# Государственное бюджетное общеобразовательное учреждение г. Москвы

**«Школа № 2086»**

КОСМИЧЕСКИЙ СИМУЛЯТОР

Автор: Ученик 10 “А” класса ГБОУ «Школа № 2086», Лещук Глеб Олегович

Научный руководитель: преподаватель ГБОУ «Школа № 2086», Ярцев Александр Владимирович

# Москва, 2023

**Оглавление**

1. [Введение](#_heading=h.gjdgxs)
2. [Цели и Задачи](#_heading=h.30j0zll)
3. [Гипотеза](#_heading=h.1fob9te)
4. [Методика выполнения работы](#_heading=h.3znysh7)
5. [Результаты](#_heading=h.2et92p0)
6. [Описание продукта](#_heading=h.tyjcwt)
7. [Ссылки](#_heading=h.3dy6vkm)
8. [Список литературы](#_heading=h.1t3h5sf)

# Введение

С развитием космической отрасли человек начинает все чаще обращать свой взгляд к звездам. Простому обывателю уже не чужды знания о космическом пространстве, планетах и устройстве вселенной. Активно развиваются научно- популярная литература, интернет-ресурсы, посвященные космосу.

Однако, далеко не у каждого есть возможность самостоятельно постигать основы астрономии и космической механики. Многие источники требуют довольно высокого порога вхождения для изучения их содержимого. Этот тезис стал основой для моих дальнейших рассуждений в этом направлении.

На уроках астрономии в школе я заинтересовался основами взаимодействия космических тел. До этого я уже ознакамливался с программами, позволяющими моделировать движение планет, однако почти все из них требовали установки и знания ПО. С этими убеждениями я принял решение о создании собственного сайта-симулятора космического пространства с простым и удобным интерфейсом.

Объект разработки - браузерный симулятор космоса, с возможностью симуляции взаимодействия физических тел в межзвездном пространстве. В расчет при создании программы будут браться все существующие законы физики, описывающие поведения объектов в космосе. Перспективой разработки такого рода проекта может стать использование в образовании и воссоздании реальных планет в 3D анимациях.

Я рассчитываю на простоту в эксплуатации и эргономичность проектируемого софта, а также на стабильное соотношение амбициозности и собственных возможностей в создании этого проекта.

# Цели и Задачи

**Цель проекта:**

Разработать востребованное в образовательных целях приложение для демонстрации космического движения тел.

# Задачи проекта:

1. Разработка концепции проекта и получение необходимых навыков для создания полноценного сайта с симуляцией. Определение целевых групп пользователей приложения.
2. Изучение принципов работы законов физики в пространстве. Разработка прототипов с помощью ООП, доступного на языке Javascript. Интеграция их в 3D пространство с помощью ThreeJS.
3. Тестирование версий программы. Улучшение графики, пользовательского интерфейса, добавление возможности обучения с помощью программы.
4. Загрузка приложения на сервер. Публичное тестирование и дальнейшее развитие проекта.

# Гипотеза

Разработка браузерного космического симулятора позволит решить проблему отсутствия интерактивности при усвоении учениками теоретического

материла и каждому без затраты больших объемов времени самостоятельно наблюдать и ставить опыты по взаимодействию небесных тел.

