

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего
профессионального образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт математики и компьютерных наук

Кафедра прикладной математики

Система удаленной визуализации трехмерных объектов с использованием сред виртуальной реальности. Описание модели

Квалификационная работа
на соискание степени магистра наук
по направлению
02.04.01 Компьютерные науки
Магистерская программа «Математическая кибернетика»
студента гр. МКМ-240602
Левчука Георгия Ивановича

Задачи

- ◆ Реализация приложения с возможностью формирования трехмерной сцены;
- ◆ Подключение 3D моделей из готовых библиотек в сцену, с дальнейшей возможностью корректирования трехмерных объектов пользователям;

WebGL, Three.js

Для решения задач использовались технологии WebGL и Three.js. Выбор на этих библиотеках остановился в силу некоторых причин:

- ◆ отсутствует зависимость от ОС и присутствует кроссбраузерность, главное, чтобы ваш браузер поддерживал WebGL;
 - ◆ использование языка JavaScript, который достаточно распространен;
 - ◆ управление памятью – автоматическое, то есть не надо выполнять специальные действия для выделения и очистки памяти;
 - ◆ для рендеринга графики используется графический процессор на видеокарте, поэтому для этой технологии характерна высокая производительность.
- 

План действий

- ◆ реализация трехмерной сцены в web-приложение;
 - ◆ подключение 3D моделей в сцену;
 - ◆ формирование возможности изменения трехмерных моделей внутри сайта.
- 

Сцена в 3D-пространстве

Тег `<canvas>` - формирует сцену и определяет ее свойства.



Загрузка 3D моделей в сцену

`THREE.STLLoader()` – объект, с помощью которого происходит загрузка файлов формата `stl`, содержащих информацию для 3D моделирования



Меню

- ◆ `dat.GUI()` – объект, образующий простой функционал для изменения свойств объекта.





skullName	<input type="checkbox"/>
skullOpacity	<input type="checkbox"/>
SaveScene	
LoadScene	
scaleX	<input type="range" value="3"/>
scaleY	<input type="range" value="3"/>
scaleZ	<input type="range" value="3"/>
rotationX	<input type="range" value="0"/>
rotationY	<input type="range" value="0"/>
rotationZ	<input type="range" value="0"/>
Close Controls	

Вывод

- ◆ в результате работы были выполнены все поставленные цели;
- ◆ исследовались полезные, для работы с трехмерной графикой, свойства WebGL и Three.js
- ◆ увеличение скорости загрузки сложных трехмерных объектов можно увеличить с помощью редупликации полигонов

Спасибо за внимание!

