

PLAN DE TRABAJO DETALLADO – PROYECTO: “YOYO – ASISTENTE TERAPÉUTICO INTELIGENTE”

Duración: 8 semanas (3 encuentros semanales: martes, jueves y sábado)

Lugar de trabajo: Colegio Técnico / Polo Creativo / Trabajo remoto

Equipo: Prof. Cussi Nicolás + 2 estudiantes (6° año)

Herramientas de gestión: Trello + GitHub + Google Drive



Objetivo General

Diseñar y construir un asistente terapéutico robótico para adultos mayores, compuesto por un brazo robótico tipo Yarli, sensores, cámara y micrófono omnidireccional, controlado por comandos de voz y gestos, con una interfaz web accesible y simple.



Objetivos semanales (metodología ágil + Scrum simplificado)

Cada semana se trabajará con un **objetivo principal**, tareas asignadas en **Trello**, y gestión del código/documentos en **GitHub**.

Cada sábado se hace una **revisión semanal** del avance.



SEMANA 1 – Diagnóstico, planificación e investigación

Objetivo: comprender el problema, investigar y planificar el trabajo.

Actividades:

- Análisis de la consigna: ¿Qué es un asistente terapéutico? ¿Qué necesita un adulto mayor?
- Búsqueda de proyectos similares.
- Lectura de referentes: Yarli, asistentes de IA, brazos robóticos simples.
- Alta del repositorio en **GitHub**: estructura del proyecto, README, licencias.
- Creación del tablero **Trello**: columnas (Ideas, To Do, Doing, Done).

- Carga de los 8 objetivos semanales como listas.
- Roles: Programador 1, Electrónica 2, Documentación compartida.

Entregables:

- Trello y GitHub configurados.
 - Carpeta de campo: fundamentación, propósito, objetivos iniciales.
-



SEMANA 2 – *Diseño físico, materiales y prototipo en fibrofacil*

Objetivo: diseñar y construir la base física del brazo.

Actividades:

- Dibujo a mano/boceto del brazo robótico.
- Medidas reales: articulaciones, base, altura total.
- Corte de piezas en **fibrofacil de 2 mm** (con cutter/laser si es posible).
- Ensamble básico con pegamento o tornillos.
- Diseño modular (que se pueda desarmar si falla un servo).
- Pruebas de montaje de servos y soporte de cables.

Entregables:

- Fotos del brazo montado (sin electrónica).
 - Registro en carpeta de campo: bocetos + fotos + herramientas usadas.
-



SEMANA 3 – *Conexión de ESP32, servos y sensores*

Objetivo: integrar electrónica básica y probar control desde código.

Actividades:

- Conexión de la placa **ESP32** (placa + expansión).
- Montaje de servos SG90 o MG996.

- Pruebas con código Arduino para mover servos (grados, posiciones).
- Conexión del micrófono, sensor táctil y pantalla TFT (ojos).
- Configuración del entorno de desarrollo (Arduino IDE + librerías).
- Primeros commits en GitHub: código base + wiring diagram.

Entregables:

- Código de prueba subido a GitHub.
 - Carpeta de campo: diagrama de conexiones y explicación.
-

SEMANA 4 – Desarrollo de la app web de control (MVP)

Objetivo: construir una interfaz web sencilla para controlar el brazo.

Actividades:

- Elección de tecnologías web: HTML + JS (o App Inventor si van a mobile).
- Diseño UI con botones de control.
- Conexión vía WiFi entre app y ESP32 (WebSocket o HTTP).
- Pruebas de latencia y respuesta.
- Crear rama “web” en GitHub para separar código.

Entregables:

- App funcional + video mostrando control remoto.
 - Documentación: cómo se conecta la app con la ESP32.
-

SEMANA 5 – Cámara e inteligencia de gestos

Objetivo: integrar la cámara (ESP32-CAM o PiCam) y procesar imágenes.

Actividades:

- Conexión y configuración de la cámara.

- Streaming básico de video en red local.
- Análisis de gestos simples (levantar mano = activar saludo).
- Comunicación bidireccional: cámara ve → app acciona brazo.

Entregables:

- Video de demo de cámara funcionando.
 - Código de visión por computadora (si se usa OpenCV).
 - Registro de pruebas + dificultades.
-

SEMANA 6 – Control por voz y respuesta con sintetizador

Objetivo: que el usuario pueda hablarle al robot y recibir respuesta.

Actividades:

- Configuración del **micrófono omnidireccional**.
- Uso de Google Speech API o voz offline.
- Reconocimiento de comandos simples: "Hola", "Levanta brazo", etc.
- Respuesta por voz sintetizada: parlante + voz generada.
- Testing en ambiente real.

Entregables:

- Video del sistema reconociendo voz.
 - Lista de comandos disponibles.
 - Documentación: diagrama de entrada/salida del módulo de voz.
-

SEMANA 7 – Integración total y pruebas reales

Objetivo: unir todo en un sistema funcional y probar con usuarios.

Actividades:

- Unificación de todos los sistemas: app + voz + cámara + brazo.
- Ajuste de energía: batería, fuentes, consumo de servos.
- Pruebas con adultos mayores (simulados).
- Documentación del comportamiento: fallas, aciertos.
- Preparación del pitch y presentación.

Entregables:

- Prototipo completo funcionando.
- Video demostrativo.
- Documento con correcciones y aprendizajes.



SEMANA 8 – Documentación final y preparación para la feria

Objetivo: preparar carpeta, informe final y materiales de exposición.

Actividades:

- Redacción final del **informe técnico** (PDF).
- Cierre de **carpeta de campo** (con fotos, códigos QR, bitácoras).
- Preparación del **stand**: cartel, cartelera, logo, presentación Canva.
- Ensayo oral de presentación (tiempo máximo 7 minutos).
- Cierre del Trello y repositorio en GitHub con release final.

Entregables:

- Carpeta de campo lista para imprimir.
- Informe en PDF + QR al repositorio GitHub.
- Kit visual del stand.



Gestión en GitHub

- Branch principales: `main`, `dev`, `web`, `hardware`.
- Commits cada clase: siempre comentar lo hecho.
- Subir fotos, bocetos, PDF dentro del repositorio (`/docs`).
- Usar **Issues** para anotar errores detectados o mejoras.