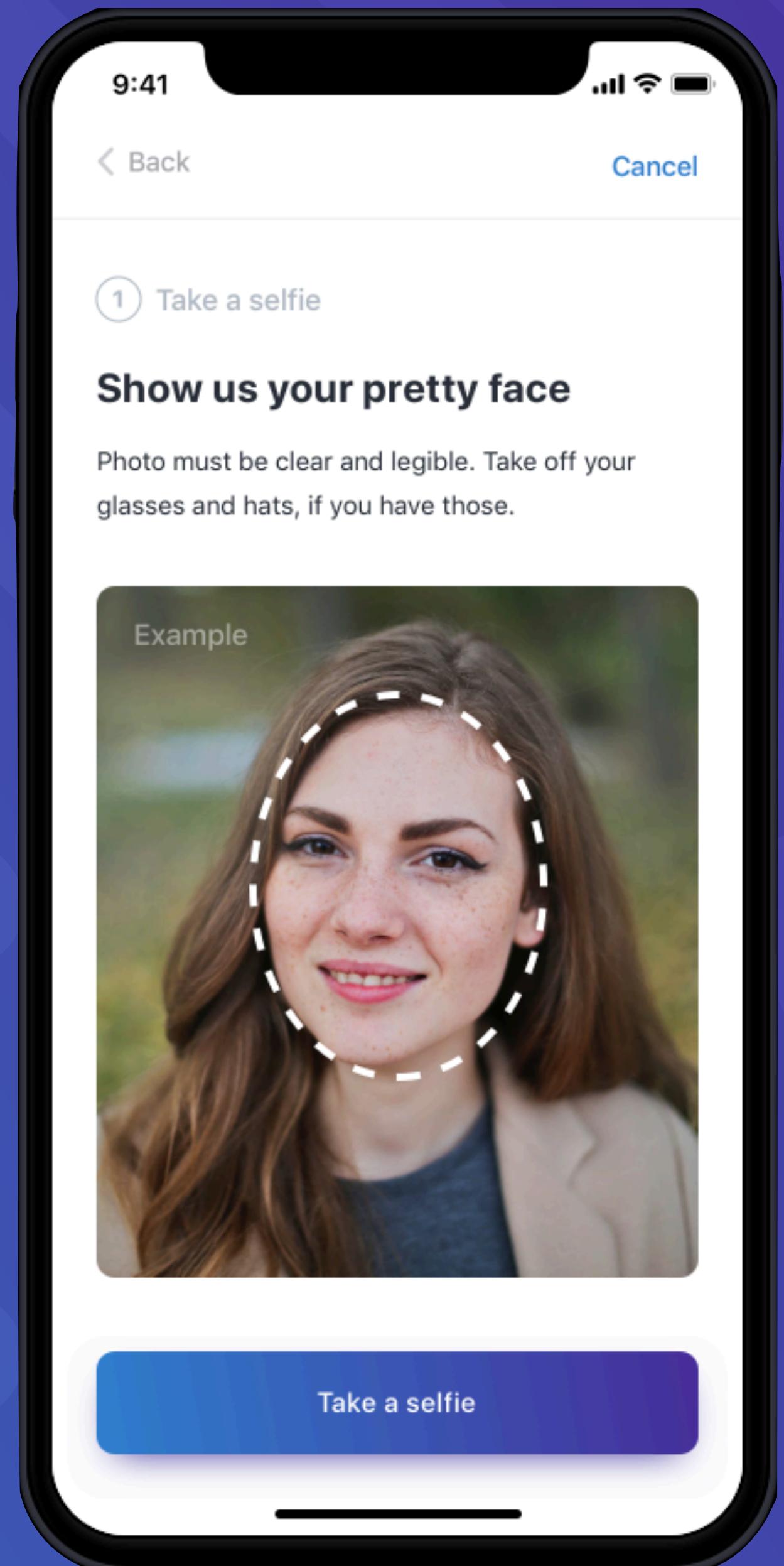
