

# Demostración de Proyectos de Realidad Aumentada y Virtual

Dr. Marco Aurelio Nuño Maganda

Universidad Politécnica de Victoria  
Laboratorio de Sistemas Inteligentes  
mnunom@upv.edu.mx

Marzo de 2024

# Contenido

- 1 Introducción a la programación móvil
- 2 Conceptos de Realidad Virtual y Aumentada
- 3 Modelado 3D
- 4 Prototipos de Realidad Aumentada
- 5 Prototipos de Realidad Virtual
- 6 Conclusiones

# Outline

- 1 Introducción a la programación móvil
- 2 Conceptos de Realidad Virtual y Aumentada
- 3 Modelado 3D
- 4 Prototipos de Realidad Aumentada
- 5 Prototipos de Realidad Virtual
- 6 Conclusiones

# Computadora y Programabilidad

## Computadora

Es una máquina (electrónica) programable\* que recibe y procesa datos para convertirlos en información útil. Contiene periféricos de entrada (para introducir datos) y salida (para mostrar resultados)

# Computadora y Programabilidad

## Computadora

Es una máquina (electrónica) programable\* que recibe y procesa datos para convertirlos en información útil. Contiene periféricos de entrada (para introducir datos) y salida (para mostrar resultados)

## Algoritmo

Conjunto finito de instrucciones para resolver una tarea específica

# Computadora y Programabilidad

## Computadora

Es una máquina (electrónica) programable\* que recibe y procesa datos para convertirlos en información útil. Contiene periféricos de entrada (para introducir datos) y salida (para mostrar resultados)

## Algoritmo

Conjunto finito de instrucciones para resolver una tarea específica

## Programación

El proceso de crear un software utilizando un lenguaje de programación (C, C++, Java, Python, Kotlin, etc)

# Computadora y Programabilidad

## Computadora

Es una máquina (electrónica) programable\* que recibe y procesa datos para convertirlos en información útil. Contiene periféricos de entrada (para introducir datos) y salida (para mostrar resultados)

## Algoritmo

Conjunto finito de instrucciones para resolver una tarea específica

## Programación

El proceso de crear un software utilizando un lenguaje de programación (C, C++, Java, Python, Kotlin, etc)

## Programa

Un conjunto de instrucciones que una computadora interpreta en una secuencia lógica para llevar a cabo una tarea en particular

# Computadora y Programabilidad

## Computadora

Es una máquina (electrónica) programable\* que recibe y procesa datos para convertirlos en información útil. Contiene periféricos de entrada (para introducir datos) y salida (para mostrar resultados)

## Algoritmo

Conjunto finito de instrucciones para resolver una tarea específica

## Programación

El proceso de crear un software utilizando un lenguaje de programación (C, C++, Java, Python, Kotlin, etc)

## Programa

Un conjunto de instrucciones que una computadora interpreta en una secuencia lógica para llevar a cabo una tarea en particular

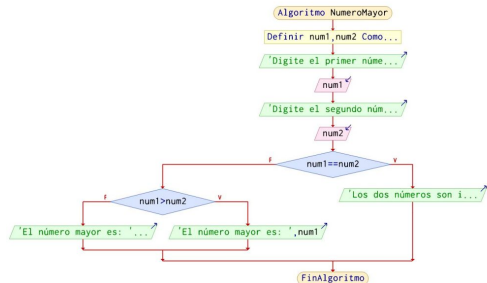


# Diagrama de Flujo

Un diagrama de flujo es una representación gráfica de un algoritmo, los pasos que los componen y la secuencia de ejecución de sus instrucciones

## Algoritmo

Diseñar un algoritmo para comparar dos numeros



# Codificación de un algoritmo en varios lenguajes de programación (Python - Kotlin)

## Codificación del Algoritmo en Python

```

1  def CompararNumeros(A,B):
2      if (A==B):
3          print ("Son Iguales ")
4      else:
5          if (A>B):
6              print ("A es mayor que B ")
7          else:
8              print ("B es mayor que A ")
9
10 As = input ("Escribe el primer numero (A): ")
11 A = int(As)
12 Bs = input ("Escribe el segundo numero (B): ")
13 B = int(Bs)
14 CompararNumeros(A,B)

