



# UNIVERSIDAD DE GRANADA

## PinguiAR

### Entrega Final CUIA

Torres Ramos, Juan Luis

[g20596044@correo.ugr.es](mailto:g20596044@correo.ugr.es)

Universidad de Granada  
ETSIIT  
23/07/2024



## Contents

<b>1</b>	<b>Aplicación</b>	<b>3</b>
1.1	Inicio de la aplicación . . . . .	3
1.2	Flujo de la aplicación . . . . .	3
<b>2</b>	<b>Tecnologías Utilizadas en Computación Ubicua e Inteligencia Ambiental</b>	<b>7</b>
2.1	Realidad Aumentada . . . . .	7
2.2	Reconocimiento Facial . . . . .	7
2.3	Reconocimiento de Voz . . . . .	7
2.4	API de OpenWeather . . . . .	7
<b>3</b>	<b>Cambios con respecto a la Entrega 1</b>	<b>8</b>



## 1. Aplicación

A través del modelo 3D de tu pingüino, podrás experimentar un reconocimiento facial que le permitirá interactuar contigo. Mediante comandos de voz, podrás alimentarlo y jugar en su entorno virtual para aumentar su felicidad. Además, te proporcionará información útil como la hora, la temperatura y la fecha actual, adaptando sus respuestas según estos datos.

### 1.1 Inicio de la aplicación

Para comenzar, asegúrate de tener instalados todos los paquetes necesarios mediante pip. Dependiendo de tu configuración, podrías necesitar un entorno virtual dedicado para estos paquetes. Descarga un marcador aruco 1, el que viene en la carpeta marcador y tenlo a mano, a papel o en un dispositivo electrónico para poder visualizar al pingüino. Para el reconocimiento facial, asegúrate de tener una cámara web conectada a tu dispositivo y sube una foto de su cara a la carpeta `/media/faces`. Para el reconocimiento de voz, verifica que tengas un micrófono conectado y configurado en tu dispositivo. Por último, asegúrate de tener configurada la API de OpenWeather con tu clave correspondiente y la ciudad adecuada. Sigue los pasos en su página para obtener tu propia clave. Se configura en el archivo `main.py` y `escenario.py`.

Una vez hecho esto, asegúrate de ejecutar el archivo `main.py` utilizando Python 3.11.2 (en mi caso), de la siguiente manera:

```
python main.py
```

### 1.2 Flujo de la aplicación

La aplicación sigue un flujo específico para interactuar con el pingüino virtual mediante reconocimiento facial y comandos de voz. A continuación se detalla el proceso:

#### 1. Reconocimiento Facial:

Primero, la aplicación solicita el reconocimiento facial. Se abre una ventana con la cámara web y se pide al usuario que se posicione frente a ella. Una vez detectada la cara, si coincide con las fotos almacenadas en la carpeta `/media/faces`, se procede a mostrar el modelo 3D del pingüino. El reconocimiento facial es casi instantáneo.

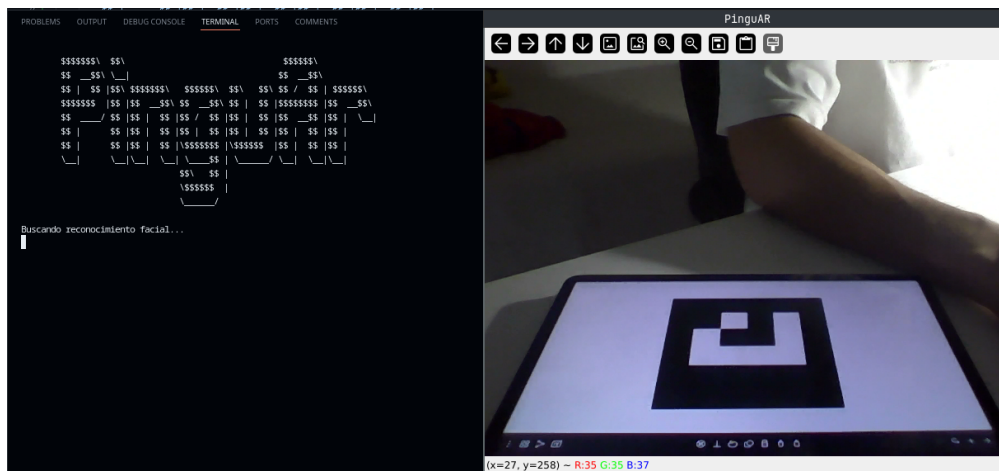


Figure 1: Reconocimiento Facial



## 2. Marcador Aruco:

Tras detectar la cara, se solicita al usuario que muestre el marcador Aruco número 1. Si el marcador es detectado, se muestra el modelo 3D del pingüino en la pantalla. La aplicación se estructura en tres escenarios:

## 3. Navegación entre Escenarios:

La navegación entre 3 escenarios se realiza mediante comandos de voz. Para ir al Escenario 1 se dice "UNO", para el Escenario 2 "DOS" y para el Escenario 3 "TRES". Cuando se está en el Escenario 2 o 3, se inicia el modo juego y se puede salir diciendo "SALIR", lo que permite volver a seleccionar otro escenario.