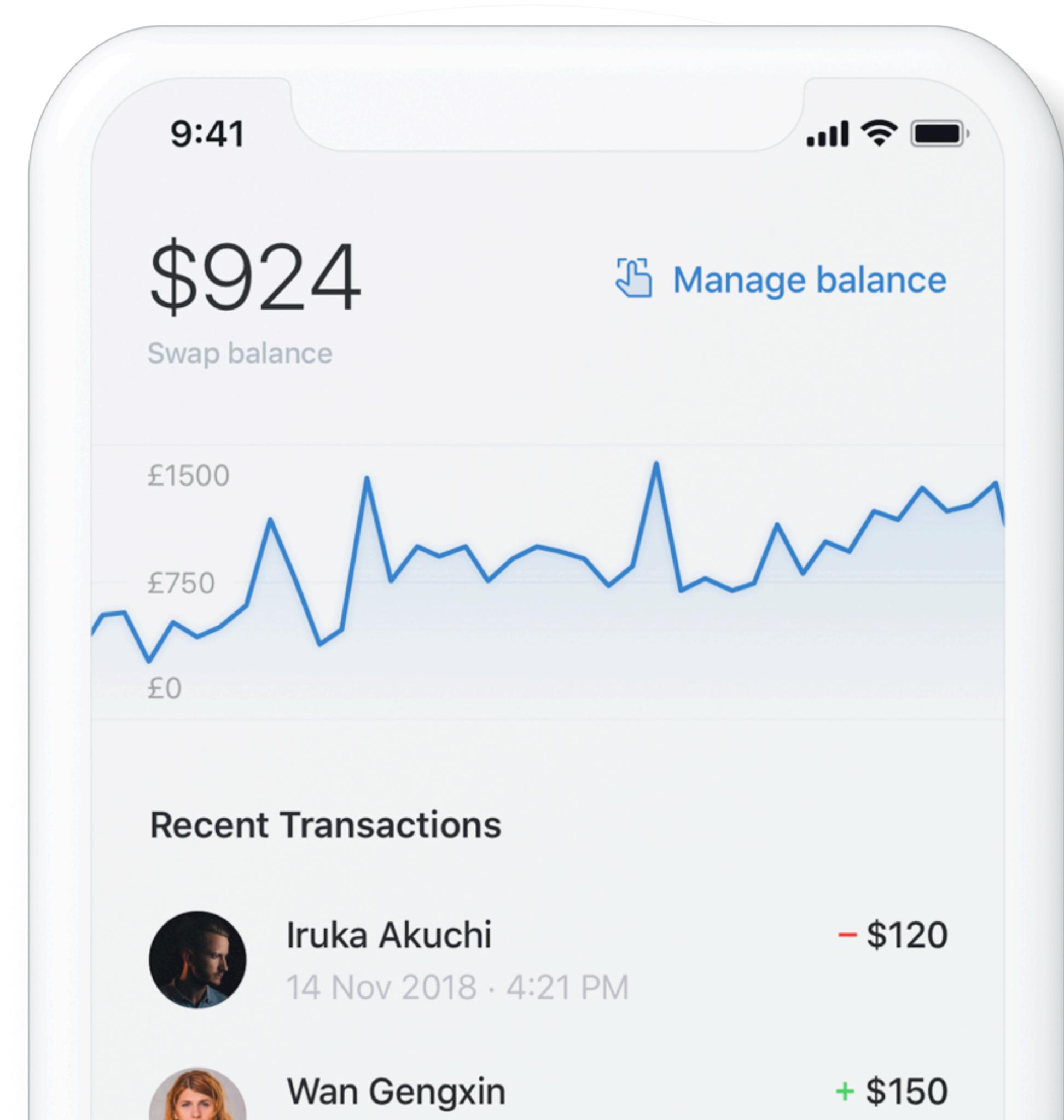


