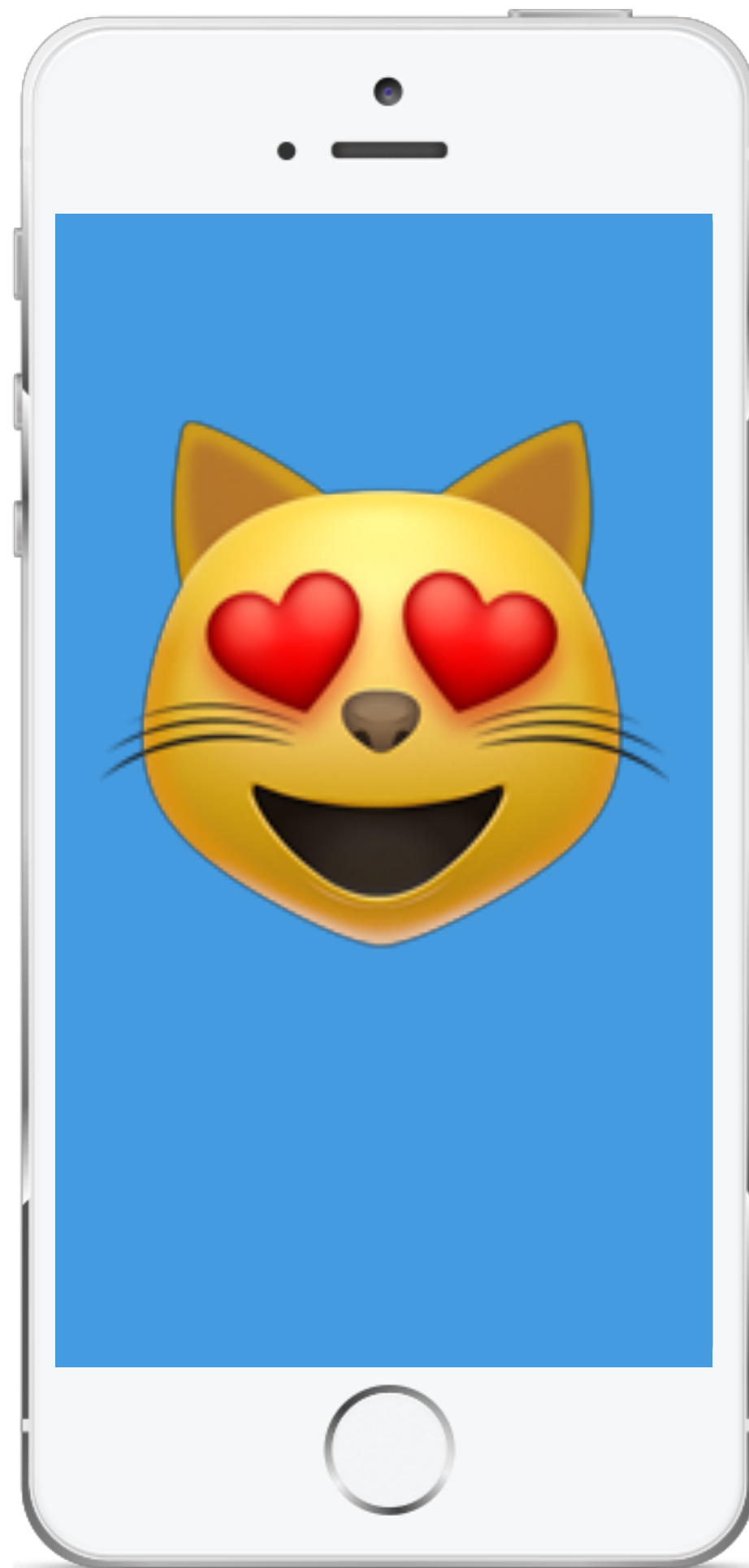


What I learned creating a social media app clone



Jen Person
@ThatJenPerson





CatCalling

An app for enjoying cats

Super simple: let's just share some cat photos!

But is it so simple?