```

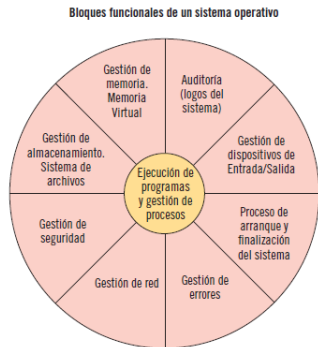
## Codificación del Algoritmo en Kotlin

```

1  package com.example.myapplication
2  fun CompararNumeros (A : Int, B: Int) {
3      if (A==B) {
4          print ("Son Iguales")
5      } else {
6          if (A > B) {
7              print ("A es mayor que B")
8          } else {
9              print ("B es mayor que A")
10         }
11     }
12 }
13 fun main() {
14     print("Escribe el primer numero (A):")
15     val As = readLine()!!
16     val A= Integer.parseInt(As)
17     print("Escribe el segundo numero (B):")
18     val Bs = readLine()!!
19     val B= Integer.parseInt(Bs)
20     CompararNumeros(A,B)
21 }

```

### Estructuración de los servicios del sistema operativo



# Sistemas Operativos para PCs



# Telefono Celular No-inteligente vs Telefono Celular Inteligente

## Teléfono No-inteligente

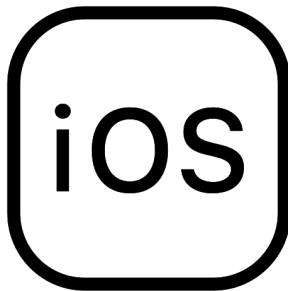
- Su funcionalidad principal era la comunicación (llamadas o mensajes) a través de la red celular (GSM)

## Teléfono inteligente

- Interfaz de entrada: Pantalla Touch (a color, de alta definición)
- Conexión a Internet: WiFi, GSM (4G o 5G)
- Comunicación con otros dispositivos: Bluetooth, NFC
- Cámaras (Frontal y Posterior)

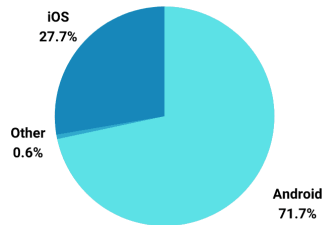


# Sistemas Operativos para Telefonos Inteligentes



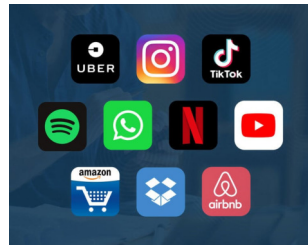
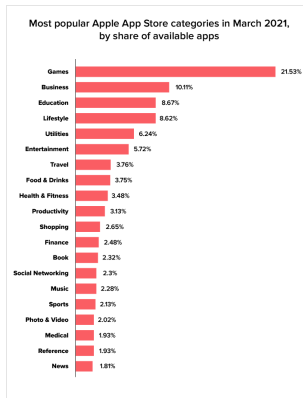
# Android

- Android es un sistema operativo móvil basado en Linux
- Principalmente orientado a dispositivos de pantalla táctil (Smartphone, tablets, smartwatches, etc)
- Fue desarrollado por Android Inc (Adquirida por Google en 2005)
- Vinculado con un grupo de empresas (HTC, Sony, Motorola, Samsung, LG, Lenovo, entre otras) para la creación de un SO común para sus dispositivos
- A la fecha (Q1 2023), los teléfonos con SO Android concentran mas del 70% del mercado global.



