

# Initialize Firebase Storage

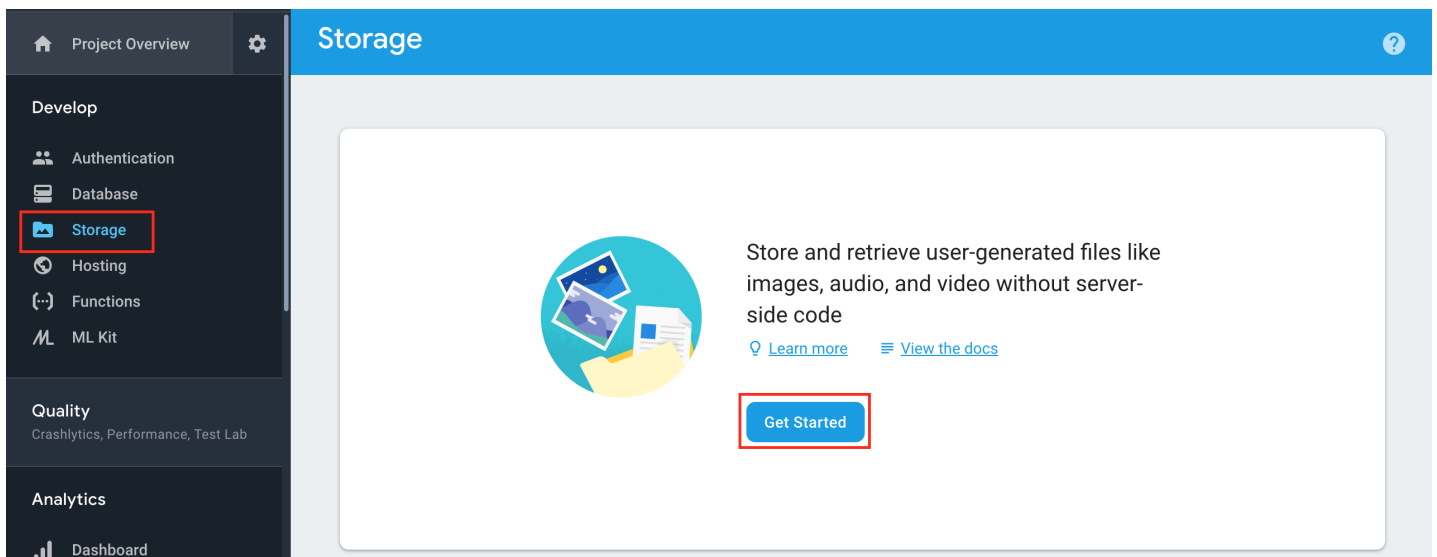
In this lesson, you learn how to turn Firebase Storage on from the console and configure security rules so that we can upload images in the next lesson.

## We'll cover the following

- Enable Storage From the Firebase Console
  - Agree to the Default Security Rules
  - Agree to the Storage Location
- You are Now Ready to Use Firebase Storage
  - Change the Rules
- Required Firebase Script Includes
- The Photo Sharing Application Starting Point.