CatCalling

Store cats names and descriptions

Store cat photos

Verify users

Verify cat photos

Download photos



Develop



Realtime Database



Cloud Firestore



Authentication



Cloud Functions



Cloud Storage



Hosting



Test Lab



Performance



Crashlytics



Crash Reporting

Grow



Analytics



Cloud Messaging



Dynamic Links



Remote Config



Invites



App Indexing



AdMob



AdWords



A/B Testing



Predictions



Cloud Firestore
Firebase Authentication
Google Cloud Storage for Firebase
Imgix
ML Kit for Firebase



Cloud Firestore

Firebase Authentication

Google Cloud Storage for Firebase

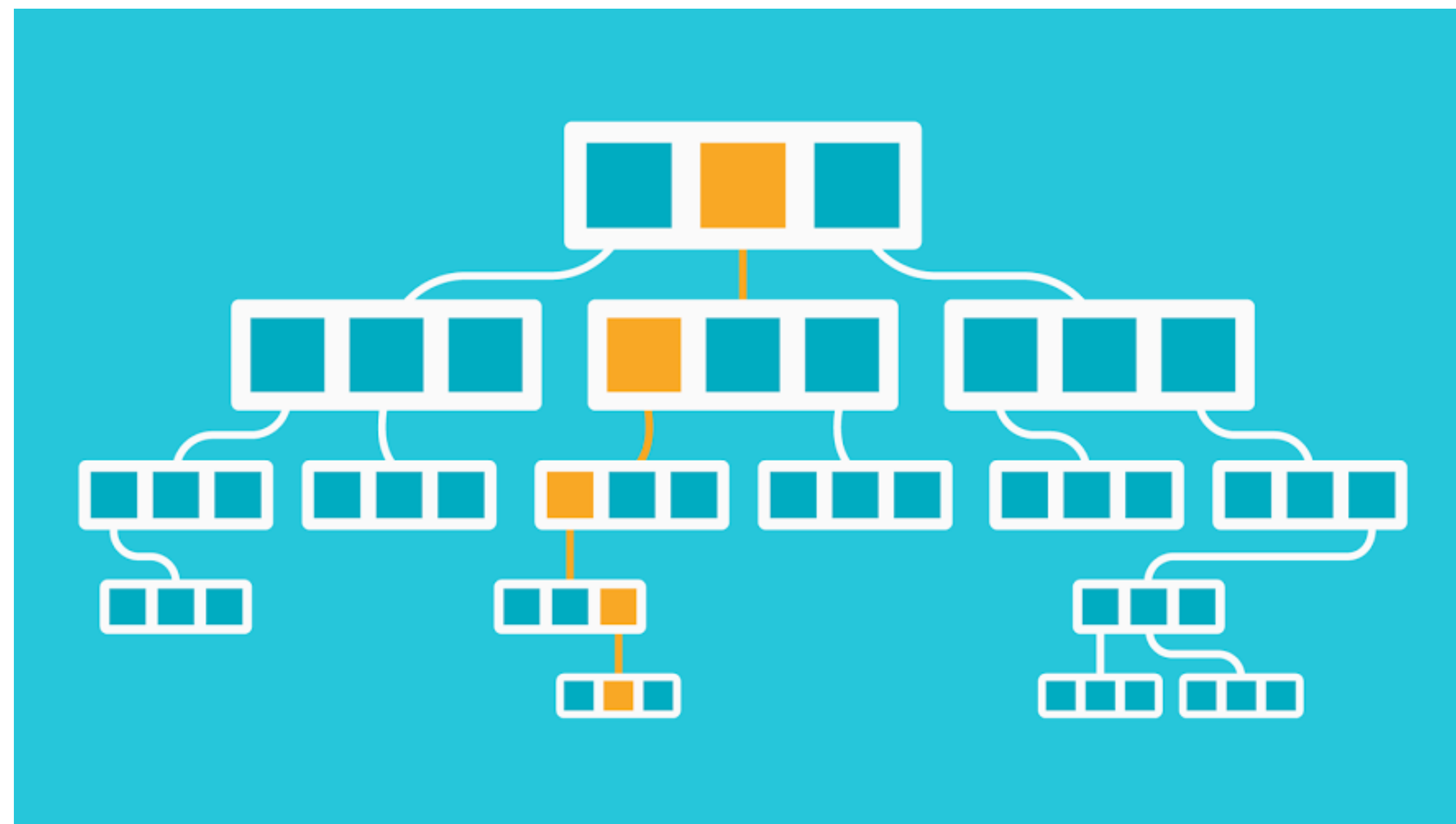
Imgix

ML Kit for Firebase



Cloud Firestore

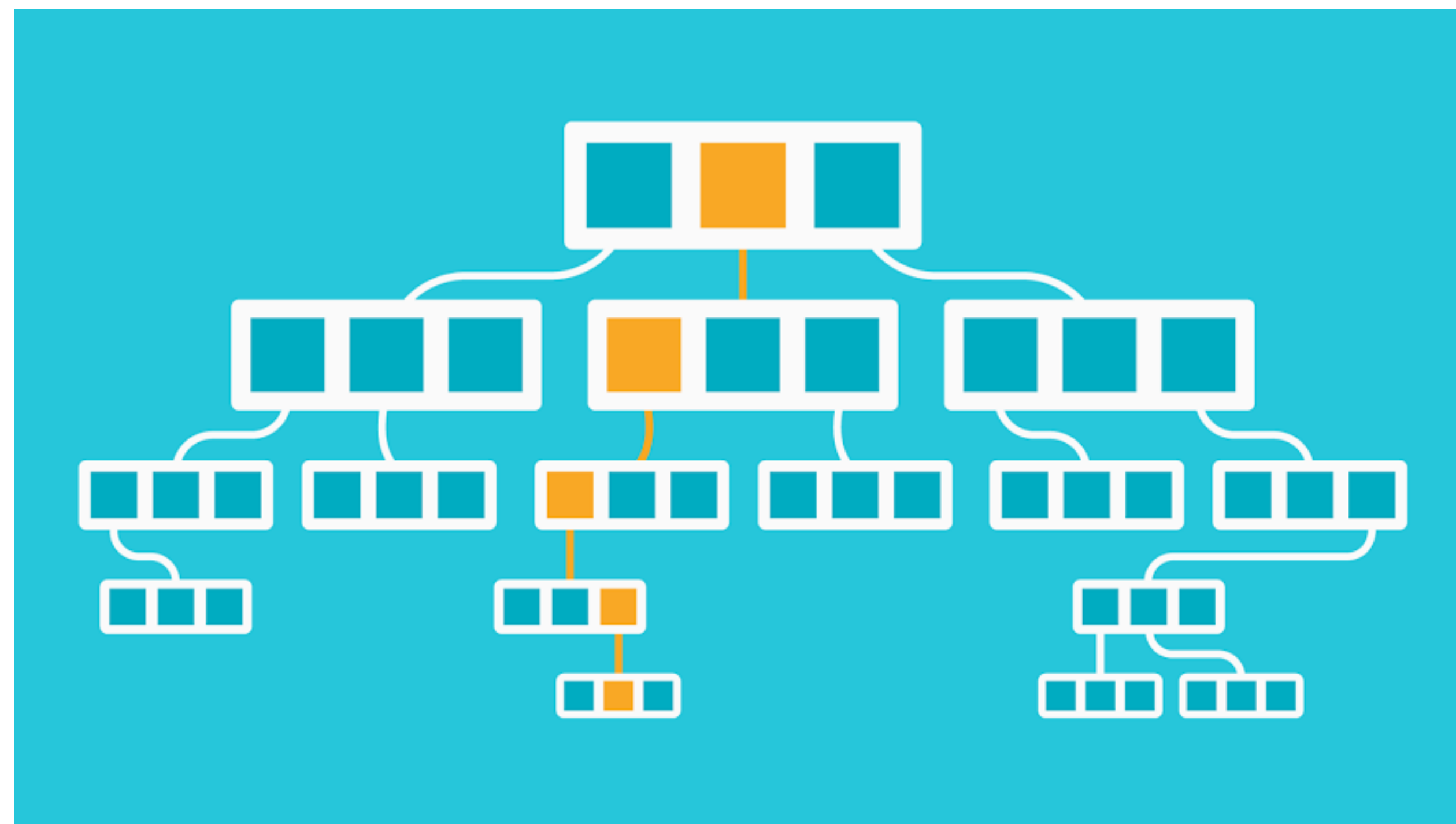
- NoSQL database
- Organized by collections with documents
- Expressive querying (multiple where clauses!)
- Effortless scaling





Cloud Firestore

- Text String
- Date and Time
- Integer
- Boolean
- Bytes
- Map
- Array
- Subcollections



Cats

```
4btcA7LEvcMb4A5fF910: {  
  ...  
}
```

```
5CBZc7fxUrsRw0nSId3g : {  
  ...  
}
```

```
5Z889yXWWXF1wdo812Kr: {  
  ...  
}
```

Users

```
KtX8NP0tNi68vzJckcDUjp1: {  
  ...  
}
```

```
5cg7sSVwfJYyT7LemgUtgEn: {  
  ...  
}
```

```
keJk1ojG2qQKxEtweDd1Uj2: {  
  ...  
}
```