# Методика выполнения работы Список использованных ресурсов (Таблица 1):

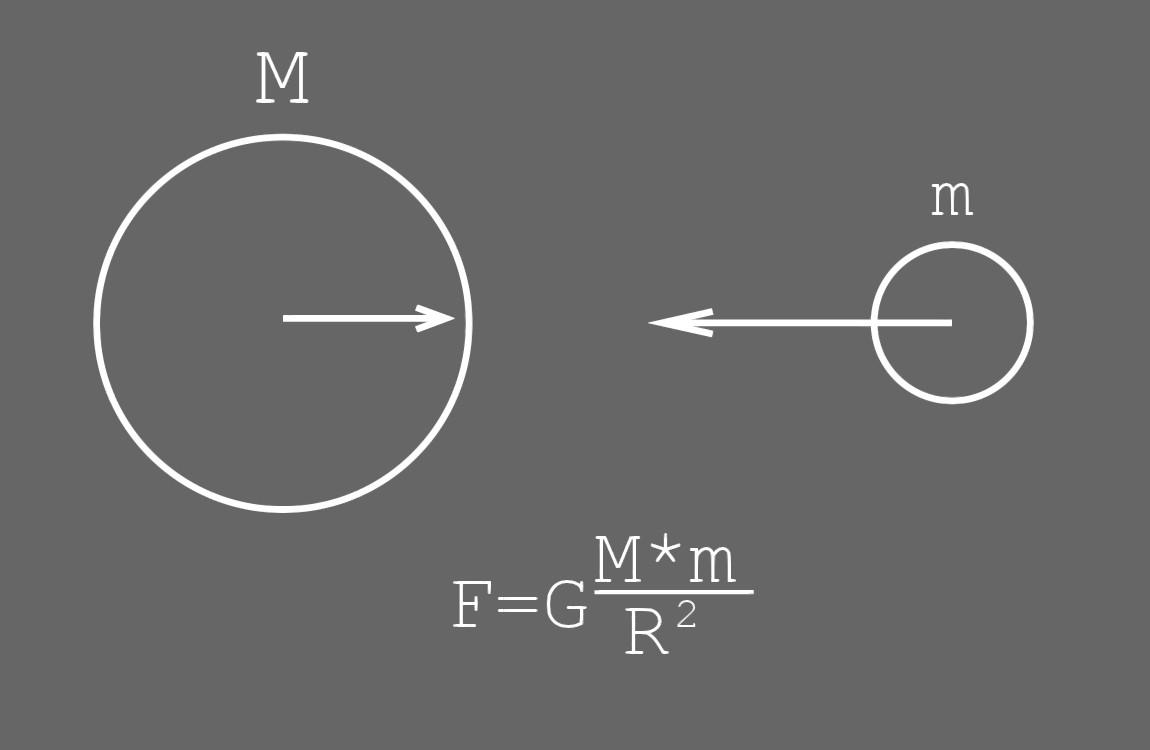
*Таблица 1. Материалы и ресурсы*

| **Название** | **Назначение** |
| --- | --- |
| Среда разработки Visual Studio  Code | Программирование и разработка  дизайна интерфейса |
| Браузер Google Chrome | Тестирование и отладка  прототипов |
| Компьютер на os Windows | Разработка программы |

# Описание алгоритма моделирование взаимодействия объектов в пространстве:

Все силы, действующие на тела в межзвездном пространстве, могут быть описаны с помощью закона всемирного тяготения. Зная массу и расстояние планет друг от друга, можно рассчитать попарно действующие на них силы притяжения. (Рисунок 1)

