
Colégio Técnico da Universidade Federal de Minas Gerais

Nome: Lorena de Miranda Pacheco

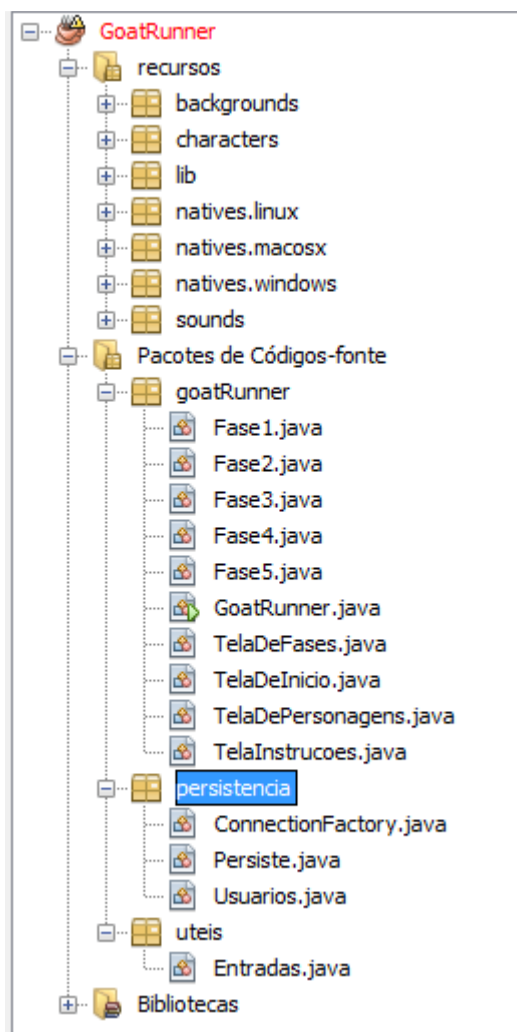
Turma: 303

Data: 24/11/2014

Checkpoint Final de Tecnologia de Programação

Sobre o código:

- O trabalho sofreu modificações do projeto inicial, mas os objetivos traçados inicialmente foram cumpridos
- O código de todas as classes está bem comentado
- As classes estão separadas em pacotes contendo o conteúdo do jogo (goatRunner), uteis (uteis) e comunicação com o banco de dados (persistencia).
 - Dentro do pacote do jogo, tenho as classes de todas as telas utilizadas ao longo do jogo
 - Dentro do pacote uteis, tenho métodos disponibilizados pelo Leandro para auxiliar na utilização de JoptionPanes
 - Dentro do pacote persistência, tenho classes de conexão, a classe usuário, para que possamos tratá-la como objeto posteriormente, e os métodos que contactam diretamente o banco de dados na classe persistência.



- A maior parte das imagens usadas para o jogo foi personalizada por mim

- Pra que o jogo funcione, é necessária a conexão com internet e com um banco de dados. Dentro de tal banco é necessária a criação da seguinte tabela:

“
`create table if not exists usuarios (
 id int(4) auto_increment,
 login varchar(255),
 senha varchar(255),
 estrelas varchar(255),
 fase1 varchar(255),
 fase2 varchar(255),
 fase3 varchar(255),
 fase4 varchar(255),
 fase5 varchar(255),
 cabra1 varchar (255),
 cabra2 varchar(255),
 cabra3 varchar(255),
 cabra4 varchar(255),
 cabraAtual varchar(255)
);`
 “

O Jogo:

- Na primeira tela, o usuário pode escolher entre logar em uma conta já existente, cadastrar-se e, desse modo, criar uma conta nova, ou ir a tela de instruções. Nesta mesma tela já se inicia a música de fundo que tocará em *loop* durante todo o jogo.
- Ao fazer login, o jogador é redirecionado para a tela onde ele escolhe o personagem em que ele irá jogar. Apenas a primeira cabra é liberada para usuários novos. Quando o jogador já possui o personagem ou acabou de o comprar, seu fundo fica verde:



- Para que o jogador compre um dos personagens, ele deve obrigatoriamente possuir o anterior.

