Fundação Getulio Vargas

Trabalho A2 Linguagem de Progamação

The Last Minute Coding

Lucas Batista, Samyra Mara, Gabriel Carneiro

Professor Matheus Telles Werner

5 de dezembro de 2024

Relatório da A2

Lucas Batista Pereira, Samyra Mara, Gabriel Carneiro

Resumo

O presente relatório concerne ao trabalho realizado pelo grupo constituído pelos alunos Gabriel Carneiro, Samyra Mara e Lucas Batista da FGV-EMAp, válido para a segunda avaliação da disciplina do 2° período Linguagens de Programação, disciplina ministrada pelo Prof. Mateus Werner. O foco do trabalho foi a aplicação dos conceitos de programação orientada a objetos (classe, objeto, herança, relações entre classes, encapsulamento etc.) por meio do desenvolvimento de um jogo a partir da biblioteca `Pygame` do Python.

Sumário

1 Introdução

As aulas da disciplina de linguagens de programação da segunda parte de 2024.2 se focaram especialmente na introdução de conceitos do paradigma de programação de orientação a objetos e na introdução ao Pygame. Foi requisitado para a avaliação que os alunos produzissem um jogo utilizando a biblioteca `Pygame` do Python, implementando os conceitos vistos em sala na segunda parte do semestre e solidificando os conhecimentos em desenvolvimento de projetos.

O jogo desenvolvido pelo grupo chama-se **The Last Minute Coding**, e se baseia na seguinte história:

> Você é um aluno da FGV da matéria de "Litografia Promissora", e em uma das suas sessões de cochilo diárias você se lembra que tinha um trabalho **MUITO** importante (para o qual você tinha um mês para fazer) para ser entregue para daqui 1 hora! Para manter seu CR e não reprovar na matéria, você precisa passar por todos os desafios, eles são:
>
> - Esquivar dos anúncios e pressionar todos os comandos para conseguir entrar no site da FGV;
> - Fazer as tarefas das outras matérias (que, por um acaso, você também esqueceu);
> - Batalhar contra o Python para fazer com que seu código funcione;
> - Documentar seu código usando Sphynx, um desafio do qual poucos saem vivos;
> - Usar uma famosa plataforma de controle de versão (que você deveria ter aprendido a usar no primeiro semestre, mas...)
>
> Será que você tem o que é preciso para ajudar nossos heróis? Se divirta!

2 Organização do trabalho

O trabalho foi organizado da seguinte forma:

```
trabalho-lp-a2/
    assets/             # Diretório que contém as imagens e sprites
    config/             # Diretório que contém os arquivos das informações de cada fase
    src/                # Diretório que contém as classes e os códigos que definem o jogo
    tests/              # Diretório contendo os testes unitários
    docs/               # Diretório que contém o relatório do grupo
    readme.md           # Arquivo contendo as informações do projeto e de como executá-lo
    requirements.txt    # Arquivo contendo as bibliotecas usadas
```

O grupo organizou 2 reuniões semanais para acordar na divisão das tarefas e na confecção geral do código (ou seja, o que seria colocado e como iríamos implementar cada parte do jogo), de forma que a divisão foi a seguinte:

- Samyra Mara
  - Design das sprites e dos fundos
  - Configuração da Fase 4
  - Configuração dos Personagens
- Gabriel Carneiro
  - Confecção do Relatório
  - Confecção dos testes unitários
  - Configuração das fases 1 e 5
- Lucas Batista
  - Configuração do Menu
  - Configuração das fases 2 e 3
  - Configuração dos NPC's
  - Configuração dos objetos

Como o núcleo do projeto está no diretório `src` e o restante dos diretórios têm funções autoexplicativas ou que são melhores explicadas contextualizados, começaremos o explicando.

2.1 Organização do Projeto

Ao abrir o diretório `src`, essa será a configuração encontrada:

```
src/
    armas/            # Define armas e ataques
    boss/             # Define os chefes e suas habilidades
    game/             # Define o jogo e as classes relacionadas à execução
    interagiveis/     # Define os objetos que podem interagir com o personagem
    manager/          # Gerencia as fases e concatena os módulos do jogo
    mapa/             # Define o mapa e seus métodos de carregar cada fase
    personagem/       # Define os personagens e seus inimigos
    ui_menu/          # Define caixas de textos e blocos de seleção
```

Os conteúdos de cada fase estão em arquivos json na pasta `config`. Cada um deles determina os objetos e inimigos de uma fase, e o mapa de cada fase é construído pelas classses presentes no módulo `mapa.py`.

As classes relacionadas às movimentações, ataques e interações tanto do jogador quanto dos inimigos comuns estão em `personagem.py`. Para construí-las, usamos uma classe base `personagem`, que é herdada pela classe `perso_Controle`,

para o jogador, e pela classe `Inimigo` para os inimigos comuns. Os chefes também herdam a classe `personagem`, mas a organização de seus métodos e sua classe estão no arquivo `boss.py`.

Os módulos `manager.py` e `ui_menu` são responsáveis pelo menu e pelo relacionamento entre menu, fases e jogo.

A organização do jogo está no módulo `game.py`, onde os vários objetos definidos no restante dos módulos são unidos para formar o jogo de maneira sequencial: menu, fases, transição entre fases, morte etc.