# Aplicaciones Móviles

- Ejecutadas en el teléfono
- La entrada de datos es mediante un teclado “virtual”
- El apuntador del raton es la pantalla
- Incluyen una interfaz de usuario gráfica (GUI)
- Es posible descargar miles de éstas en nuestros dispositivos



<https://www.netsolutions.com/insights/top-10-most-popular-apps-2018/>



# Android Studio

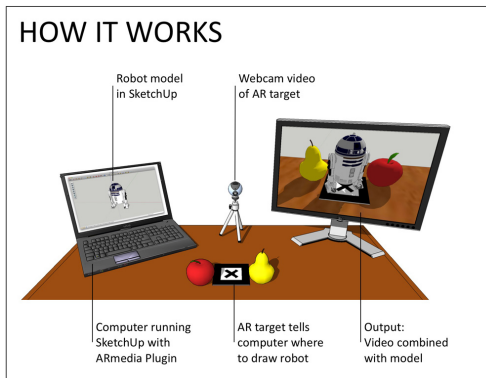
- Android Studio es un entorno oficial de desarrollo integrado (IDE) para el sistema operativo Android de Google
- La primera versión se libera en el año 2013, siendo el lenguaje de programación Java
- En 2019, se reemplaza el lenguaje oficial de desarrollo por Kotlin, aunque Java todavía es soportado
- Es gratis, se puede descargar e instalar en cualquier computadora sin importar el sistema operativo (Windows, Linux y MacOS)  
<https://developer.android.com>

# Outline

- 1 Introducción a la programación móvil
- 2 Conceptos de Realidad Virtual y Aumentada**
- 3 Modelado 3D
- 4 Prototipos de Realidad Aumentada
- 5 Prototipos de Realidad Virtual
- 6 Conclusiones

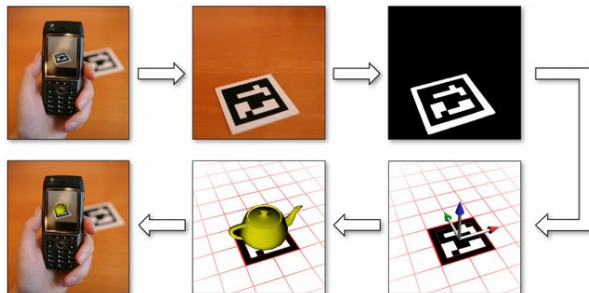
# Realidad Aumentada (AR)

- La AR es una experiencia que traslapa elementos digitales (modelados por computadora) con el mundo físico del usuario (adquirido mediante una cámara).
- Los elementos digitales se combinan con las vistas del mundo real.



# Pasos en la detección de marcadores de AR

- 1 Umbralización.
- 2 Detección del Marcador.
- 3 Estimación de Pose y Posición.
- 4 Empalme del modelo 3D.

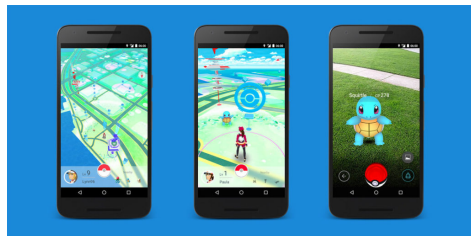


# Tipos de Aplicación de AR

- Basadas en Localización. Están basada en sensores GPS para determinar la ubicación del dispositivo para crear objetos AR
- Basadas en Visión – Utilizan una cámara, aunque también es posible incorporar sensores (compass, acelerómetros, giroscopios, etc).
  - Requieren Marcadores (Marker) – Localizan un patrón o marcador QR y renderizan un objeto 3D basado en su localización en el espacio real.
  - No requieren marcadores (Markerless)– Se emplean esquinas y puntos característicos del espacio real

# Aplicaciones de RA

- Aplicaciones principales: Arquitectura, Cosméticos, Contenido social, Marketing, Juegos, etc
- Houzz
- Pokemon Go

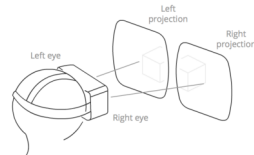
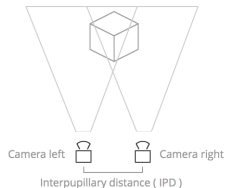


