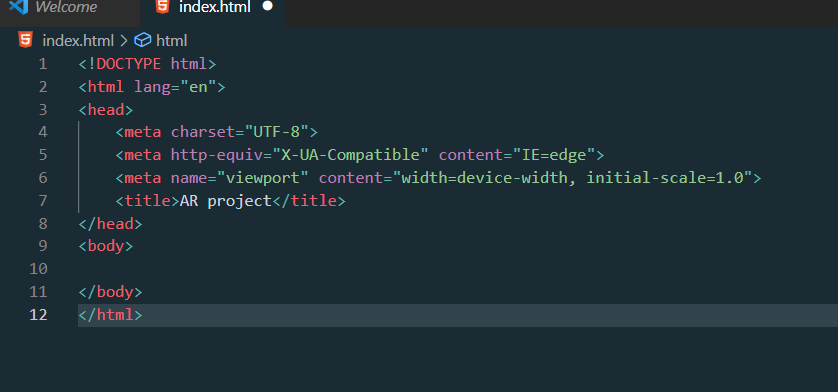
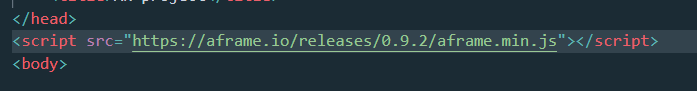
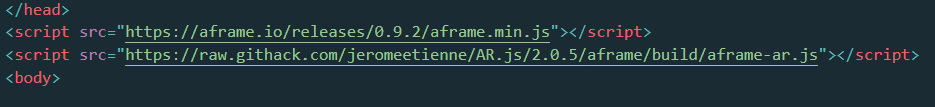
