

Игра Pygame “Witch adventure”

Авторы проекта:

Болдин Николай - Team Lead

Орлов Александр

Петрова Полина

Идея:

Создать красивую игру развлекательного характера. В нашей игре мы хотели реализовать несколько уровней, в каждом из которых наш главный герой - ведьма Моргана обучается разным способностям и борется с врагами с помощью “заклинаний”. Кроме самой сути игры, мы хотели сделать хорошую графику и озвучку. Игра должна была получиться красивой и гармоничной с множеством анимаций. Должен быть функционал, например, индикаторы здоровья, способностей и усталости. Также хотелось оформить с помощью нескольких экранов - главный экран, описание, управление и отдельные экраны под каждый уровень.

Реализация:

Для реализации нашей идеи нам понадобилось несколько файлов:

- в `methods.py` содержатся стандартные, базовые функции для работы `pygame`;
- в главном файле `main.py` находится класс `Game()`, отвечающий за переход между экранами
- Файл `player.py` отвечает за главного игрока и все его способности, например, класс `FireBall()` отвечает за “заклинания”, то есть огненные шарики, которые главный герой бросает во врагов
- файл `enemies.py` содержит классы разных врагов

Из интересных приемов можно отметить, что мы использовали библиотеку для работы с `tmx`-файлами, которой нет в `pip`. Эта библиотека облегчает работу с игрой, появляется возможность работать с экраном, как с картой и добавлять различные свойства к объектам карты. Данная библиотека была позаимствована здесь:

<https://github.com/renfredxh/tmx>

Описание технологий и библиотеки:

Для запуска программы понадобились библиотеки:

- `pygame`
- `tmx`
- `os`
- `sys`

Технологии:

При запуске программы открывается начальный экран с названием игры и кнопкой “начать игру”. За это в классе Game() отвечает функция main_menu(), там есть цикл while, отвечающий за работу окна и анимацию. После нажатия кнопки button() мы переходим к следующей функции start_screen2(), которая отвечает за работу второго экрана, на котором также происходит анимация фона, имеется описание и две кнопки: button_control() открывает экран “управление”, а button_game() запускает саму игру - функцию main(). Также имеются функции action1-4(), которые отвечают за смену флагов - переходы к другим экранам с помощью кнопок и функция terminate() отвечает за закрытие экрана. В функции main() содержится главный игровой цикл, отвечающий за прорисовку всех объектов на экране (главный герой, враги, карта, полосы здоровья, способностей и усталости). При окончании игры (проигрыше) появляется “GAME OVER” с помощью полупрозрачного затемнения экрана. В __init__() класса Game мы загружаем все картинки для работы всех экранов, карту и спрайты героев.

В файле enemies.py мы описываем всех врагов, загружаем их изображения, функции set_frame() отвечает за анимацию и update() отвечает за их направление перемещения и скорость.

В классе Player(), в __init__() загружаем всё, что связано с персонажем - его изображения, скорость обновления, различные способности, его размеры, характеристики, а также звуки действий героя. Функция set_frame() отвечает за всевозможные анимации; cast_spell() изменяет направление героя; gravity() задает гравитацию прыжка; update() отвечает за все движения героя - перемещение, прыжок, столкновение и отвечает за анимацию этих движений. В классе FireBall() описывается броски огненных, их скорость, анимации. Функции set_frame() отвечает за анимацию и update() отвечает за их направление перемещения и скорость. В классе Book() описываются способности, получаемые персонажем.

