



C mo poner a trabajar Figma   Lovable   Google Stitch

Descripci n

  Qu  es cada uno?

- **Figma**: plataforma para **dise o UI/UX y prototipado** colaborativo (componentes, Auto Layout, variantes, prototipos).
- **Lovable**: plataforma para **construir apps** con ayuda de IA (**auth, BD, CRUD, deploy**) partiendo de descripciones y prompts.
- **Google Stitch**: experimento de Google orientado a **generar interfaces y c digo** con IA, muy alineado con **Material Design** y el ecosistema Google (p. ej., Web/Flutter/Compose). No es un reemplazo de Figma; se sit a entre el dise o y el c digo.

Comparaci n r pida

Criterio	Figma	Lovable	Google Stitch
Prop�sito	Dise�ar y prototipar UI/UX	Construir un MVP o panel interno funcional	Generar UI + c�digo con IA (�nfasis Material/Google)
Entregable	Mockups, design system, prototipos	App corriendo: login, DB, CRUD, rutas	Pantallas y componentes de UI + snippets/proyecto base
Colaboraci�n	Multijugador, comentarios, handoff	Equipo itera con IA sobre features/datos	Enfocado en generar y ajustar UI con IA
Curva de aprendizaje	Baja�media (Auto Layout, componentes)	Baja para empezar; media al personalizar l�gica	Baja si sigues patrones Material; media si sales del ��happy path��

Criterio	Figma	Lovable	Google Stitch
Personalización visual	Máxima (tokens, variantes, motion)	Depende de cómo estilos y componentes se apliquen	Alta cuando te mantienes en Material; fuera de eso, más esfuerzo
Rol ideal	Diseñadores, PMs, devs	Founders, PMs técnicos, devs lean	Devs/UI engineers que quieren código de UI rápido
Licenciamiento	Free + planes	Free + planes	Experimental/preview (cambios frecuentes)

¿Cuándo usar cada uno?

- Necesitas validar la experiencia y la marca → **Figma**.
- Necesitas poner algo usable en manos de usuarios YA (auth, BD, formularios, dashboards) → **Lovable**.
- Quieres acelerar la capa de UI en **código** (especialmente si vives en ecosistema Google/Material) → **Stitch**.

Mezclas ganadoras (workflows)

1. Figma → Lovable

- Diseñas el flujo, componentes y estados en Figma.
- Construyes en Lovable respetando tokens y specs.
- Resultado: app **funcional** con **UI consistente**.

2. Figma → Stitch → Repo → (Lovable opcional)

- Prototipas en Figma, defines tokens/espaciados.
- Usas Stitch para **materializar UI en código** (p. ej., React/Flutter/Compose).

- Integras ese código en un repo y conectas lógica/datos (con Lovable o stack propio).
- Resultado: **código de UI** de alta calidad + lógica a medida.

3. Lovable + Figma (loop de refinamiento)

- Levantas MVP en Lovable.
 - Si la UI no está al nivel visual, la rediseñas en Figma y aplicas los estilos/estructura.
 - Resultado: mejoras rápidas sin frenar el producto.
-

Recomendación para Geniales.co (práctica y eficiente)

- Si el proyecto es comercial/corporativo y multi-plataforma:

1. Define design system en **Figma** (tokens, componentes, variantes).
2. **Stitch** para obtener UI en **código** alineado a Material, si tu target es Android/Web Material o Flutter.
3. Integra lógica (auth, datos, reglas) con **Lovable** o tu backend propio.

- Si el objetivo es validar negocio en días:

1. Wireframes rápidos en **Figma** (lo máximo).
 2. Construcción directa en **Lovable** (auth, BD, CRUD, paneles).
 3. Pulir visual con **Figma** cuando haya feedback real.
-

Puntos finos a tener en cuenta

- **Figma** te da control total de la **identidad visual** (marca, spacing, tipografía, motion).
- **Lovable** te da **time-to-value**: autenticación, base de datos, paneles en horas.
- **Stitch** brilla si adoptas **Material Design** y el ecosistema Google; acelera la **UI en código** y reduce la brecha diseño-desarrollo.
- Para **SEO/marketing** y **apps data-driven**, la pareja **Figma + Lovable** es hoy la más directa para clientes que necesitan resultados rápidos.
- Para **productos Android/Flutter/Compose** con guÃa Material estricta, **Figma + Stitch** produce UI limpia y mantenible desde el día 1.

Autor
alvaroabril

default watermark