Collections

Cats

```
4btcA7LEvcMb4A5fF910: {  
  ...  
}
```

```
5CBZc7fxUrsRw0nSId3g : {  
  ...  
}
```

```
5Z889yXWWXF1wdo8l2Kr: {  
  ...  
}
```

Users

```
KtX8NP0tNi68vzJCkcDUjp1: {  
  ...  
}
```

```
5cg7sSVwfJYyT7LemgUtgEn: {  
  ...  
}
```

```
keJk1ojG2qQKxEtweDd1Uj2: {  
  ...  
}
```

Collections

Documents

Cats

```
4btcA7LEvcMb4A5fF910: {  
  ...  
}
```

```
5CBZc7fxUrsRw0nSId3g : {  
  ...  
}
```

```
5Z889yXWWXF1wdo8l2Kr: {  
  ...  
}
```

Users

```
KtX8NP0tNi68vzJCkcDUjp1: {  
  ...  
}
```

```
5cg7sSVwfJYyT7LemgUtgEn: {  
  ...  
}
```

```
keJk1ojG2qQKxEtweDd1Uj2: {  
  ...  
}
```

Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44  
}
```

Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44  
}
```

Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44  
}
```

Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44  
}
```


Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44  
}
```

Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44  
}
```

Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44  
}
```

Cats

```
4btcA7LEvcMb4A5fF910: {  
  cloudImagePath: "https://firststorage.com/535.jpg",  
  imagePath: "https://cat.imgix.net/cat/535.jpg",  
  name: "Kitty",  
  note: "I love my cat!",  
  timestamp: June 3, 2018 at 1:36:29 PM UTC-7,  
  likes: 44
```

Likes

```
KtX8NP0tNi68vzJCkcDUjp1: true
```

```
}
```

Likes

```
KtX8NP0tNi68vzJCkcDUjp1: true
```

```
5cg7sSVwfJYyT7LemgUtgEn: true
```

```
keJk1ojG2qQKxEtweDd1Uj2: true
```

```
let dbRef = Firestore.firestore().collection("cats")
var cats = [Cat]()

dbRef.getDocuments { snapshot, error in
    if let error = error {
        print(error)
        return
    }
    for doc in snapshot!.documents {
        let cat = Cat(snapshot: doc)
        self.cats.append(cat)
    }
}
```

```
let dbRef = Firestore.firestore().collection("cats")
var cats = [Cat]()
```

```
dbRef.getDocuments { snapshot, error in
    if let error = error {
        print(error)
        return
    }
    for doc in snapshot!.documents {
        let cat = Cat(snapshot: doc)
        self.cats.append(cat)
    }
}
```

```
let dbRef = Firestore.firestore().collection("cats")
var cats = [Cat]()
```

```
dbRef.getDocuments { snapshot, error in
    if let error = error {
        print(error)
        return
    }
    for doc in snapshot!.documents {
        let cat = Cat(snapshot: doc)
        self.cats.append(cat)
    }
}
```



```
let dbRef = Firestore.firestore().collection("cats")
var cats = [Cat]()
```

```
dbRef.getDocuments { snapshot, error in
    if let error = error {
        print(error)
        return
    }
    for doc in snapshot!.documents {
        let cat = Cat(snapshot: doc)
        self.cats.append(cat)
    }
}
```

```
let dbRef = Firestore.firestore().collection("cats")
var cats = [Cat]()
```

```
dbRef.getDocuments { snapshot, error in
    if let error = error {
        print(error)
        return
    }
    for doc in snapshot!.documents {
        let cat = Cat(snapshot: doc)
        self.cats.append(cat)
    }
}
```

Closure



```
let dbRef = Firestore.firestore().collection("cats")
var cats = [Cat]()
```

```
dbRef.getDocuments { snapshot, error in
    if let error = error {
        print(error)
        return
    }
    for doc in snapshot!.documents {
        let cat = Cat(snapshot: doc)
        self.cats.append(cat)
    }
}
```

```
print(cats) // what will this print?
```

Closure



```
let dbRef = Firestore.firestore().collection("cats")
var cats = [Cat]()
```

```
dbRef.getDocuments { snapshot, error in
    if let error = error {
        print(error)
        return
    }
    for doc in snapshot!.documents {
        let cat = Cat(snapshot: doc)
        self.cats.append(cat)
    }
    print(cats) // what will this print?
}
```

Closure