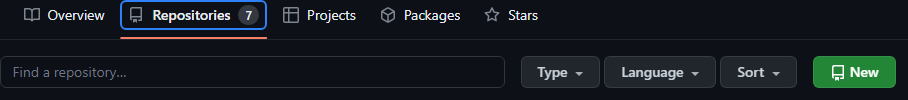
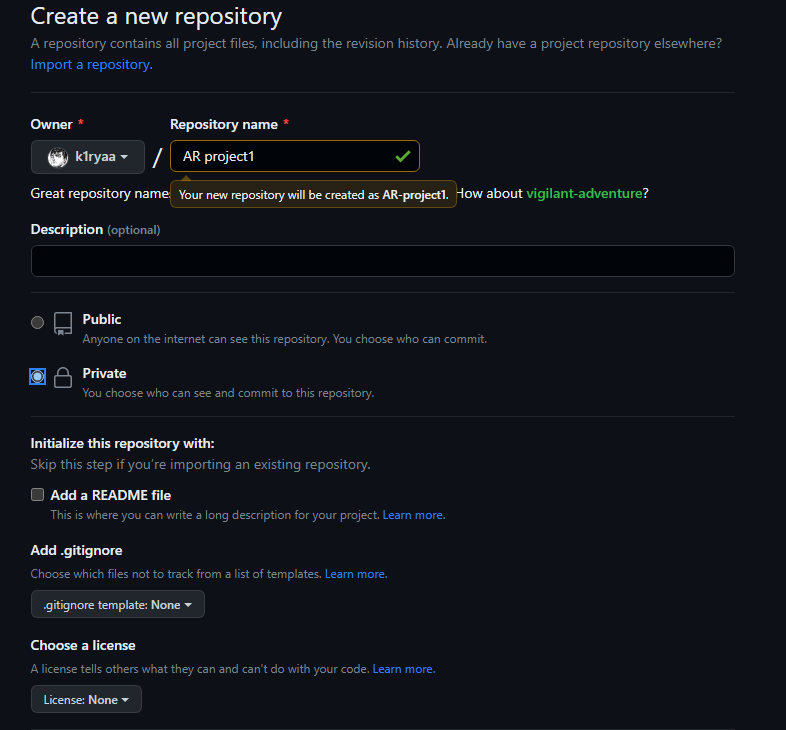
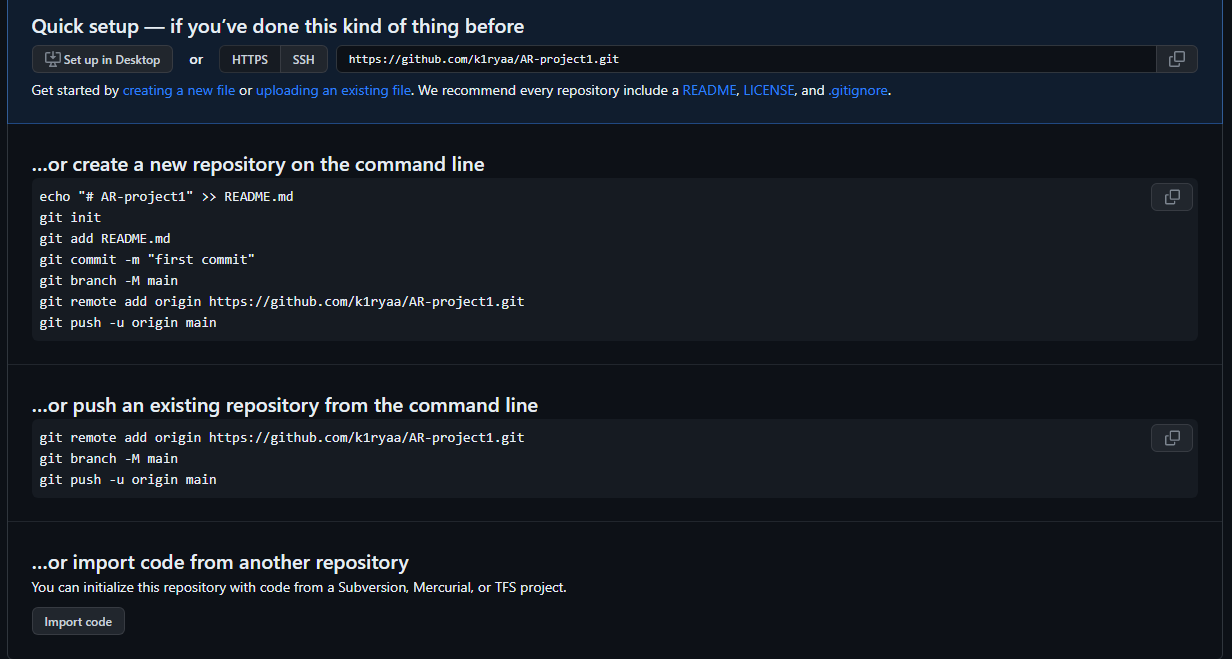
1. Создаём файл index.html

