



# Práctica. Evidencia integradora de Sistemas Inteligentes

8A ITI - Sistemas Inteligentes

Mendieta Chimal Sony Luis - MCSO220598

Nava Sánchez Axel - NSAO220388

Torres Flores Jesús Antonio - TFJO220203

Universidad Politécnica del Estado de Morelos

Fecha de Entrega: 20 de abril de 2025

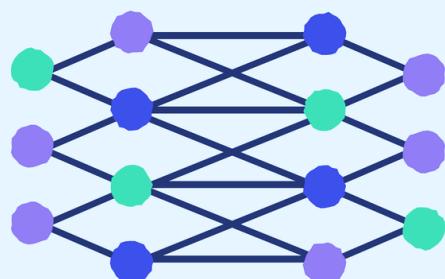
# ÍNDICE

- 1** Introducción
- 2** Problematica
- 3** Desarrollo
- 4** Pruebas
- 5** Conclusiones

# Introducción



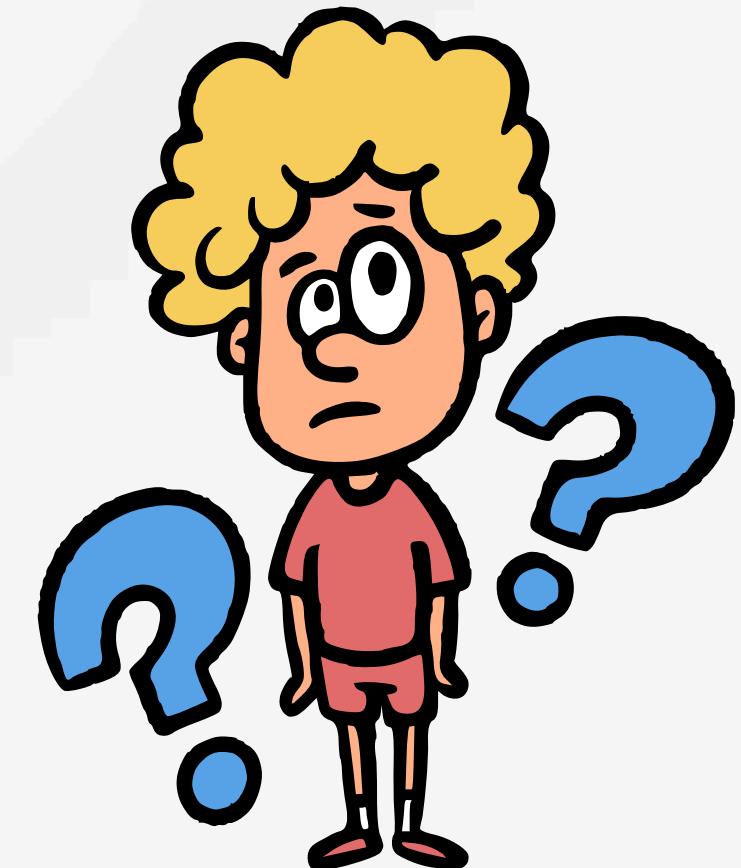
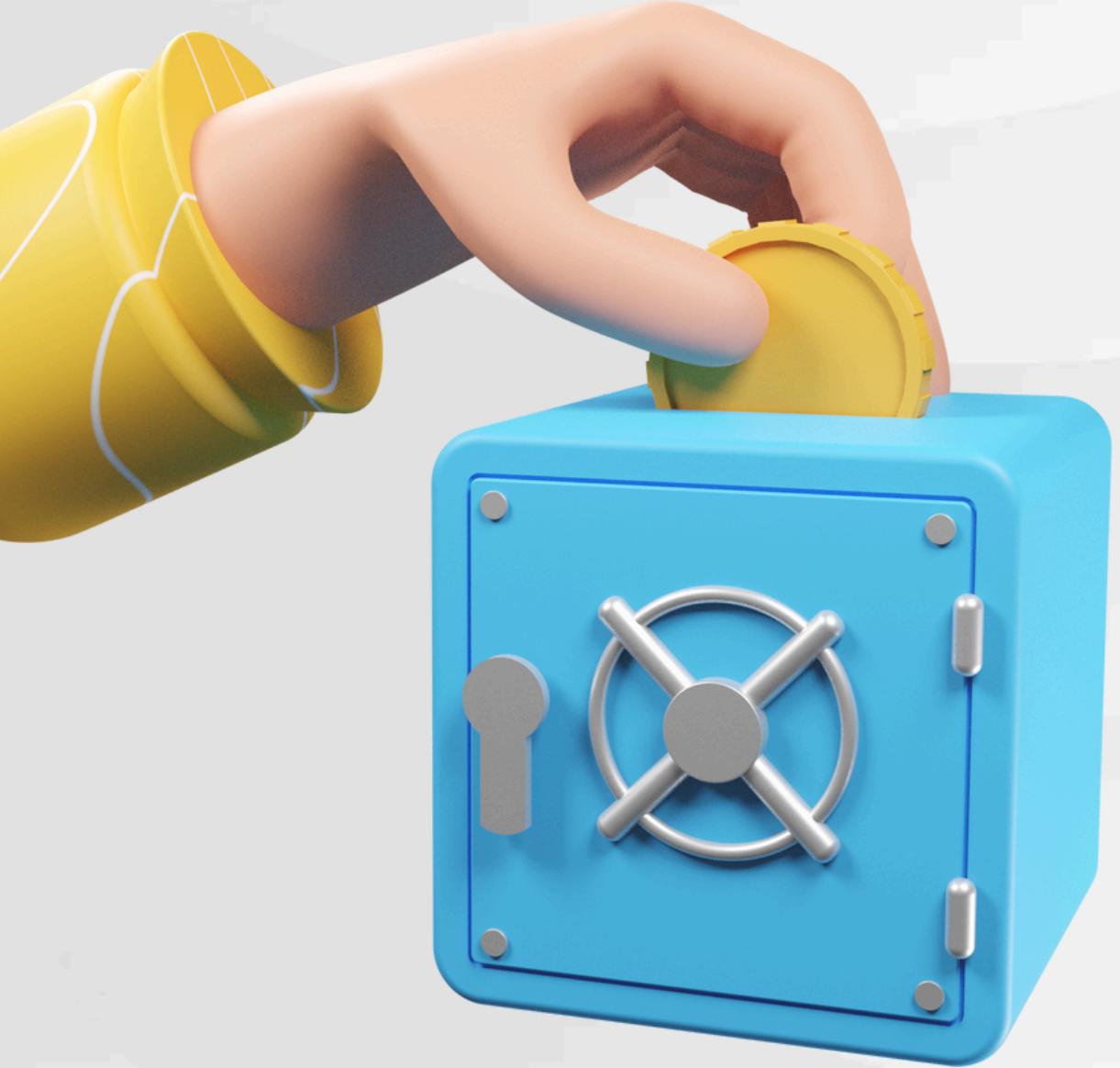
Monedas Mexicanas



Redes Neuronales Convolucionales

# Problematica

Muchos niños tienen dificultad para diferenciar las monedas y contarlas correctamente.