```
let dbRef = Firestore.firestore().collection("cats")

func downloadCats(completion: @escaping ([Cat]) -> Void) {
    dbRef.getDocuments { snapshot, error in
        var cats = [Cat]()
        if let error = error {
            print(error)
            return
        }
        for doc in snapshot!.documents {
            let cat = Cat(snapshot: doc)
            self.cats.append(cat)
        }
        completion(cats)
    }
}
```

```
catDownloader.downloadCats() { cats in  
    // yay much better!  
    self.cats = cats  
    tableView.reloadData()  
}
```



Cloud Firestore

Firebase Authentication

Google Cloud Storage for Firebase

Imgix

ML Kit for Firebase

Auth is hard.

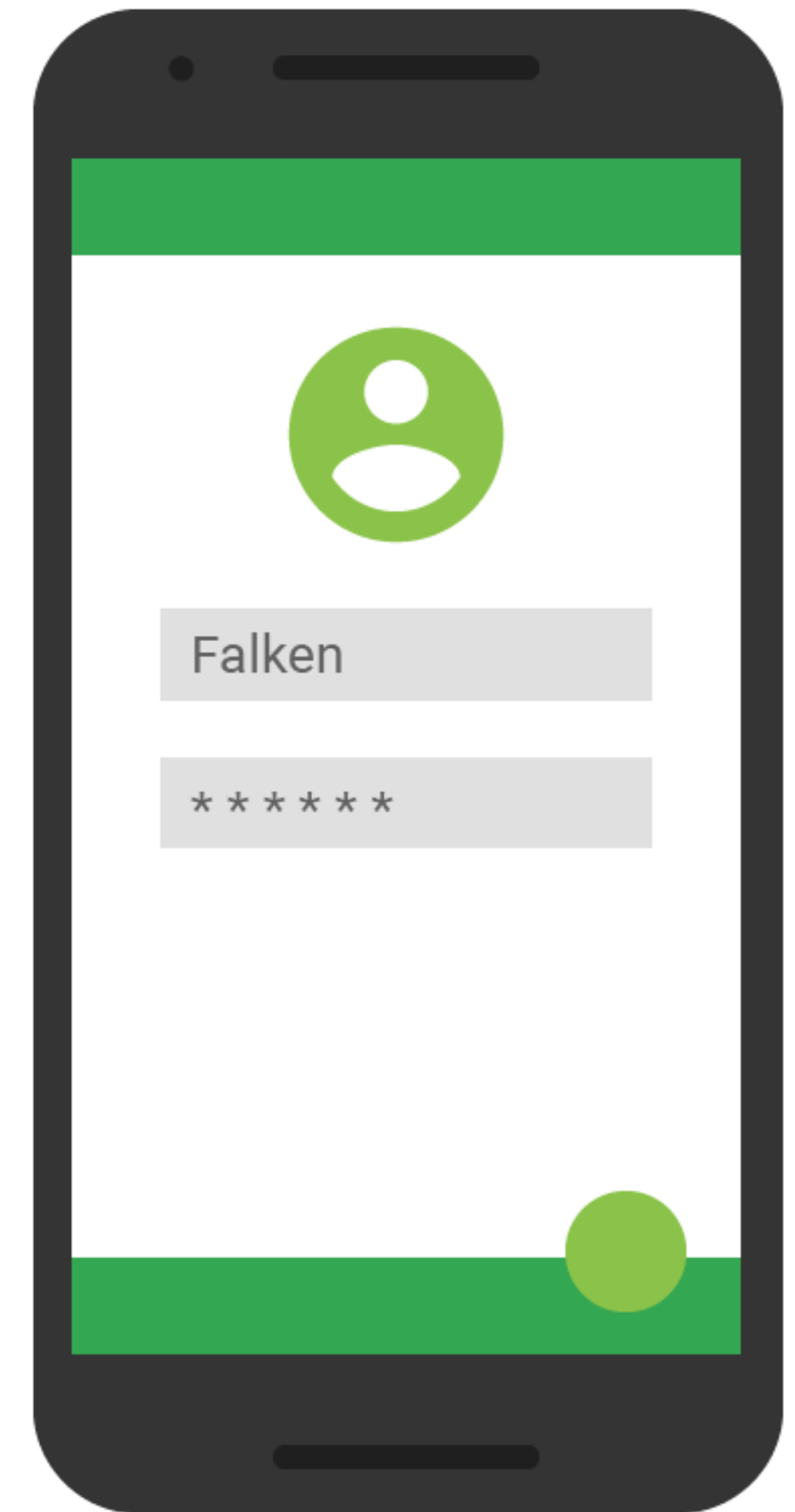
Firestore Auth makes it easier.



Authentication

Supports:

- Email & password
- Google
- Facebook
- Twitter
- GitHub
- Anonymous
- Phone number
- Existing auth systems



firestore.rules

