



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

Especificação de Requisitos de Software Sushi Go

Peter Michael Claes Krause
Ubiratan da Rosa Rodrigues

Florianópolis
2019

Versão	Autor(es)	Data	Ação
1.0	Peter Krause e Ubiratan Rodrigues	28/09/2019	Estabelecimento dos Requisitos
1.1	Peter Krause e Ubiratan Rodrigues	22/10/2019	Atualização de Requisitos
1.2	Peter Krause e Ubiratan Rodrigues	18/11/2019	Adição de regras e pontuação

Sumário

- I. Introdução
- II. Visão Geral
- III. Requisitos de Software
- IV. Interface Gráfica

I Introdução

I.I Resumo

Este software tem por objetivo tornar possível jogar o jogo **Sushi Go** de maneira digital e distribuída.

Para projetar este software, será utilizado todo o conhecimento obtido até então em matérias passadas para a construção lógica do código somado ao aprendizado da documentação UML obtida na matéria de Análise e Projeto de Sistemas.

O resultado disto é a aplicação do jogo junto de toda sua documentação estrutura.

I.II Contexto

Sushi Go é um jogo de cartas de no mínimo dois jogadores onde você está comendo em um restaurante de sushi e tentando pegar a melhor combinação de pratos. Cada prato possui uma pontuação específica, a qual define o vencedor no final do jogo.

A jogatina se dá através da dinâmica de “comprar e passar”, mantendo um mínimo de regras. O jogador deve avaliar o destino da rodada e decidir que tipo de sushi irá priorizar.

As diferentes combinações de pontuação permitem algumas jogadas inteligentes - ou combos desagradáveis.

I.III Regras

1. Cada jogador começa com o número máximo de cartas na mão, sete.
2. O jogador só tem visão da sua mão no momento que estiver com ela.
3. A cada rodada do jogo, um jogador deve escolher, dentre todas as cartas em sua mão, apenas **uma**. Esta carta escolhida é reservada pelo jogador e colocada em seu “monte”, e este então “passa” o resto de sua mão para o outro jogador.

4. Algumas cartas possuem habilidade especial. Estas podem ser ativadas durante a etapa de escolha de carta.
5. O “monte” do jogador é visível somente para ele.
6. O jogo acaba quando ambos jogadores tiverem “passado” sua mão ao outro seis vezes, resultando em um monte com **sete** cartas. É neste momento então que as cartas são contabilizadas.
7. A contabilização dos pontos se dá no final do jogo, quando ambos jogadores tiverem passado suas mãos ao outro seis vezes.
8. As regras de contabilização se dão através das cartas selecionadas, sendo elas:
 - a. **Sashimi**: Pontua 10 pontos para cada trio de Sashimi pego
 - b. **Maki**: Pontua 2 pontos pra cada Maki de Dois, e pontua 3 pontos para cada Maki de Três.
 - c. **Tempura**: Pontua 5 pontos para cada dupla de Tempura pego.
 - d. **Bolinho (Dumpling)**: Pontua 1 ponto para um bolinho; 3 pontos para dois bolinhos; 6 pontos para três bolinhos; 10 pontos para quatro bolinhos; 15 pontos para cinco bolinhos
 - e. **Pudim (Pudding)**: Pontua 1 ponto para cada pudim.
 - f. **Nigiri**: 2 pontos para cada nigiri de ovo; 3 pontos para cada nigiri de salmão
9. As cartas habilidade pontuam da seguinte maneira, quando ativadas:
 - a. **Wasabi**: Se a próxima carta escolhida for do tipo *Nigiri*, o Wasabi triplica os pontos do *Nigiri*.
 - b. **Hashi (Chopsticks)**: Adicione +2 pontos extras à próxima carta.

I.IV Componentes

1. Um deque de cartas para cada jogador, com limite de até 7 cartas.
2. Um elemento para confirmar seleção de carta.
3. Um elemento para confirmar o passe de mão.
4. Um elemento para ativar habilidade especial, quando possível.

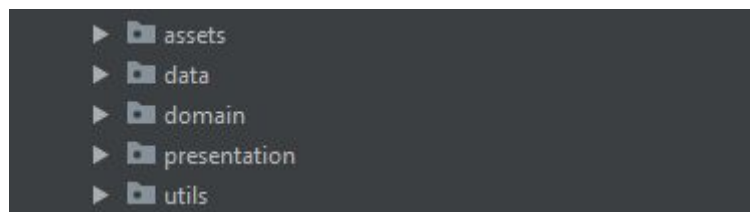
II Visão Geral

II.I Detalhes Técnicos

O sistema será desenvolvido na linguagem Java, utilizando a biblioteca Swing para construção da interface gráfica e seguindo o paradigma de programação orientado-a-objetos.

A arquitetura da aplicação será a Cliente-Servidor, sendo o NetGames o servidor, e a aplicação do Sushi Go o cliente.

O projeto possui cinco módulos principais, como mostra a figura a seguir:



Na pasta `assets` encontram-se miscelâneas em geral, como arquivos `.png` ou `.jpg` para planos de fundos utilizados na interface.

A pasta `data` contém as entidades do projeto e suas abstrações.

A pasta `domain` supervisiona os atores e os controladores - responsáveis por conectar a estrutura do sistema e operar as regras de negócio.

A pasta `presentation` contém todos elementos gráficos utilizados para compor e interface do programa.

Por último, a pasta `utils` contém classes utilitárias do programa.

II.II Premissas

1. Utilizar o framework NetGamesNRT para a conectividade do jogo.
2. Modelagem do projeto baseada em UML 2 através da ferramenta Visual Paradigm.
3. Desenvolvimento totalmente realizado na linguagem Java.

III Requisitos de Software

III.I Requisitos Funcionais

1. Conectar

Jogadores devem possuir a opção de conectarem-se ao jogo.

2. Iniciar partida

Jogadores devem possuir a capacidade de iniciar a partida quando ambos estiverem conectados.

3. Escolher carta

Um jogador deve poder escolher uma carta de sua mão.

4. Ativar habilidade da carta

Um jogador deve poder ativar a habilidade de uma carta que possua habilidade especial se assim desejar.

5. Passar mão

Um jogador deve ser capaz de passar sua mão para o outro após ter selecionado uma carta.

6. Receber mão

Um jogador deve receber a mão do oponente simultaneamente ao passar sua mão para ele.

III.II Requisitos Não-funcionais

1. Especificação do projeto

O programa deve ser escrito em Java, e a modelagem deve ser feita baseada em UML 2.

2. Interface Gráfica

O programa deve apresentar uma interface gráfica minimamente complexa, não sendo tolerável “interface de terminal”.

IV Interface Gráfica