# Impacto



Educación  
Infantil

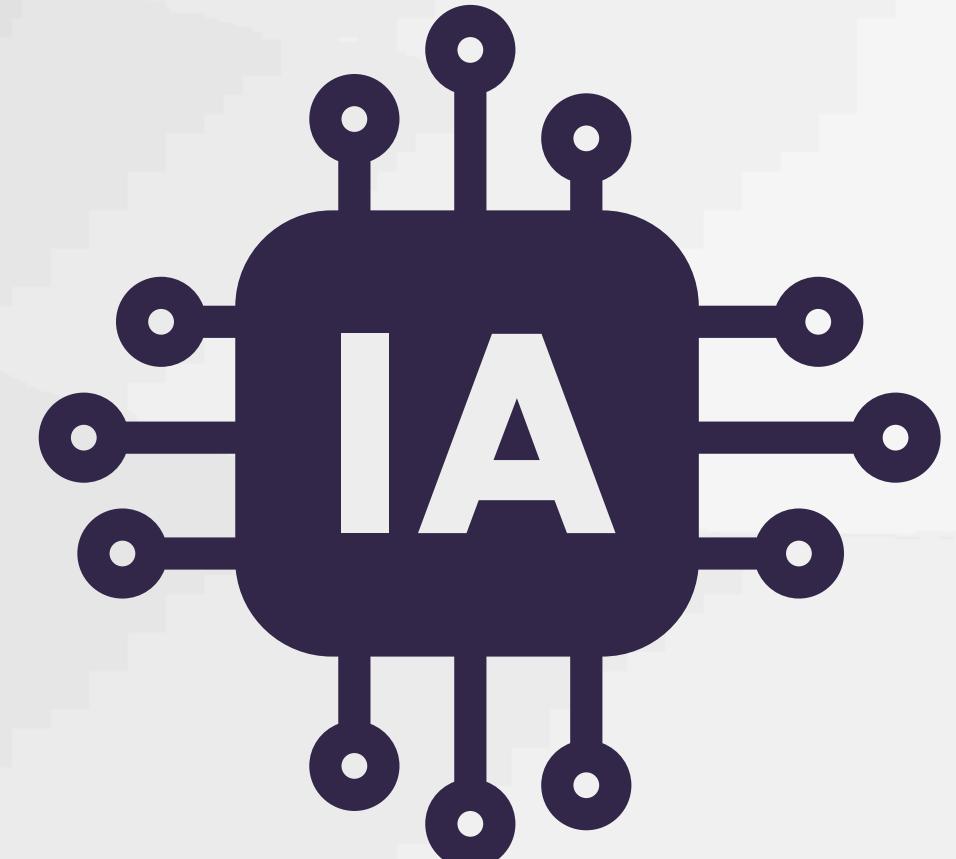


Apoyo Didáctico

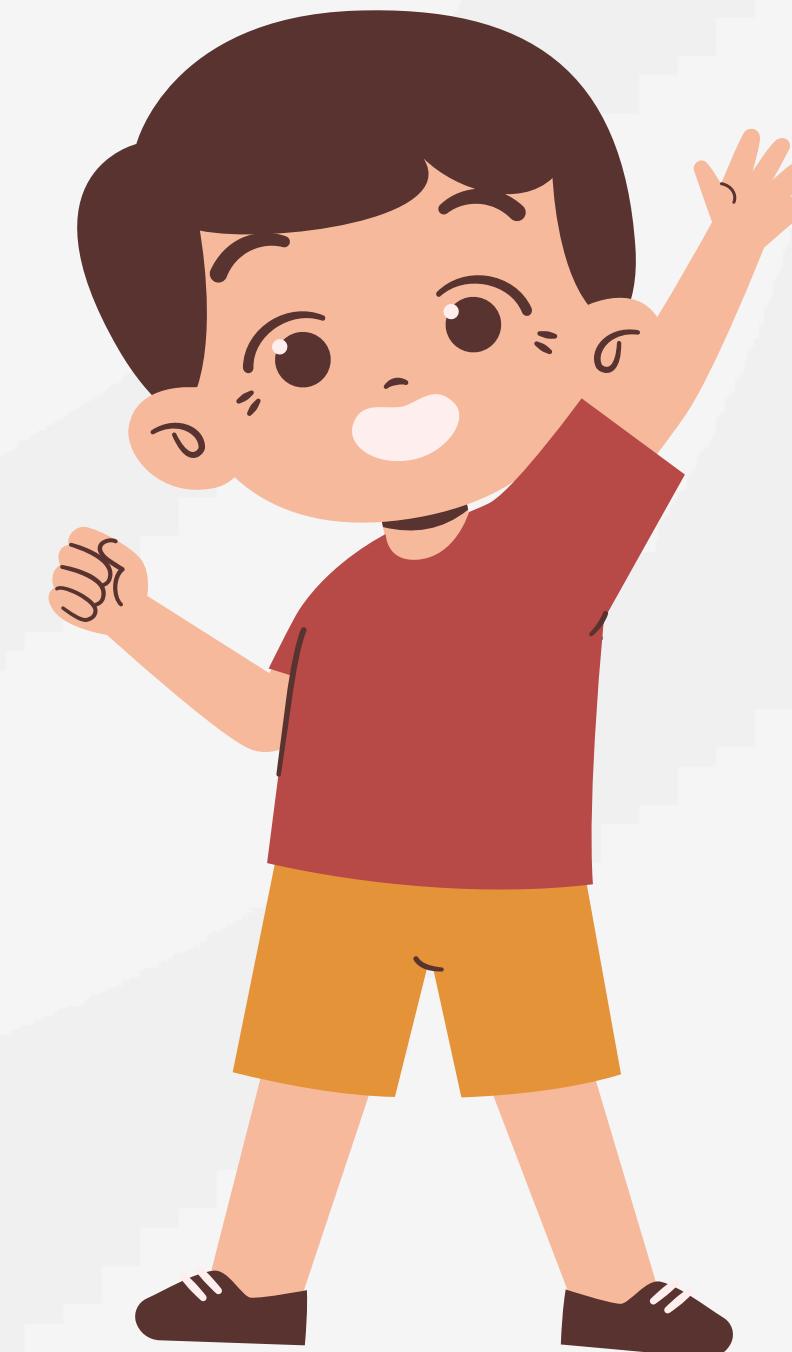


Uso de la  
tecnología

Desarrollar un sistema capaz de reconocer monedas y calcular su valor total, con el fin de apoyar a