

Projeto: Clone do Letterpress

Especificação de Requisitos

Versão 1.4

16/04/2016

Versão	Autor(es)	Data	Ação
1.0	Guilherme João Matheus	14/04/2016	Definição dos requisitos.
1.1	Guilherme João Matheus	14/04/2016	Nova seção com o esboço da interface
1.2	Guilherme João Matheus	15/04/2016	- Descrições mais detalhadas sobre os requisitos funcionais e não funcionais. - Novo Requisito não-funcional 7 - Indicar vez - Referências.
1.3	Guilherme João Matheus	16/04/2016	Novo Requisito funcional 6 - Limpar palavra.
1.4	Guilherme João Matheus	16/04/2016	Correção de erros ortográficos.

Conteúdo

1. Introdução
2. Visão Geral
3. Requisitos de software

1 Introdução

1.1. Objetivo do desenvolvimento

Clone do jogo Letterpress em português.

1.2. Definições

- **Letterpress:** é um jogo distribuído disponível apenas para o sistema operacional iOS(sistema operacional para celulares iPhone e tablets iPad) similar ao caça palavras e apenas em inglês e espanhol.
- **Matriz MxN:** uma tabela de M linhas e N colunas.
- **Concatenar:** adicionar a sequência.
- **Adjacente:** ao lado de.

1.3 Referencias

- <http://www.atebits.com/letterpress/> (site do jogo original em inglês)
- <http://pastie.org/private/0ilsutafjby7eamqeu1ckg> (regras do jogo traduzidas para o português)
- <http://imgur.com/a/3RIb1> (capturas de tela do jogo original)

1.4 Regras traduzidas do jogo original

1.4.1 O básico: 2 Jogadores formam palavras com letras de uma matriz 5x5 em turnos. Clique em letras de qualquer posição da matriz formando uma palavra(concatenando as letras) e clique em “Enviar” para enviar a palavra formada e assim colorir essas letras com a sua cor. Cada letra colorida vale 1 ponto. O vencedor será quem tiver a maior pontuação no final do jogo.

1.4.2 Formação de palavras: A palavras que satisfaça as seguintes condições será aceita:

- possuir mais do que 2 letras.
- não ter sido jogada ainda.
- não ser parte inicial de uma palavra já jogada.
- não ser um pronome, abreviação ou acrônimo.

1.4.3 Defendendo letras: Para defender uma letra, um jogador deve ter colorido todas as letras adjacentes a ela. O outro jogador ainda pode usar a letra, mas não colori-la.

Note que letras defendidas deixarão de ser defendidas assim que o outro jogador colorir alguma das letras adjacentes.

1.4.4 Terminando o jogo: O jogo termina quando todas as letras forem coloridas. O jogador com mais letras vence.

2 Visão Geral

2.1 Arquitetura do programa

- **2.1.1 Paradigma de programação:** Linguagem orientada a objetos.
- **2.1.2 Tipo de aplicação:** Distribuída com arquitetura cliente-servidor.

2.2 Premissas de desenvolvimento

- **2.2.1 Linguagem:** O programa deve ser feito em Java.
- **2.2.2 Interface:** O programa deve ter interface gráfica bidimensional.
- **2.2.3 Conectividade:** O programa utilizara a ferramenta NetGamesNRT.
- **2.2.4 Modelagem:** A modelagem será feita através da linguagem UML2 utilizando a ferramenta Visual Paradigm Community Edition.

3 Requisitos de Software

3.1 Requisitos Funcionais

Requisito funcional 1 - Conectar: O software deve apresentar em seu menu principal a opção de "Conectar", para estabelecer conexão com o servidor. Após conectado deverá aparecer uma nova janela com um texto indicando que foi bem sucedida a conexão ou não.

Requisito funcional 2 - Iniciar: O programa deve apresentar em seu menu principal a opção de "Iniciar partida", para procurar um adversário e assim dar início a uma nova partida. A opção só ficara disponível após ter sido feita a conexão(Requisito funcional 1). Ao iniciar a partida, a matriz de letras será preenchida com letras pseudo-aleatórias seguindo um algoritmo que escolhe letras mais comuns, para que, por exemplo, a letra "a" apareça mais do que a letra "y". O jogador inicial será aquele iniciou primeiro.

