

# **Universidad Autónoma de Baja California**



## **Facultad de Ciencias de la Ingeniería y Tecnología**

### **Propuesta De Dispositivos Móviles.**

#### **Alumnos:**

Cruz Hernandez Juan Carlos

Medrano Barrera Victor Yahir

Rodriguez Cardenas Marshall

Cortes Hernández Baku Lenin

**Nombre de la Profesora:** Jose Manuel Muños Contreras

**Materia:** Dispositivos Móviles

**Grupo:** 564

# **Propuesta del Proyecto: Math AI Studio**

## **Resumen Ejecutivo**

Math AI Studio es una aplicación móvil desarrollada en Flutter que combina una potente graficadora matemática (2D y 3D) con un asistente de Inteligencia Artificial integrado. Su objetivo es democratizar la comprensión de las matemáticas complejas, permitiendo a los estudiantes no solo visualizar ecuaciones, sino recibir explicaciones contextuales instantáneas sobre el comportamiento de las funciones, dominios, rangos y puntos críticos, todo desde su dispositivo móvil.

## **Problema y Oportunidad**

- **El Problema:** Las herramientas actuales (como GeoGebra o Desmos) son excelentes para graficar, pero **no explican**. Los estudiantes ven una línea roja en la pantalla pero no entienden por qué la curva se comporta así. Existe una brecha entre "ver el resultado" y "entender el proceso".
- **La Oportunidad:** Aprovechar el auge de los LLMs (Modelos de Lenguaje) para actuar como un tutor personal. Integrar la capacidad de cálculo simbólico con la capacidad explicativa de la IA en una sola interfaz móvil es un nicho desatendido en la educación STEM.

## Solución: Math AI Studio

Nuestra solución es una aplicación "todo en uno" que ofrece:

1. **Editor Matemático Intuitivo:** Entrada de ecuaciones naturales (ej.  $\sin(x) + x^2$ ).
2. **Visualizador Híbrido:** Motor de renderizado optimizado para gráficos cartesianos en 2D y superficies en 3D.
3. **Tutor IA (Context-Aware):** Un chat inteligente que "ve" la gráfica del usuario y responde dudas como "*¿Por qué la gráfica corta en cero?*" o "*¿Cuál es el dominio de esta función?*" sin costos excesivos por consulta.

## Público Objetivo

- **Estudiantes Universitarios:** Carreras de Ingeniería, Física, Economía y Matemáticas que requieren análisis de funciones multivariadas.
- **Estudiantes de Bachillerato:** Para comprender los fundamentos del cálculo y la trigonometría.
- **Docentes:** Como herramienta de apoyo visual en el aula para explicar conceptos abstractos.

## Modelo de Negocio

- **Freemium (B2C):**
  - *Gratis*: Graficación 2D ilimitada, Graficación 3D básica, 10 consultas a la IA por día (usando modelo local o cacheado).
  - *Premium*: Graficación 3D de alta resolución, exportación a LaTeX/PDF, consultas ilimitadas a la IA con razonamiento avanzado.
- **Publicidad No Intrusiva:** Anuncios visuales solo al momento de guardar o exportar proyectos, nunca durante el estudio.

## Limitaciones y Soluciones

- **Limitación:** El renderizado 3D en tiempo real consume mucha batería y recursos de GPU en móviles de gama baja.
  - *Solución*: Uso de librerías optimizadas como [ditredi](#) que simplifican polígonos y control de cuadros por segundo (FPS) dinámico.
- **Limitación:** La IA puede "alucinar" resultados matemáticos numéricos.
  - *Solución*: Sistema híbrido. La app calcula los números exactos con el motor matemático ([math\\_expressions](#)) y la IA solo se usa para *explicar* la teoría, no para calcular el resultado, reduciendo el error.

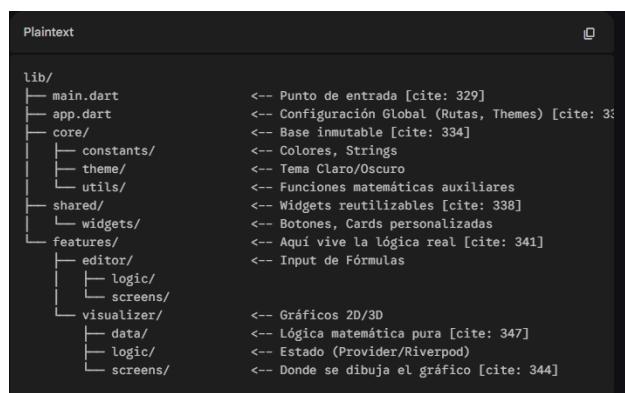
## Riesgos y Mitigaciones

- **Riesgo:** Dependencia de conexión a internet para la IA más potente.
  - *Mitigación:* Implementación de modelos "On-Device" (Gemini Nano o TensorFlow Lite) para explicaciones básicas offline.
- **Riesgo:** Curva de aprendizaje de la interfaz 3D.
  - *Mitigación:* Interfaz de usuario minimalista basada en gestos (pinch-to-zoom, rotación con un dedo) familiar para usuarios de móviles.

## Arquitectura Técnica

Se utilizará una arquitectura "**Feature-First**" (basada en características) con separación estricta de responsabilidades:

- **Capa de Presentación (UI):** Widgets de Flutter (Screens).
- **Capa de Lógica de Negocio (BLoC/Provider):** Gestión del estado de la ecuación y la respuesta de la IA.
- **Capa de Datos:** Repositorios que conectan con el motor matemático local y la API de IA.



```
PlainText
```

```
lib/
  main.dart          --- Punto de entrada [cite: 329]
  app.dart           --- Configuración Global (Rutas, Themes) [cite: 329]
  core/              --- Base inmutable [cite: 334]
    |--- constants/  --- Colores, Strings
    |--- theme/      --- Tema Claro/Oscuro
    |--- utils/       --- Funciones matemáticas auxiliares
  shared/             --- Widgets reutilizables [cite: 338]
    |--- widgets/   --- Botones, Cards personalizadas
  features/           --- Aquí vive la lógica real [cite: 341]
    |--- editor/    --- Input de Fórmulas
      |--- logic/   --- Gráficos 2D/3D
      |--- screens/ --- Lógica matemática pura [cite: 347]
  visualizer/         --- Estado (Provider/Riverpod)
    |--- data/     --- Donde se dibuja el gráfico [cite: 344]
    |--- logic/
```

## Stack Tecnológico

- **Lenguaje:** Dart 3.0+ (Tipado fuerte y nulo seguro).
- **Framework:** Flutter (Multiplataforma Android/iOS/Web).
- **Gestión de Estado:** [provider](#) (Sencillo y escalable para este alcance).
- **Librerías Clave:**
  - [math\\_expressions](#): Para el parsing y evaluación de fórmulas.
  - [fl\\_chart](#): Para gráficos 2D interactivos.
  - [ditredi](#): Para visualización 3D ligera.
  - [google\\_generative\\_ai](#): SDK para conexión con Gemini.
- **Backend (Opcional):** Firebase para autenticación de usuarios y guardado de historial de gráficas en la nube.