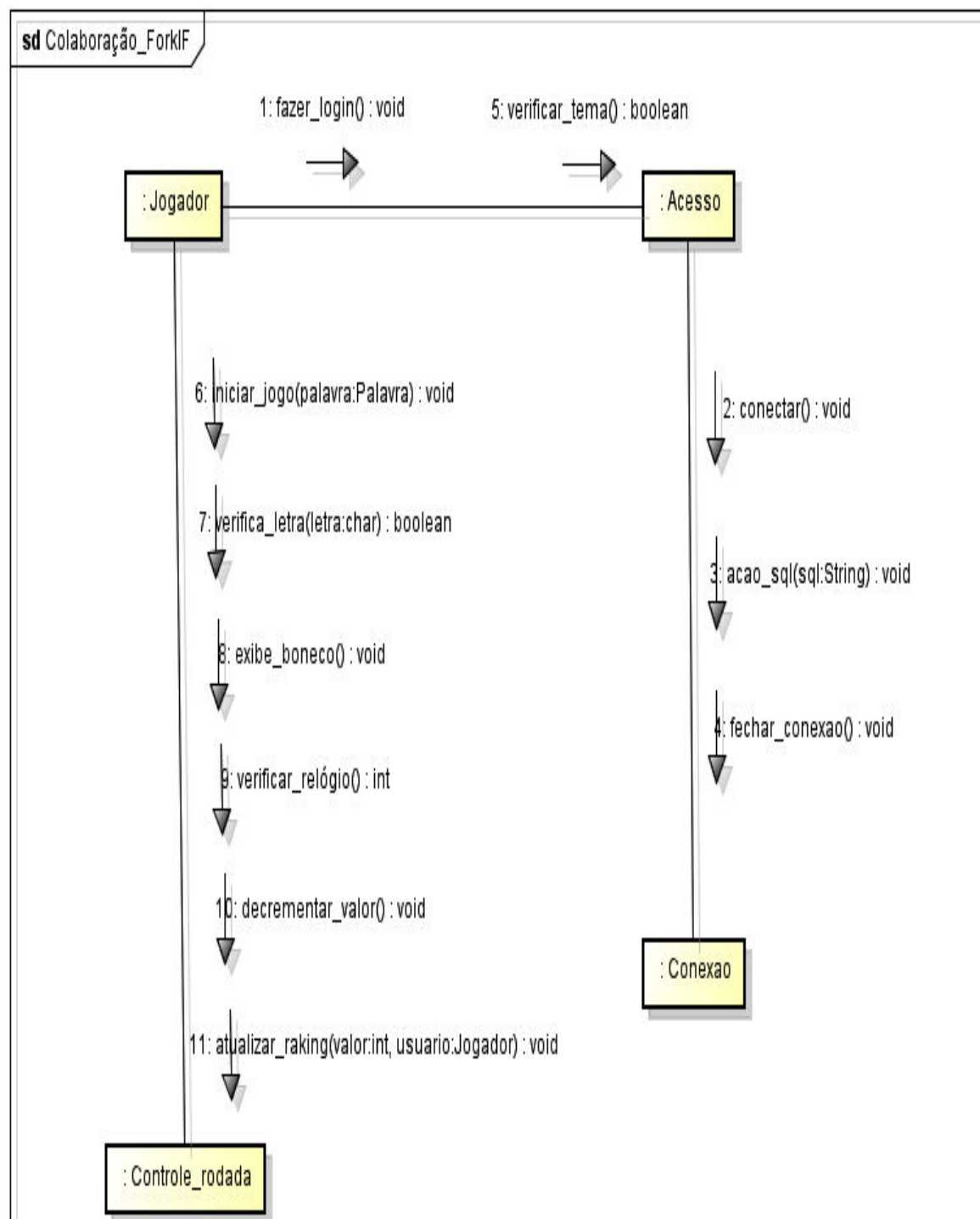


Fig. 6 Diagrama de colaboração do ForkIF