## Enable Storage From the Firebase Console #

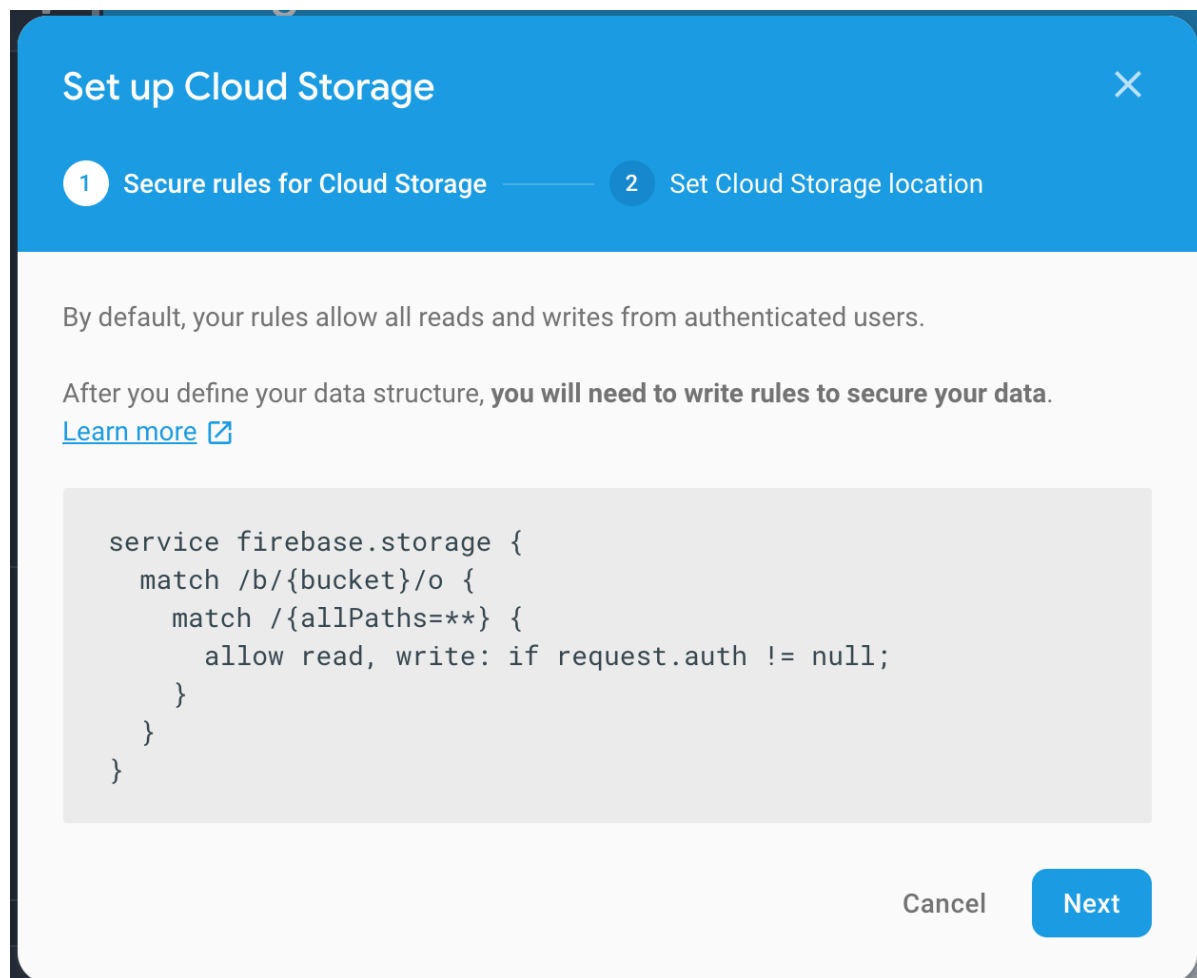
From the Firebase console, select your project. You will see a main sidebar on the left. Click on **Storage**. On the right, click on **Get Started**.