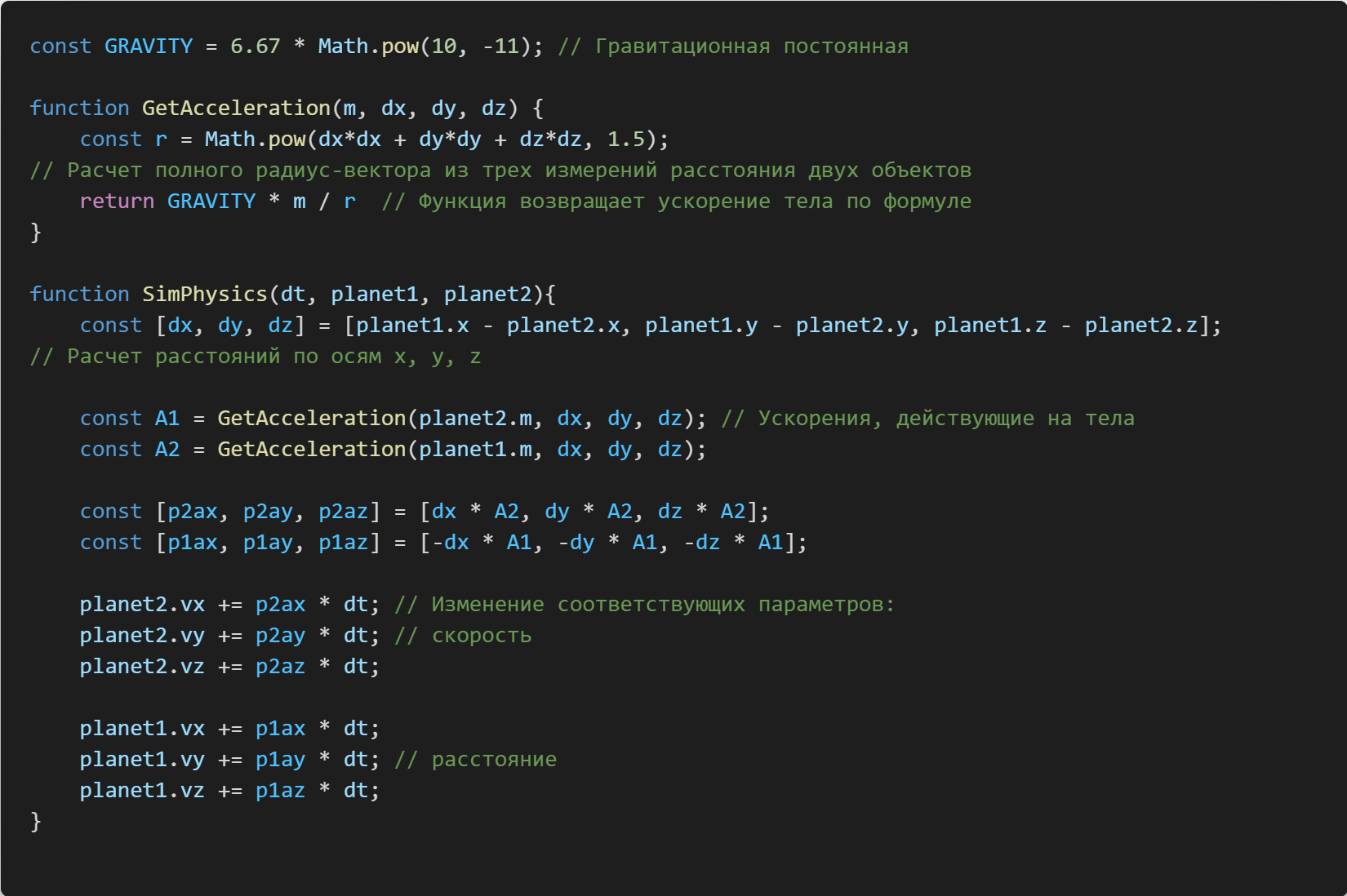
*Рисунок 1. Закон всемирного тяготения*



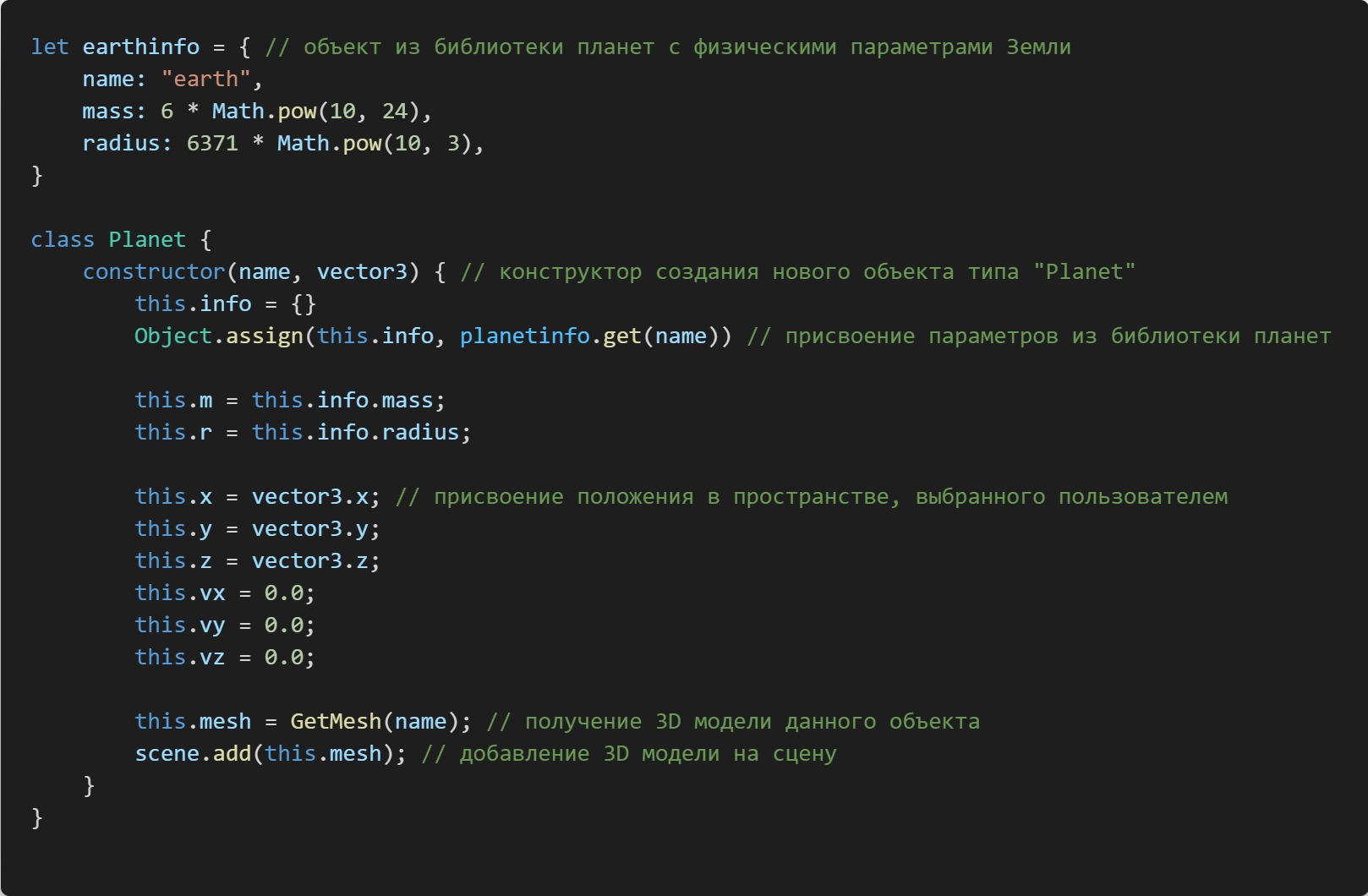
Таким образом, с помощью языка программирования Java Script, можно воссоздать функцию для расчета текущего положения тела в пространстве в зависимости от момента времени и действующих сил. (Рисунок 2)