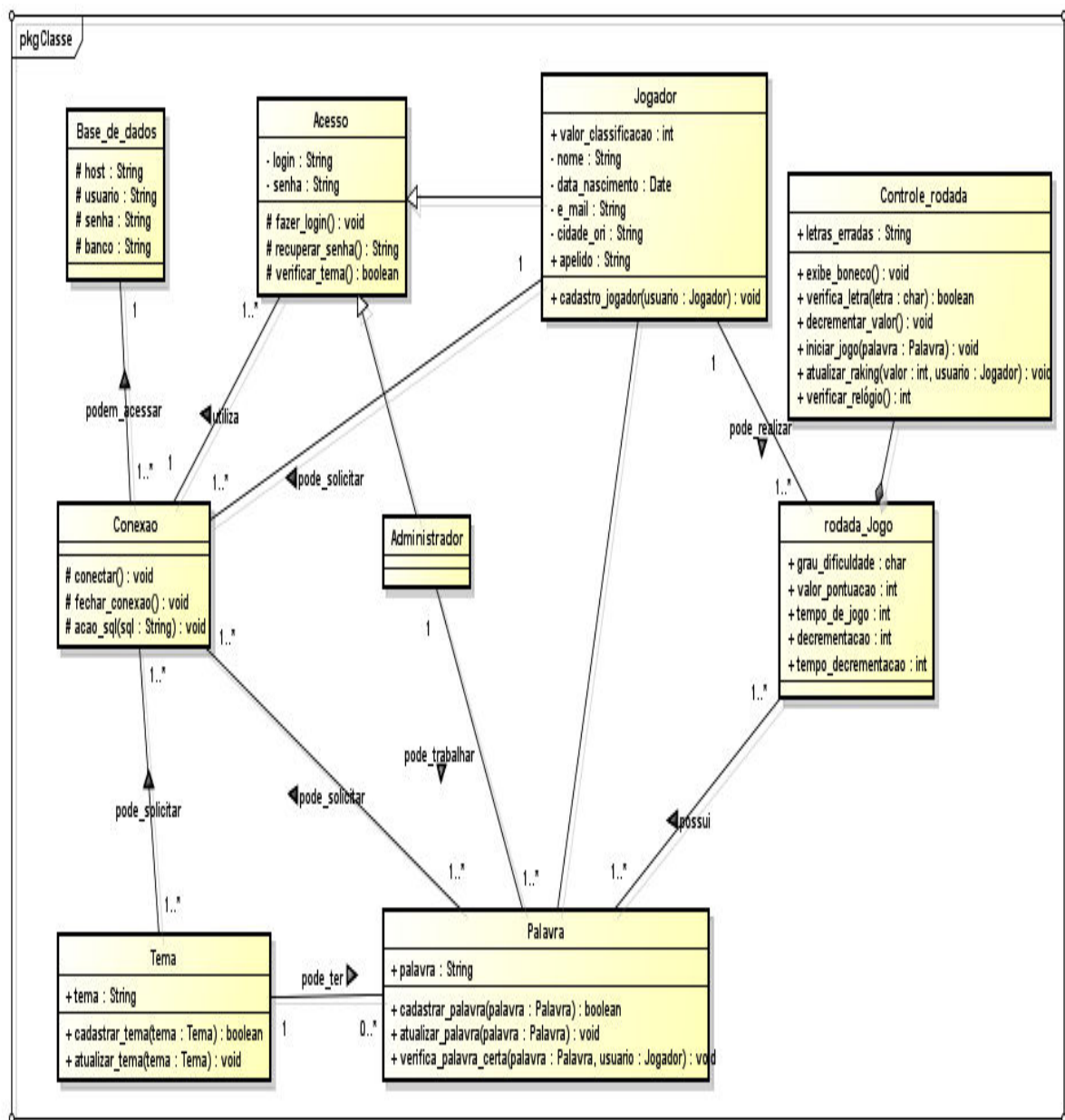


Fig. 7 Diagrama de classes do ForkIF