# Validación de Identidad con Google Vision

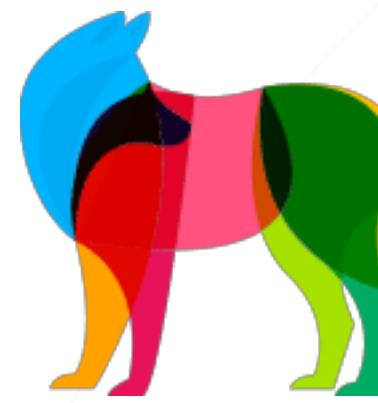


# ¿Qué es Swap?

Regulación FinTech



# Plataformas de Validación de Identidad



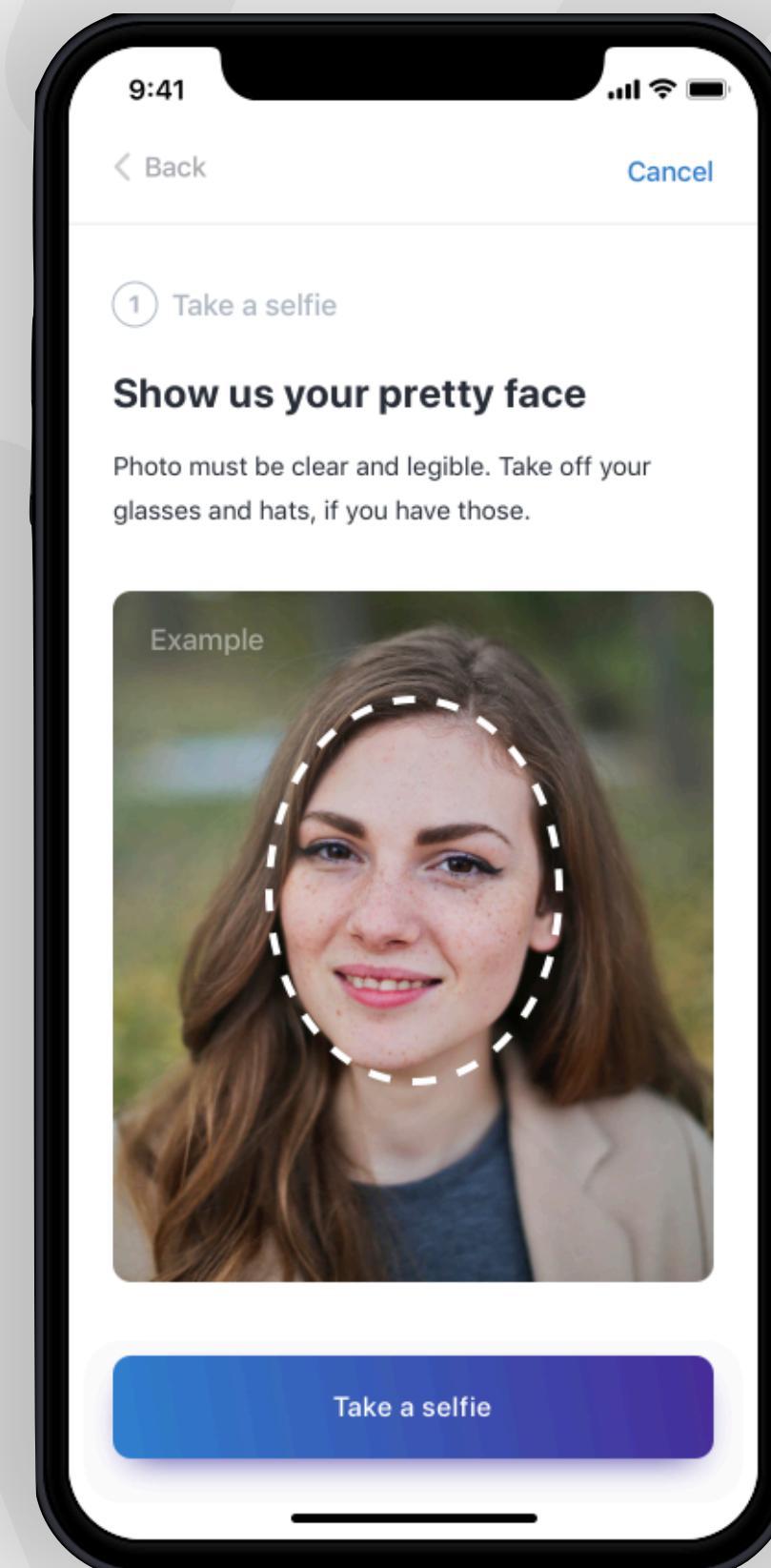
mati

JUMIO®

Trulioo

Cobran más de \$20 por validación

1. Crear el **backend**: Back4App
2. **Cloud Code**: Funciones API
3. Herramienta **PostMan**
4. **Google Vision**
5. **Módulos** en Cloud Code



BRUNO RAMOS BERHO

# Crear el BackEnd



# BACK4APP

Back4App es Backend as a Service. Montado sobre la ex-plataforma Parse, que ahora es Opensource, Back4App te permite crear tu backend de manera sencilla. Hosteada en AWS, escalable, fácil de usar, base de datos en MongoDB.