Requisito funcional 3 - Ver regras: O programa deve apresentar em seu menu principal a opção de "Regras", para que o jogador possa consultar as regras a qualquer momento. Ao clicar na opção, aparecerá uma nova janela contendo apenas o texto das regras.

Requisito funcional 4 - Formar palavra: Após a partida ser iniciada e ter aparecido as letras(Requisito funcional 2), o programa deve permitir que qualquer jogador forme uma palavra, independente de ser a sua vez ou não. O jogador formará a palavra, escolhendo qualquer letra de uma matriz 5x5, pertencendo a ele ou ao outro jogador. Ao escolher uma letra, ela desaparecerá da matriz e será concatenada à um campo de texto com as letras já escolhidas(ou seja a palavra que o jogador está formando) e o placar será atualizado localmente(não visível para outro jogador) para mostrar ao jogador como ficaria caso ele enviasse a palavra(Requisito funcional 5). Enquanto a palavra esta sendo formada ela não é visível para o outro jogador.

Requisito funcional 5 - Enviar palavra: Após ter sido formada uma palavra(Requisito funcional 4), o programa deve ter um botão de "Enviar Jogada" para o jogador da vez. Ao clicar no botão, o programa verificará se a palavra formada é uma palavra válida, e caso seja, enviará a palavra. Serão enviadas, as palavras que:

- possuírem mais do que 2 letras.
- não terem sido jogada ainda.
- não serem parte inicial de uma palavra já jogada.
- não serem um pronome, abreviação ou acrônimo.
- pertencerem a língua portuguesa(regra não pertencente ao jogo original).

Ao enviar a palavra, o placar será atualizado para ambos os jogadores. E, para o jogador que não é o da vez, aparecerá uma nova janela, com um texto informando a palavra enviada do jogador da vez, e que o turno mudou, e é a sua vez.

Se a palavra não for válida, aparecerá uma nova janela com um texto informando o motivo da não-validação.

Requisito funcional 6 - Limpar palavra: Enquanto seja formada uma palavra(Requisito funcional 4), o programa deve ter um botão "Limpar Palavra" para o jogador limpar a palavra que está formando.

Requisito funcional 7 - Passar a vez: Após a partida ser iniciada(Requisito funcional 2), o programa deve ter a opção de "Passar a vez", caso o jogador decida não efetuar uma jogada naquele turno. Ao clicar na opção aparecerá uma nova janela, confirmando se o jogador realmente quer passar o turno e não clicou na opção por engano. Caso ele confirme que sim, a seta que indica a vez será

atualizada para os dois jogadores. Para o outro jogador, aparecerá uma nova janela, informando que o jogador passou a vez e que agora é a sua vez.

Requisito funcional 8 - Ver palavras já jogadas: Após palavras terem sido jogadas, ou seja enviadas(Requisito funcional 5), o menu do programa deve ter a opção de "Palavras jogadas", para que o jogador possa consultar todas as palavras que já foram utilizadas naquela partida. Ao clicar na opção aparecerá uma nova janela, com um texto das palavras já jogadas separadas por vírgula.

Requisito funcional 9 - Desistir: Após a partida ser iniciada(Requisito funcional 2), o programa deve ter a opção de "Desistir", para que qualquer jogador, independente de ser sua vez ou não, possa desistir da partida atual e iniciar uma nova partida, resultando na vitória do adversário. Ao clicar na opção aparecerá uma janela, confirmando se o jogador realmente quer desistir da partida e não clicou na opção por engano. Para o outro jogador aparecerá uma nova janela, informando que o outro jogador desistiu. Para os dois jogadores o programa será reinicializado, como se tivesse sido fechado e aberto novamente.

3.2 Requisitos Não-funcionais

Requisito não-funcional 1 - Especificação de Projeto: A codificação da aplicação deve ser feita em Java e a especificação de projeto em UML2.

