

UNIVERSIDADE FEDERAL DE SANTA CATARINA - UFSC
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA - INE
INE 5608 - ANÁLISE E PROJETO DE SISTEMAS - Turma 04238 (2011.1)
PROF. RICARDO PEREIRA E SILVA

Projeto: Jogo Scrabble

Especificação de Requisitos

Grupo:

Areli Andreia dos Santos
Tiago Jaime Nascimento

Histórico de Versões

Data	Versão	Autores	Ação
17/04/11	1.0	Tiago, Areli e Marlon	1ª versão do documento
02/06/11	1.1	Tiago, Areli	Revisão das regras do jogo, interface e tecnologias usadas

Conteúdo

1. Introdução	
1.1. Objetivo do Desenvolvimento.....	3
1.2. Referências.....	3
2. Visão Geral do Sistema	
2.1. Arquitetura da Aplicação.....	3
2.2. Descrição do Jogo.....	3
2.3. Premissas de Desenvolvimento.....	4
3. Requisitos de Software	
3.1. Requisitos Funcionais.....	5
3.2. Requisitos Não Funcionais.....	6
Anexo A - Esboço da Interface.....	7

1. Introdução

1.1 Objetivo do Desenvolvimento

Desenvolvimento de um programa que implemente o jogo de tabuleiros Scrabble®, permitindo disputas entre dois jogadores em rede.

Este trabalho faz parte da avaliação final da disciplina Análise e Projeto de Sistemas (INE5608), ministrada pelo Professor Ricardo Pereira e Silva.

1.2 Referências

<<http://en.wikipedia.org/wiki/Scrabble>>

<<http://www.scrabble.com/>>

2. Visão Geral do Sistema

2.1 Arquitetura da Aplicação

Programa orientado a objetos escrito na linguagem Java.

2.2 Descrição do Jogo

Scrabble é um jogo em que jogadores ganham pontos formando palavras a partir de letras aleatórias em um tabuleiro de 15 x 15 casas, onde cada casa comporta uma única letra. As palavras criadas permanecem no tabuleiro e suas letras podem ser reutilizadas em futuras palavras encaixadas ao seu redor, sendo que palavras só são válidas através da leitura das letras nos sentidos acima-abaixo e esquerda-direita.

A pontuação é calculada por palavra colocada a cada turno, sendo que são considerados os pontos de cada letra - uma pontuação distinta dependente da sua raridade na língua usada, e multiplicadores de pontuação espalhados pelo tabuleiro - letras colocadas em determinadas casas podem multiplicar a pontuação da letra (2 ou 3 vezes) ou da palavra completa (2 ou 3 vezes). No total são utilizadas 117 letras, guardadas escondidas dos jogadores no *Banco de Letras*.

A cada turno cada jogador pode:

UNIVERSIDADE FEDERAL DE SANTA CATARINA - UFSC
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA - INE
INE 5608 - ANÁLISE E PROJETO DE SISTEMAS - Turma 04238 (2011.1)
PROF. RICARDO PEREIRA E SILVA

1. Receber 7 letras aleatórias (na primeira rodada , ou caso não possua nenhuma) ou o número de letras que faltam para completar 7;

2. Formar uma palavra no tabuleiro;

3. Passar o turno (sem ganhar qualquer ponto) ;

4. (opcionalmente) Desafiar a última palavra colocada pelo oponente, que deverá ser checada em um dicionário de palavras. Caso a palavra conste no dicionário o jogador desafiante perderá 50 pontos, caso não conste a jogada do jogador desafiado será anulada e os pontos ganhos no turno cancelados.

