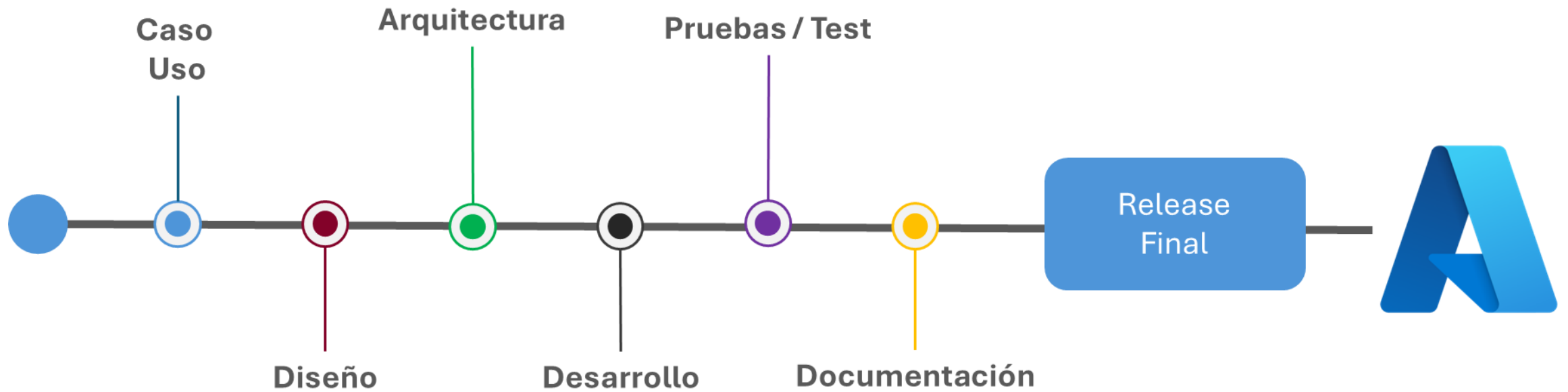


# Plan de Practica

(08-05-24 → 20-06-24))



# Caso de Uso (13-05-24)

## **Mapa de Calor**

- Usar Azure Vision para detectar puestos ocupados, e implementar facial recognition usando Azure Face API para documentar entrada y salida de cada persona
- Generar un mapa de calor de areas communes

## **Usos**

1. Análisis y optimización del espacio
2. Saber quien esta presente para facilitar contacto

# Entregables

1. Puestos Libres/Mapa de Ocupados
2. Identificar personas
  1. Nombre
  2. Edad
  3. Etc.
3. Mapa de Calor

# Arquitectura

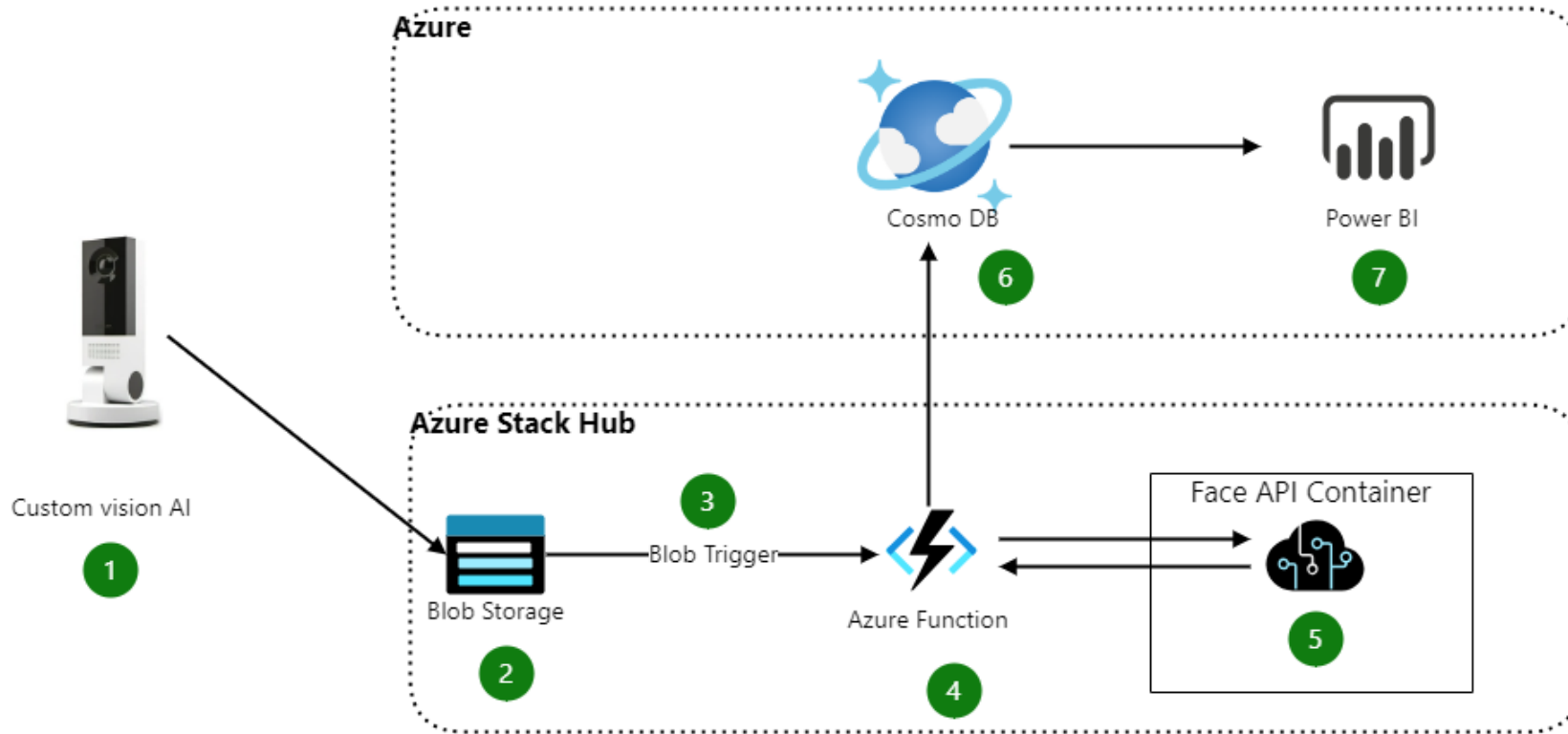
## Componentes:

1. Visual Studio Code/Python
2. Azure Vision y Face
3. Power BI
4. Camera
5. Fotos de Entrenamiento

## Recursos

- <https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/>
  - Guías de Microsoft
- <https://github.com/pablosabaterlp/FaceRecognitionAzure.git>
  - Repositorio GitHub

# Diseño



## Dataflow:

1. Si se detecta una persona, la cámara toma una foto (JPEG, PNG, o BMP) y la sube un Azure Stack Hub Blob Storage
2. El blob storage provoca una function de Azure
3. La funcion llama a un contenedor con el API de Face para obtener la información de la persona
4. Los datos se envian a un Cosmos DB
5. Los datos se represtan visualmente usando Power BI

# Resumen Avance Diario

## Pasos

### Primer Testing

1. Reconocer Caras en Foto
2. Reconocer Caras en Video

### Spatial Analysis

1. Entrenar modelo para detectar puestos en una oficina
2. Crear codigo para detectar puestos ocupados usando modelo
3. EXTRA: Añadir aspecto reconocimiento facial

### Integración Cloud

1. Implementar utilización de blob storage
2. Implementar Cosmos DB, function, etc.
3. Testing y Optimización

## Programa

1. 13/5/24 -> 15/5/24

2. 15/5/24 -> 16/5/24

1. 17/5/24 -> 20/5/24

2. 17/5/24 -> 22/5/24

3. 22/5/24 -> 27/5/24

1. 2/5/24 -> 5/6/24

2. 5/6/24 -> 12/6/24

3. 12/6/24 -> 19/6/24

# Documentación:

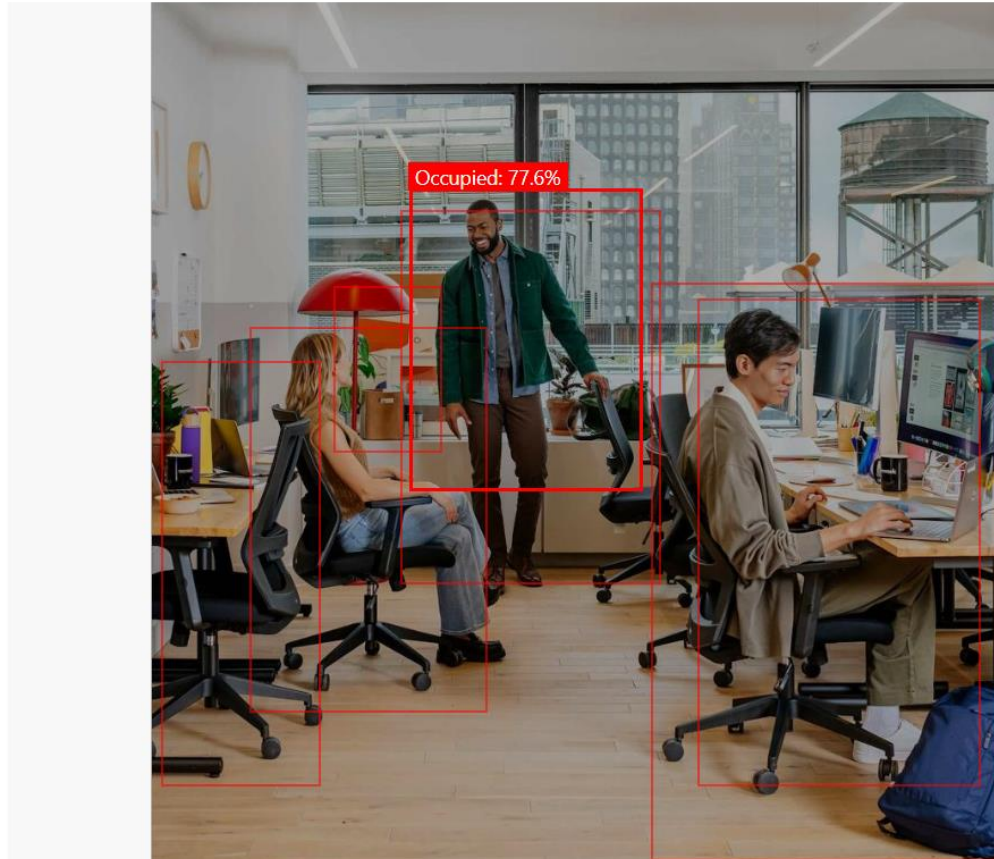
## Versión Uno:

1. Entrenado con 15 fotos de oficinas ocupadas y 15 desocupadas
2. Tardo 35 minutos en entrenar
3. Detecta los puestos con gran certeza (75%), pero aun se equivoca en si están ocupados o no

Image Detail

↶ Undo Changes

☒ Regions Shown



### My Objects



Only show suggested objects if the probability is above the selected threshold.

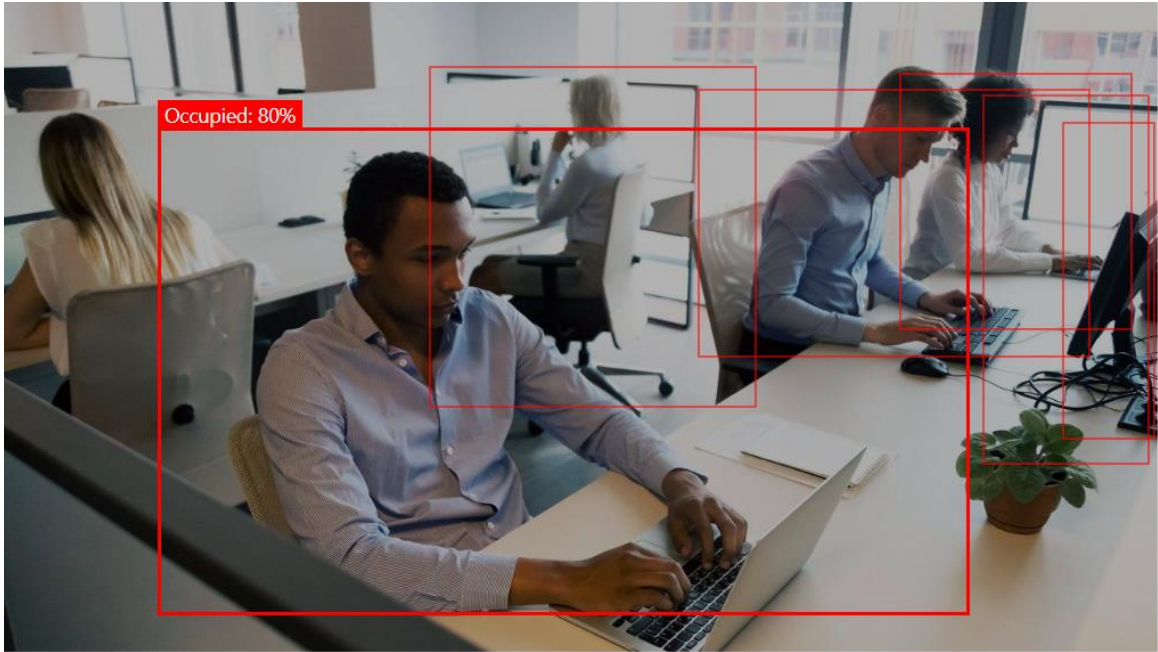
Threshold Value: 74%



### Predictions

Predictions are shown in red

Tag	Probability
Unoccupied	97.5%
Unoccupied	95.1%
Unoccupied	90.4%
Occupied	86.3%