```
service cloud.firestore {  
  match /databases/{database}/documents {  
    match /cats/{catId} {  
      allow read, write: if request.auth.uid != null;  
    }  
    match /users/{uid} {  
      allow read: if request.auth.uid != null;  
      allow write: if request.auth.uid == uid;  
    }  
  }  
}
```

firestore.rules

```
service cloud.firestore {  
  match /databases/{database}/documents {  
    match /cats/{catId} {  
      allow read, write: if request.auth.uid != null;  
    }  
    match /users/{uid} {  
      allow read: if request.auth.uid != null;  
      allow write: if request.auth.uid == uid;  
    }  
  }  
}
```

firestore.rules

```
service cloud.firestore {  
  match /databases/{database}/documents {  
    match /cats/{catId} {  
      allow read, write: if request.auth.uid != null;  
    }  
    match /users/{uid} {  
      allow read: if request.auth.uid != null;  
      allow write: if request.auth.uid == uid;  
    }  
  }  
}
```

firestore.rules

```
service cloud.firestore {  
  match /databases/{database}/documents {  
    match /cats/{catId} {  
      allow read, write: if request.auth.uid != null;  
    }  
    match /users/{uid} {  
      allow read: if request.auth.uid != null;  
      allow write: if request.auth.uid == uid;  
    }  
  }  
}
```

firestore.rules

```
service cloud.firestore {  
  match /databases/{database}/documents {  
    match /cats/{catId} {  
      allow read, write: if request.auth.uid != null;  
    }  
    match /users/{uid} {  
      allow read: if request.auth.uid != null;  
      allow write: if request.auth.uid == uid;  
    }  
  }  
}
```

firestore.rules

