**OkayJet**

**OkayJet** — аркадная инди-игра, вдохновленная легендарной мобильной игрой *Jetpack Joyride*.

**История создания**

Мы долго рассматривали идеи для проекта, искали примеры в Google Play и AppStore. Например, сначала мы хотели сделать игру, подобную Knife Hit, и даже уже создали репозиторий. Однако, в последний момент мы вспомнили про игру Jetpack Joyride. После недолгих рассуждений, было принято решение реализовать игру наподобие этой.

**Авторы**

* [Protocs (Уткин Никита)](https://github.com/Protocs)
* [makeitokay (Васильев Андрей)](https://github.com/makeitokay)
* [nikitakosatka (Усатов Никита)](https://github.com/nikitakosatka)

**Основные классы**

**Основная логика**

*Game* (ядро игры) – класс, отвечающий за экран самой игры. На нем помимо самой игры отображается количество собранных монет, пройденный путь и количество имеющихся дополнительных жизней.

**Остальные окна**

*Start* – класс экрана заставки. На нем отображается приглашение к игре, логотип и кнопка выключения музыки;

*Pause* – класс экрана паузы. Присутствует четыре кнопки: возобновить игру, начать заново, выйти в главное меню и выйти из игры;

*Death* – класс экрана смерти; появляется после того, как у игрока закончатся дополнительные жизни и он столкнется с препятствием. На этом экране присутствует информация о пройденном пути, собранных монетках и о лучшем результате игрока (был побит рекорд или нет?). Есть кнопка рестарта, выхода в главное меню и выхода из игры.

**Игровые объекты**

*GameObject* – основной класс объекта, от которого наследуются все игровые объекты;

*XSlideObject* – класс объекта, скользящего вместе с фоном;

*AnimatedSprite* – класс анимированного спрайта, атрибут image которого меняется каждый определенный промежуток времени;

*Obstacle* – основной класс препятствия, от которого наследуются все имеющиеся препятствия;

* *Rocket* – класс ракеты;
* *YellowLaser* – класс желтого лазера;

*Player* – основной класс игрока;

*CoinStructure* – класс, объединяющий монеты в одну структуру;

*Coin* – класс монетки;

*ExtraLife* – класс дополнительной жизни;

*MetresCounter* – класс-счетчик пройденных метров, отображающийся на экране игры и на экране смерти;

*CoinCounter* – класс-счетчик собранных монет, отображающийся на экране игры и на экране смерти;

*MusicButton* – класс кнопки выключения музыки.

**Вспомогательные файлы**

settings.py – файл, содержащий основные настройки игры (константы);

util.py – файл, содержащий вспомогательные функции;

events.py – файл, содержащий информацию о событиях.

Главный запускаемый файл – *okayjet.py*

**Особенности приложения**

* Код проекта разбит на множество файлов. Каждый класс располагается в отдельном файле.
* Большинство дизайна и музыка разработаны нами.
* Уровень генерируется случайно.
* Есть стартовое окно, экран смерти, дополнительные жизни.
* Лучшие результаты игроков сохраняются.

**Использованные библиотеки**

* [**PyGame**](https://pygame.org)

**Использованное программное обеспечение**

* [**PyCharm 2018 Community Edition**](https://www.jetbrains.com/pycharm/?fromMenu)
* [**Python 3**](https://python.org)
* [**Adobe Photoshop CC**](https://www.adobe.com/ru/products/photoshop.html)

**Скриншоты**

Рисунок 1. Стартовый экран

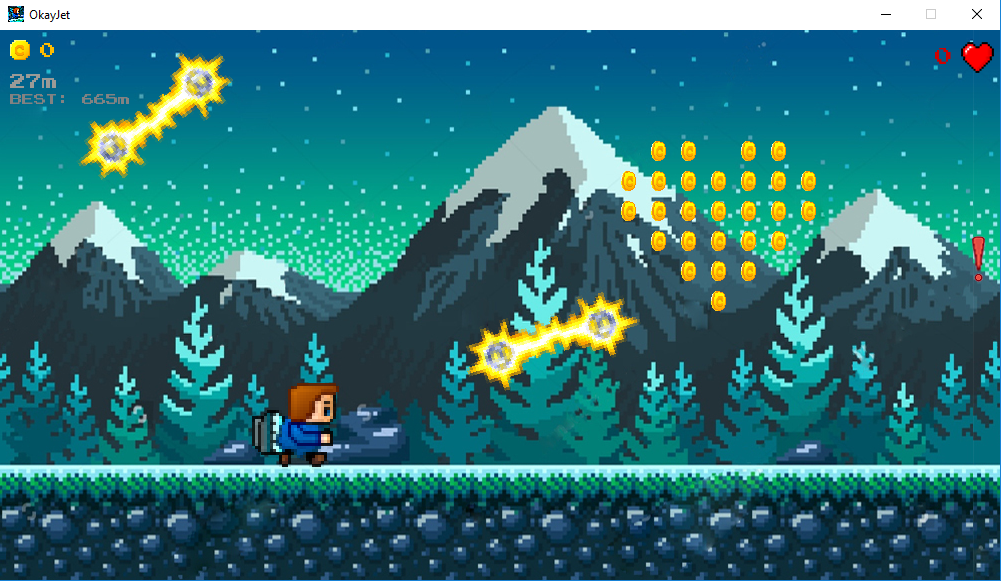
****

Рисунок . Игровой процесс

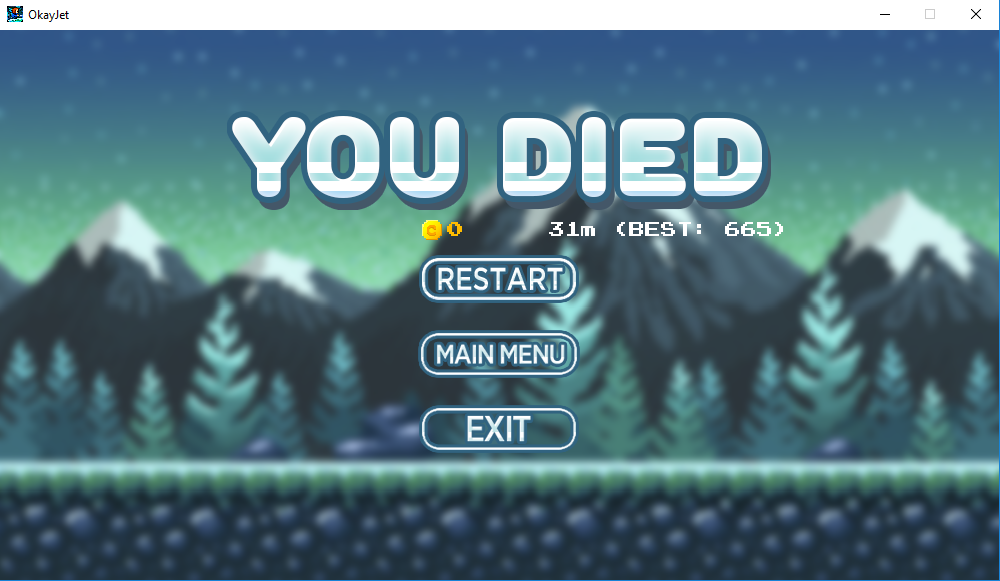


Рисунок . Экран смерти

**Заключение**

* Подведем итоги. У нас получилось реализовать весь функционал, однако всё же осталось несколько недоработок. Например, магазин игровых скинов.
* В процессе работы мы использовали доску задач на GitHub, что значительно облегчило нам задачу.