Ao término do jogo são comparados os pontos de cada jogador para determinar o ganhador. O jogo acaba em duas situações:

a) Um jogador utiliza todas as letras que possui, e não existem mais letras no *Banco*;

b) Se passaram 6 turnos em que jogadores não receberam pontos (*passaram* turno);

As letras consideradas no jogo, bem como as quantidades iniciais e seus valores em pontos são:

Letra	Quantidade	Valor	Letra	Quantidade	Valor
A	14	1	N	6	2
B	3	4	O	10	1
C	4	4	P	4	4
D	5	3	Q	1	8
E	11	1	R	6	2
F	2	5	S	8	2
G	2	5	T	5	3
H	2	5	U	7	2
I	10	1	V	2	5
J	2	5	X	1	10
L	5	3	Z	1	10
M	6	2			

2.3 Premissas de Desenvolvimento

- O programa será desenvolvido com a linguagem de programação Java, sob o paradigma de Orientação a Objetos, devendo ser compatível com qualquer plataforma que possuir uma Máquina Virtual Java (JVM);
- O programa deverá apresentar uma interface bidimensional, onde cada jogador irá interagir com o uso do *mouse* e do teclado;
- O jogo deverá ser implementado em Português, tanto a interface quanto o dicionário de palavras;
- O jogo deverá ser distribuído, para ser jogado pela Internet por dois jogadores;
- O programa deve implementar regras similares às do jogo Scrabble de forma que seja intuitivo para um jogador do jogo de tabuleiro utilizá-lo.

3 Requisitos de Software

3.1 Requisitos Funcionais

Requisito funcional 1 - iniciar partida: o programa deve oferecer a opção de iniciar uma nova partida, pedindo informações dos jogadores como o Nome;

Requisito funcional 2 - conexão com servidor: o programa deverá oferecer pela sua interface uma forma de conectar-se a um servidor central para que possam ser disputadas partidas pela internet;

Requisito funcional 3 - procedimento de lance: o programa deve permitir que os jogadores executem ações correspondentes a lances possíveis do jogo, como descritos na seção 2.2 (Descrição do jogo). O programa deve permitir:

3.1. que o jogador *arraste com o mouse* as letras que possui ao tabuleiro para formar palavras;

3.2. que o jogador passe o turno clicando em um botão;

3.3 que o jogador desafie a palavra colocada pelo jogador adversário, clicando em um botão. Se qualquer das palavras do adversário forem inválidas, elas devem ser revertidas: as letras

UNIVERSIDADE FEDERAL DE SANTA CATARINA - UFSC
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA - INE
INE 5608 - ANÁLISE E PROJETO DE SISTEMAS - Turma 04238 (2011.1)
PROF. RICARDO PEREIRA E SILVA

voltarem para o banco de letras e as pontuações de deduzidas - mas se todas constarem no dicionário o jogador desafiante deve perder 50 pontos.

A cada jogada o programa deve verificar se a ação é válida e pedir que o jogador refaça-a caso necessário. Após o fim da jogada, o programa deve verificar se o jogador ganhou a partida;

Requisito funcional 4 - banco de letras: o programa deve gerenciar as 117 letras disponíveis no jogo, distribuindo-as aos jogadores em cada turno. O programa deve deixar claro aos jogadores quando acabarem as letras;

Requisito funcional 5 - dicionário de palavras: o programa deve manter em arquivo o registro de um conjunto de palavras da língua portuguesa consideradas válidas, e a pedidos de um jogador desafiante, consultar se uma determinada palavra é correta;

Requisito funcional 6 - comunicação do estado da partida: a cada jogada o programa deve atualizar o estado do tabuleiro, informar ao jogador seguinte que é sua vez de jogar e mostrar aos jogadores quais palavras foram feitas e quantos pontos foram obtidos com elas, assim como atualizar e mostrar o total de pontos de cada jogador.

