

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по курсовой работе
по дисциплине «Web-технологии»
Тема: Разработка игр на языке JavaScript

Студент гр. 0382

Азаров М.С.

Преподаватель

Беляев С. А

Санкт-Петербург

Цель работы.

Используя навыки web-разработки полученные в течении курс разработать браузерную игры.

Задание.

Необходимо выполнить курсовую работу в соответствии с учебным пособием Беляев С.А. «Разработка игр на языке JavaScript».

1. Минимум 2 уровня игры
2. Реализованы все менеджеры в соответствии с учебным пособием (УП)
3. Есть таблица рекордов
4. Есть препятствия
5. Есть «интеллектуальные» противники и «бонусы»
6. Используются tiles с редактором Tiled (www.mapeditor.org) в соответствии с УП

Выполнение работы.

В данной работе будет разработана игра “Танчики” в которой есть танк игрока , и задача игрока уничтожить все вражеские танки используя возможность передвигаться, уворачиваться от снарядов и стрелять. Также в игре будет присутствовать бонус восполнения здоровья и увеличения урона. После убийства всех врагов игрок переходит на следующий уровень. В течении игры игрок накапливает счет за убийство противников и подбор бонусов , который в конце игры сохраняется в общую таблицу результатов .

Для реализации данных механик были созданы следующие классы:

- EventManager – отвечает за обработку команд пользователем .
Именно здесь обрабатывается нажатие клавиш игроком .

Посредством этого класса другие объекты знают какую клавишу нажал пользователь и нажал ли вообще.

- MapManager – отвечает за загрузку карты и текстур , а также за взаимодействие с ней.
- PhysicManager – отвечает за физику игры , здесь обрабатываются столкновение объектов и законы по которым они двигаются .
- SoundManager – отвечает за загрузку и воспроизведение звуков игры.
- SpriteManager – отвечает за анимации объектов и загрузку спрайтов.
- GameManager – отвечает за всю игру , именно он инициализирует все менеджеры упомянутые выше и описывает взаимодействие между ними , также имеет функцию update() которая обновляет игру.

Также были реализованы классы игровых сущностей:

- B
- B
- Boom – класс взрыва после смерти танка
- Bullet – класс пули
- EnemyTank – класс вражеского танка
- E
- H
- H

Также в ходе выполнения работы была создана следующая файловая архитектура:

h

h класс бонуса повышающего уровень игрока

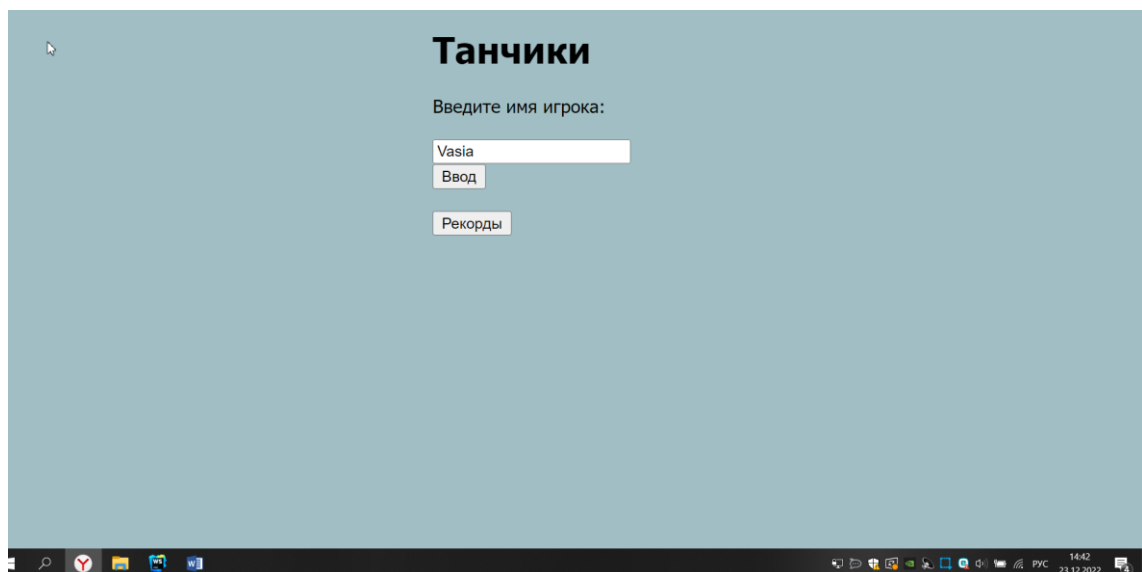
= класс бонуса повышающего здоровье.

