

PLAN DE IMPLEMENTACIÓN: COMPANIONS ANIMADOS Y EXPRESIVOS






OBJETIVO FINAL

Transformar los companions de Hogara en personajes vivos, expresivos y naturales que:

- Hablan con voces dulces, tiernas y expresivas (no robóticas)
- Se animan de forma sutil y natural (parpadeo, boca, movimientos leves)
- Responden al toque directo sin abrir chat
- Permiten elegir entre modo voz y modo texto en ajustes

PRECAUCIONES DE SEGURIDAD

Antes de empezar:

1.  **Crear backup completo** del código actual
2.  **Checkpoint de seguridad** antes de cualquier cambio
3.  **Mantener PNG originales** sin modificar
4.  **Implementación incremental** (fase por fase, con pruebas)
5.  **Reversibilidad total** - poder volver atrás en cualquier momento

Archivos que NO se modificarán:

- `/public/images/companions/*.png` (imágenes originales)
- Estructura de base de datos existente
- Layout general de la aplicación

Archivos que se modificarán (con backup):





- `components/companion/floating-companion.tsx` (interacción)
- `components/companion/improved-voice-chat.tsx` (voz mejorada)
- `components/companion/companion-settings.tsx` (preferencias)
- `lib/voice-config.ts` (configuración de voces)
- `hooks/useVoiceChat.ts` (lógica de voz)

FASE 1: VOCES NATURALES Y EXPRESIVAS

1.1 Integración de TTS Avanzado

Opciones de implementación:

Opción A: Abacus.AI LLM APIs (Recomendada)

-  Ya tienes acceso
-  No requiere configuración externa
-  Integración simple
-  Sin costos adicionales

- ⚠ Verificar calidad de voces en español

Opción B: ElevenLabs

- ✓ Voces ultra realistas y expresivas
- ✓ Control total de emociones
- ✓ Clonación de voces posible
- ⚠ Requiere API key
- ⚠ Costos por uso

Opción C: Play.ht

- ✓ Buena calidad en español
- ✓ Voces expresivas
- ✓ Precios competitivos
- ⚠ Requiere API key

Decisión necesaria: ¿Qué servicio prefieres?

1.2 Configuración por Personaje

Cada companion tendrá:

```
{
  voiceId: string,           // ID de voz específica
  speed: number,             // Velocidad (0.8-1.2)
  pitch: number,             // Tono (0.8-1.2)
  emotion: string,           // "happy" | "calm" | "playful" | "wise"
  expressiveness: number     // Nivel de expresividad (0-1)
}
```

Personalidades definidas:

- **Lumi:** Voz dulce, optimista, juvenil (pitch: 1.15, speed: 1.05, emotion: cheerful)
- **Nimbo:** Voz serena, tranquila, maternal (pitch: 0.95, speed: 0.9, emotion: calm)
- **Fabel:** Voz cálida, amigable, juguetona (pitch: 1.1, speed: 1.0, emotion: playful)
- **Hada:** Voz etérea, melodiosa, mágica (pitch: 1.2, speed: 0.95, emotion: magical)
- **Elfo:** Voz sabia, pausada, reflexiva (pitch: 0.9, speed: 0.85, emotion: wise)
- **Draguito:** Voz enérgica, valiente, animada (pitch: 1.05, speed: 1.1, emotion: excited)
- **Unicornito:** Voz esperanzadora, dulce, inspiradora (pitch: 1.15, speed: 0.95, emotion: hopeful)
- **Sprig:** Voz natural, tranquila, conectada (pitch: 1.0, speed: 0.9, emotion: grounded)
- **Human:** Voz empática, comprensiva, cálida (pitch: 1.05, speed: 1.0, emotion: empathetic)

1.3 Implementación de Expresividad Emocional

Detección de emociones en conversación:

- Analizar el contexto emocional del mensaje del usuario
- Ajustar tono y velocidad dinámicamente
- Añadir pausas expresivas
- Variar inflexiones según emoción detectada

Estados emocionales:

- alegre : velocidad +10%, pitch +5%
- triste : velocidad -15%, pitch -5%
- sorprendido : velocidad +15%, pitch +10%

- **pensativo** : velocidad -10%, pausas más largas
- **emocionado** : velocidad +20%, énfasis en palabras clave