## Agree to the Default Security Rules #

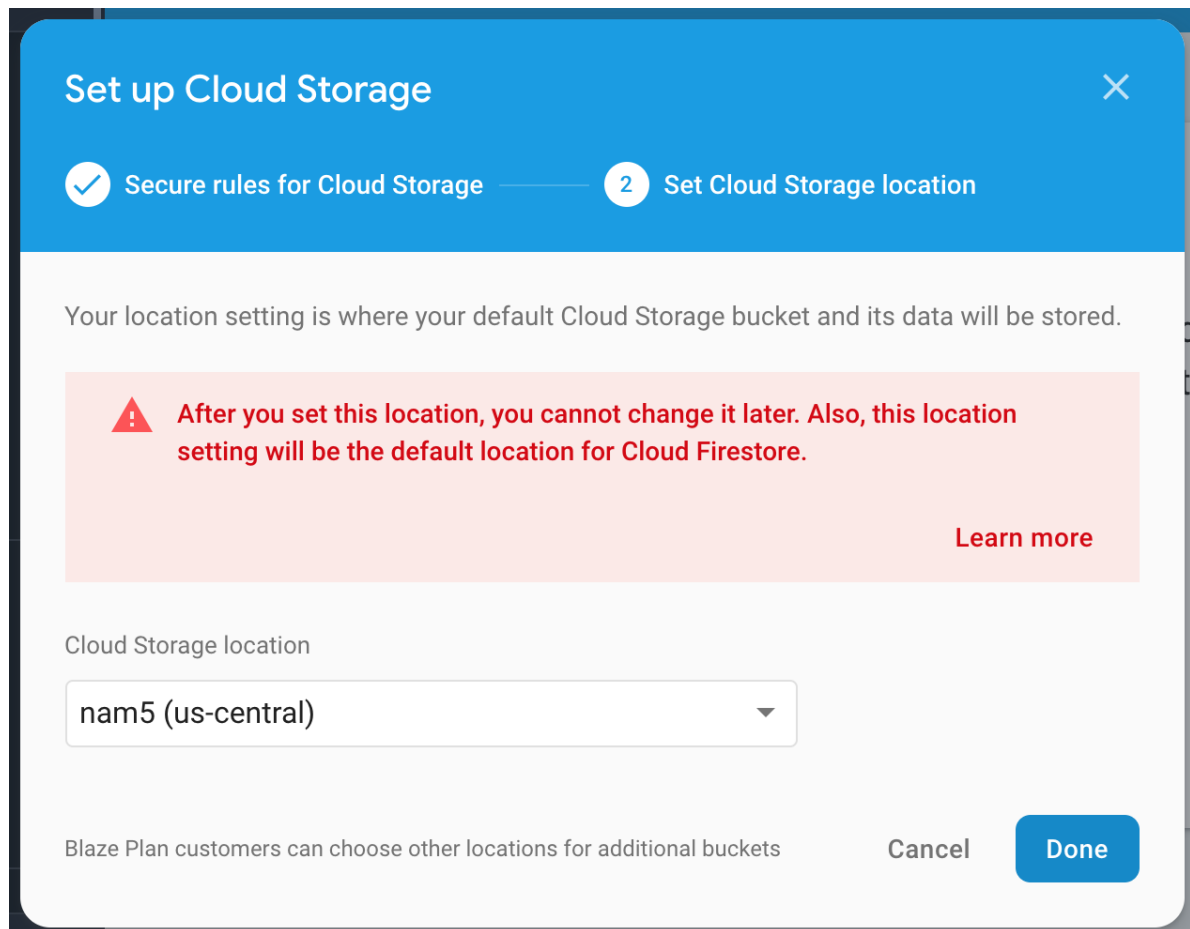
These rules state that in order to upload files you have to be authenticated. You can agree to these now by clicking on **Next** but we will change these in a moment

because you haven't learned how to authenticate a user yet. We will cover that later in the course.