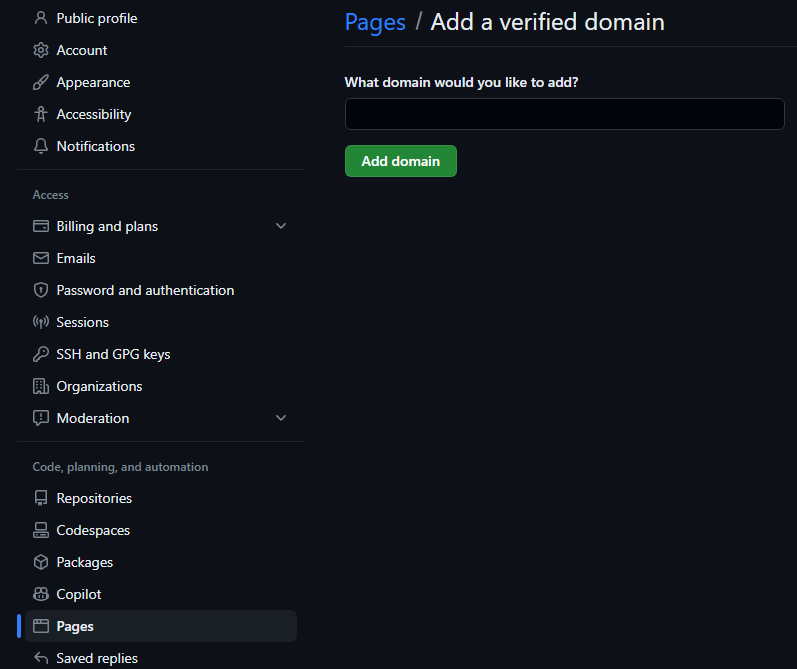
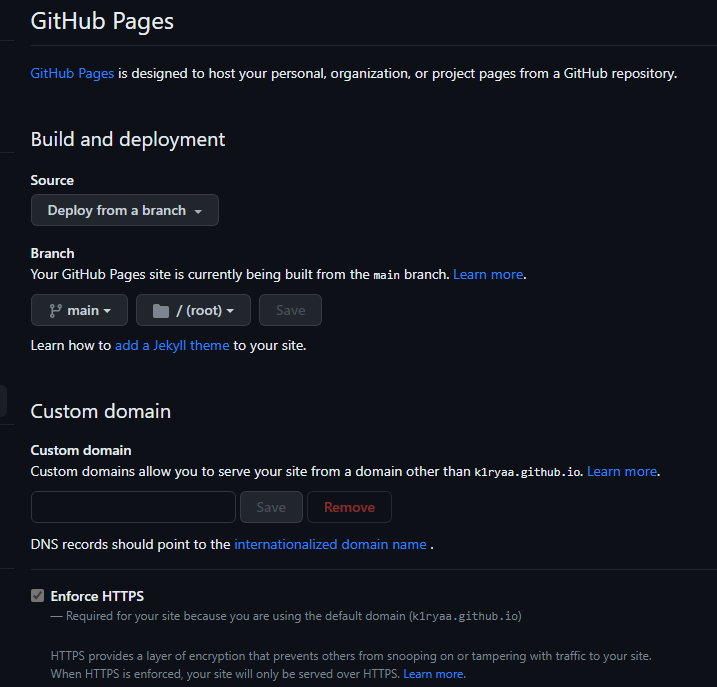
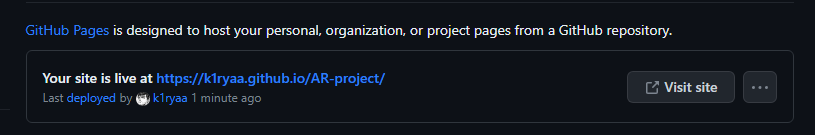
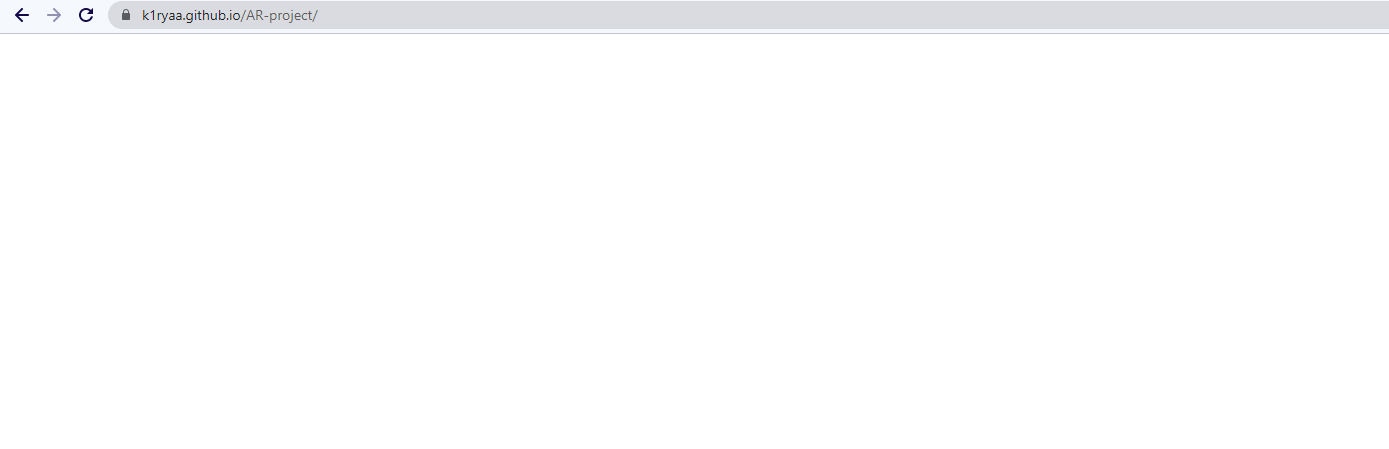


