



Universidade Federal de Pernambuco - Campus Recife

Centro de Informática - CIn

Aluno: Marcos da Silva Barreto (msb5)

Relatório do projeto final para a disciplina Programação Funcional

1. Proposta

Controlar via web um jogo de mapa 2D.

2. O jogo

Arca é um jogo de mapa 2D 20x20. A ideia do jogo veio do filme Indiana Jones e Os Caçadores da Arca Perdida, e o seu objeto é levar o herói até o tesouro, passando por um mapa cheio de obstáculos. O mapa consiste em uma matriz 20x20 (representado por quadrados), onde cada quadrado tem uma coloração e significado diferentes:

- O quadrado preto é o tesouro;
- O quadrado branco é o herói;
- Os quadrados marrons representam o chão, não tem dano associado;
- Os quadrados vermelho escuro representam espinhos, e têm dano igual a 15 de vida;
- Os quadrados verdes representam flecha, e têm dano igual a 25 de vida;
- Os quadrados azuis representam buracos, e têm dano igual a 40 de vida;
- Os quadrados amarelos representam chamas, e têm dano igual a 60 de vida.

O herói inicialmente tem 300 pontos de vida, e estes são descontados a medida em que passa por algum obstáculo mencionado anteriormente. O objetivo do jogo é passar pelo melhor caminho possível, e chegar com vida no tesouro. Durante qualquer momento do jogo o jogador pode utilizar a poção da vida, dando ao seu herói 60 pontos de vida, sendo limitado a 5 aplicações durante todo o jogo.

O jogo dispõe de 3 níveis diferentes de dificuldade: fácil, médio e difícil. As dificuldades alteram as probabilidades dos obstáculos, favorecendo aqueles com maiores danos à medida em que a dificuldade aumenta. Inicialmente o jogo inicia na dificuldade fácil, e passa para as próximas dificuldades assim que o herói chega no tesouro, e chega ao fim após o herói enfrentar todas as dificuldades e alcançar o último tesouro, ou se acabar a sua vida.

3. Desenvolvimento

O projeto foi desenvolvido em três partes: Arca, servidor e frontend. A primeira se refere ao jogo com a lógica descrita na seção anterior juntamente com os recursos gráficos. A segunda e terceira partes representam o serviço web proposto para controlá-lo, com a interface web e servidor respondendo a requisições HTTP.

3.1. Arca

O jogo foi desenvolvido utilizando OpenGL e GLUT para a interface gráfica, escrito em Haskell, e dividido em 5 arquivos, nomeados como: Main, Mapa, Environment, Ganhou e GameOver. Os dois últimos arquivos foram criados para desenhar na interface as mensagens do final do jogo, quando o jogador consegue finalizá-lo, e quando perde. Environment define algumas propriedades do jogo e tipos utilizados em sua construção, como cor, vértices, vida e dano e os objetos (obstáculos). O arquivo Mapa traz algumas funções de renderização do mapa do jogo como por exemplo `mapa` a qual recebe uma matriz de objetos (definidos em Environment) e desenha cada quadrado. Seu código é semelhante ao do arquivo GameOver, mas mais específico e utilizado no fluxo principal.

Por fim, o arquivo Main é onde ocorre a lógica do jogo Arca. Carrega as definições de herói, posição no mapa, cor do herói, nível de dificuldade, a configuração das probabilidades associadas ao nível e um registro da poção. Sua função principal é o fluxo do jogo, criando a janela, inicializando os valores e definindo os *callbacks* de display e teclado para controle do jogo.

Algumas funções interessantes são: *callback* do teclado, escolha do nível e as direções do herói. A primeira é uma função do GLUT a qual recebe entre outros parâmetros a tecla pressionada e na implementação recebe informações de contexto definidas neste arquivo, e de acordo com a tecla é executado uma ação diferente: movimentação 2D do herói, aplicação da poção e mudança de dificuldade. A segunda, de acordo com a entrada entre 1, 2 e 3, modifica a configuração das probabilidades associadas à dificuldade, esta utilizada na renderização do mapa. A última realiza operações com a posição 2D alterando a vida do herói caso passe por um obstáculo e altera o nível caso necessário ao chegar no tesouro.

3.2. Servidor

O servidor web foi desenvolvido utilizando o framework Yesod. Foram criados algumas rotas para requisições HTTP GET representando as ações essenciais do jogo: movimentação 2D, nível de dificuldade e aplicação da poção. Para realizar as ações no jogo de acordo com as requisições, foi utilizado o simulador de eventos de teclado Robot. A depender da ação requisitada, é enviado através do Robot uma ação no teclado que refletirá no jogo.

3.3. Frontend

O frontend da aplicação se refere a interface web que será utilizada pelo jogador para controlar Arca remotamente, enviando através do servidor os comandos

essenciais do jogo. Desenvolvido utilizando HTML, JS, CSS e jQuery Arca necessariamente estará sendo executado no mesmo ambiente que o servidor, por fins de limitação do controle via Robot, e o controle do jogo como movimentação do herói, alteração do nível de dificuldade e a utilização da poção será realizado via o frontend.

4. Dificuldades

- Execução do servidor juntamente com o jogo: como as funções do jogo necessitam das informações de contexto passadas nos argumentos impossibilitava a chamada ao receber uma requisição HTTP GET, dessa forma foi necessário o uso do Robot como forma de interação com o Arca;
- Informações em tempo real do jogo para o usuário via interface web: foi utilizado um arquivo de texto auxiliar para registrar as informações da quantidade de vidas e a mensagem indicando o término do uso das poções escritas pelo jogo, e consumidas pelo servidor sempre que houver uma requisição web específica para tal, retornando ao frontend esses valores, porém as requisições feitas pelo cliente estava rejeitando o retorno e não foi possível contornar essa situação.

