

Avance Preliminar del Proyecto: AirType

Información General del Proyecto

Nombre del Proyecto: **AirType**

Equipo:

- Richard Ruiz Soto
- Michelle Arauz Martínez
- Jimena León Bogantes
- Raúl Quesada Solórzano

2. Roles del Equipo

1. Líder de Hardware / Integración Electrónica

Jimena León

¿Qué hace?

- Diseña e implementa conexiones físicas (sensores, actuadores, ESP32, batería).
- Define el prototipo físico (estructura, impresión 3D, corte láser).
- Asegura la funcionalidad física y la seguridad.

Perfil Ideal:

- Conocimiento en electrónica básica/intermedia.
- Experiencia con soldadura, protoboard, circuitos.
- Creatividad para resolver problemas físicos.

2. Desarrollador de Software / Programador de Firmware

Richard Ruiz

¿Qué hace?

- Programa la lógica del ESP32.
- Integra sensores, actuadores y APIs (BLE/Wi-Fi).
- Asegura respuestas correctas del sistema.

Perfil Ideal:

- Dominio de C++.
- Experiencia con APIs y modelos de IA.
- Capacidad para estructurar código modular y claro.

3. Diseñador de Prototipo / Fabricación Digital

Raúl Quesada

¿Qué hace?

- Diseña la estética y ergonomía del dispositivo.
- Crea planos para impresión 3D / corte láser.
- Asegura una presentación visual atractiva y funcional.

Perfil Ideal:

- Dominio de software 3D (Fusion 360, SolidWorks).
- Creatividad con materiales y ensamblaje.
- Conocimientos básicos en estructuras físicas.

4. Líder de Documentación / Comunicación

Michelle Arauz

¿Qué hace?

- Documentación técnica (README, manuales).
- Coordina tiempos y prepara presentaciones.
- Garantiza que el equipo explique bien el proyecto.

Perfil Ideal:

- Excelente comunicación oral y escrita.
- Organización y visión global del proyecto.
- Capacidad para unir partes técnicas en una narrativa clara.

Resumen de Roles

Rol	Persona
Desarrollador de Software	Richard Ruiz
Líder de Hardware	Jimena León
Diseñador de Prototipo	Raúl Quesada
Líder de Documentación	Michelle Arauz

3. Descripción y Justificación

Problema

Los teclados físicos no son portátiles ni accesibles en muchos contextos. En entornos móviles o con limitaciones físicas, resultan ineficientes. AirType propone una solución sin contacto, cómoda y ligera, que mejora la accesibilidad y portabilidad.

Importancia y Contexto

AirType sigue la tendencia de interfaces gestuales y dispositivos wearables. Puede usarse en AR/VR, educación, medicina, industrias limpias o situaciones sin contacto físico, fomentando el aprendizaje interdisciplinario.

Usuarios/Beneficiarios

- Personas con movilidad reducida
- Profesionales en movimiento
- Estudiantes y educadores
- Makers y desarrolladores de tecnología
- Industrias médicas, tecnológicas y de manufactura
- Usuarios de AR/VR

4. Objetivos del Proyecto

Objetivo General

Desarrollar un teclado invisible por gestos integrado en un guante, que permita escribir sin contacto físico usando sensores, con transmisión de datos vía Bluetooth.

Objetivos Específicos

1. Diseñar un sistema de detección de gestos con sensores flexibles e IMU.
2. Programar el ESP32 para interpretar señales y traducirlas en texto.
3. Establecer comunicación Bluetooth eficiente con dispositivos externos.

5. Requisitos Iniciales del Sistema

1. Captar movimientos/flexión con sensores en un guante.
2. Procesar señales en tiempo real desde el ESP32.
3. Traducir gestos a caracteres mediante un algoritmo.
4. Enviar caracteres por Bluetooth.
5. Simular un teclado sin necesidad de superficie física.
6. Ser portátil, con batería recargable.
7. Compatible con múltiples plataformas (Android, Windows, etc.).

