**Техническое задание проекта на pygame, выполненного Исаевым Владиславом и Черноусовым Владимиром**

**Название проекта:** Игра «Танчики»

**Краткое описание:** Данный проект является компьютерной игрой, вследствие чего предусматривается одна категория пользователей - игроки. В процессе работы приложения пользователь является непосредственным участником игрового процесса и оказывает непосредственное влияние на него.

Программа должна обладать следующим функционалом:

а) звуковой функционал:  
 1) регулировка общей громкости;  
 2) регулировка громкости музыки;  
 3) регулировка громкости звуков в игре;  
б) внутри игровой функционал:  
 1) система взаимодействия игровых объектов;

2) боевая система;

Игра должна быть создана с использованием модуля pygame, в ней можно будет управлять своим танком. В ней будет таблица лучших результатов, которая будет храниться в файле.

**Классы:**

1. Button()
   1. Создание кнопок
2. Bullet()
   1. Класс пуль
3. Tile() и Wall()
   1. Классы для прорисовки карты
4. Player()
   1. Класс игрока
5. Enemy()
   1. Класс врагов
6. Explosion()
   1. Класс анимаций
7. InputBox()
   1. Класс для создания полей ввода данных

**Функции:**

1. main\_menu()
   1. Функция главного меню
2. size\_menu()
   1. Функция нужна для более удобной работы программы
3. except\_hook()
   1. Функция для вывода ошибок программы
4. terminate()
   1. Закрытие всех окон, выход из игры
5. load\_image()
   1. Функция для загрузки необходимых для работы программы изображений
6. print\_text()
   1. Функция выводит на экран необходимые сообщения
7. sign\_in()
   1. Вход пользователя в игру
8. sign\_up()
   1. Регистрация пользователя в игре
9. blok\_start\_game()
   1. Блокирует запуск игры пока игрок не авторизовался
10. restart\_auth()
    1. Запускает заново процесс авторизации игрока, если он вышел из аккаунта
11. save()
    1. Если игрок выиграл или проиграл, то полученные очки сохраняются в базе данных
12. score\_update()
    1. Если игрок убивает вражеский танк, то в зависимости от прошедшего времени с начала игры игроку присваиваются баллы
13. hit\_player()
    1. Если вражеский танк попадает в игрока, то у него отнимаются жизни и если количество = 0 игрок «Умирает»
14. load\_level()
    1. Загружает необходимый уровень игры
15. generate\_level()
    1. Генерирует уровень
16. start\_game()
    1. Сама игра
17. check\_sounds()
    1. Проверяет громкость всех звуков игры
18. how\_to\_play()
    1. При запуске пользователю рассказывается как играть
19. options\_menu()
    1. Настройки игры