

# Proyecto Algoritmos, Sistema Solar Interactivo Tecnológico para Educación STEM

**Objetivo General:** Desarrollar un sistema solar interactivo usando tecnologías modernas que combine automatización, visualización de datos y una maqueta interactiva, con fines educativos y de simulación.

## Sistema solar en maqueta con:

1. Motores que hacen girar los planetas
2. Luces LED que indican las características (Luna, Sol, Planetas, Asteroides, etc.)
3. Panel de control web para interactuar con los planetas (Encender luces, hacer girar, etc.)
4. Arduino UNO o ESP32
5. Servomotores
6. Node.js + Express
7. Html + Css + Js
8. Puerto COM para conectar Arduino y node.js

## Distribución del trabajo

Miembro	Rol	Actividades principales
Yuam	Backend Developer	Node.js, Express, Comunicación con Arduino. API Rest para manejar comandos y sensores
Goob	Frontend Developer	HTML, CSS, JavaScript, diseño de interfaz web con botones, feedback visual, conexión a API
Moisés	Electrónico	Cableado, conexión de LEDs y motores, codificación del Arduino, pruebas de circuito
Adrián	Diseñador y maquetador	Construcción física, planetas, sistema giratorio, estructura estable, organización de componentes físicos
José	Documentador y Analista	Documentación técnica, algoritmos, diagrama de flujo, manual del usuario, presentación gráfica, defensa teórica del proyecto

### Cronograma del trabajo

Semana	Actividades	Responsable
Semana 1	Investigación, planificación, lista de materiales, diagrama de flujo y algoritmo	Todos, liderado por documentador(José)
	Boceto de maqueta, diseño visual del Frontend, arquitectura del sistema (Node.js – Arduino)	Yuam, Goob, Adrián
Semana 2	Cableado y ensamblado básico: luces, motores	Moisés, Adrián
	Backend: Servidor Node.js + Conexión serial con Arduino	Yuam
	Frontend: Estructura de la página + prueba de botones	Goob
Semana 3	Integración Backend - Arduino	Yuam, Moisés
	Integración Frontend – API Rest	Yuam, Goob
	Finalización de maqueta y ajustes físicos	Maquetador
Semana 4	Pruebas finales, solución de bugs, presentación en PowerPoint	Todos
	Documentación final: algoritmo, diagrama de flujo y manual de usuario	José
	Video demostrativo y grabación de funcionamiento	José, Goob

### Funcionalidades del sistema

Función	Tecnología	Descripción
Movimiento de planetas	Arduino + Servomotores	Cada planeta puede girar individualmente
Encendido de LEDs	Arduino + LEDs	Un LED por planeta indica condiciones como atmosfera
Control Web	Node.js + HTML/CSS/JS	WebApp con botones por planeta
API Rest	Node.js + Express	Ruta para enviar comandos a Arduino
Comunicación Arduino <-> Web	Comunicación Serial (USB)	Arduino escucha comandos y los ejecuta

### OPCIONAL EN DADO CASO FUESE NECESARIO

- **Base de datos ligera** con la info de cada planeta que se consulta desde la web.
- **Dashboard educativo** con gráficas simples (ej. tamaño relativo de planetas).

### Presupuesto:

#### Opciones de Arduino:

- Página oficial 45.24\$ - 352Q aproximadamente; 8 a 14 días de entrega
- La electrónica 299Q; 126 existencias, entrega rápida
- Tettsa 450Q; 6 disponibles

### Sensores y componentes:

Componente	Uso	Aprox. Precio
LEDs	Luz por planeta	10-20 Q
Resistencias 220	Limitar la corriente del LED	Q5
Servomotores SG90	Girar planetas	15-25 Q cada uno
Protoboard + Jumpers	Conexión sin soldadura	20Q
Fuente 9V o Powerbank	Alimentación	20-40 Q

### Softwares de trabajo:

Documento	Herramienta	Justificación
Informe técnico, manual	Microsoft Word	Más presentable y fácil exportación a PDF
Algoritmo y Diagrama de flujo	Draw.io	Gratis, online, Exportación a PNG/SVG/PDF
Código y control de versiones	Github + Git + Visual Studio Code	Profesional, backups, colaboración
Presentación	PowerPoint, Canva o Google slides	Accesible, buen diseño visual
Notas y tareas del grupo	Notion	Organización de tareas, fechas y pendientes

### Herramientas y software:

Área	Herramienta	Listo	Usos
Programación Arduino	Arduino IDE	<input type="checkbox"/>	Subir código al Arduino
Backend/API Rest	Node.js + Express	<input type="checkbox"/>	Crear servidor y comunicar con Arduino
Frontend	HTML/CSS/JS + VsCode	<input type="checkbox"/>	Página de control del sistema solar
IDE principal	Visual Studio Code	<input type="checkbox"/>	Código
Repositorio	Github	<input type="checkbox"/>	Guardar código, documentación e historial
Diagramas	Draw.io	<input type="checkbox"/>	Algoritmo, flujo y arquitectura
Documentos	Word	<input type="checkbox"/>	Informe, documentación y manual de usuario
Presentación	Canva / PowerPoint	<input type="checkbox"/>	Para defender el proyecto
Gestión del proyecto	Notion	<input type="checkbox"/>	Asignar tareas y organización