# Realidad Virtual - Antecedente histórico

- Un estereoscopio proporciona imágenes separadas para cada ojo mediante lentes individuales, donde cada imagen tiene una variante en el ángulo de captura y un desplazamiento horizontal.
- El cerebro de una persona con una percepción binocular normal de la profundidad al utilizar el estereoscopio “mezcla” ambas imágenes para crear una “ventana estereoscópica”



How to create stereoscopic 3D images







## Costos de Dispositivos HeadSets para VR

- Meta Quest 3 (499 USD)
- Sony PlayStation VR2 (599 USD)
- Meta Quest Pro (900 USD)
- Valve Index VR Kit (1350 USD)
- HTC Vive Pro 2 (1400 USD)



# Outline

- 1 Introducción a la programación móvil
- 2 Conceptos de Realidad Virtual y Aumentada
- 3 Modelado 3D**
- 4 Prototipos de Realidad Aumentada
- 5 Prototipos de Realidad Virtual
- 6 Conclusiones

# Gráficos por Computadora

- Es la rama de las CC encargada de la producción de imágenes y animaciones empleadas en juegos de computadora y simulaciones, en algunos casos incluyendo elementos fotorealísticos.
- Se requieren conocimientos de geometría, álgebra, cálculo, física, programación (estructura de datos).
- Existen librerías de bajo nivel (OpenGL) hasta frameworks (Unity, Unreal).



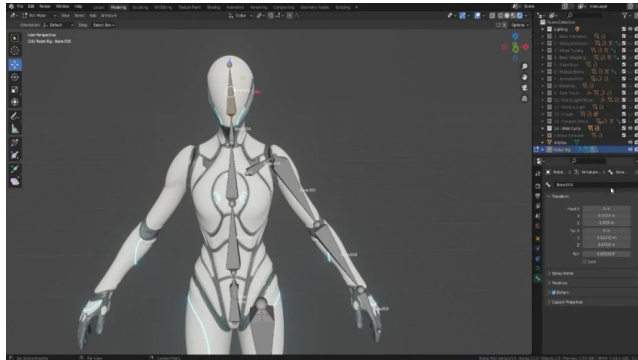
# ¿Qué es Blender?

- Blender es un software de modelado, animación y renderizado 3D de código abierto.
- Es utilizado en diversas industrias como animación, efectos visuales, videojuegos y diseño industrial.
- Soporta múltiples plataformas y cuenta con una comunidad activa de desarrolladores y artistas.



# Funcionamiento de Blender

- Basado en un flujo de trabajo no destructivo con modificadores y nodos.
- Permite modelado poligonal, esculpido digital, animación y simulaciones físicas.
- Motor de renderizado Cycles y Eevee para resultados realistas y en tiempo real.
- Compatibilidad con scripting en Python para automatización y personalización.



# Aplicaciones de Blender

- **Animación y cine:** Usado en películas y cortos animados como "Spring" y "Sintel".
- **Videojuegos:** Creación de modelos y animaciones para motores como Unity y Unreal Engine.
- **Arquitectura y diseño:** Visualización de espacios en 3D con realismo.
- **Impresión 3D:** Modelado y exportación de archivos para fabricación.



# ¿Qué es MediaPipe?

- Es una biblioteca de código abierto de Google para procesamiento de señales en tiempo real.
- Permite la implementación de modelos de Machine Learning en dispositivos móviles y de escritorio.
- Soporta múltiples plataformas como Android, iOS, y Python.

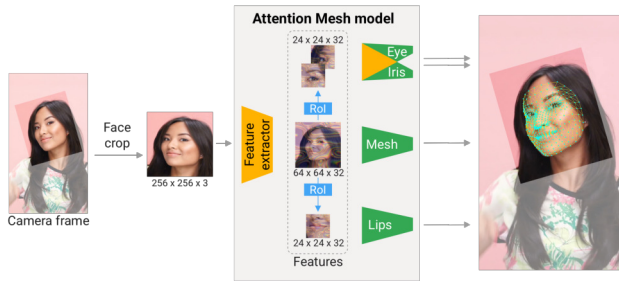
Flujo de Procesamiento:

- 1 Captura de datos desde una cámara o archivo.
- 2 Procesamiento mediante el gráfico de MediaPipe.
- 3 Obtención de resultados como coordenadas de puntos clave o segmentación.