3. 2 Requisitos não Funcionais

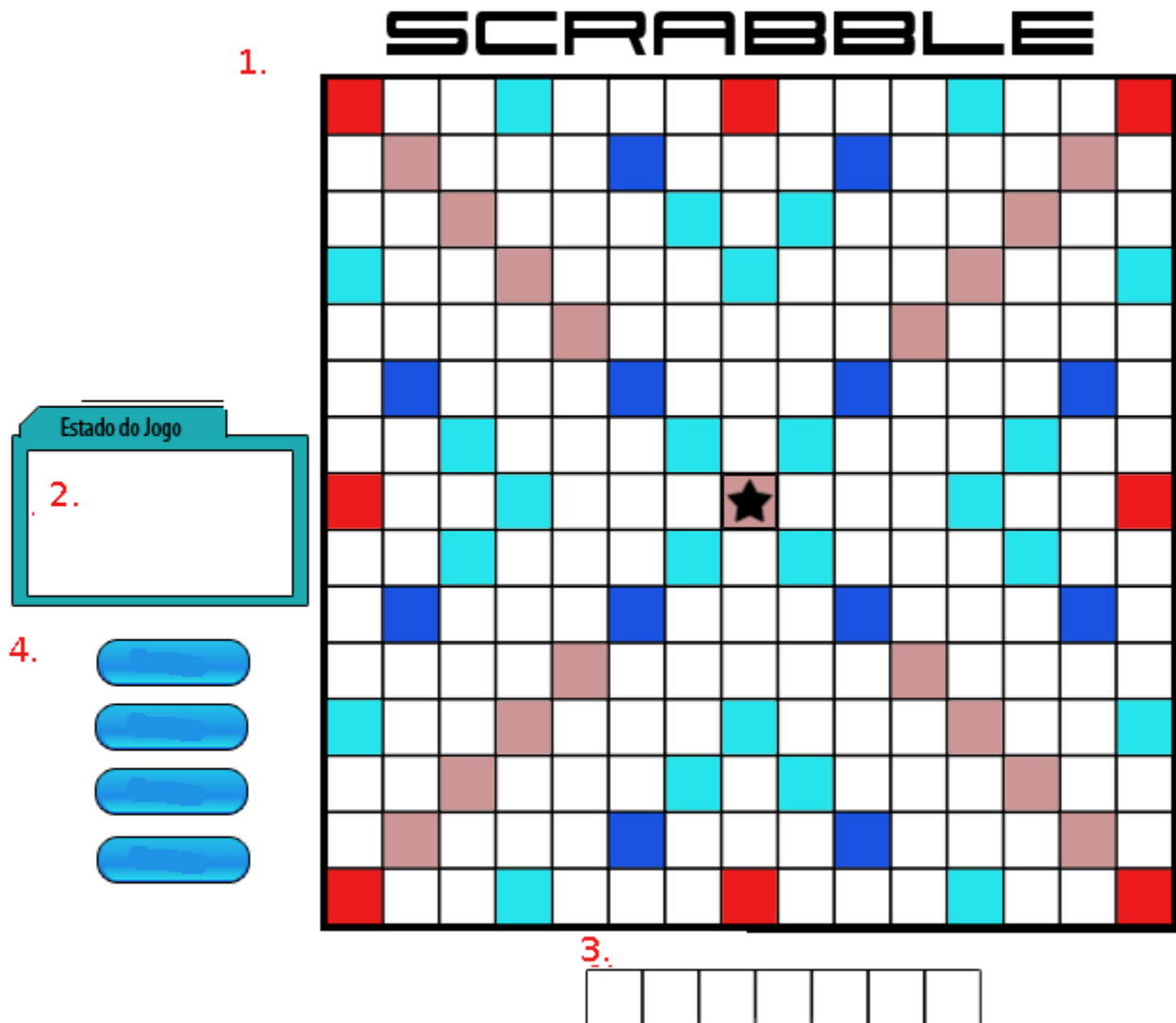
Requisito não funcional 1 - especificação de projeto: além do código Java, deve ser produzida especificação de projeto baseada em UML2, utilizando a ferramenta *Visual Paradigm*;

Requisito não funcional 2 - interface gráfica para usuário: o programa deverá ter interface gráfica única, partilhada pelos usuários, desenvolvida pela tecnologia *Java Swing*;

Requisito não funcional 3 - tecnologia de programação distribuída: o programa deverá ser implementado utilizando o *framework NetGamesNRT*;

Requisito não funcional 4 - tecnologia de banco de dados: o dicionário de palavras deve ser implementado em banco dados utilizando a tecnologia de banco de dados embutido *Apache Derby*.

ANEXO A - Esboço da Interface



Legenda

1. Tabuleiro;

2. Informações do Jogo: como totais de pontos de cada jogador e informações da situação do jogo;

3. Letras disponíveis ao jogador em cada turno.

4. Botões de ações: possibilitam o jogador realizar cada ação possível na rodada;