



UNIVERSIDADE FEDERAL DE SANTA CATARINA – UFSC
CENTRO TECNOLÓGICO – CTC
CURSO DE SISTEMAS DE INFORMAÇÃO
ANÁLISE E PROJETO DE SISTEMAS

PROJETO SKIP-BO

Graduandos: Alexandre Beirith Neto – 11201006
Fellipe Bratti Pasini - 11203854
Marcos Demétrio - 11201041

FLORIANÓPOLIS - SC
15/04/2013

1. Especificação de Requisitos

Versão 1.0 – 09/04/2013

Versão	Autores	Data	Ação
1.0	Alexandre Beirith Neto Fellipe Bratti Pasini Marcos Demétrio	09/04/2013	Definição dos requisitos do software.

2. Conteúdo

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

3. Objetivo

Projeto e implementação de um jogo de cartas chamado SKIP-BO, que pode ser disputado entre dois jogadores via rede. O baralho é composto por 162 cartas, 12 de cada um dos números de 1 a 12, e 18 cartas do tipo "SKIP-BO" que podem ser jogadas como qualquer número.

Cada jogador recebe 30 cartas para sua pilha de estoque apenas com o cartão superior visível, e uma mão de cinco cartas, e as cartas restantes são colocadas com a face para baixo para criar um baralho comum.

A área de jogo permite construir até quatro pilhas de cartas, que devem ser iniciadas usando uma carta de número "1" ou uma SKIP-BO, e cada jogador também tem até quatro pilhas de descarte pessoais.

O jogador da vez deve jogar a próxima carta em ordem sequencial ou um cartão de SKIP-BO (coringa), usando as cartas da mão, a carta de cima de sua pilha de estoque, ou a carta de cima de qualquer uma de suas quatro pilhas de descarte. Se o jogador conseguir jogar todas as cinco cartas da mão, ele chama mais cinco e continua jogando. Quando não há mais descartes disponíveis, o jogador descarta uma carta para uma pilha vazia ou em cima de uma já existente e passa o jogo para o próximo jogador.

Quando uma pilha de construção chega a 12, ela é removida do tabuleiro e o espaço torna-se vazio para outra pilha ser iniciada. Quando o primeiro jogador usar todas as cartas da sua pilha de reserva, vence.

4. Visão Geral

4.1. Arquitetura do programa

O software será desenvolvido utilizando o paradigma de Orientação à Objetos, com a linguagem de programação Java. O suporte a rede será implementado usando o framework NetGames de Leonardo de Souza Brasil (<http://www.inf.ufsc.br/~netgames/>).

4.2. Premissas de desenvolvimento

- O programa será desenvolvido utilizando a linguagem de programação Java (Orientado a Objetos e rodará em qualquer dispositivo que possuir a JVM)

instalada.

- O jogo possuirá todas as regras presentes no SKIP-BO original.
- O programa terá interface 2D, e a interação será feita através do mouse que poderá arrastar as cartas até o centro da partida.
- O idioma do jogo será Português do Brasil.
- O software será utilizado através da Internet para a disputa entre dois jogadores remotos.

5. Requisitos de Software

5.1. Requisitos Funcionais

Requisito funcional 1 - Jogo: o software permitirá partidas entre dois jogadores e irá controlar as ações dos mesmos ao longo da partida, podendo levar a vitória de somente um dos jogadores;

Requisito funcional 2 - Conexão com servidor: o software deverá permitir a conexão com o servidor para que as partidas sejam disputadas entre os jogadores via rede;

Requisito funcional 3- Cadastro: o software deverá permitir o cadastro de informações dos dois jogadores para a disputa da partida;

Requisito funcional 4 - Iniciar partida: o software terá a opção “Iniciar partida”, para o início de uma nova partida;

Requisito funcional 5 - Desistir da partida: o software terá a opção “Desistir da partida”, permitindo a desistência da partida por parte dos jogadores que estarão disputando a mesma;

Requisito funcional 6 - Ações do jogador: a cada rodada o software deverá informar o jogador da vez, permitindo que este tenha como possibilidade de jogada todas as ações de jogadas possíveis para a condição atual da partida;

Requisito funcional 7 - Informar vencedor: o software deverá informar qual foi o jogador que venceu a disputa quando o mesmo tiver utilizado todas as cartas da pilha reserva;

Requisito funcional 8 - Sair: o software conterá a opção “sair”, permitindo a saída do jogo por parte dos jogadores.

5.2. Requisitos Não Funcionais

Requisito não funcional 1 - Conectividade: Para suporte a partidas em rede deverá ser utilizado o framework NetGamesNRT.

Requisito não funcional 2 – Ferramenta de criação do projeto técnico: Será utilizada a ferramenta Visual Paradigm para criação do projeto técnico em UML 2.0.

Requisito não funcional 3 – Tecnologia para interface gráfica: O software será desenvolvido utilizando a linguagem de programação Java e mais especificamente a biblioteca SWING para a parte de interface gráfica.

6. Esboço gráfico