## BACKEND: BACK4APP

swap

The screenshot shows the 'My Apps' section of the Back4App dashboard. At the top left is a large blue button labeled 'Build new app' with a white plus sign icon. To its right are four application cards:

- AFORE Movil**: Dedicated Plan, 40 Req/s. Buttons: DASHBOARD, SERVER SETTINGS, PLAN USAGE.
- AFORE Movil D...**: Free Plan, 10 Req/s. Buttons: DASHBOARD, SERVER SETTINGS, PLAN USAGE.
- Avante**: Free Plan, 10 Req/s. Buttons: DASHBOARD, SERVER SETTINGS, PLAN USAGE.
- Cardello**: Free Plan, 10 Req/s. Buttons: DASHBOARD, SERVER SETTINGS, PLAN USAGE.

In the second row, there are four more cards:

- Millas Dev**: Free Plan, 10 Req/s. Buttons: DASHBOARD, SERVER SETTINGS, PLAN USAGE.
- Millas Para El ...**: Basic Plan, 30 Req/s. Buttons: DASHBOARD, SERVER SETTINGS, PLAN USAGE.
- Swap**: Standard Plan, 30 Req/s. Buttons: DASHBOARD, SERVER SETTINGS, PLAN USAGE.
- Swap Dev**: Free Plan, 10 Req/s. Buttons: DASHBOARD.

In the third row, there are four cards:

- Swap FS**: Basic Plan, 30 Req/s. Buttons: DASHBOARD.
- Swap FS Dev**: Free Plan, 10 Req/s. Buttons: DASHBOARD.
- SwapKey**: Free Plan, 10 Req/s. Buttons: DASHBOARD.
- Help**: A blue button with a question mark icon.

At the top right of the dashboard, there is a 'Hello, bruno' dropdown menu and a green 'NEW APP' button. The overall interface is clean and modern, using a light gray background with dark blue and white accents for the cards.



BRUNO RAMOS BERHO

# Cloud Code JavaScript



## Cloud Code

Cloud Code son las funciones que pueden correr en el Back End. El lenguaje de programación es JavaScript y pueden llamarse desde cualquier dispositivo por llamadas REST o con el SDK. También se pueden crear funciones que son activadas por eventos, como:

After Save o Before Save

**<https://github.com/pocketdeveloper/identidad>**

# Sign Up

La función creará un objeto usuario nuevo en la base de datos.

```
Parse.Cloud.define("signUp", function(request, response) {
    var name = request.params.name;
    var email = request.params.email;
    var password = request.params.password;

    var user = new Parse.User();
    user.set("username", email);
    user.set("password", password);
    user.set("email", email);

    // Se pueden agregar los campos adicionales
    user.set("name", name);

    // Seguridad del objeto ¿Quién puede leer y modificar?
    var acl = new Parse.ACL();
    acl.setPublicReadAccess(true);
    acl.setPublicWriteAccess(false);
    user.setACL(acl);

    try {
        await user.signUp();
        response.success('¡Sign Up exitoso!');
    } catch (error) {
        response.error("¡Ups! Salió algo mal. Error: " + error.code + " " + error.message);
    }
});
```

BRUNO RAMOS BERHO

# PostMan



## **PostMan**

Es una herramienta de Google que permite realizar llamadas REST.

## Llamada REST API con PostMan

swap

The screenshot shows the Postman application interface. The top navigation bar includes 'Runner', 'Import', 'Builder' (which is selected), 'Team Library', and various status indicators like 'IN SYNC' and user profiles. Below the navigation is a toolbar with icons for different operations and a 'Production' dropdown.

The main workspace displays a single API endpoint:

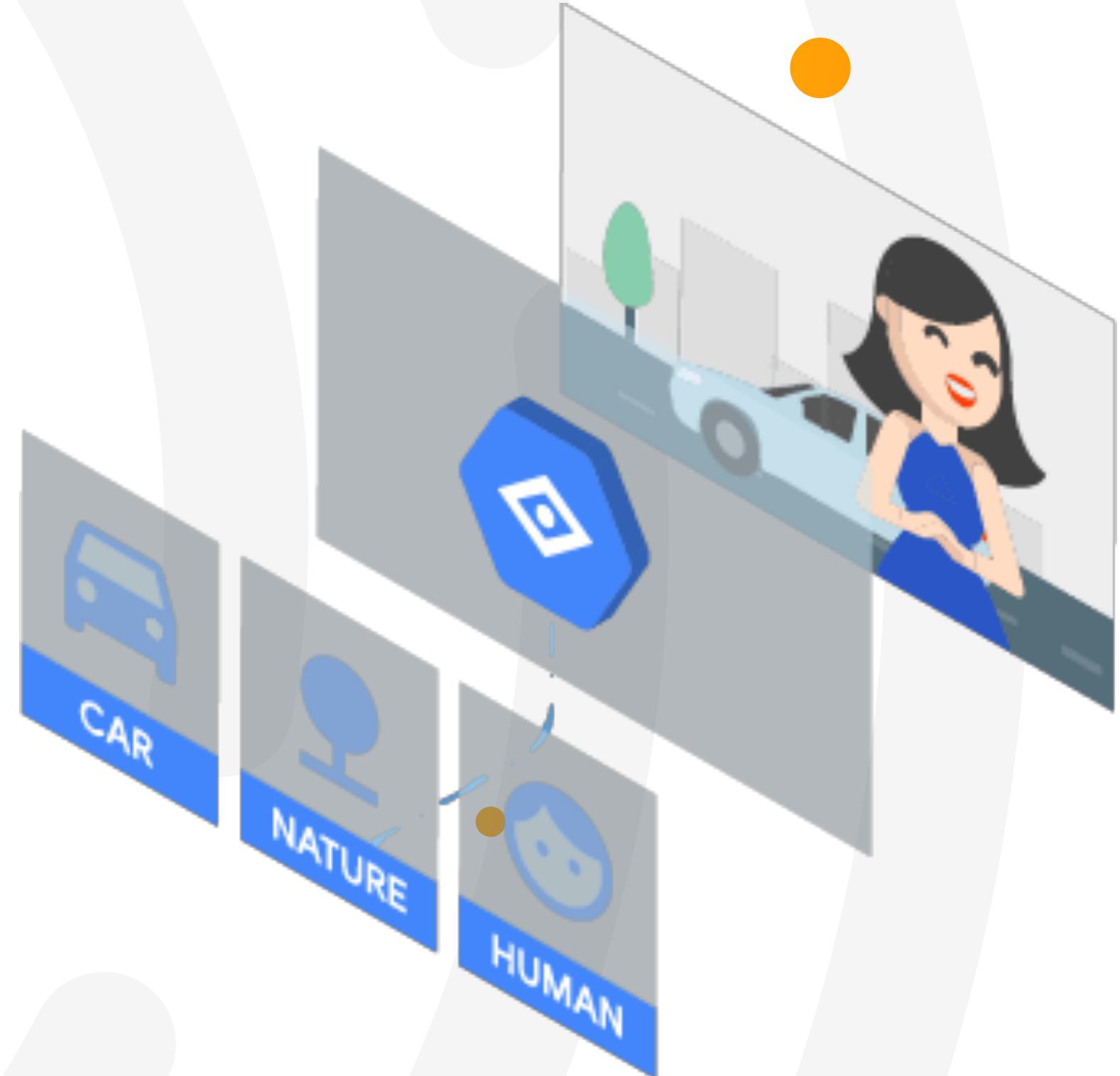
- Method:** POST
- URL:** https://parseapi.back4app.com/functions/signUp
- Body:** (selected tab) JSON (application/json)  
The JSON payload is:

```
1 {  
2   "name": "Bruno Ramos",  
3   "email": "brunoramosberho@me.com",  
4   "password": "Anonymous:)"  
5 }
```

At the bottom right are 'Send' and 'Save' buttons. The bottom of the interface features a footer with the Postman logo and version information.