## 4. Detalles de los Escenarios:

- En el Escenario 1, el pingüino saluda al usuario por su nombre, muestra la hora, fecha, temperatura actual y su nivel de felicidad.

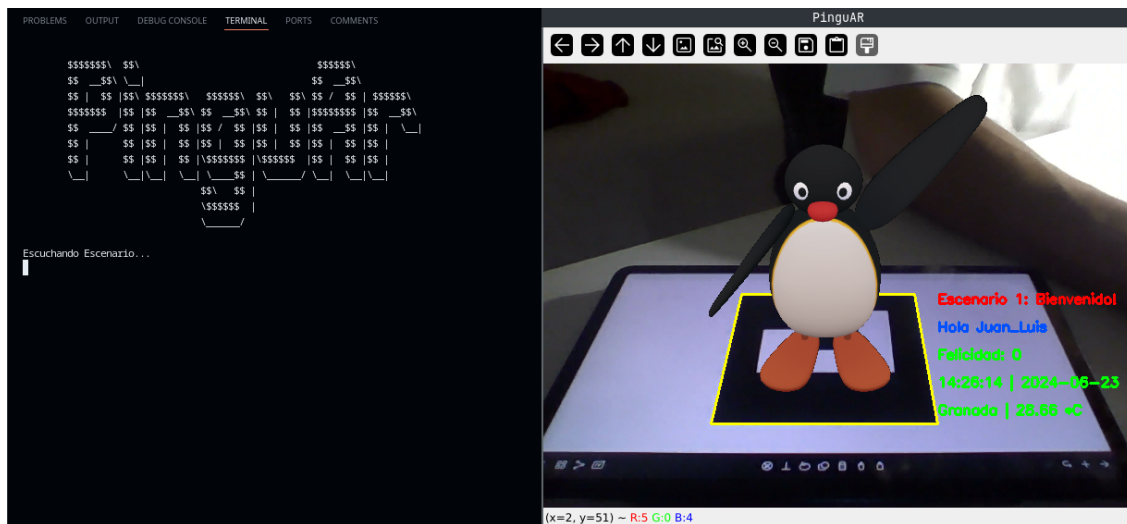
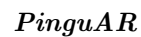


Figure 2: Escenario 1



The image is a composite of two screenshots. The left screenshot shows a terminal window with a dark background. At the top, there are tabs labeled 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is active), 'PORTS', and 'COMMENTS'. The terminal displays a large ASCII art penguin made of dollar signs (\$) and backslashes (\). Below the penguin, the text 'Escuchando Escenario...' is visible, followed by a prompt character and the text 'Escuchando: COME'. The right screenshot shows a tablet displaying a 3D penguin model. The penguin is black with a white belly and red feet. It is standing on a yellow rectangular base. To the right of the penguin, there is a list of statistics: 'Escenario 2: Dale de Comer', 'Sacado: 50%', 'Felicidad: 0', '14:29:00 | 2024-06-23', and 'Granada | 28.66 °C'. At the bottom of the tablet screen, there is a status bar showing '(x=326, y=214) ~ R:27 G:28 B:31'. Above the tablet, there is a toolbar with various icons for navigation and editing.

The image is a screenshot of a computer screen divided into two main sections. The top section is a terminal window with a dark background. It has a menu bar at the top with the following items: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is highlighted with a red underline), PORTS, and COMMENTS. The terminal displays a large ASCII art drawing of a penguin made of dollar signs (\$). Below the drawing, the text "Escuchando Escenario..." is visible, followed by a single vertical bar character "|".

The bottom section of the screen shows a game interface. At the top, the text "PinguAR" is displayed. Below it is a toolbar with icons for back, forward, up, down, home, search, and other controls. The main area of the interface shows a 3D rendering of a penguin standing on a tablet. The penguin is black and white with orange feet and a red beak. It is standing on a black square with a white crosshair. To the right of the penguin, there is a red text overlay that reads "Escenario 2: Dale de Comer". Below this, there is a blue text overlay that reads "El pinguino esta lleno!". Further down, there is a green text overlay that reads "Felicidad: 1". Below that, there is a green text overlay that reads "14:30:02 | 2024-08-23". At the bottom right, there is a green text overlay that reads "Granada | 28.65 °C".