1. Прописываем основную структуру (Метаданные, название проекта)
2. В теге html подключим веб-фреймворк A-Frame
3. Там же подключим основной фреймворк AR.js, с помощью которого VR и будет работать
4. Нам понадобится веб-сервер, чтобы был доступ к камере устройства. Воспользуемся для этого GitHub Pages. Загрузим наш проект на GitHub, во вкладке Repositories нажимая на кнопку New
5. Устанавливаем следующие настройки проекта, нажимаем Create repository
6. Воспользуемся вторым способом для загрузки готового проекта в репозиторий
7. Вставляем следующий код в терминал

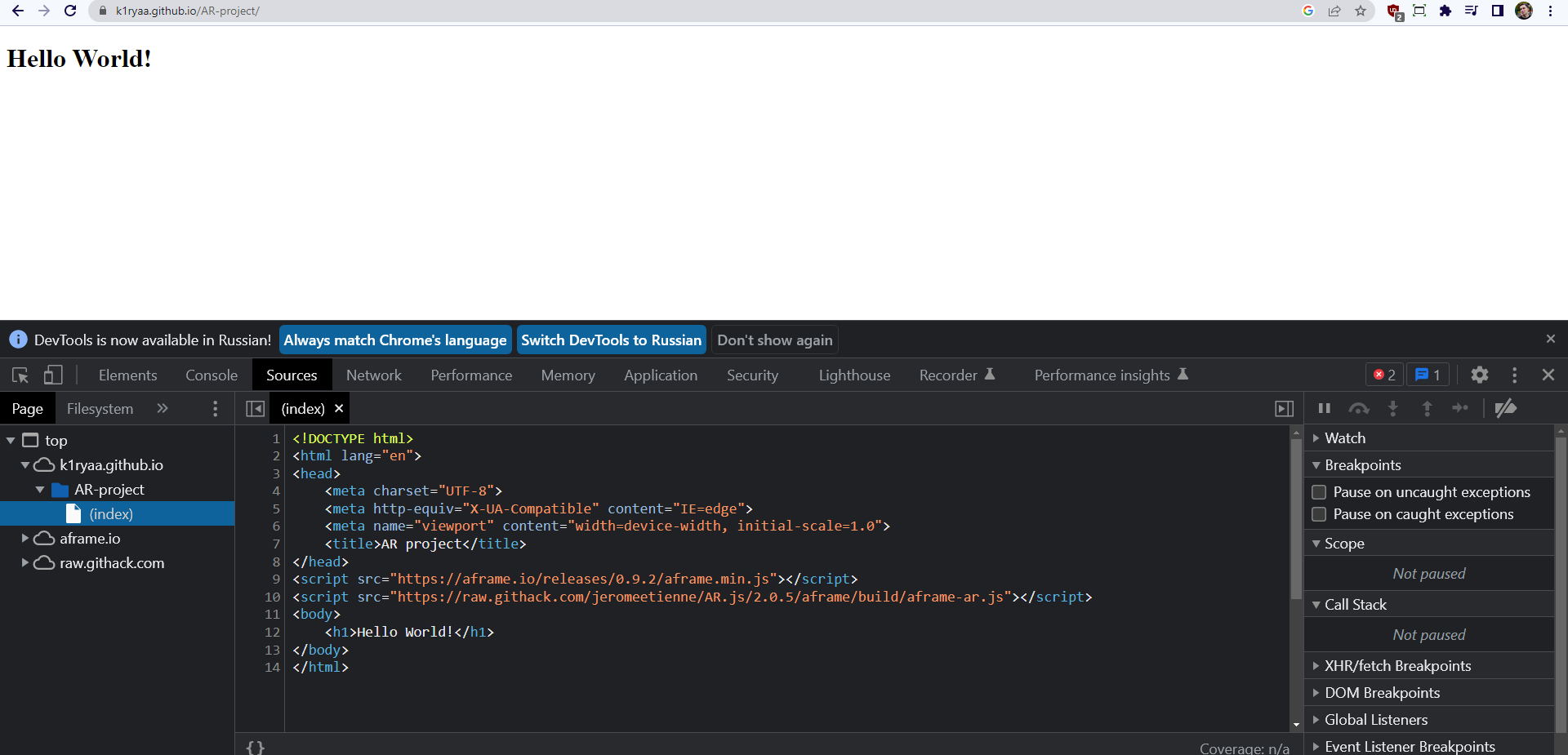
git remote add origin https://github.com/k1ryaa/AR-project1.git

git branch -M main

git push -u origin main

1. Заходим в настройки репозитория и находи вкладку pages  
   
2. Устанавливаем данные настройки, сделав перед этим репозиторий публичным
3. Через некоторое время появится ссылка на наш рабочий сайт  
4. Для проверки добавим заголовок в файл index.html, сделав коммит и запушив его в репозиторий

<h1>Hello World!</h1>



Теперь протестируем возможности ar-фреймворка  
Вставим следующий код на место тега body:

<body style='margin : 0px; overflow: hidden;'>

  <a-scene embedded arjs>

    <a-box position='0 0.5 0' material='opacity: 0.75;'></a-box>

    <a-marker-camera preset='hiro'></a-marker-camera>

  </a-scene>

</body>

Разберём построчно

<body style='margin : 0px; overflow: hidden;'>

В стилях тега body мы задаём margin: 0, чтобы не было внешних отступов и изображение растягивалось на весь экран, overflow: hidden не позволит пользователю скроллить страницу, так как в данном случае это не нужно.

  <a-scene embedded arjs>

Тег a-scene задаёт пространство для загрузки моделей, атрибут embedded удаляет полноэкранные стили в области, ограниченной тегом a-scene, тег arjs включает дополненую реальность в области, ограниченной тегом a-scene.

