**基于Three.js库的网页版小车游戏的开发**

开发小组成员：陈诺言[1](组长) 戴梓莘[2]

开发小组所属学院：教育科学与技术学院

专业：教育技术学(师范)[1] 计算机科学与技术(师范)[2]

学号：202005720304[1] 202005720307[2]

指导老师：张维泽

摘要：鉴于目前Three.js库有强大的网页开发功能，基于本学期Web前端开发课程所学知识，以及部分WebGL、NodeJS、Blender建模知识，编写了一个高自由度的网页版小车游戏。该游戏实现了利用方向键移动、碰撞检测、切换第一第三人称视角、重置游戏、显示当前速度等功能，具有较好的实践意义。

关键词：Three.js；游戏

**目录**

1. **引言**

§1.1 项目背景

§1.2 参考资料

§1.3 版权声明

1. **项目概述**

§2.1 项目工作内容

§2.2 项目运行环境与限制

§2.3 项目功能

§2.4 项目实现的不足与反思

1. **其他**

§3.1 人员分工

§3.2 附言

**§1.1 项目背景**

小组成员本学期进行了Web前端开发技术课程的学习，并且在制作期末作业期间进行了Three.js的自学。通过对部分同学群体的调研，发现有意向制作三维网页小游戏的小组较少，故本小组针对三维网页小游戏方向开发。本项目采取易上手与理解游戏模式的模拟赛车游戏形式。

**§1.2 参考资料**

1、Webpack THREE.js Template：https://github.com/brunosimon/webpack-three-js-template

2、车辆模型：<https://sketchfab.com/3d-models/fallout-car-2-cf54e5b166644fc7ade7bbaac502a04f>

**§1.3 版权声明**

本项目所有内容及参考资料均遵循MIT协议。

**§2.1 项目工作内容**

1. 利用由法国著名开发者Bruno Simons提供的Webpack-THREE.js开发模板搭建了NodeJS环境，解决同源策略引起的无法读取模型等问题，并且在本地服务器上实现实时刷新。
2. 使用基本的Web前端开发技术，采用轻量化开发策略，并对代码进行了一定的优化。
3. 使用Blender进行模型的制作，并且在赛道上添加了贴图，使得环境显得更为合理。
4. 碰撞反馈和按键事件皆为即时响应，实现了较好的用户操作体验。
5. 受Bruno Simons的个人首页网站启发，在网页标签页利用计时器实现了小车移动动画。

**§2.2 项目运行环境与限制**

测试阶段均采用Windows操作系统，使用的浏览器包括Chrome、Firefox等现阶段主流浏览器，皆可以正常运行。

运行环境要求用户下载全部文件后，安装NodeJS并启用npm服务，然后使用浏览器访问localhost:8080以开始体验。无法在未启用npm服务的电脑上使用。亦可以选择将所有文件部署于服务器，即可通过服务器域名访问。

**§2.3 项目功能**

1. 通过键盘上的方向键实现小车向前、后退、转弯操作；
2. 通过键盘F键按下事件响应实现第一第三人称视角切换；
3. 通过键盘R键按下事件响应实现快捷重置游戏；
4. 通过键盘Space键按下事件响应实现刹车；
5. 通过函数全自动监测，实现在与赛道模型碰撞时提示并自动重置游戏、实时速度显示与最大速度限制等功能；
6. 其他包含彩蛋在内的功能。

**§2.4 项目实现的不足与反思**

1. 项目未实现赛道模型的合理接缝与贴图的全景化，有较为明显的接缝痕迹；
2. 项目未实现基于CANNON.js的真实物理碰撞检测反馈；
3. 项目未实现按键事件并发的同时响应效果；
4. 项目未实现真实的物理阴影与反光效果；
5. 项目的可玩性、目标性较低。

**§3.1 人员分工**

陈诺言：实现小车模型的导入，碰撞、操纵事件的模拟，部署环境的配置，文档编写以及项目的总体管理；

戴梓莘：赛道模型与贴图的制作，场景的架设，部分交互代码的实现，文档编写及项目调试。

**§3.2 附言**

第一个真正意义上自己制作的小游戏，基于Web前端开发技术的创意实现，既是探索也是学习。