BRUNO RAMOS BERHO

# Google Vision



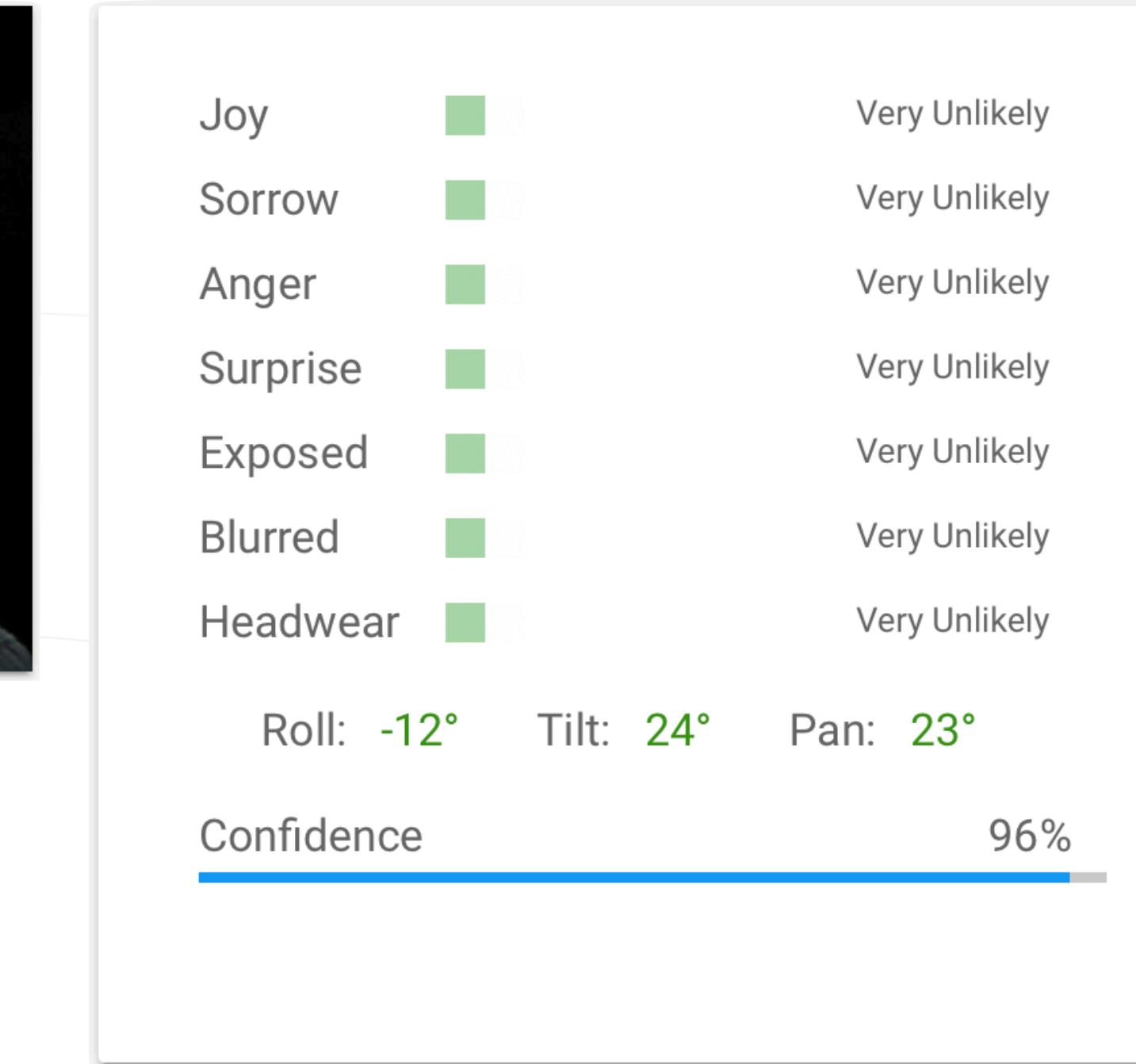