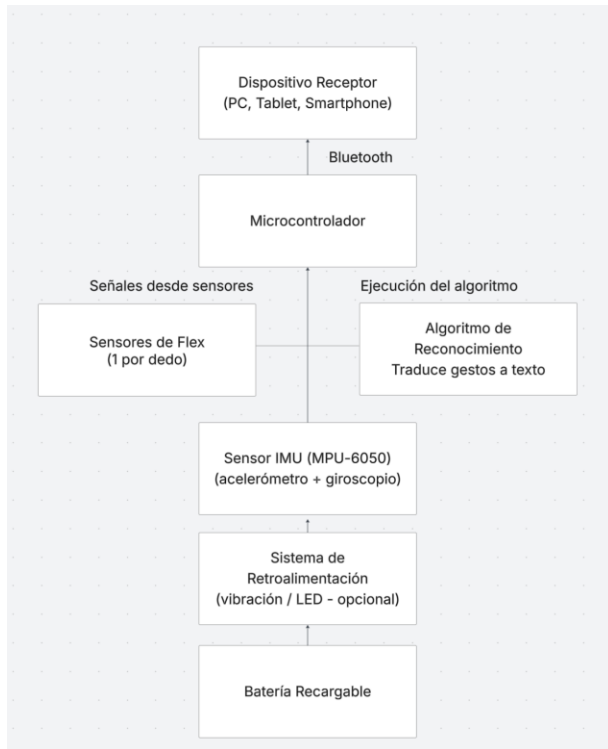
6. Diseño Preliminar del Sistema

Arquitectura Inicial

Módulos principales:

1. **Guante con sensores**
 - Sensores flex (x5)
 - IMU (MPU-6050)
2. **ESP32**
 - Procesa señales
 - Ejecuta algoritmo
 - Maneja Bluetooth
3. **Algoritmo de Reconocimiento**

- Traductor de gestos a texto
 - Posible integración de IA (TinyML o API)
4. **Dispositivo Receptor**
- Recibe caracteres
 - Visualiza o calibra gestos



Componentes Previstos

Componente	Función
ESP32	Microcontrolador central con BLE/Wi-Fi
Sensores Flex	Detectan flexión de los dedos
MPU-6050	Mide aceleración y rotación
Batería Li-Po	Fuente de energía portátil
Actuadores	Feedback con vibración/LED
TinyML o API	(Opcional) Reconocimiento de texto

Herramientas y Librerías

- Arduino IDE / PlatformIO
- Librerías: Adafruit_MPU6050, FlexSensor, Wire, BluetoothSerial
- Python/Processing (para visualización en PC)
- MIT App Inventor / Flutter (opcional)

7. Plan de Trabajo

Cronograma

Semana	Actividad Principal
1	Investigación y revisión de tecnologías

2	Compra de componentes, pruebas iniciales de ESP32
3	Integración de sensores y lectura en tiempo real
4	Desarrollo del algoritmo de reconocimiento
5	Traducción de gestos a texto vía Bluetooth
6	Interfaz en PC/móvil, calibración del sistema
7	Pruebas con usuarios, recolección de feedback
8	Optimización final y preparación de presentación

Riesgos Identificados

Riesgo 1: Lecturas inestables de sensores

Mitigación: Filtros digitales, calibración, tolerancia al error.

Riesgo 2: Problemas de conexión Bluetooth

Mitigación: Pruebas multiplataforma, librerías estables, reconexión automática.

8. Prototipos Conceptuales

Código de prueba básico (Aun pendiente por desarrollar)

cpp

CopyEdit

```
#include "BluetoothSerial.h"
```

```
BluetoothSerial SerialBT;
```

```
void setup() {
  Serial.begin(115200);
  SerialBT.begin("AirType_ESP");
  Serial.println("Bluetooth iniciado. Esperando conexión...");
}
```

```
void loop() {
  SerialBT.println("Prueba desde AirType");
  delay(2000);
}
```

¿Qué verifica este código?

- El ESP32 funciona correctamente
- Se conecta vía Bluetooth
- Envía texto al receptor cada 2 segundos

9. Evidencia Visual (*Agregar próximamente*)

- Bocetos del guante y su estructura
- Diagramas de conexión de sensores
- Capturas de lecturas en tiempo real
- Fotos del prototipo inicial

Nota: Este documento refleja el estado preliminar del proyecto y el enfoque del equipo en la etapa inicial. No se requiere que el sistema esté completamente funcional en este punto.