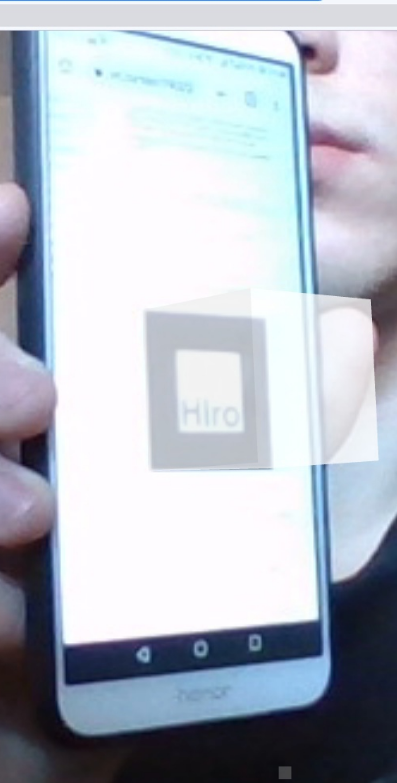
    <a-box position='0 0.5 0' material='opacity: 0.75;'></a-box>

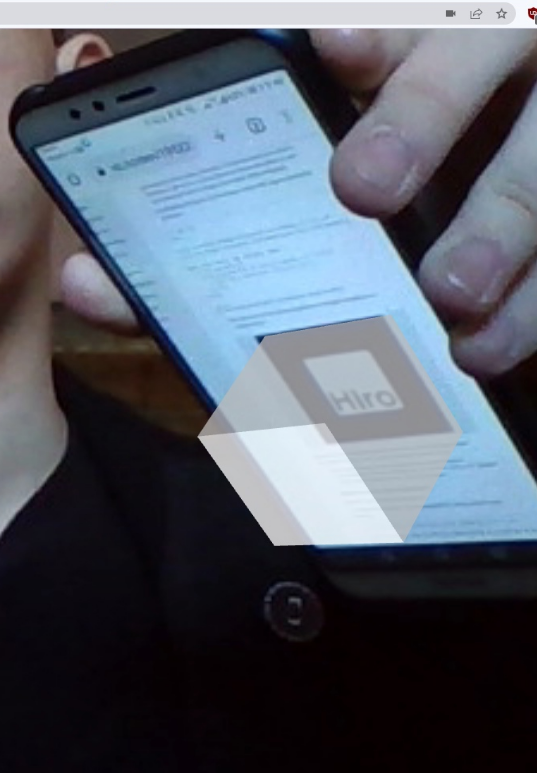
Тег a-box позволяет нам создавать примитивы A-Frame фреймворка, атрибут position задаёт положение объекта относительно маркера (x, y, z соответственно), в атрибуте material задаётся материал объекта, в данном случае только прозрачность.

    <a-marker-camera preset='hiro'></a-marker-camera>

Тег a-marker-camera устанавливает маркер, относительно которого объект будет проецироваться в дополненной реальности. Атрибут preset, равный “hiro”, означает, что мы будем использовать стандартный маркер, предусмотренный фреймворком. 

Как это выглядит: 

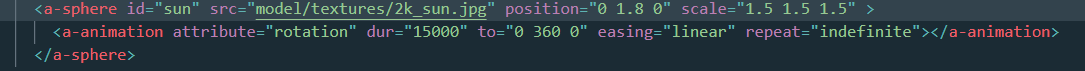




В данном варианте, маркер движется относительно камеры, и объект – за ним, с мобильных устройств работает аналогично, только там движется камера относительно маркера.

1. Приступим к созданию моделей. Так как мы делаем солнечную систему, то мы можем использовать примитивы в виде сфер

Рассмотри код для создания Солнца:

Тег a-sphere создаёт примитив сферы. Рассмотрим атрибуты:

* id – селектор для обозначения названия элемента в DOM структуре
* src – указывает путь материала примитива
* position – задаёт положение примитива в пространстве относительно маркера
* scale – задаёт размеры примитива (Размеры солнца не в том же соотношении с планетами, что и в реальности, так как оно бы просто не уместилось)

Тег a-animation предназначен для создания анимации. Мы будем создавать анимацию вращения солнца вокруг своей оси. Рассмотрим атрибуты:

* attribute – указывает, что нужно анимировать
* dur – время выполнения анимации
* to – указывает значения, к которым нужно стремиться
* easing – функция плавности выполнения анимации
* repeat – повторение анимации