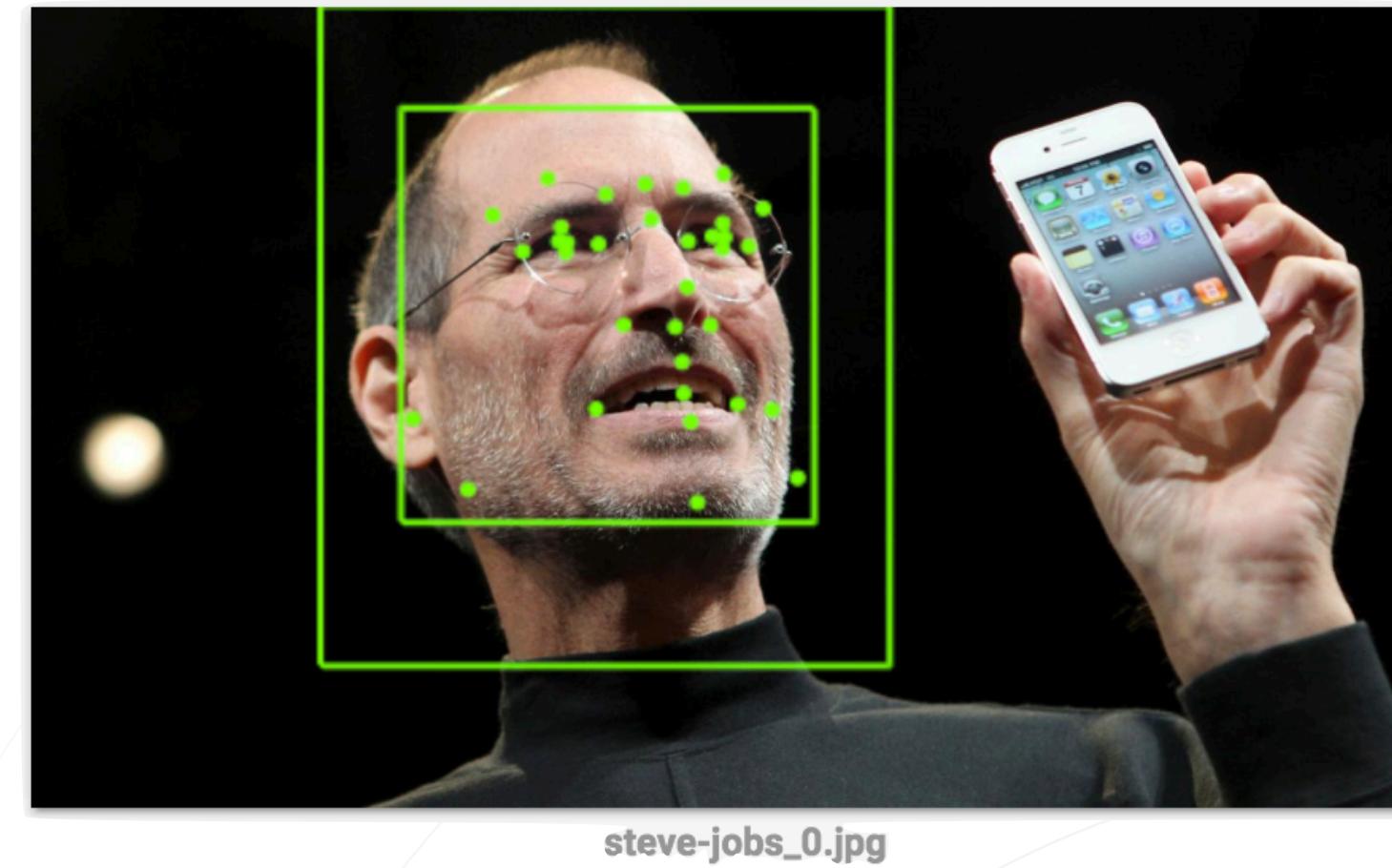
## Cloud Vision