5

- En el juego "¿Dónde está?" (Escenario 3), el pingüino se tapa la cara y el usuario debe preguntar "¿Dónde estás?". El pingüino reacciona al jugar con el, lo que aumenta su felicidad.

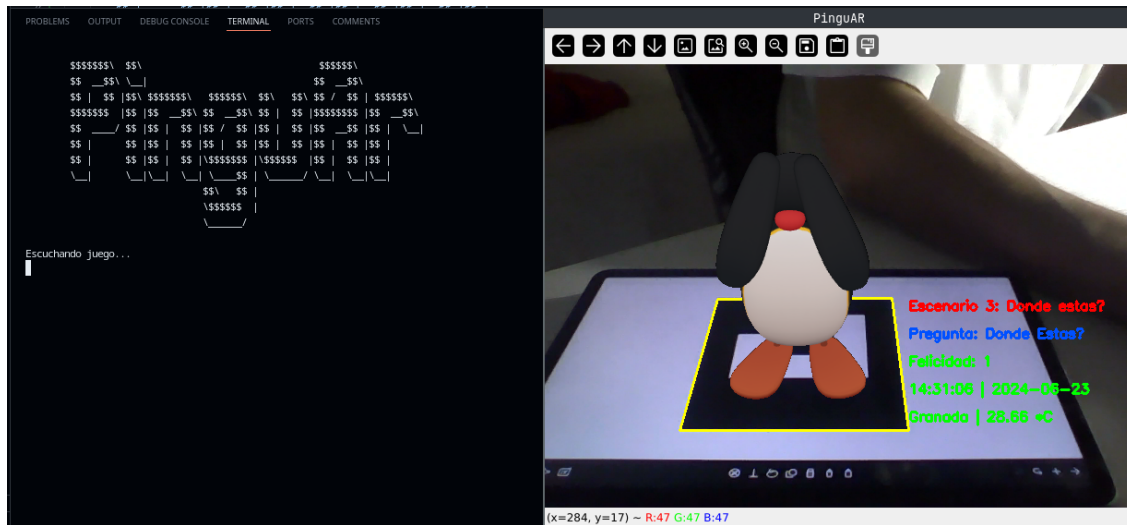


Figure 5: Escenario 3: ¿Dónde está?

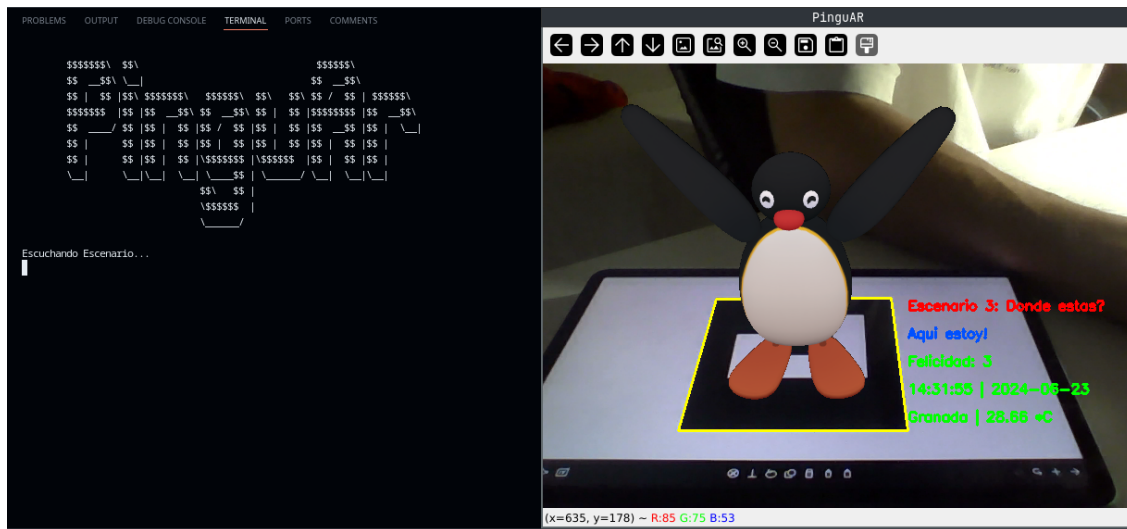


Figure 6: Escenario 3: Aquí esta



## 2. Tecnologías Utilizadas en Computación Ubicua e Inteligencia Ambiental

En el ámbito de la computación ubicua e inteligencia ambiental, se han utilizado diversas tecnologías para desarrollar esta aplicación, integrando características avanzadas como la realidad aumentada, el reconocimiento facial y el reconocimiento de voz.

