

GUÍA DEFINITIVA: NOMBRES Y VOCES DE COMPANIONS

Fecha: 1 de Noviembre 2025

Estado: COMPLETADO Y VERIFICADO

TABLA MAESTRA DE COMPANIONS ESTE ES EL ÚNICO MAPEO VÁLIDO - NO HAY OTROS

Nombre Real	Type en BD	Género	Sistema TTS	Voz	Pitch
Ada	hada	Femenino	Puter.js	Mia (generative)	1.9
Luna	lumi	Femenino	Puter.js	Lucia (neural)	1.7
Coral	nimbo	Femenino	Puter.js	Conchita (neural)	1.7
Aurora	human	Femenino	Puter.js	Lupe (neural)	1.85
Sprig	fabel	Masculino	Web Speech	Voz española	0.85
Willow	willow	Masculino	Web Speech	Voz española	1.0
Orion	unicornito	Masculino	Web Speech	Voz española	0.9
Ken	ken	Masculino	Puter.js	Enrique (neural)	0.75
Ember	draguito	Masculino	Web Speech	Voz española	0.85
Sage	elfo	Masculino	Web Speech	Voz española	0.75

REGLA DE ORO

SIEMPRE QUE AGREGUES O MODIFIQUES UN COMPANION:

1. **Base de datos:** Usa el type de la tabla (ej: hada, nimbo)
2. **voice-config.ts:** La key debe ser el type (ej: hada)
3. **puter-tts-service.ts:** La key debe ser el type (ej: hada)
4. **companion-stories.ts:** El COMPANION_ID_MAP debe tener el type → ID en minúsculas

NO inventes nombres nuevos. NO uses variaciones. Usa EXACTAMENTE los tipos de la tabla.

ARCHIVOS IMPORTANTES

1. /lib/voice-config.ts

```
export const companionVoices: Record<string, VoiceConfig> = {
  hada: {      // ← SIEMPRE USA EL TYPE, NO EL NOMBRE
    realName: 'Ada',
    gender: 'female',
    usePuter: true,
    pitch: 1.9,
    // ...
  },
  nimbo: {      // ← SIEMPRE USA EL TYPE, NO EL NOMBRE
    realName: 'Coral',
    gender: 'female',
    usePuter: true,
    pitch: 1.7,
    // ...
  }
  // ...
}
```

2. /lib/puter-tts-service.ts

```
const magicalVoiceConfigs: Record<string, PuterTTSOptions> = {
  hada: {      // ← SIEMPRE USA EL TYPE
    voice: 'Mia',
    engine: 'generative',
    realName: 'Ada',
    // ...
  },
  nimbo: {      // ← SIEMPRE USA EL TYPE
    voice: 'Conchita',
    engine: 'neural',
    realName: 'Coral',
    // ...
  }
  // ...
}
```

3. /lib/companion-stories.ts

```
export const COMPANION_ID_MAP: Record<string, string> = {
  'hada': 'ada',           // type → ID en minúsculas
  'nimbo': 'coral',        // type → ID en minúsculas
  'human': 'aurora',
  // ...
}
```

CHECKLIST ANTES DE CAMBIAR ALGO

Cuando vayas a modificar o agregar un companion:

- ¿El `type` en la base de datos es el correcto? (ej: `hada`, `nimbo`)
- ¿La configuración en `voice-config.ts` usa el `type` como key?
- Si usa `Puter.js`, ¿está configurado en `puter-tts-service.ts` con el `type`?
- ¿El `COMPANION_ID_MAP` en `companion-stories.ts` tiene el mapeo correcto?
- ¿El género (`gender`) coincide con la voz asignada?

Si todas las respuestas son SÍ, entonces está bien.

CÓMO VERIFICAR QUE TODO FUNCIONA

1. Verifica la Base de Datos

```
cd /home/ubuntu/hogara_planner/nextjs_space
yarn tsx << 'EOF'
import { config } from 'dotenv'
config()
import { PrismaClient } from '@prisma/client'
const prisma = new PrismaClient()
async function check() {
  const companions = await prisma.companion.findMany({
    select: { name: true, type: true }
  })
  companions.forEach(c => console.log(`"${c.name} → type="${c.type}"`))
  await prisma.$disconnect()
}
check()
EOF
```

Resultado esperado:

```
Ada → type="hada"
Coral → type="nimbo"
```

```
Aurora → type="human"  
Sprig → type="fabel"  
Ken → type="ken"
```

2. Prueba las Voces

1. Limpia la caché del navegador: Ctrl + Shift + R
2. Ve a /premium/acompañante
3. Selecciona **Coral**
4. Activa el micrófono
5. Di: "Hola Coral, ¿cómo estás?"

En la consola del navegador (F12) deberías ver:

```
Usando Puter.js TTS para Coral (FEMENINA)  
Llamando a puter.ai.txt2speech...  
- Voz AWS Polly: Conchita  
- Motor: neural  
Coral está hablando...  
Coral terminó de hablar
```

ERRORES COMUNES Y CÓMO EVITARLOS

Error 1: "La voz suena masculina"

Causa: El type en la BD no coincide con la configuración de voz.

Solución:

```
# Verificar el type en la BD  
yarn tsx << 'EOF'  
import { config } from 'dotenv'; config()  
import { PrismaClient } from '@prisma/client'  
const prisma = new PrismaClient()  
async function check() {  
  const coral = await prisma.companion.findFirst({  
    where: { name: 'Coral' },  
    select: { type: true }  
  })  
  console.log(`Coral type="${coral?.type}"`)  
  await prisma.$disconnect()  
}  
check()  
EOF  
  
# Si NO es "nimbo", corrígetelo:  
yarn tsx << 'EOF'
```

```

import { config } from 'dotenv'; config()
import { PrismaClient } from '@prisma/client'
const prisma = new PrismaClient()
async function fix() {
    await prisma.companion.updateMany({
        where: { name: 'Coral' },
        data: { type: 'nimbo' }
    })
    console.log(' Coral corregida a type="nimbo"')
    await prisma.$disconnect()
}
fix()
EOF

```

Error 2: “No escucho ninguna voz”

Causa: Puter.js no está disponible en el navegador.

Solución: 1. Abre la consola (F12) 2. Escribe: `window.puter` 3. Si ves `undefined`, recarga la página con `Ctrl + Shift + R`

Error 3: “Veo logs de Web Speech en lugar de Puter.js”

Causa: La caché del navegador tiene la configuración antigua.

Solución: 1. Abre DevTools (F12) 2. Ve a **Application** → **Storage** → **Clear site data** 3. Recarga la página

SI ALGO SALE MAL

1. **NO INVENTES SOLUCIONES.** Sigue esta guía al pie de la letra.
 2. Verifica la tabla maestra al inicio de este documento.
 3. Usa los scripts de verificación de esta guía.
 4. Si el problema persiste, limpia la caché del navegador.
-

ESTADO ACTUAL (1 Nov 2025)

Base de datos: - Coral corregida: `coral` → `nimbo` - Todos los companions tienen el `type` correcto

Configuraciones: - `voice-config.ts` tiene todos los `types` correctos - `puter-tts-service.ts` tiene las voces femeninas configuradas - `companion-stories.ts` tiene los mapeos correctos

Compilación: - Build exitoso - Checkpoint guardado: “Coral corregida con voz femenina”

RECUERDA: Esta guía es la ÚNICA fuente de verdad. Si algo no coincide con esta guía, la guía tiene razón. Corrígelo.