Однако, для расчета движения n-го количества тел, что подразумевает формат взаимодействия пользователя с приложением, требуется использовать более сложные структуры. Благодаря возможностям ООП (Объектно- ориентированного программирования) языка Java Script, я реализовал базовый класс космического тела. Именно к нему будут обращаться все остальные компоненты и функции в программе для получения свойств, изменения параметров и рендеринга графики. (Рисунок 3)

*Рисунок 2. Реализация расчетов на JS*



*Рисунок 3. Реализация структуры с данными*



# Рендеринг 3D графики в браузере:

В целях обработки 3D в браузерных приложениях используется интерфейс WebGl, который с помощью элемента canvas обеспечивает API HTML и графики.

Для взаимодействия с WebGl мною была выбрана библиотека Three.js. Она позволяет моделировать ускоренное видеокартой изображение, без подключения сторонних плагинов. Код для Three.js можно реализовать на JavaScript, импортируя модуль библиотеки. Отдельно в HTML-документ добавляется скрипт, содержащий Three.js объекты, отображаемые в элементе canvas.

Ниже приведен пример базовой 3D сцены на Three.js. (Рисунок 4)

*Рисунок 4. Базовая сцена (куб)*



В основном цикле обработки событий Three.js, я реализовал функции добавления объекта и изменения его положения, согласуясь с приведенными выше расчетами. Рендерер Three.js фиксирует эти изменения и выводит результат на страницу браузера. (Рисунок 5)