# Soluciones preentrenadas de MediaPipe: Detección facial

Seguimiento de rasgos faciales para aplicaciones como realidad aumentada y reconocimiento facial.

- Utiliza una malla 3D con 468 puntos clave en el rostro.
- Permite el reconocimiento detallado de expresiones faciales y contornos.
- Funciona en tiempo real para aplicaciones como filtros de realidad aumentada y seguimiento facial.
- Proporciona una representación tridimensional sin necesidad de sensores de profundidad.





# Soluciones preentrenadas de MediaPipe: Detección de pose

Identificación de puntos clave en el cuerpo humano para análisis de movimiento.

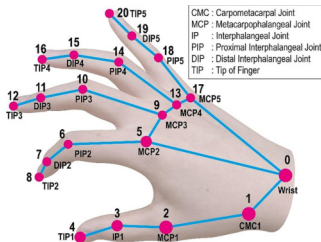
- Utiliza modelos de aprendizaje profundo para identificar 33 puntos clave en el cuerpo humano.
- Procesa imágenes en tiempo real desde la cámara o archivos de video.
- Genera un esqueleto virtual basado en las coordenadas de los puntos detectados.
- Se emplea en aplicaciones como deportes, rehabilitación y realidad aumentada.



## Soluciones preentrenadas de MediaPipe: Detección de manos

## Segmentación y seguimiento de manos en tiempo real para gestos y control por movimientos.

- Detecta 21 puntos clave en cada mano utilizando modelos de Machine Learning.
- Funciona en tiempo real, permitiendo el seguimiento preciso de los movimientos de la mano.
- Se utiliza en aplicaciones como control por gestos, realidad aumentada y comunicación en lenguaje de señas.
- Proporciona coordenadas tridimensionales sin necesidad de sensores adicionales.



# Outline

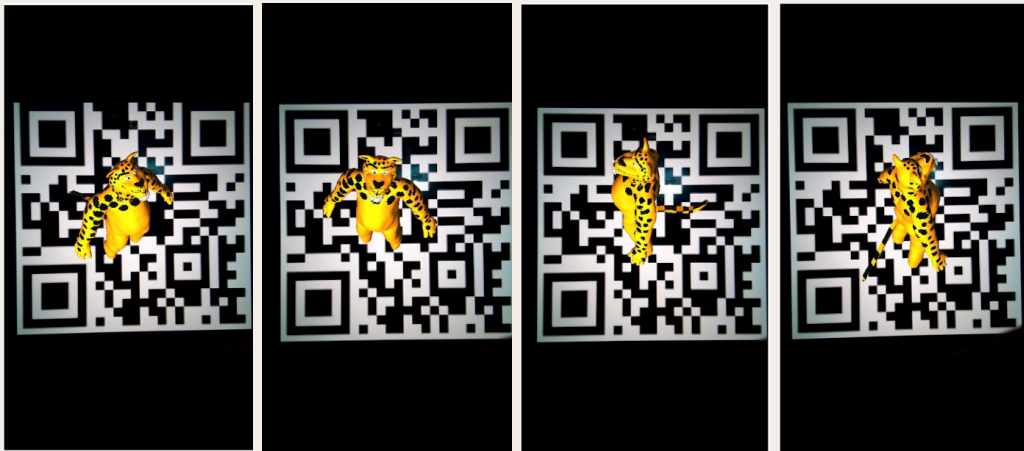
- 1 Introducción a la programación móvil
- 2 Conceptos de Realidad Virtual y Aumentada
- 3 Modelado 3D
- 4 Prototipos de Realidad Aumentada**
- 5 Prototipos de Realidad Virtual
- 6 Conclusiones