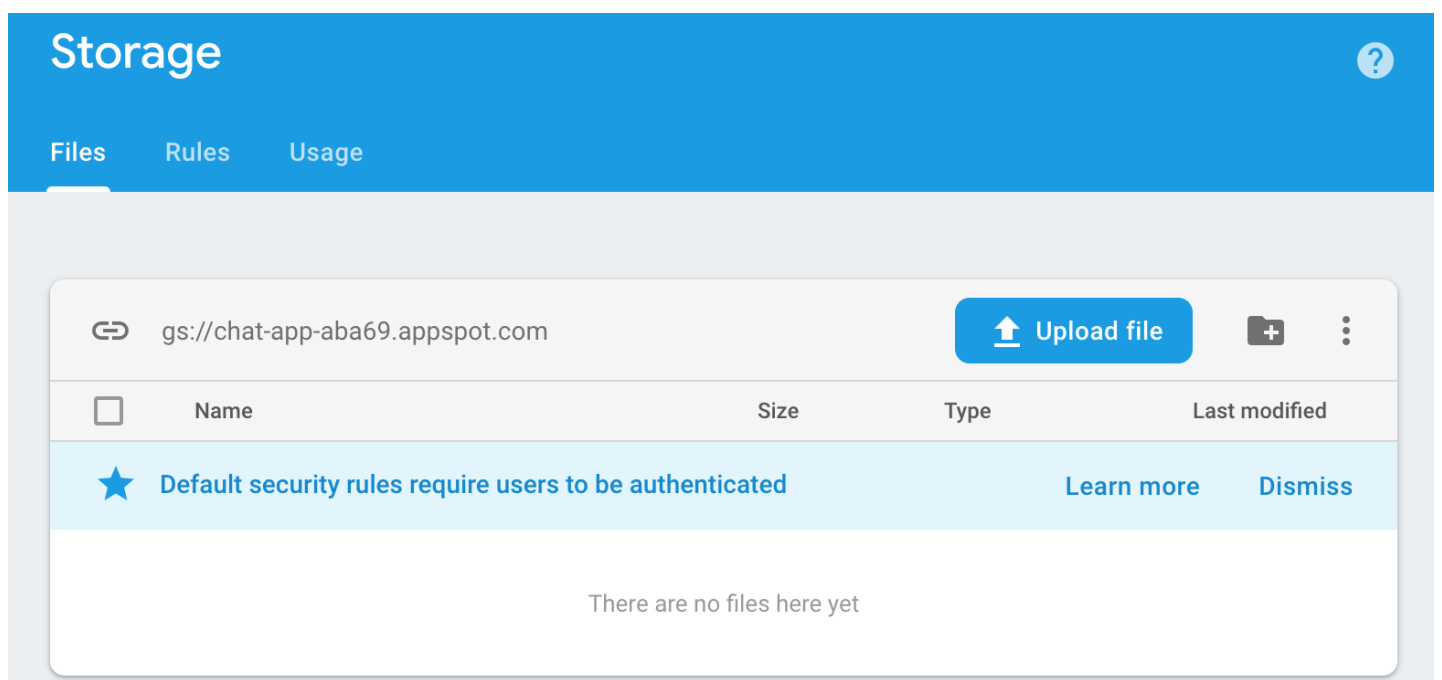
## Agree to the Storage Location #

Firebase will suggest a location that is near you. Using their suggestion is generally the best way to go.