*Рисунок 5. Объекты в 3D пространстве*



# 

# 

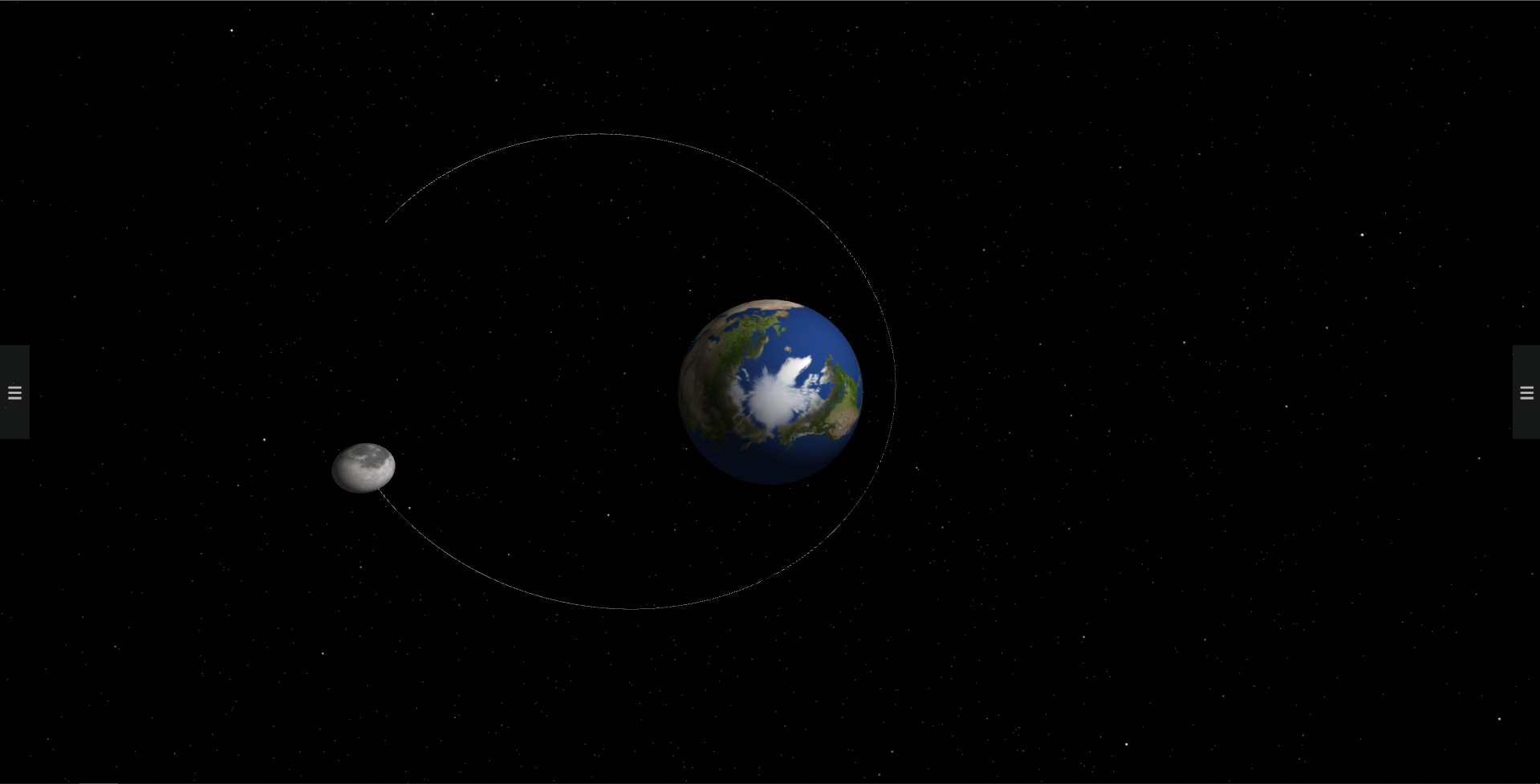
# Визуальная составляющая и юзер-интерфейс симулятора:

Для того, чтобы пользователь мог взаимодействовать со сценой, необходимо добавить меню, которые позволили бы это сделать. С помощью HTML, я расположил два контейнера с элементами меню. (Рисунок 6) Первое дает возможность добавлять выбранные объекты на сцену, а второе - менять их параметры и изучать теоретическую информацию об объекте или явлении. (Рисунок 7) В дальнейшем раздел информации планируется доработать.

*Рисунок 6. Земля. Информация об объекте*

****

*Рисунок 7. Движение тела по эллиптической орбите*

****

# Результаты

Проведен анализ существующих решений. Создана локальная бета модель веб-приложения, дизайн, пользовательский интерфейс. Продумана детальная карта дальнейшего развития проекта.

В дальнейшем планируется расширение функционала приложения и добавление теоретической информации и “пресетов” - моделей реальных звездных систем для изучения.

# Описание продукта

Создана симуляция космического пространства для изучения астрономии с доступом из браузера и удобным пользовательским интерфейсом.

Разработанный проект может стать уникальной средой для визуализации космических процессов, понимание которых трудно выстроить лишь на почве теоретической базы.

Перспективой разработки такого рода проекта может стать использование в школьном образовании и самообучении.

# Ссылки

Код проекта: <https://github.com/leshless/Project/tree/main/Test>

Демонстрация работы:

<https://drive.google.com/file/d/1cEd-1XERzr-yqjgnq7TFzlmwgzZTOg87/view?usp=sharing>

# Список литературы

1. Учебник по JavaScript – URL: <https://learn.javascript.ru/>
2. Документация по Three.js – URL: <https://threejs.org/>
3. Рой, А. Движение по орбитам / А. Рой. — Москва : Мир, 1981. — 544 c.
4. Левантовский, В. И. Механика космического полёта в элементарном изложении / В. И. Левантовский. — Москва : Наука, 1980. — 512 c.