FASE 2: ANIMACIONES SUTILES Y NATURALES

2.1 Estrategia de Animación (SIN modificar PNG)

Técnica: CSS animations + Framer Motion sobre imágenes completas

Animaciones implementadas:

1. Parpadeo Natural

```
// Parpadeo aleatorio cada 3-6 segundos
{
  animation: "blink",
  duration: 0.15,
  interval: random(3000, 6000)
}
```

2. Movimiento de Boca (Sincronizado con voz)

```
// Durante TTS, animar escala vertical de zona de boca
{
  scaleY: [1, 0.95, 1, 0.92, 1],
  transformOrigin: "center 75%",
  synced: true // Sincronizado con audio
}
```

3. Respiración Sutil

```
// Movimiento de cuerpo muy sutil
{
  scale: [1, 1.02, 1],
  duration: 4,
  repeat: Infinity,
  ease: "easeInOut"
}
```

4. Inclinación Leve (Reacción al toque)

```
// Al hacer hover o click
{
  rotate: [-2, 0, 2, 0],
  duration: 0.8,
  ease: "easeInOut"
}
```

5. Expresiones Faciales (Con filtros CSS)

```
/* Alegre */
filter: brightness(1.15) saturate(1.2) hue-rotate(10deg);

/* Triste */
filter: brightness(0.85) saturate(0.8) hue-rotate(-10deg);

/* Sorprendido */
filter: brightness(1.25) saturate(1.4) contrast(1.1);

/* Pensativo */
filter: brightness(0.95) saturate(0.9);
```

2.2 Implementación Técnica

Archivo nuevo: components/companion/animated-companion-character.tsx

Este componente:

- ☒ Renderiza el PNG original sin modificarlo
- ☒ Aplica animaciones CSS y Framer Motion
- ☒ Sincroniza boca con audio (Web Audio API)
- ☒ Cambia expresión según emoción
- ☒ Mantiene todos los efectos visuales originales

Estructura:

```
<AnimatedCharacter
  imagePath={companionImage}
  emotion={currentEmotion}
  isSpeaking={isSpeaking}
  isListening={isListening}
  onTouch={() => activateVoice()}
/>
```



FASE 3: INTERACCIÓN DIRECTA (Sin Chat Visible)

3.1 Nuevo Modo de Interacción

Flujo actual:

1. Click en personaje → Abre chat → Click en tab voz → Habla

Flujo nuevo:

1. Click en personaje → Voz se activa directamente
2. Personaje responde visualmente (animación + voz)
3. Sin ventanas ni paneles adicionales

3.2 Componente de Voz Compacta

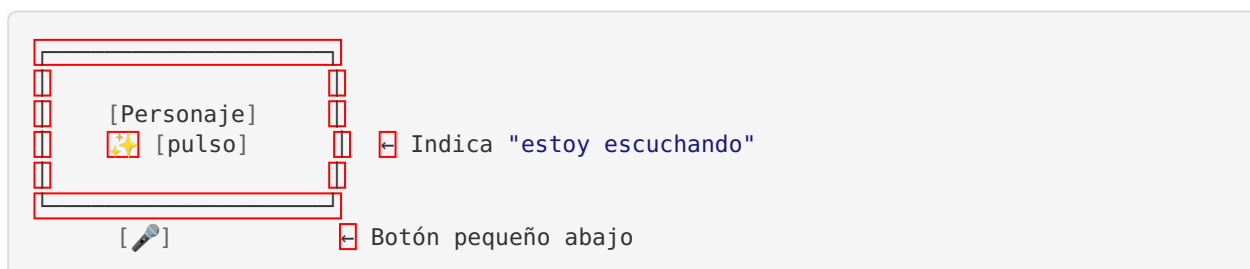
Archivo nuevo: components/companion/direct-voice-interaction.tsx

Características:

- Botón de micrófono flotante pequeño
- Indicador visual en el personaje (anillo de pulso)

- Transcripción temporal en tooltip
- Sin chat visible
- Modo "siempre listo" (un solo click para hablar)

Interfaz minimalista:



3.3 Visualización de Estado

Indicadores visuales:

- **Idle:** Personaje con respiración suave
- **Escuchando:** Anillo dorado pulsante
- **Procesando:** Partículas flotantes (✨)
- **Hablando:** Movimiento de boca + brillo
- **Error:** Sacudida suave + tooltip de error



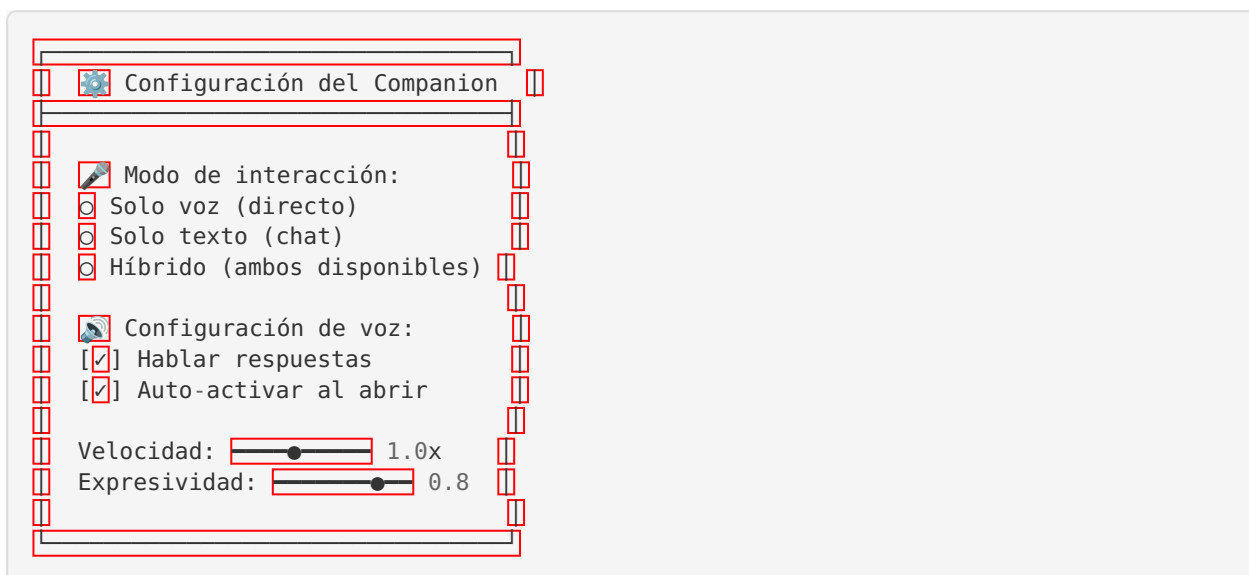
FASE 4: PREFERENCIAS EN AJUSTES

4.1 Nuevas Opciones en Configuración

Panel de ajustes ampliado:

```
// Agregar al schema de Prisma
model Companion {
  // ... campos existentes
  interaction_mode: String @default("voice") // "voice" | "text" | "hybrid"
  voice_enabled: Boolean @default(true)
  auto_speak: Boolean @default(true)
  voice_speed: Float @default(1.0)
  expressiveness_level: Float @default(0.8)
}
```

UI de ajustes:



4.2 Persistencia de Preferencias


- Guardar en base de datos (tabla `companion`)
- Sincronizar entre dispositivos
- Aplicar inmediatamente al cambiar

PLAN DE IMPLEMENTACIÓN PASO A PASO


CHECKPOINT 0: Backup Inicial

- [x] Crear rama de backup
- [x] Documentar estado actual
- [x] Guardar checkpoint “Pre-Animaciones”

PASO 1: Voces Mejoradas (1-2 horas)

1. Integrar servicio TTS elegido
2. Configurar personalidades por companion
3. Probar voces con texto de ejemplo
4. Ajustar parámetros
5.  CHECKPOINT: “Voces mejoradas implementadas”

PASO 2: Animaciones Básicas (2-3 horas)

1. Crear componente `AnimatedCharacter`
2. Implementar respiración y parpadeo
3. Probar con todos los companions
4. Ajustar tiempos y suavidad
5.  CHECKPOINT: “Animaciones básicas funcionando”

PASO 3: Sincronización Voz-Boca (2-3 horas)

1. Integrar Web Audio API
2. Detectar frecuencias de voz
3. Animar boca en sincronía

4. Ajustar sensibilidad
5. ☒ CHECKPOINT: “Sincronización voz-boca lista”

PASO 4: Interacción Directa (2-3 horas)

