**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

2022

## 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 .

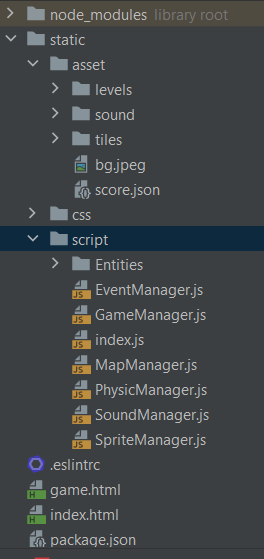
Для реализации данных механик были созданы следующие классы:

* EventManager – отвечает за обработку команд пользователем . Именно здесь обрабатывается нажатие клавиш игроком . Посредством этого класса другие объекты знаю 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:

* BonusDamage – класс бонуса повышающего урон игрока
* BonusHealth – класс бонуса повышающего здоровье.
* Boom – класс взрыва после смерти танка
* Bullet – класс пули
* EnemyTank – класс вражеского танка
* Entity – класс объектов, все объекты игры наследуются от этого класса.
* Player – класс игрока.

Также в ходе выполнения работы была создана следующая файловая архитектура:

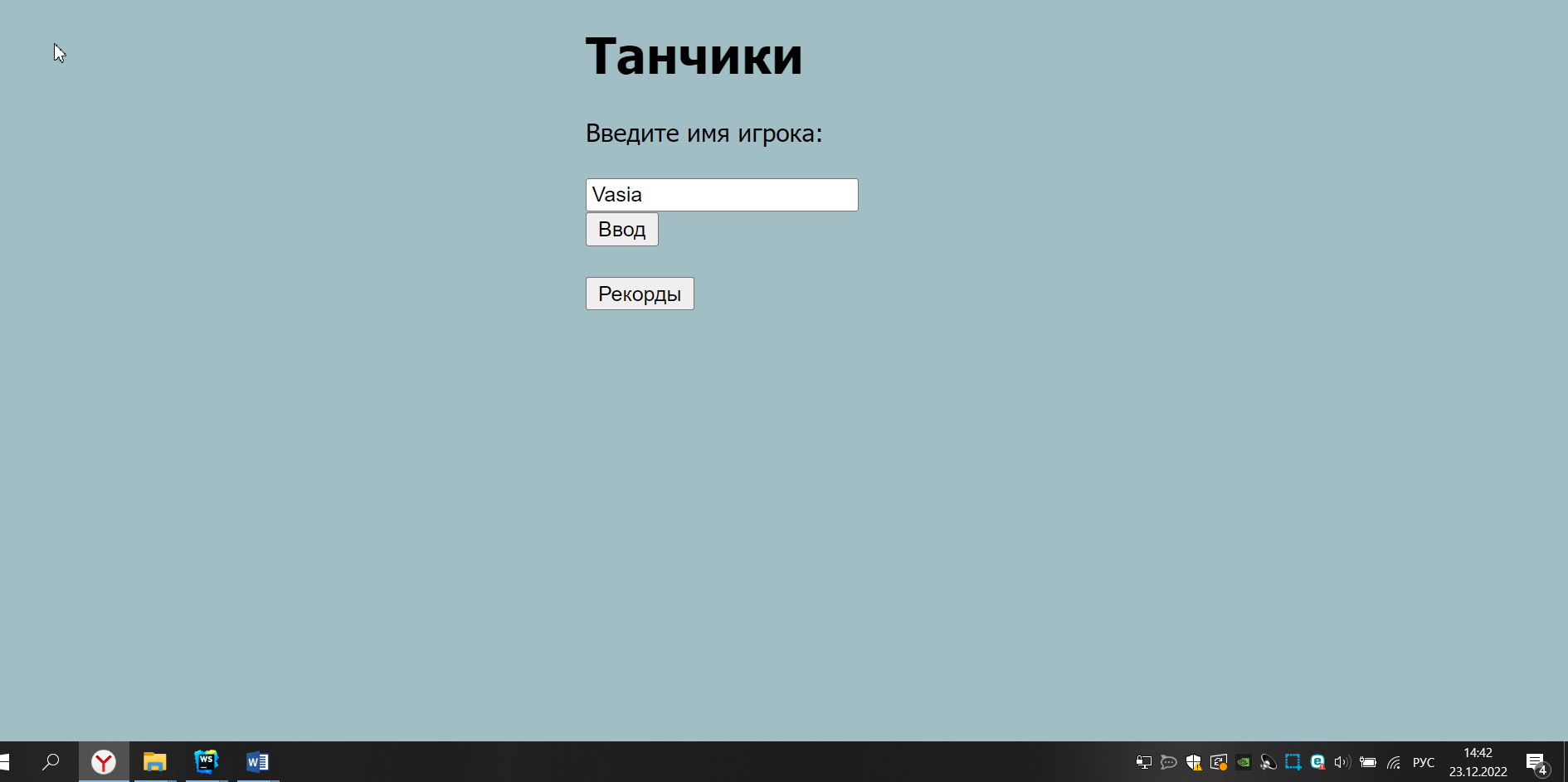


В папке static хранятся статические файлы которые выдаются сервером при обращении к ним. Все фалы в этой папке делятся на levels – json файлы карты , sound – папка со звуками, tiles – хранит все текстуры игры.

В папке script хранятся все скрипты игры.

Также на внешнем уровне присутствуют html фалы начальной страницы index.html и страницы игры game.html. Здесь же хранится серверный код в файле server.js

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



После нажатия клавиши ввод пользователь переход 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.