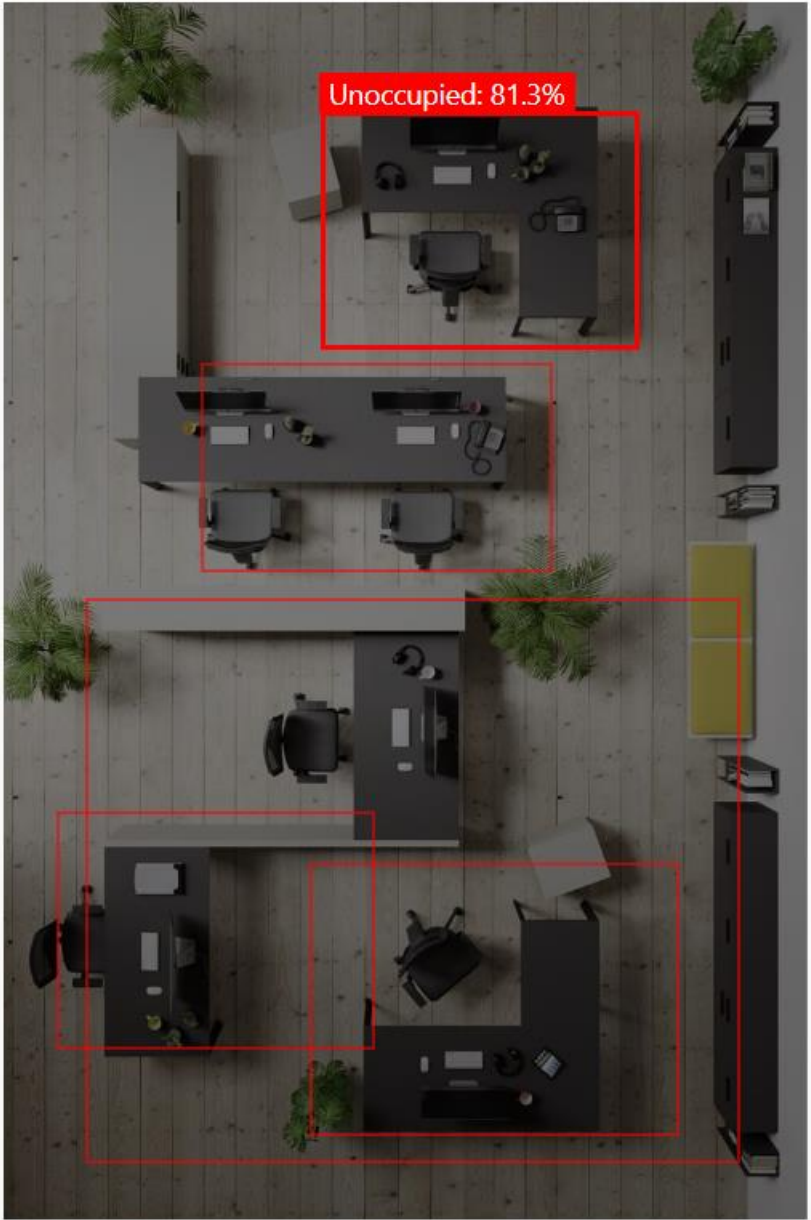
Only show suggested objects if the probability is above the selected threshold.

Threshold Value: 74%

### Predictions

Predictions are shown in red

Tag	Probability
Occupied	99.4%
Occupied	95.9%
Unoccupied	94.9%
Occupied	94.6%



Iterations

Probability Threshold: 65% ⓘ

Overlap Threshold: 30% ⓘ

Iteration 1

PUBLISHED

Trained : 1 hours ago with General [A1] domain

Unpublish

Prediction URL

Delete

Export

### Iteration 1

Finished training on 17/5/2024, 13:23:16 using **General [A1]** domain

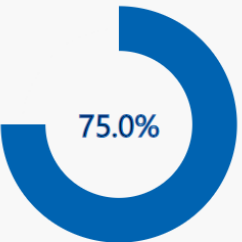
Iteration id: e2d81309-a790-4cc3-8d55-21531e0af5c4

Published as: Iteracion1 - Basico

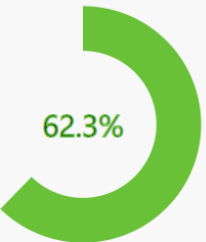
Precision ⓘ



Recall ⓘ



mAP ⓘ





# Demostración de Cada Entregable

## Reconocimiento en Fotos (15/5/24)