# Modelo 3D del Jaguar de la UPV sobre un código QR (1)

- Detección de Códigos QR
- Modelado de la mascota en 3D mediante el Software Blender
- Sobreposición del modelo 3D dependiendo de la posición del código QR



## Modelo 3D del Jaguar de la UPV sobre un código QR (2)



# *Aplicación móvil para el seguimiento de puntos caraterísticos de una mano y la visualización simultánea de su modelo 3D (1)*

## Propuesta

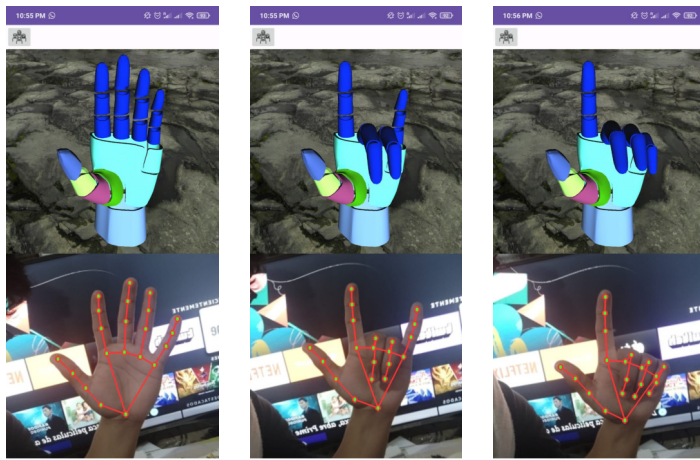
Un modelo 3D dinámico de una Mano controlado por el usuario implementado en un teléfono inteligente

- Utiliza la cámara del teléfono inteligente para detectar la mano (Libreria MediaPipe)
- Utiliza OpenGL ES para crear el entorno 3D de una mano humana modelada en Blender
- En base a los movimientos de los dedos del usuario, el modelo se actualiza

---

\* Kency Marisol Saldaña Martínez, César Zavala-López, Humberto Erubiel Ortega-Lujano, Jose Alan Gonzalez-Perales, and Anibal Gonzalez-Tovar. *Aplicación móvil para el seguimiento de puntos caraterísticos de una mano y la visualización simultánea de su modelo 3D*. Informe técnico proyecto de asignatura "Programación Móvil". Universidad Politécnica de Victoria, 2023

# Aplicación móvil para el seguimiento de puntos característicos de una mano y la visualización simultánea de su modelo 3D (2)



# Filtro de Cabeza de Jaguar UPV (1)

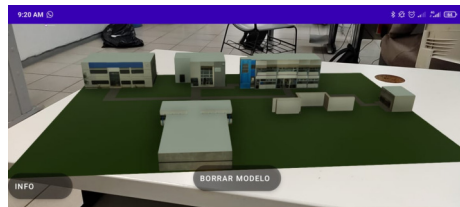
- La mascota institucional de la UPV es el Jaguar
- Se diseñó un filtro que detecta el rostro de una persona y superpone la imagen de un Jaguar





# Mapa de Realidad Aumentada UPV (1)

- Se diseñó una aplicación de Realidad Aumentada que muestra un mapa de la UPV
- Se modeló el mapa en Blender y se integró a la aplicación



\* Jorge Luis Charles-Torres, Paullette Monserrat Esparza-Gonzales, Juan Emmanuel Laurian-Mendoza, and Francisco Gael Sustaita-Reyna Humberto Erubiel Ortega-Lujano and. *Mapa de Realidad Aumentada UPV*. Proyecto Final de la Asignatura Programación Móvil. Universidad Politécnica de Victoria, 2023

# Outline

- 1 Introducción a la programación móvil
- 2 Conceptos de Realidad Virtual y Aumentada
- 3 Modelado 3D
- 4 Prototipos de Realidad Aumentada
- 5 Prototipos de Realidad Virtual**
- 6 Conclusiones