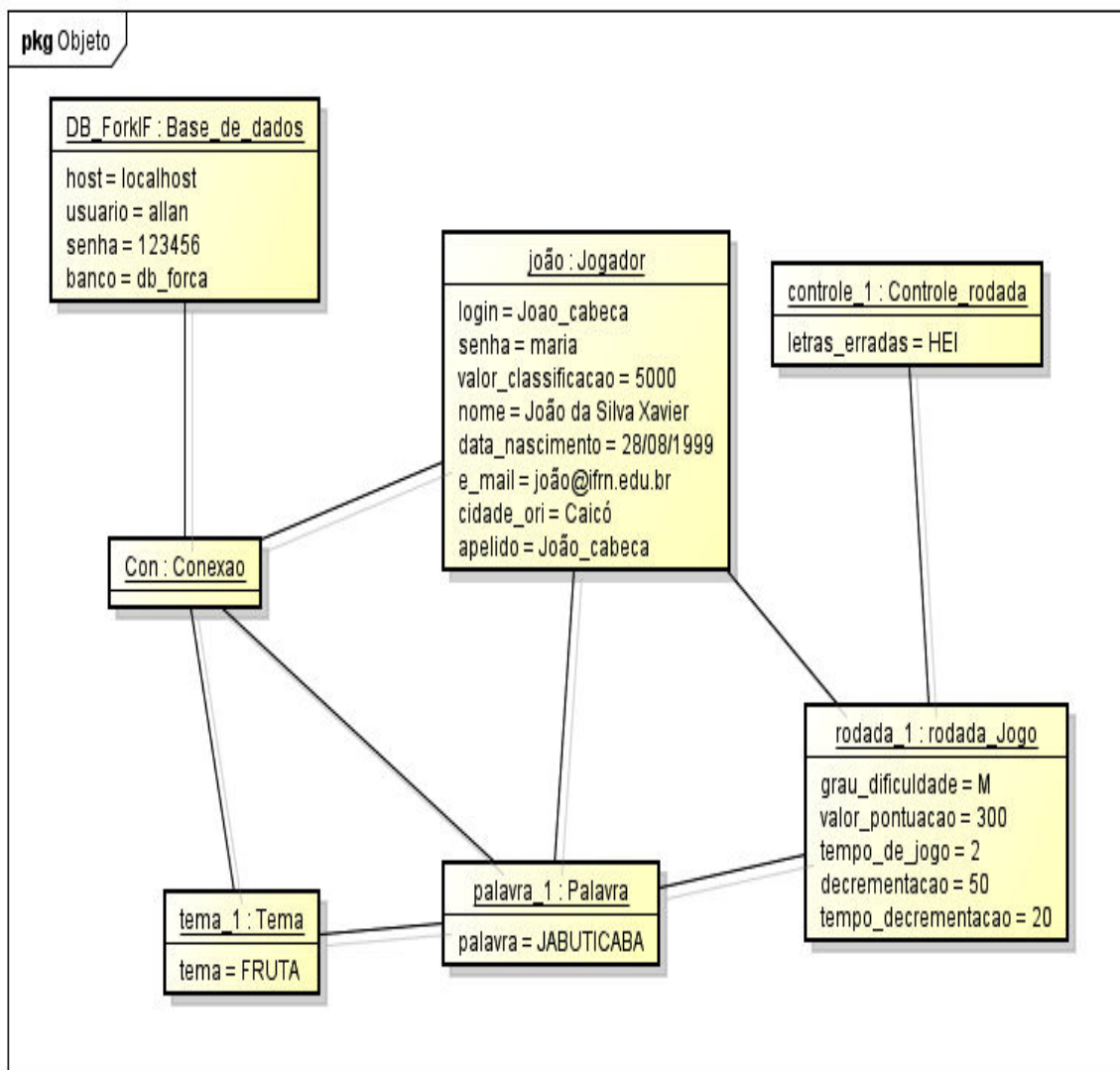


Fig. 8 Diagrama de objetos do ForkIF