Cloud Vision API permite a los desarrolladores comprender el contenido de una imagen al encapsular poderosos modelos de aprendizaje automático en una API REST fácil de usar. Clasifica imágenes, detecta objetos y rostros y lee palabras.

# Cloud Vision



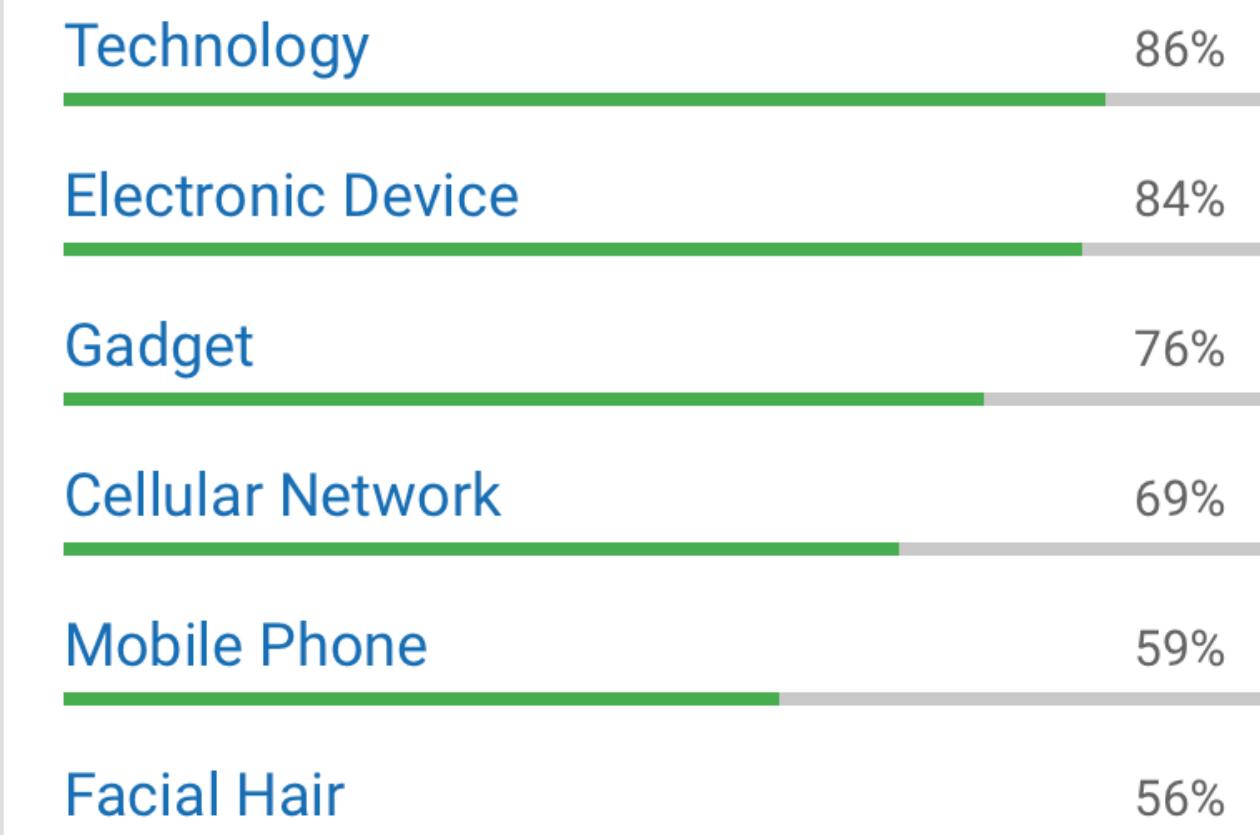
# Cloud Vision - Labels



# Cloud Vision - Faces



steve-jobs\_0.jpg



# Cloud Vision - Web



steve-jobs\_0.jpg

## Web Entities

Steve Jobs	14.982
	0.7027
Apple	0.7027
2011	0.5506
Death	0.5452
United States of America	0.4761
October 5	0.4503
	0.3248
	0.3217
Chief Executive	0.2936
Barack Obama	0.06119
Steve Wozniak	0.06078
Lisa Brennan-Jobs	0.05683

# API REST de Cloud Vision

The screenshot shows the Postman application interface for making a POST request to the Google Cloud Vision API.

**Request (Body):**

```
1 [ {  
2   "requests": [  
3     {  
4       "features": [ {"type": "LABEL_DETECTION"}, {"type": "FACE_DETECTION"}, {"type": "TEXT_DETECTION"}, {"type": "DOCUMENT_TEXT_DETECTION"} ],  
5       "image": {  
6         "source": {  
7           "imageUri": "https://cdn.tn.com.ar/sites/default/files/styles/1366x765/public/2017/10/05/steve-jobs_0.jpg"  
8         }  
9       }  
10     ]  
11   }  
12 }]
```

**Response:**

Status: 200 OK | Time: 1124 ms | Size: 18.06 KB

```
1 [ {  
2   "responses": [  
3     {  
4       "faceAnnotations": [ [ ] ],  
5       "labelAnnotations": [ [ ] ],  
6       "textAnnotations": [ [ ] ],  
7       "fullTextAnnotation": { [ ] }  
8     }  
9   ]  
10 }  
11 }  
12 }
```

BRUNO RAMOS BERHO

# Módulos



# Módulos

Para Cloud Code, es necesario tener un archivo **main.js**. Es posible crear Módulos u otros archivos JS para tener más claridad en el código y mejor documentación.

Para hacer esto, es necesario crear otro archivo JS y hasta arriba del archivo **main.js** poner **require("./archivo.js")**



## Face++

Este servicio permite comparar dos caras y da una probabilidad de que sean la misma persona. En este caso en particular es útil para identificar si la persona de la selfie es la misma que la persona de la INE proporcionada y así tomar una decisión.