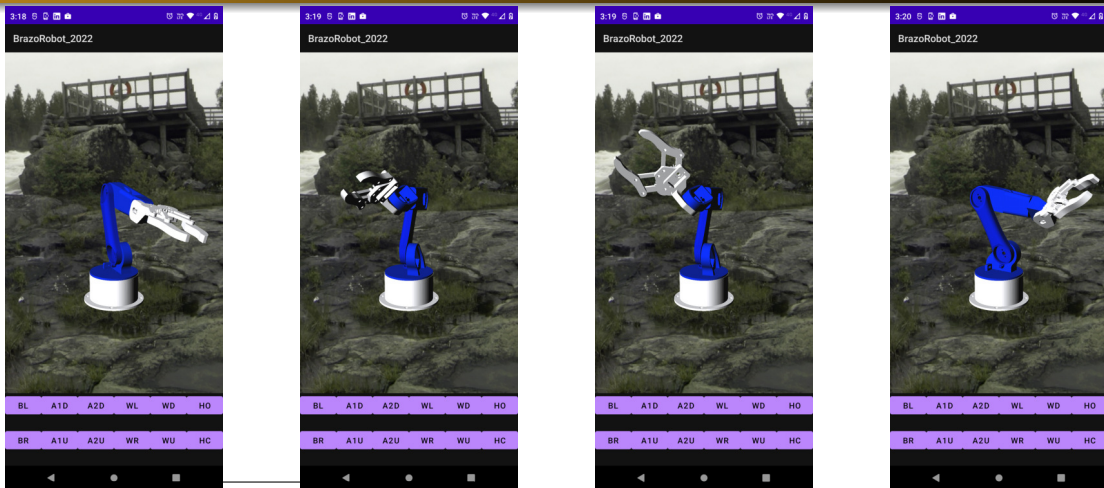
# Implementación de un Modelo 3D de un Brazo Robótico en una Aplicación Android (1)

- Se retoma un diseño previamente realizado para WebGL.
- Los componentes del robot son movidos mediante motores, y puede ser visto desde difentes perspectivas.



\* José Carlos Morin-García, Froylán Melquiades Wbario-Martínez, and Mariela Georgina Reyes-Fonseca. *Implementación de un Modelo 3D de un Brazo Robótico en una Aplicación Android*. Informe técnico proyecto de asignatura "Graficación por Computadora Avanzada". Universidad Politécnica de Victoria, 2019

# Simulación de brazo robótico en dispositivos móviles utilizando OpenGL ES (1)



\* Dino Tonatíuh Gutierrez-Martinez, Marly Yuleth Cantú-Charles, and Jesus Eduardo Uriegas-Ibarra an Marco Aurelio Nuño-Maganda. *Simulación de brazo robótico en dispositivos móviles utilizando OpenGL ES*. Informe técnico proyecto de asignatura "Programación Móvil". Universidad Politécnica de Victoria, 2022

# Recorrido UPV Virtual en teléfonos inteligentes

Demo incremental, que emplea OpenGL ES 2.0 (compatible con el 100% de los smartphones).

- Versión 1: Solo mundo virtual (no inmersivo). El usuario se movía con presionando teclas de la interfaz de usuario
- Versión 2: Mundo virtual inmersivo (integrado a unos lentes). El usuario movía la vista mediante los datos obtenidos por el sensor giroscopio del teléfono inteligente y avanzaba usando un manos libres alámbrico.
- Versión 3: Controlado por voz. El usuario se movía dentro del entorno mediante comandos de voz.
- Versión 3.5: Controlado mediante control de videojuegos (Bluetooth o USB)