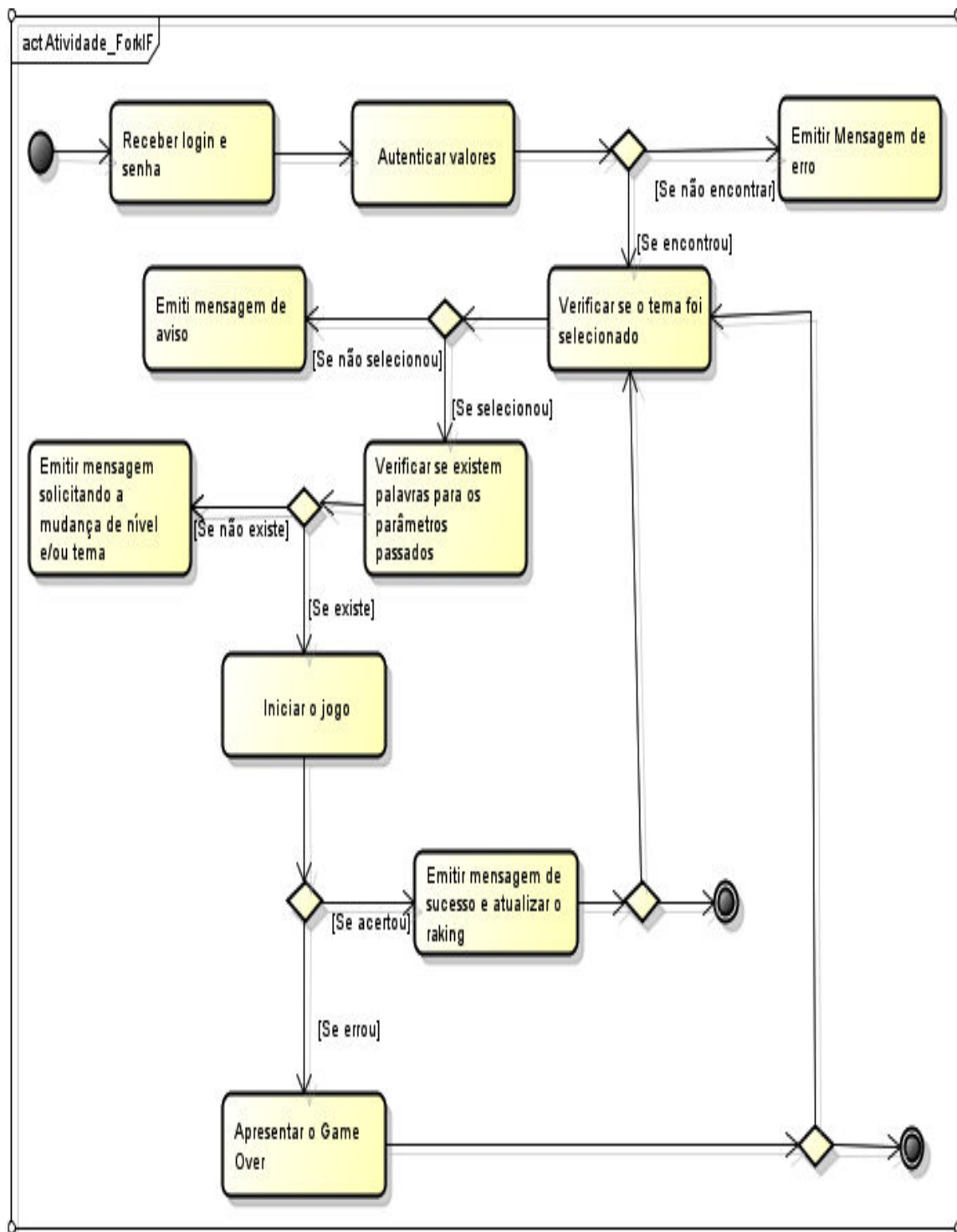


Fig. 9 Diagrama de atividade do ForkIF