niños en el aprendizaje de la diferenciación y conteo.



# Objetivo



# Desarrollo

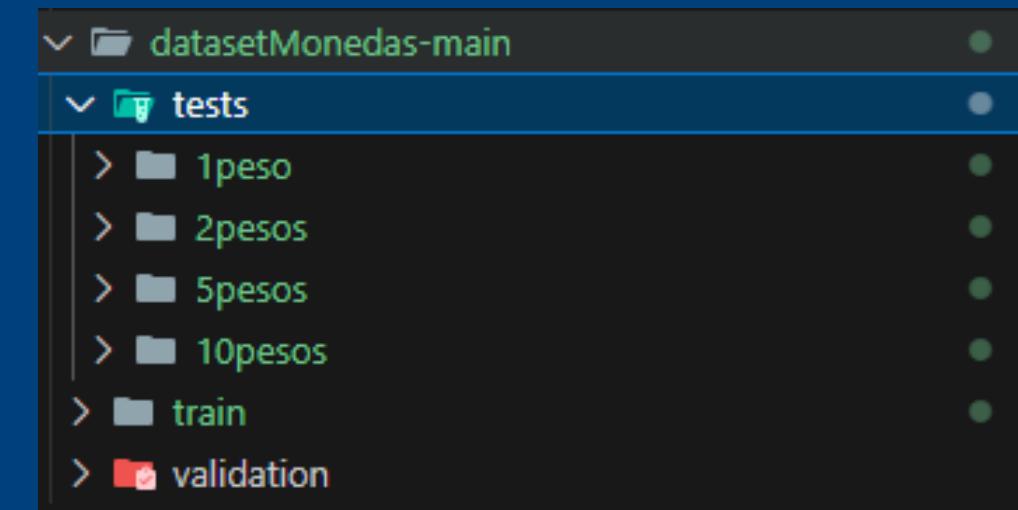


TensorFlow/Keras : Framework de aprendizaje automático utilizado para diseñar, entrenar y evaluar la Red Neuronal Convolucional (CNN) .