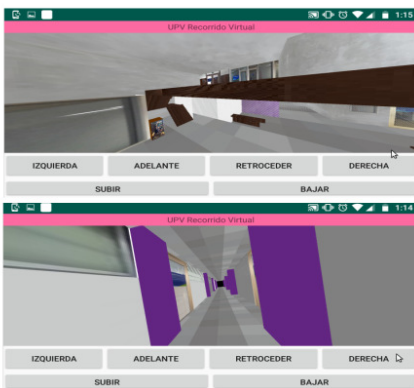
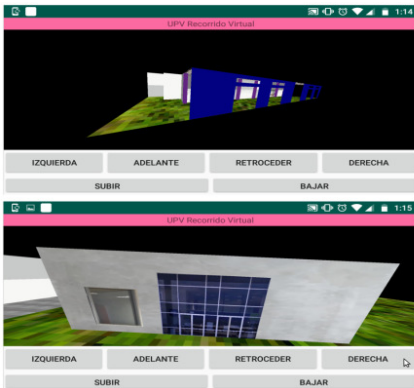
# Recorrido UPV en Android (1)

- La navegación es mediante botones (adelante, atrás, izquierda, derecha, subir, bajar)
- Una potencial mejora es mediante eventos de toque en pantalla



## Recorrido UPV en Android (2)

- Vistas de los diferentes edificios



## Aplicación UPV-Virtual (2)

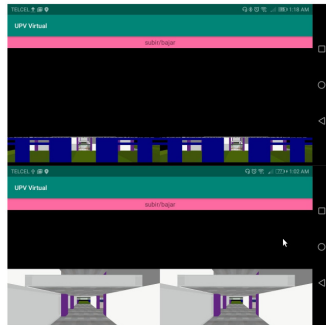
- Se extendió el demo 1.0 para generar una vista dual, requerida para su uso en conjunto con un armazón de VR.
- La vista cambia en base a lo obtenido por el giroscopio, y el movimiento se controla mediante el botón del manos libres.





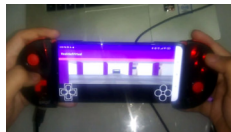
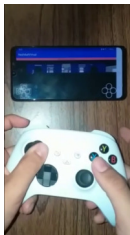
# Recorrido UPV Virtual en Android con Controles de Voz. (1)

- Se eliminó el uso del botón del manos libres para incluir comandos de voz
- La aplicación respondía a los comandos de voz, de tal forma que el usuario no debería mover nada



# Incorporación de Controles de Consola de Video Juego al Recorrido Virtual de la UPV (1)

- Se incorporaron varios controles de consolas de videojuego para la navegación.



\* Mario Alberto Delarbre-Quintanilla, Gerardo Loperena-Bustillos, Alejandro Lumbreras-Olvera, Jorge Eduardo Monita-Rodríguez, Christian Alejandro Saldaña-Calderon, Juan Ramiro Soto-Gómez, Claudia Corina Vásquez-Pérez, Hector Yahude Vallejo-Cepeda, Andrea Medrano-Salvidar, Julio Alberto Zuniga-Gallegos, and Darianna Carreon-Gomez. *Incorporación de Controles de Consola de Video Juego al Recorrido Virtual de la UPV*. Informe técnico proyecto de asignatura "Cómputo en Dispositivos Móviles". Universidad Politécnica de Victoria, 2021.

# Outline

- 1 Introducción a la programación móvil
- 2 Conceptos de Realidad Virtual y Aumentada
- 3 Modelado 3D
- 4 Prototipos de Realidad Aumentada
- 5 Prototipos de Realidad Virtual
- 6 Conclusiones**

# Conclusión

- Se presentaron los fundamentos de algoritmos, computadoras y lenguajes de programación
- Se vieron conceptos relacionados con la RA y la RV
- Se presentaron proyectos de RA y RV, principalmente desarrollados por estudiantes de la UPV (Ingeniería en Tecnologías de la Información y de Maestría en Ingeniería)
- Los asistentes pudieron interactuar con algunos de los demos presentados

# Agradecimiento

Se agradece al Consejo Tamaulipeco de Ciencia y Tecnologia (COTACYT) por la invitación a este evento  
Comentarios o Dudas: [mnunom@upv.edu.mx](mailto:mnunom@upv.edu.mx)