**Пояснительная записка**

Название проекта: Run or die

Создан в рамках проекта Яндекс Лицей по модулю PyGame.

Авторы проекта: Мясникова Дарья, Исаков Иван.

Идея:

Игрок при помощи стрелок передвигается по карте и собирает очки. Его цель добраться до портала, собрав как можно больше очков. Получать их можно следующим образом:

1. Собирать предметы на карте. Маленькие предметы дают 5 очков, большие – 50 очков.
2. Проходить игру как можно быстрее. В конце игры за каждую секунду добавляя дополнительно 2 очка.
3. Сохранить как можно больше жизней. В конце игры за каждую жизнь начисляется 100 очков. Жизни отнимают препятствия и враги, их следует избегать.

Если в конце игры кол-во очков игрока больше, чем результат в рейтинговой таблице, то в рейтинговую таблицу записывается текущий результат игрока.

Если игрок не успевает добраться до выхода пока или теряет все жизни, он проигрывает. Очки за уровень при этом не сохраняются.

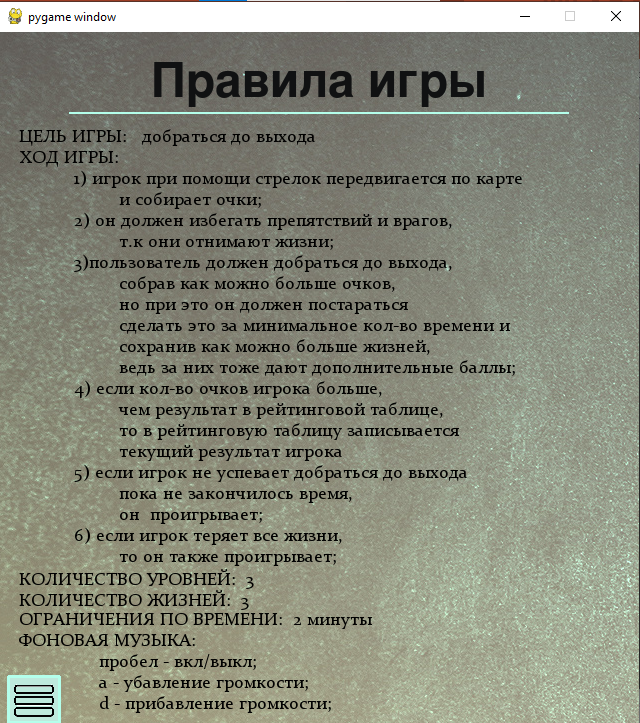
Приложение состоит из следующих элементов:

* Меню

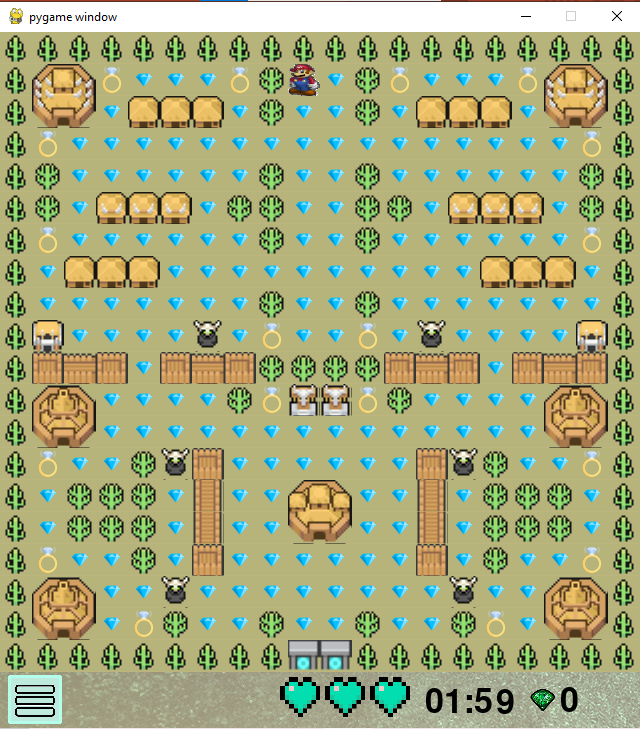








* Уровни







* Окна победы и проигрыша





Описание реализации:

* В игре реализованы классы игрока, врагов, карты, очков, игры (класс, объединяющий предыдущие 4 класса), звёзд и кнопок
* Были использованы технологии обработки событий мыши и клавиатуры; спрайты, изображения, музыка.
* Уровни были разработаны в программе Tiled Map Editor, игра работает на PyGame и встроенных библиотеках Python
* Информация о количестве очков хранится в базе данных.

Используемые библиотеки:

* pygame
* pytmx
* sqlite3
* random
* sys
* os
* time

Описание некоторых технологий:

* Пользователь вводит информацию в QLineEdit (логин, пароль, события и т.д). Из QLineEdit информация считывается и обрабатывается
* Информация заноситься и хранится в базе данных
* Если введённая пользователем информация неверна, то в QLable вылезает сообщение об ошибке
* Из QCalendarWidget считывается и заносится в базу данных выбранная дата
* Из QTimeEdit считывается и заносится в базу данных выбранное время
* В базе данных осуществляется поиск событий относительно выбранной даты, а затем найденная информация записывается в QPlainTextEdit
* С помощью pytmx к коду привязаны .tmx файлы, в которых находится карты для уровней. К ним в свою очередь привязаны .tsx и .png файлы, которые содержат информацию об объектах на этих картах.
* Обновление анимации и перемещения противников, а также таймер работают с помощью событий (pygame.event).

Чеклист:

* requirements.txt
* Стартовое окно
* Финальное окно
* Подсчет результатов
* Спрайты
* collide
* Анимация
* Несколько уровней
* Хранение данных (txt, БД)