5. Anexos

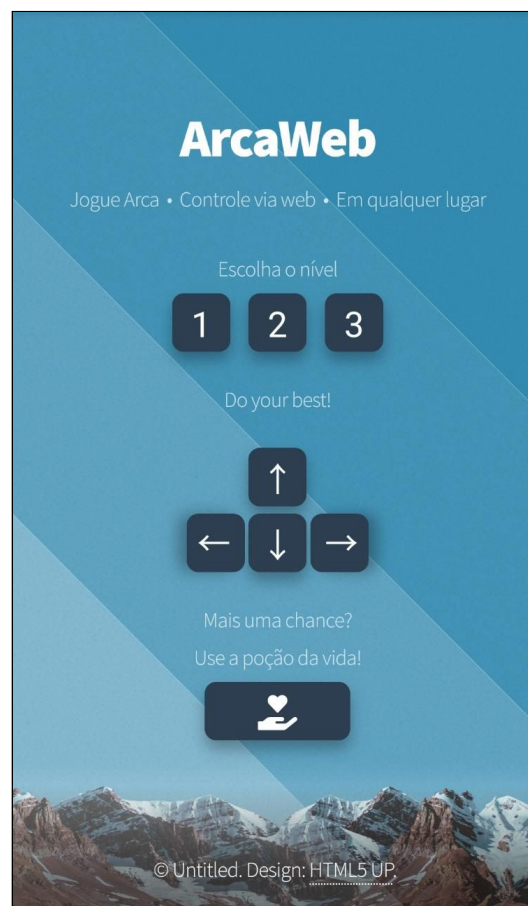


Figura 1: Interface web para controlar o jogo Arca

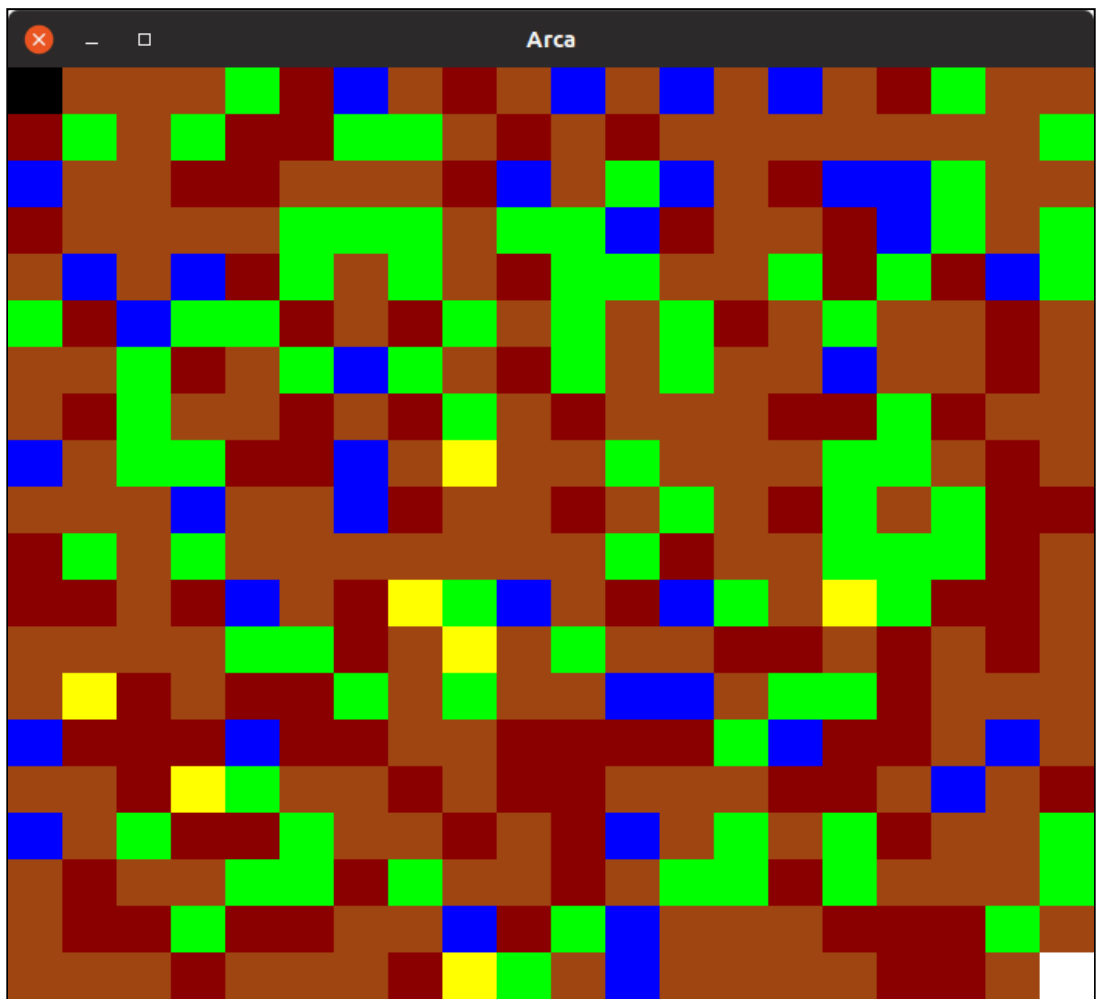


Figura 2: O jogo Arca (nível fácil)