### 2.1 Realidad Aumentada

1. **Marcadores ArUco:** Se ha implementado el uso de marcadores ArUco debido a su robustez y fiabilidad en la detección y el seguimiento de objetos en entornos de realidad aumentada. Los marcadores ArUco son una fácil identificación y posicionamiento de modelos 3D en el espacio real, proporcionando una experiencia inmersiva. En futuras versiones, se pueden añadir más marcadores para representar simultáneamente múltiples modelos, decoraciones en la habitación y otros pingüinos, mejorando la interactividad y personalización del entorno, por lo que su escalabilidad de cara al futuro es alta.
2. **Representación del Modelo 3D:** Para la representación del modelo 3D del pingüino, se ha utilizado la librería `pyrender 0.1.45` en combinación con `trimesh 4.4.0`. `Pyrender` facilita la renderización de modelos 3D de manera sencilla y eficiente, mientras que `trimesh` proporciona herramientas para trabajar con mallas y geometrías complejas. Sin embargo, estas librerías presentan ciertas limitaciones, como la falta de soporte para animaciones, lo cual restringe la capacidad de crear movimientos dinámicos y naturales en los modelos 3D, lo cual daría más inmersión de la aplicación. He creado los modelos con Blender, y los he exportado a formato `glb`, que es el formato que soporta `pyrender`. Se encuentran en la carpeta `media`

### 2.2 Reconocimiento Facial

Para el reconocimiento facial, se ha utilizado la librería `face-recognition 1.3.0`. Esta librería, basada en `dlib`, es conocida por su precisión y facilidad de uso en la detección y comparación de rostros. He seguido una guía detallada para su implementación, disponible en <https://www.youtube.com/watch?v=t12eEBFEHqM>.

### 2.3 Reconocimiento de Voz

Para el reconocimiento de voz, se han utilizado las librerías `vosk 0.3.45` y `PyAudio 0.2.14`.

1. **Vosk:** Vosk es una herramienta de reconocimiento de voz que funciona offline, lo que garantiza la privacidad y seguridad de los datos del usuario. Para su funcionamiento, se requiere la descarga de modelos específicos, en este caso, el modelo `vosk-model-small-es-0.42`, disponible en este enlace <https://alphacephei.com/vosk/models>.
2. **PyAudio:** PyAudio se utiliza para la captura de audio desde el micrófono. Facilita la integración de entrada de audio en aplicaciones Python, asegurando una comunicación efectiva entre el usuario y la aplicación.

### 2.4 API de OpenWeather

Para obtener información meteorológica precisa y actualizada, se ha utilizado la API de OpenWeather. Esta API permite que el pingüino interactúe de manera dinámica con nuestro entorno. Por ejemplo, se puede implementar una lógica condicional donde, si la temperatura desciende por debajo de un cierto umbral, el modelo del pingüino cambie su apariencia y muestre un mensaje relacionado con el frío. Igual con la hora, si es de noche, el pingüino se va a dormir, si es de día, se despierta, con lo que conlleva cambiar sus respectivos modelos y mensajes.

Estas tecnologías juntas hacen que la experiencia sea fluida y natural, siguiendo los principios de la computación ubicua. Permiten al usuario interactuar fácilmente con el entorno virtual del pingüino.



### 3. Cambios con respecto a la Entrega 1

Inicialmente consideramos la idea de introducir cambios cosméticos para el pingüino. Sin embargo, desde mi perspectiva, implementar múltiples modelos 3D requeriría mucho tiempo y complicaría la diferenciación visual entre el pingüino y sus cosméticos en la escena. Por esta razón, hemos decidido mantener al pingüino como el personaje principal y en lugar de eso, agregarle más funcionalidades.

No he podido implementar botones debido a limitaciones en `pyrender`. He visto que hay frameworks mas intuitivos y preparados para entornos de realidad virtual con Unity y Vuforia. Es más práctico realizar interacciones a través del reconocimiento de voz. Para representar el cuidado del pingüino a lo largo del tiempo, he optado por utilizar una variable de felicidad como métrica principal donde nuestro objetivo es maximizar la felicidad del pingüino jugando con el.

La idea de que el pingüino crezca podría implementarse cambiando su modelo por uno más grande según ciertos criterios como la hora del día. Sin embargo, la implementación de variaciones basadas en la hora y temperatura actual no está disponible actualmente debido a la necesidad de crear y añadir más modelos 3D, lo cual requiere más tiempo del que he podido dedicar hasta ahora. En este proyecto me he centrado más en las funcionalidades relacionadas con la representación del modelo, reconocimiento de voz y facial antes que la creación de modelos 3D en Blender.