```
service cloud.firestore {  
  match /databases/{database}/documents {  
    match /cats/{catId} {  
      allow read, write: if request.auth.uid != null;  
    }  
    match /likes/{userId} {  
      allow read, write: if request.auth.uid == userId;  
    }  
  }  
  match /users/{uid} {  
    allow read: if request.auth.uid != null;  
    allow write: if request.auth.uid == uid;  
  }  
}
```




Cloud Firestore

Firebase Authentication

Google Cloud Storage for Firebase

Imgix

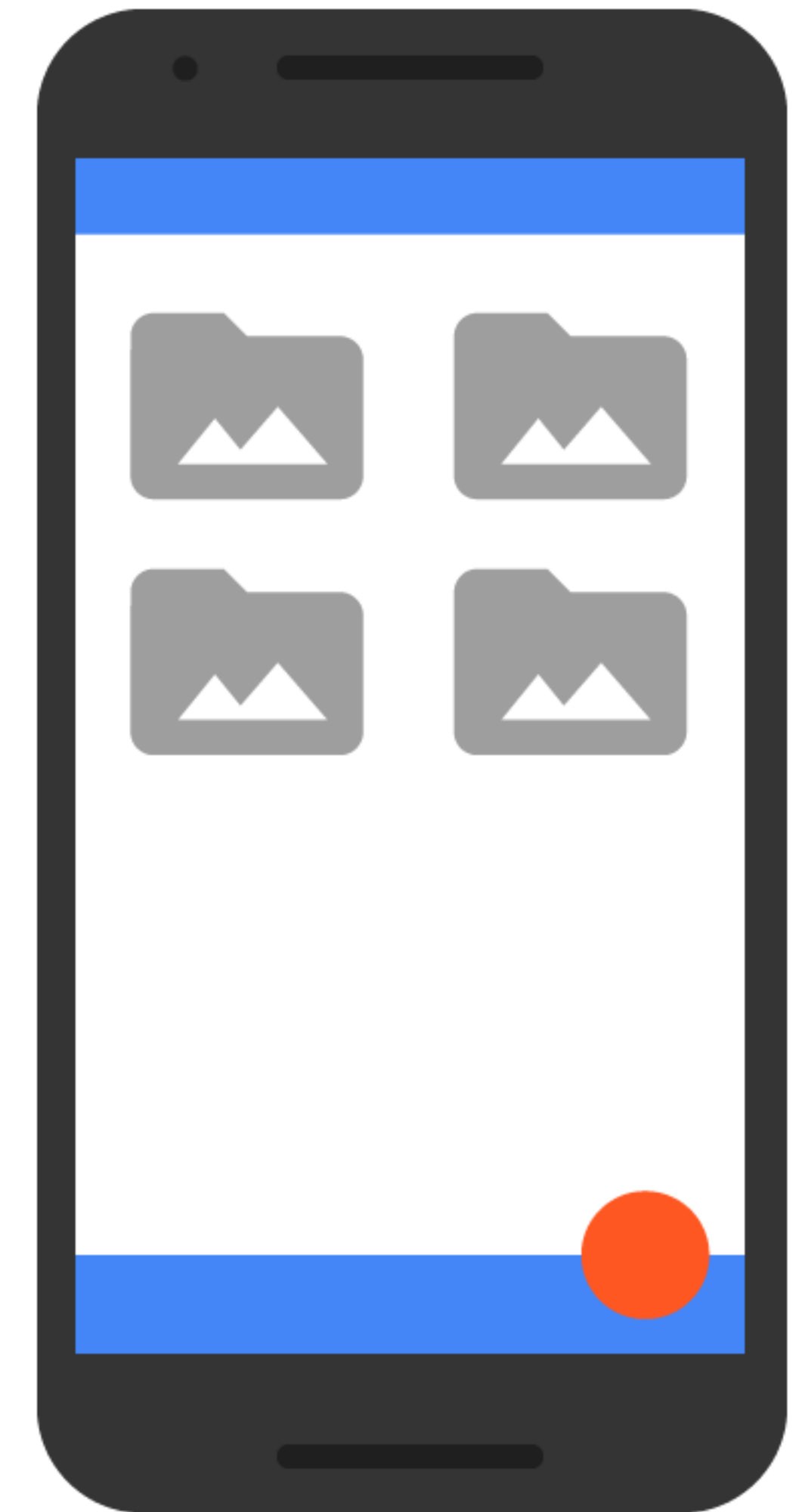
ML Kit for Firebase



Cloud Storage

for Firebase

- Same product as Google Cloud Storage
- Client SDK for mobile clients
- Handles poor connectivity
- Multiple buckets + region selection





Cloud Firestore
Firebase Authentication
Google Cloud Storage for Firebase
Imgix
ML Kit for Firebase

imgix

"imgix gives you a full range of creative imaging solutions, so you can put every pixel to work impressing your customers."

imgix

- Resizing and Cropping
- Compression and Performance
- Responsive Design
- Faster Delivery for Your Images



Cloud Firestore
Firebase Authentication
Google Cloud Storage for Firebase
Imgix
ML Kit for Firebase



ML Kit

for Firebase

Base APIs

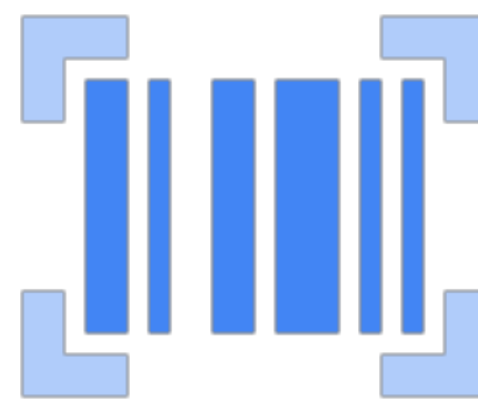
Simple APIs that solve practical use cases



Text recognition



Image labeling



Barcode scanning



Face detection



Landmark recognition

Coming soon: High density face contour feature and smart reply API



iOS. On-device image labeling.



```
vision = Vision.vision()
let labelDetector = vision?.labelDetector()

// set up image metadata ...
let image = VisionImage(image: UIImage)

labelDetector?.detect(in: image) { (labels, error) in
    guard error == nil, let labels = labels, !labels.isEmpty else {
        // ...
        return
    }

    // Handle extracted entities
}
```



Open Source

<https://github.com/firebase/firebase-ios-sdk>

<https://github.com/firebase/FirebaseUI-iOS>



Alpha program

firebase.google.com/alpha

Thank You!



Jen Person
@ThatJenPerson