Requisito não-funcional 2 - Interface Gráfica: A interface gráfica deve ser intuitiva, única e compartilhada para ambos os jogadores, apenas alternando a cor das letras: o jogador verá suas letras na cor azul e as letras de seu adversário na cor vermelho.

A interface consistirá de:

- 2 placares com os indicativos “Eu” e “Ele”.
- 1 seta para indicar a vez.
- 1 botão para limpar a palavra.
- 1 campo de texto para indicar a palavra.
- 1 botão para enviar a palavra.
- 1 matriz 5x5 de letras para formar a palavra.

A interface também deverá ter a fonte em tamanho grande, para que o jogo seja acessível para pessoas com deficit na visão, ou seja aquelas que usam óculos.

Requisito não-funcional 3 - Cor das letras: As letras serão coloridas da seguinte forma:

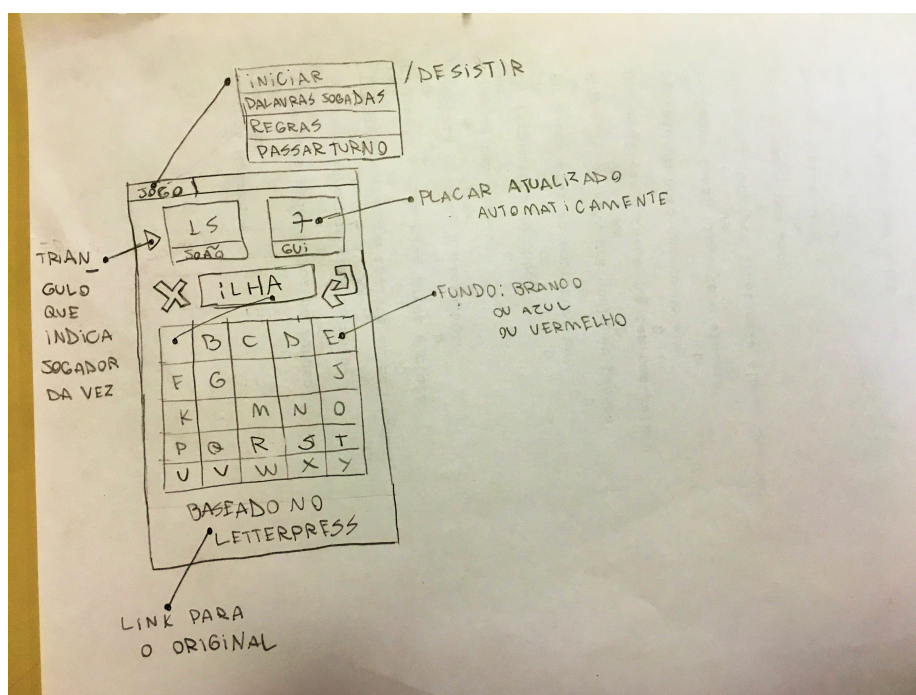
- **azul:** letras do jogador.
- **azul forte:** letras defendidas pelo jogador. Segundo as regras, é aquela em que todas as letras adjacentes pertencem ao jogador.
- **vermelho:** letras do jogador adversário.
- **vermelho forte:** letras defendidas pelo jogador adversário. Segundo as regras, é aquela em que todas as letras adjacentes pertencem ao adversário.
- **branco:** letras que não pertencem a nenhum jogador.

Requisito não-funcional 4 - Indicar vez: O programa deve indicar de quem é a vez com uma pequena seta apontando para o placar do jogador da vez.

Requisito não-funcional 5 - Tecnologia de interface gráfica: A interface gráfica deve ser baseada em Java-Swing.

Requisito não-funcional 6 - Banco de dados aberto: O banco de dados, contendo as palavras da língua portuguesa, deve ter licença aberta e grátis, que permita a disponibilização do código e o banco de dados do programa final pela internet.

4. Esboço da Interface Gráfica



OBS: A ideia de ter um link para o jogo original foi descartada, os jogadores não terão mais nomes, serão apenas indicados como “Eu” e “Ele”, e também terá no menu, a opção “Conectar”.