1. Crear componente de voz compacta
2. Modificar `floating-companion.tsx`
3. Implementar activación directa
4. Probar flujo completo
5. ☒ CHECKPOINT: “Interacción directa implementada”

PASO 5: Preferencias en Ajustes (1-2 horas)

1. Actualizar schema de Prisma
2. Migrar base de datos
3. Añadir UI de preferencias
4. Implementar persistencia
5. ☒ CHECKPOINT: “Sistema de preferencias completo”

PASO 6: Expresividad Emocional (2-3 horas)

1. Implementar detección de emociones
2. Ajustar voces dinámicamente
3. Sincronizar con animaciones
4. Probar diferentes emociones
5. ☒ CHECKPOINT: “Expresividad emocional activa”

PASO 7: Pulido y Optimización (1-2 horas)

1. Optimizar rendimiento
2. Ajustar timings
3. Mejorar transiciones
4. Testing exhaustivo
5. ☒ CHECKPOINT FINAL: “Companions animados completos”



TIEMPO ESTIMADO TOTAL

- **Mínimo:** 11 horas
- **Máximo:** 18 horas
- **Promedio:** 14 horas

Distribución recomendada:

- Sesión 1 (4h): Pasos 1-2
 - Sesión 2 (4h): Pasos 3-4
 - Sesión 3 (4h): Pasos 5-6
 - Sesión 4 (2h): Paso 7 + ajustes finales
-

DECISIONES NECESARIAS ANTES DE EMPEZAR

1. Servicio de TTS

- ☐ **Opción A:** Abacus.AI LLM APIs (gratis, integrado)
- ☐ **Opción B:** ElevenLabs (alta calidad, requiere API key)
- ☐ **Opción C:** Play.ht (buena calidad, requiere API key)

Recomendación: Empezar con Abacus.AI, evaluar calidad, cambiar si es necesario

2. Nivel de Animación

- ☐ **Básico:** Solo parpadeo + respiración + boca
- ☐ **Medio:** + Inclinación + cambios de expresión
- ☐ **Avanzado:** + Partículas + efectos adicionales

Recomendación: Medio (equilibrio entre naturalidad y rendimiento)

3. Modo de Interacción por Defecto

- ☐ **Solo voz** (directo, sin chat)
- ☐ **Híbrido** (voz por defecto, opción de texto en ajustes)

Recomendación: Híbrido (más flexible para diferentes usuarios)

CRITERIOS DE ÉXITO

Al finalizar, los companions deberán:

1. ☒ Hablar con voces naturales, dulces y expresivas
 2. ☒ Mostrar animaciones sutiles y fluidas
 3. ☒ Sincronizar movimiento de boca con voz
 4. ☒ Responder al toque directo sin abrir chat
 5. ☒ Mantener todos los efectos visuales originales
 6. ☒ Permitir elegir modo voz/texto en ajustes
 7. ☒ Funcionar fluidamente en todos los navegadores
 8. ☒ Mantener rendimiento óptimo
-

PLAN DE REVERSIÓN

Si algo sale mal en cualquier paso:

1. Detener implementación
2. Restaurar último checkpoint
3. Analizar problema
4. Ajustar enfoque
5. Reintentar

Checkpoints de seguridad:

- Pre-Animaciones (estado actual)
- Post-Voces
- Post-Animaciones-Básicas
- Post-Sincronización

- Post-Interacción-Directa
 - Post-Preferencias
 - Final
-



NOTAS FINALES

Lo que NO cambiará:

- ❌ PNG originales de companions
- ❌ Diseño visual y estilo artístico
- ❌ Estructura de base de datos principal
- ❌ Layout general de la app
- ❌ Funcionalidad de chat de texto

Lo que SÍ cambiará:

- ✅ Calidad y naturalidad de las voces
 - ✅ Movimientos y animaciones del personaje
 - ✅ Forma de interactuar (más directa)
 - ✅ Opciones de configuración
 - ✅ Experiencia inmersiva
-



ESTADO ACTUAL

- [x] Plan documentado
- [] Aprobación del usuario
- [] Backup creado
- [] Implementación iniciada

¿Listo para empezar? 🤖✨