# Desarrollo



Dataset



Entrada



# Desarrollo

```
monedas / monedero / templates / monedero.html
{# load static #}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Monedero</title>
    <link href="https://fonts.googleapis.com/css2?family=Albert+Sans:wght@400;700&display=swap" rel="stylesheet">
    <link href="https://fonts.googleapis.com/css2?family=Fredoka+One&display=swap" rel="stylesheet">
    <link rel="stylesheet" href="{% static 'css/style.css' %}">
</head>
<body>
    <div class = "Contenedor_principal">
        <h1>Monedas</h1>
        <p>Descubre el valor de tus monedas con ayuda de Don Monedon</p>
        
        <div class = "Contenedor_formulario">
            <div id="upload-area" class="upload-area" >
                <form id="upload-form" action="/upload/" method="post" enctype="multipart/form-data">
                    {% csrf_token %}
                    <label for="coinImage" class="upload-label">
                        
                        <p>Arrastra tu imagen de tu moneda aquí o haz clic para seleccionarla</p>
                        <input type="file" id="coinImage" name="coinImage" accept="image/*" required style="display: none;">
                    </label>
                </form>
            </div>
            <div class = "Contenedor_resultado">
                <div class = "Contenedor_resultado_imagen">
                    <div id="result">
                        <p>Tu moneda <br> aparecerá aquí </p>
                    </div>
                </div>
                <div class = "Contenedor_resultado_texto">
                    <p>Tu moneda vale</p>
                    <div id="result-text" class="result-text">
                        <div>
                            <div>
                                <div>
                                    <div>
                                        <button id="discover-button" class="upload-button">Descubrir</button>
                                    <h2>Colección de monedas</h2>
                                    <p id="total_monedas">Tienes</p>
                                    <div id="total_monedas" class="coin-count">
                                        <div>
                                            <div>
                                                <script src="{% static 'js/script.js' %}"></script>
                                            </div>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</body>
```

# Monedas

Descubre el valor de tus monedas con ayuda de Don Monedon



Arrastra tu imagen de tu moneda aquí  
o haz clic para seleccionarla

Tu moneda aparecerá aquí

Tu moneda vale

Descubrir

# Desarrollo

```
def Evaluacion(request):
    if request.method == 'POST':
        imagen = request.FILES.get('coinImage')
        altura,anchura = 100,100

        imagen_bytes = BytesIO(imagen.read())

    #Leer el modelo entrenado y sus pesos
    modelo = os.path.join(MODELS_DIR, "modelo_entrenado.keras")
    pesos = os.path.join(MODELS_DIR, "pesos.weights.h5")

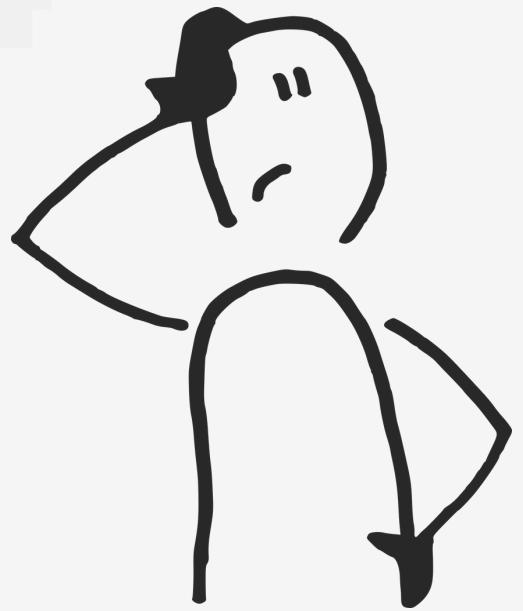
    #Cargar el modelo entrenado
    cnn = load_model(modelo)
    cnn.load_weights(pesos)

    #Transformar la imagen a clasificar
    imagen_clasificar = load_img(imagen_bytes, target_size=(altura, anchura))
    imagen_clasificar = img_to_array(imagen_clasificar)
    imagen_clasificar = np.expand_dims(imagen_clasificar, axis =0)

    #Evaluar la imagen
    clase = cnn.predict(imagen_clasificar)
    print(clase)
    arg_max = np.argmax(clase)

    resultado = ""
    valor = 0
    if arg_max == 0:
        print("10 pesos")
        valor = 10
        resultado = "10 pesos"
    elif(arg_max==1):
        print("1 peso")
        valor = 1
        resultado = "1 peso"
    elif(arg_max==2):
        print("2 pesos")
        valor = 2
        resultado = "2 pesos"
    elif(arg_max==3):
        print("5 pesos")
        valor = 5
        resultado = "5 pesos"

    return JsonResponse({'resultado': resultado , 'valor': valor})
```



