

Universidade Federal de Goiás  
Bacharelado em Ciência da computação  
Campus Goiânia - Samambaia

Projeto de Software

PAC-MAN

Dezembro  
2017

Universidade Federal de Goiás  
Bacharelado em Ciência da computação  
Campus Goiânia - Samambaia

## Projeto de Software

Trabalho desenvolvido  
na disciplina de Projeto de  
Software com intuito de  
elaborar um projeto para o  
desenvolvimento de um jogo  
no estilo PAC-MAN. Orientado  
pelo Prof. Marcelo Quinta.

Dezembro  
2017

## Escopo do Trabalho

1. Descrição do Sistema
2. Requisitos do Sistema
3. Diagramas
4. Padrão de Projeto
5. Estimativas e Considerações Finais

# 1. Descrição do Sistema

O sistema é um jogo que será disponibilizado no formato web, o jogo consiste em um estilo parecido com o do famoso Pac-Man. O jogador principal controla um personagem que deve se movimentar em quatro direções (Cima, Baixo, Esquerda e Direita) em uma mapa de duas dimensões, coletando o máximo de moedas e acumulando pontos e eventualmente escapando de inimigos que possuem características diferentes quanto a movimentação.

Será utilizado um sistema Cliente-Servidor e um padrão arquitetural do tipo MVC (Model View Controller). Para padrão de projeto foram escolhidos, Observer e Flyweight. Para facilitar as mecânicas do jogo em relação aos inimigos, como há muitas mudanças de estados, utilizaremos o padrão de projeto Observer. Como se trata de um jogo com poucos sprites mas muita repetição dos mesmos, utilizaremos o padrão Flyweight. Ambos os padrões são amplamente utilizados no mercado de jogos.

## 2. Requisitos do Sistema

### 2.1 Requisitos Funcionais

- [RF01] O usuário deve ser capaz de jogar sem ser cadastrado.
- [RF02] O usuário deve ser capaz de se cadastrar.
- [RF03] As informações de cadastro serão "nome", "email" e "senha".
- [RF04] O usuário deve ser capaz de atualizar e deletar sua conta.
- [RF05] O sistema deve ser capaz de manter os dados das pontuações dos usuários no formato de um ranking.
- [RF06] Ao final de uma partida o usuário deve ser capaz de continuar jogando ou se cadastrar.
- [RF07] O usuário deve ser capaz de movimentar o pac-man nas direções norte, sul, leste, oeste.
- [RF08] O usuário não deve ser capaz de atravessar paredes.
- [RF09] O usuário deve ser capaz de coletar moedas.
- [RF10] O usuário deve ser capaz de coletar pílulas.
- [RF11] As pílulas dão invulnerabilidade ao jogador.
- [RF12] A invulnerabilidade permite que o pac-man colete inimigos e os inimigos dão pontuação.
- [RF13] Coletar moedas garante 10 pontos, coletar pílulas garante 50 pontos, coletar frutas garante 100 pontos e coletar inimigos garante 200 pontos.
- [RF14] O usuário deve ser capaz de coletar frutas.
- [RF15] As frutas, moedas e pílulas, aparecem apenas no início do jogo e só podem ser coletadas apenas uma vez por jogo.
- [RF16] Um inimigo quando coletado, retornará ao centro do mapa e esperará o fim da invulnerabilidade do pac-man para retornar ao resto do mapa.
- [RF17] Caso o pac-man entre em contato com um inimigo e não esteja invulnerável, o jogador perde uma vida e volta para a posição inicial do jogo.
- [RF18] A posição inicial do jogo deve ser na parte inferior central do mapa.

[RF19] O jogo acaba quando o jogador coleta todas as moedas do mapa ou quando ele perde todas as vidas.

[RF20] O jogador começa com três vidas.

[RF21] A pontuação do jogo deve ficar na parte superior esquerda da tela e ser atualizada automaticamente sempre que mais pontos forem adquiridos pelo jogador.

[RF22] A pontuação inicial de cada novo jogo é zero.

[RF23] Pacman anda até parar na parede

## 2.2 Requisitos Não Funcionais

[RNF01] As paredes, inimigos, moedas, pílulas, frutas e o pac-man, possuem cores distintas.

[RNF02] O sistema deve ser de alta Responsabilidade.

[RNF03] O sistema deverá ser de alta Disponibilidade.

[RNF04] O tempo de carregamento do site não deve passar de 3 segundos.

### 3. Diagramas

Diagramas disponíveis em:

<https://github.com/viniciuspx/Pac-man>

Foram feitos alguns diagramas UML no desenvolvimento do projeto, como de Classe, Implantação, Interação, Estados e Caso de Uso. Tratando-se de um jogo, Caso de uso, Interação e Classe vem a ser alguns dos mais importantes diagramas relacionados a desenvolvimento de um jogo mas foi analisado que seria necessário também um diagrama de Interação por tratar-se de um sistema web e consequentemente um de implantação desse sistema.

## 4. Padrão de Projeto

O padrão Flyweight foi utilizado devido a característica do projeto, o projeto possui poucas sprites e muitas ocorrências das mesmas imagens, o padrão ajudará pois facilitará essa repetição de sprites, diminuindo o uso de memória do jogo.

O padrão Observer foi escolhido pois o Pac-man muda de estado toda vez que coleta uma pílula e conseqüentemente o Fantasma (Inimigo) muda também, para facilitar essa situação entra o conceito do Observer. Todas as vezes que o Pac-man muda de estado o objeto Observer notificará os Fantasmas que mudaram de estado também.



## 5. Estimativas e Considerações Finais

Aproximadamente 12 horas para desenvolvimento de Projeto, diagramas e etc.

Aproximadamente 16 horas de desenvolvimento de Código.