

Introducción a Firebase STORAGE



DOCUMENTACIÓN

<https://firebase.google.com/docs/storage>

Cloud Storage para Firebase es un servicio de almacenamiento de objetos potente, simple y rentable construido para la escala de Google.

Operaciones robustas

Las cargas y descargas son robustas (sin importar la calidad de la red), lo que significa que se reinician en el punto en el que se interrumpieron para así ahorrar tiempo y ancho de banda a los usuarios.

Seguridad sólida

Los SDK de Firebase para Cloud Storage se integran con Firebase Authentication a fin de brindar autenticación intuitiva y sencilla para los programadores.

Gran escalabilidad

Cloud Storage para Firebase está diseñado para escalar a exabytes si tu app se vuelve viral. Pasa fácilmente de la fase prototipo a la de producción con la misma infraestructura que respalda a Spotify y Google Fotos.

En Firebase Console

Storage

ARCHIVOS REGLAS

gs://testy-mcfirebaseface.appspot.com

SUBIR ARCHIVO

<input type="checkbox"/>	Nombre	Tamaño	Tipo	Última modificación
<input type="checkbox"/>	test.png	6.74 MB	application/octet-str...	4 dic. 2017
<input type="checkbox"/>	test2.jpeg	816.03 KB	application/octet-str...	4 dic. 2017
<input type="checkbox"/>	worldcup/	—	Carpeta	—

INSTALACIÓN

```
pod 'Firebase/Storage'
```



Firestore - Lectura de Storage



LEER UNA FOTO

downloadUrl:completion: es uno de los métodos que permite descargar imagenes del Storage. En este caso, dado un nodo, se obtiene la URL de descarga.

Alternativamente, se puede descargar el archivo al disco directamente o descargarlo y mantenerlo directamente en memoria, listo para ser usado.

```
let storage = Storage.storage()
let storageRef = storage.reference()
let worldCupRef = storageRef.child("worldcup.png")

worldCupRef.downloadURL(completion: { (url, error) in
    if let url = url {
        self.imageView.kf.setImage(with: url)
    }
})
```

Firestore - Escritura en Storage



GUARDAR UNA FOTO

Una de las formas de guardar un archivo es Storage es mediante la clase **Data**. Cualquier tipo de recurso (imagen, video, etc) se puede entender como Data Binaria y esa Data es la que se almacena.

Alternativamente se puede almacenar en el Storage recursos desde la memoria o desde un archivo local.

```
let storage = Storage.storage()
let storageRef = storage.reference()

let testRef = storageRef.child("myImage.png")

testRef.putData(data, metadata: nil) { (metadata, error) in
    print("success")
}
```