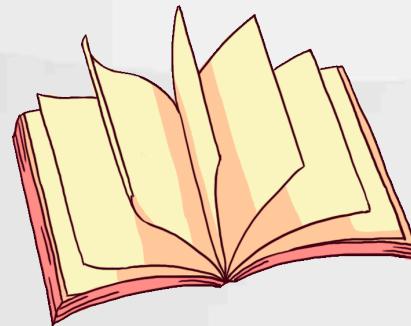
Predictión



Resultado

# Desarrollo

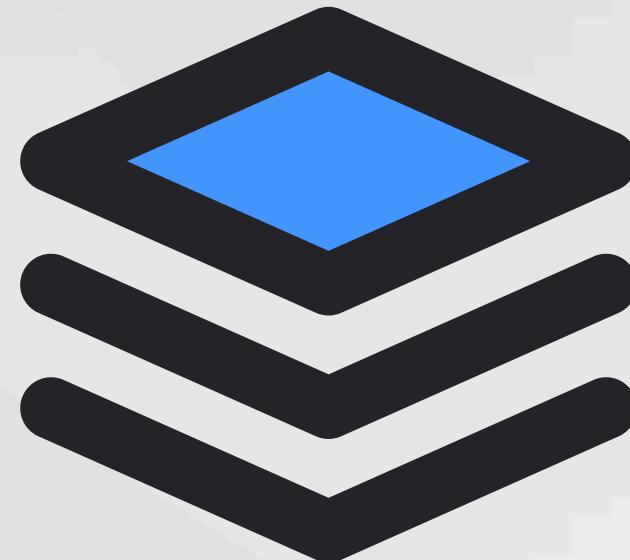
Épocas: 50



Clases: 4

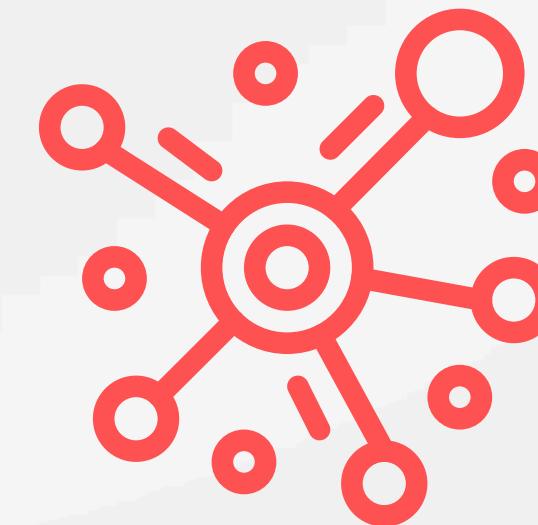


```
train_dataset = tf.keras.utils.image_dataset_from_directory(  
    entrenamiento, # Ruta a "train/"  
    seed=123,  
    image_size=(altura, anchura),  
    batch_size=batch_size  
)  
  
# Cargar datos de validación (20%)  
valid_dataset = tf.keras.utils.image_dataset_from_directory(  
    validacion, # Ruta a "validation/"  
    seed=123,  
    image_size=(altura, anchura),  
    batch_size=batch_size  
)
```