В папке static хранятся статические файлы которые выдаются сервером при обращении к ним. Все файлы в этой папке делятся на levels – json файлы карты, sound – папка со звуками, tiles – хранит все текстуры игры.

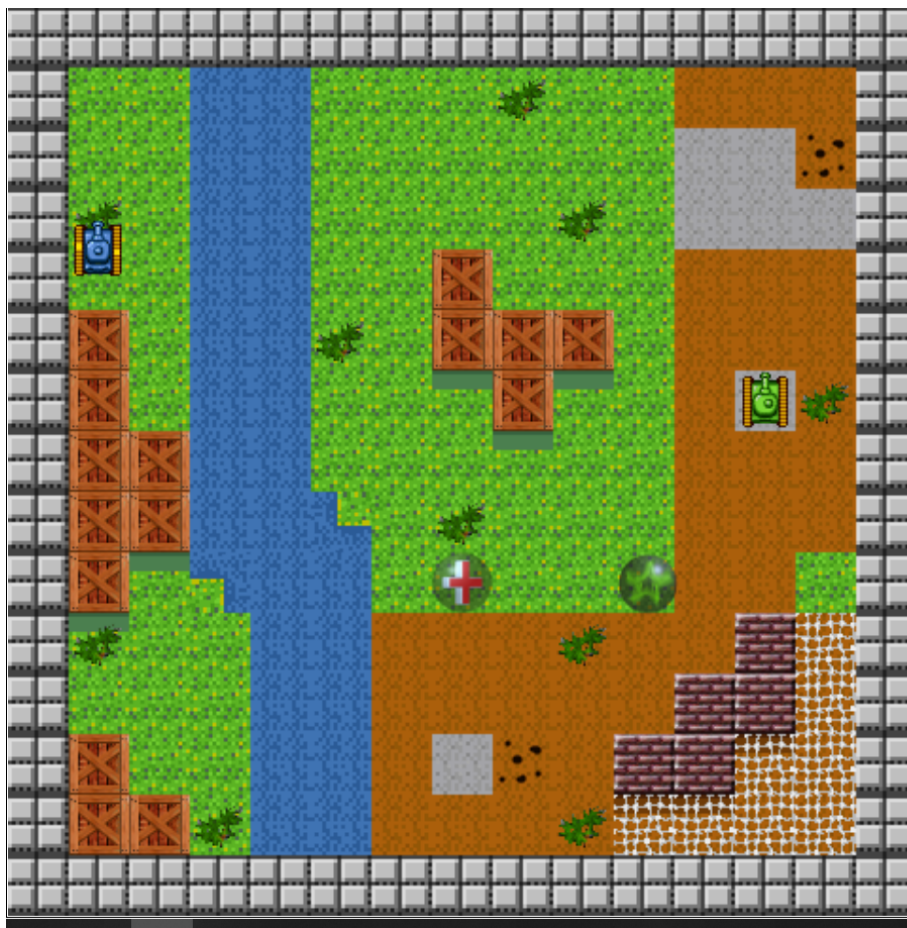
В папке script хранятся все скрипты игры.

Также на внешнем уровне присутствуют html файлы начальной страницы
файле server.js

Таким образом сайт имеет начальную страницу на которой пользователь вводит свое имя и может посмотреть общий счет.



После нажатия клавиши ввод пользователь переходит на страницу игры и может начать проходить первый уровень



В случае проигрыша выведется соответствующее сообщение и можно будет начать играть сначала.



Если игрок уничтожит всех противников выведется соответствующее сообщение и игрок перейдет на следующий уровень.



Выводы.

Был изучен процесс создания игр на языке JavaScript. Полученные знания были применены для выполнения поставленной задачи.