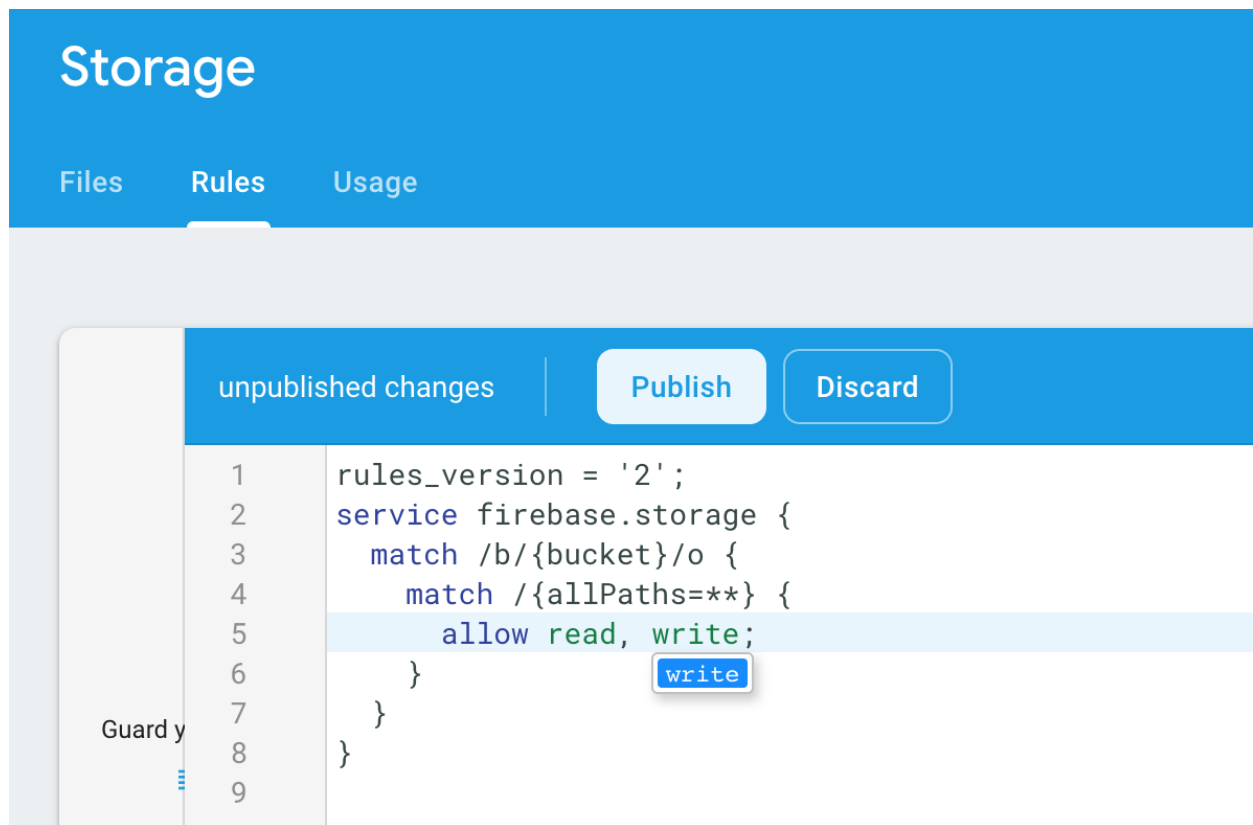
## You are Now Ready to Use Firebase Storage #

At this point, you can use the dashboard to upload files. In the next lesson, you will learn how to allow users to upload with your application.



## Change the Rules #

Navigate to the **Rules** tab and change the rules to `allow read, write;`. This allows us to use storage in a less restricted way. It's a good starting point, but it is not very secure. Finally, click on **Publish** to save your changes.



## Required Firebase Script Includes #

We add the script for Firestore and Storage in the head of our index.html file. We are going to use Firestore in the next lesson to make references to our uploads in Cloud storage.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <script src="https://www.gstatic.com/firebasejs/6.3.0/firebase-app.js"></script>
  <script src="https://www.gstatic.com/firebasejs/6.3.0/firebase-firestore.js"></script>
  <script src="https://www.gstatic.com/firebasejs/6.3.0/firebase-storage.js"></script>
</head>
<body>

</body>
</html>
```

## The Photo Sharing Application Starting Point. #

Note: When running the playground below don't expect to see anything yet! It's only ready to start working with storage. You will see visual feedback in the upcoming lessons.

This code requires the following API keys to execute: ^

apiKey	Not Specified...
authDomain	Not Specified...
databaseURL	Not Specified...
projectId	Not Specified...
storageBucket	Not Specified...
messagingSenderId	Not Specified...
appId	Not Specified...

Output

JavaScript

HTML

CSS (SCSS)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
  <script src="https://www.gstatic.com/firebasejs/6.3.0/firebase-app.js"></script>
  <script src="https://www.gstatic.com/firebasejs/6.3.0/firebase-firestore.js"></script>
  <script src="https://www.gstatic.com/firebasejs/6.3.0/firebase-storage.js"></script>
</head>
<body>

</body>
</html>
```



Now we are ready to start building the app! Join me in the next lesson to learn how to upload and store images.