Conv2D  
Capas

MaxPooling2D



255 neuronas

# Desarrollo

```
# Definir la arquitectura de la Red Neuronal Convolucional (CNN)
CNN = Sequential([
    # Primera capa convolucional con 16 filtros de tamaño 3x3
    Conv2D(16, (3, 3), padding="same", activation="relu", input_shape=(altura, anchura, 3)),
    # Capa de MaxPooling para reducir la dimensionalidad espacial
    MaxPooling2D(pool_size=(3, 3)),

    # Segunda capa convolucional con 32 filtros de tamaño 3x3
    Conv2D(32, (3, 3), padding="same", activation="relu"),
    # Otra capa de MaxPooling para reducir aún más la dimensionalidad
    MaxPooling2D(pool_size=(3, 3)),

    # Aplanar los mapas de características en un vector unidimensional
    Flatten(),

    # Primera capa densa (fully connected) con 255 neuronas y función de activación ReLU
    Dense(255, activation="relu"),

    # Segunda capa densa con 255 neuronas y función de activación ReLU
    Dense(255, activation="relu"),

    # Capa de Dropout para evitar el sobreajuste (apaga el 50% de las neuronas aleatoriamente)
    Dropout(0.5),

    # Capa de salida con 2 neuronas (una por clase) y función de activación Softmax
    Dense(clases, activation="softmax")
])

# Compilar el modelo
CNN.compile(
    loss="sparse_categorical_crossentropy", # Función de pérdida adecuada para etiquetas enteras
    optimizer="adam", # Optimizador Adam para ajustar los pesos del modelo
    metrics=["accuracy"] # Métrica principal: precisión
)

# Entrenar el modelo
historico = CNN.fit(
    train_dataset, # Datos de entrenamiento (70%)
    validation_data=valid_dataset, # Datos de validación (20%)
    epochs=epocas,
    verbose=1
)
```



# Pruebas



Arrastra tu imagen  
de tu moneda aquí  
o haz clic para  
seleccionarla



Tu moneda  
vale  
5 pesos

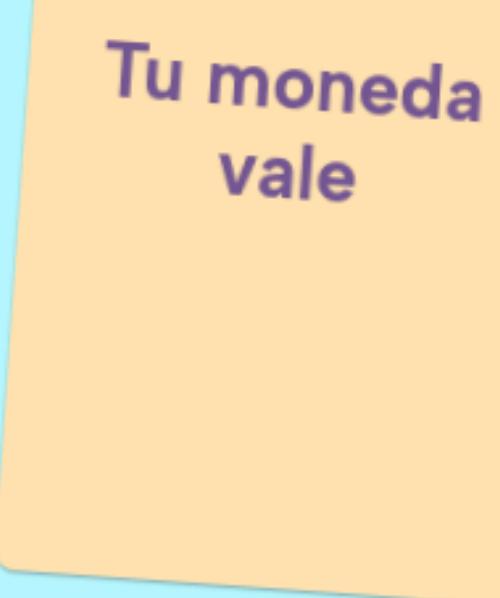
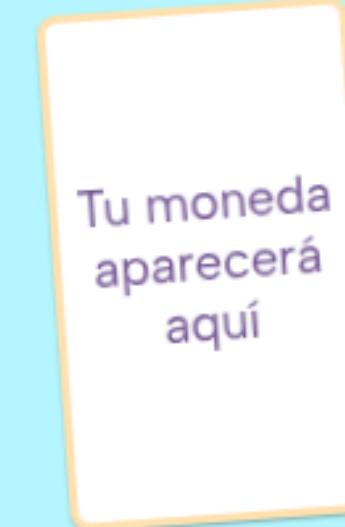
Descubrir

## Colección de monedas

Tienes 5 pesos en total



Arrastra tu imagen  
de tu moneda aquí  
o haz clic para  
seleccionarla



Descubrir

## Colección de monedas

Tienes

# Pruebas



Arrastra tu imagen  
de tu moneda aquí  
o haz clic para  
seleccionarla



Tu moneda  
vale  
5 pesos

Descubrir

## Colección de monedas

Tienes 5 pesos en total



Arrastra tu imagen  
de tu moneda aquí  
o haz clic para  
seleccionarla



Tu moneda  
vale  
1 peso

Descubrir

## Colección de monedas

Tienes 6 pesos en total

# Conclusiones