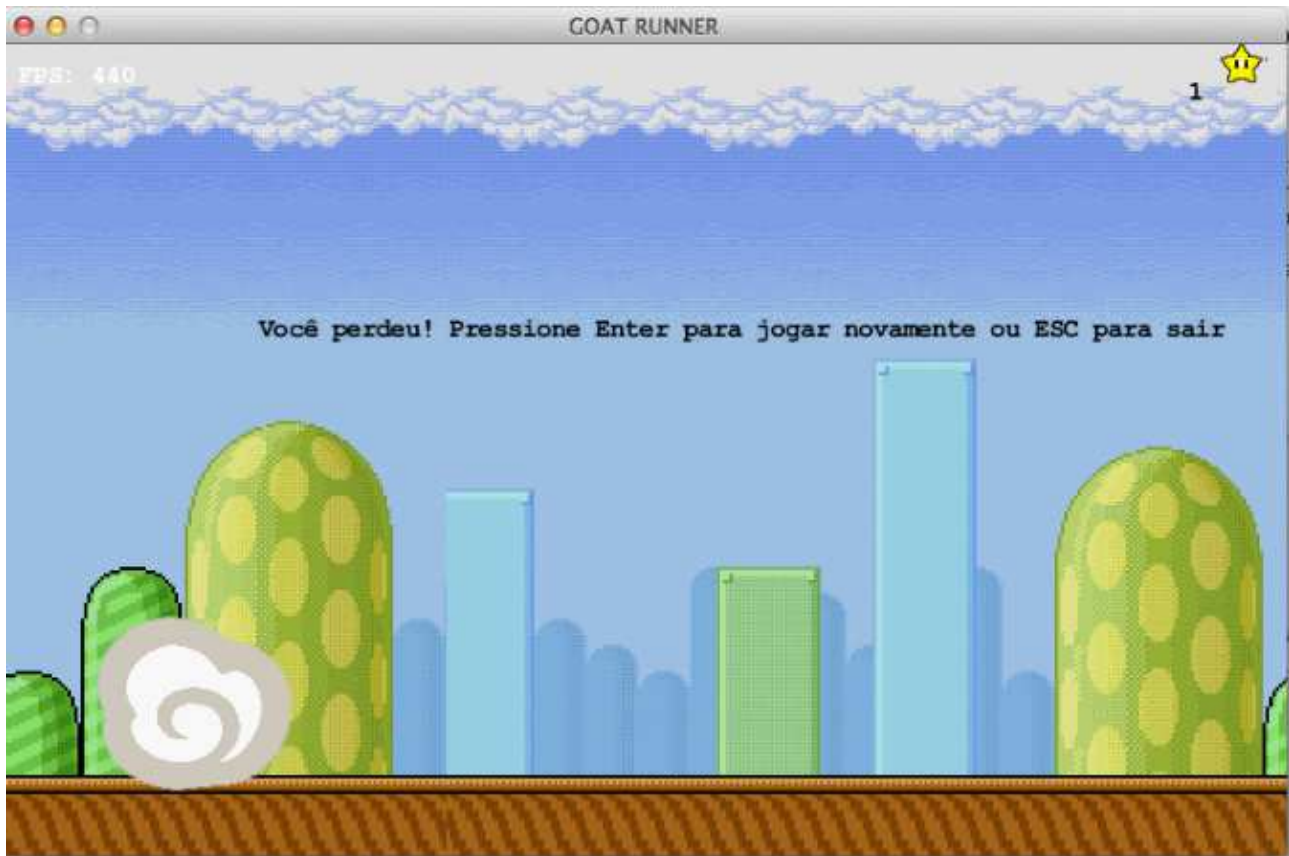
- Ao escolher um personagem válido, o jogador é redirecionado para a tela de fases.
- Na tela de fases, o usuário pode escolher qual nível ele quer jogar. O jogo apresenta 5 fases, e, para usuários novos, apenas uma delas é liberada (Fase 1). Para passar para a próxima fase, o jogador precisa recolher as estrelas do nível anterior para ir comprando as próximas.
- Novamente, só é possível comprar uma fase quando já se possui a anterior.
- As fases que o jogador não possui ficam cinza na Tela de Fases. Ao comprar uma nova, seu fundo fica vermelho.



- Também é possível trocar de personagem (ou comprar um novo) antes de jogar cada fase.
- O jogo contém 5 fases com a posição dos obstáculos e estrelas predefinidas, variando a quantidade e a frequência em cada uma das fases. Cada fase tem 5 obstáculos a mais do que a anterior, o que aumenta o nível de dificuldade.
- O objetivo do jogo é ajudar o personagem a pular os obstáculos e coletar as estrelas de cada fase para não só liberar os próximos níveis, mas também ter a opção de jogar com outras cabras.
- Ao pular, a cabra emite um som. O último personagem tem um som diferente das demais.



- Ao colidir com um obstáculo, os obstáculos seguintes são apagados e a animação da cabra andando é substituída por uma animação de nuvem de fumaça e um som de colisão é tocado.
 - O jogador pode apertar “enter” para repetir a fase
 - O jogador pode apertar a tecla “esc” para voltar a tela de níveis
 - As estrelas coletadas até o momento são perdidas.



- Há também em cada fase um valor de pontuação que mede a “distância” percorrida pelo personagem. (Observar figura 3).
- Durante os níveis, para enfatizar que os obstáculos nem as estrelas são parte do plano de fundo, as estrelas se movem por 500ms no eixo X para a esquerda e depois 500ms no eixo X para a direita.
 - A ideia original era que os obstáculos seguissem tal lógica, porém o nível de dificuldade das fases seria muito aumentado. Como, a nível de código, estrelas e obstáculos são a mesma coisa (retângulos), não houve problemas em tal troca.
- Ao terminar a fase, o jogador deve pressionar enter para ser redirecionado a tela de níveis e para que as estrelas recolhidas na fase sejam enviadas ao banco de dados, através do método atualizaMoedas(int moedasRecolhidasNaFase) e pelo método atualizaUsuario(Usuario user).
 - A atualização do usuário é necessária a cada modificação feita no banco durante o jogo, pois as informações iniciais são gravadas em um objeto estático do tipo Usuarios, o usuarioAtual. O método de atualizar pesquisa no banco o usuário e coloca os resultados no usuarioAtual, para que tais informações possam ser acessadas de outras classes.
 - A mesma atualização acontece quando se compra uma fase ou

personagens novos, pois há atualização de dados na camada de persistência.

- O jogo tem boa jogabilidade, com controles intuitivos e práticos. O código está feito de forma que adições de novos modos, fases ou personagens aconteça sem muita dificuldade.