

Acceso Completo: Modificar Sistema PMS desde Cursor

LINKS DIRECTOS PARA ACCESO INMEDIATO

PROYECTO EN GITHUB (Para Cursor)

```
https://github.com/mfconsultoresdev/hotel-pms-paseo-las-mercedes-chatllm.git
```

SISTEMA FUNCIONANDO (Para Referencia)

```
https://hotelpmercedes.abacusai.app
```

BASE DE DATOS (Información de Conexión)

```
Host: db-a4a563bb9.db001.hosteddb.reai.io  
Puerto: 5432  
Base de datos: a4a563bb9  
Usuario: role_a4a563bb9
```

OPCIÓN 1: CLONAR EN CURSOR (RECOMENDADO)

Comando Directo en Cursor Chat:

```
@terminal git clone https://github.com/mfconsultoresdev/hotel-pms-paseo-las-mercedes-chatllm.git
```

Abrir Proyecto en Cursor:

1. **Ejecutar comando** en Cursor Chat
2. **File → Open Folder**
3. **Seleccionar:** hotel-pms-paseo-las-mercedes-chatllm
4. **Navegar a:** app/ (carpeta principal del proyecto)

Setup Rápido en Cursor Chat:

```
@terminal cd hotel-pms-paseo-las-mercedes-chatllm/app && yarn install
```

Configurar Variables de Entorno:

```
DATABASE_URL="postgresql://role_a4a563bb9:1MY2RR4lrIvERMLjoTpbC9f9SlRsmxRm@db-a4a563bb9.db001.hosteddb.reai.io:5432/a4a563bb9?connect_timeout=15"  
NEXTAUTH_URL="http://localhost:3000"  
NEXTAUTH_SECRET="K0oRFBbRS19HsDXg8zVe0DyK4W7KYBfY"
```

Ejecutar Sistema:

```
@terminal npx prisma generate && yarn dev
```

✓ Acceder en Navegador:

```
http://localhost:3000
```

OPCIÓN 2: ACCESO DIRECTO A BASE DE DATOS

Panel de Administración (Prisma Studio):

```
@terminal cd hotel-pms-paseo-las-mercedes-chatllm/app
@terminal npx prisma studio
```

Resultado: Se abre `http://localhost:5555` con interface visual para la base de datos

Modificar Esquema de Base de Datos:

1. **Archivo:** `app/prisma/schema.prisma`
2. **Migrar cambios:** `@terminal npx prisma migrate dev`
3. **Regenerar cliente:** `@terminal npx prisma generate`

Queries SQL Directas:

```
@terminal npx prisma db seed # Poblar con datos de prueba
@terminal npx prisma migrate reset # Resetear y repoblar
```

OPCIÓN 3: MODIFICACIONES EN SISTEMA LIVE

⚠ IMPORTANTE:

El sistema live (`https://hotelpmercedes.abacusai.app`) **NO TIENE** acceso directo para modificaciones. Las opciones son:

Para Modificar Código:

1. **Clonar desde GitHub** → Desarrollar local → **Push cambios**
2. **El preview se actualiza automáticamente** con los cambios

Para Modificar Base de Datos:

```
# Conectar desde local usando las credenciales:
DATABASE_URL="postgresql://role_a4a563bb9:LMY2RR4lrIvERMLjoTpbC9f9S1RsmxRm@db-a4a563b-b9.db001.hosteddb.reai.io:5432/a4a563bb9?connect_timeout=15"
```

WORKFLOW COMPLETO DE MODIFICACIONES

Flujo Recomendado:

1. Clonar Proyecto:

```
@terminal git clone https://github.com/mfconsultoresdev/hotel-pms-paseo-las-mercedes-chatllm.git
@terminal cd hotel-pms-paseo-las-mercedes-chatllm/app
@terminal yarn install
```

2. Conectar a Base de Datos Live:

```
# Crear .env con conexión a base real
DATABASE_URL="postgresql://role_a4a563bb9:LMY2RR4lrIvERMLjoTpbC9f9SlRsmxRm@db-a4a563b-b9.db001.hosteddb.reai.io:5432/a4a563bb9?connect_timeout=15"
NEXTAUTH_URL="http://localhost:3000"
NEXTAUTH_SECRET="K0oRFBbRS19HsDXg8zVe0DyK4W7KYBfY"
```

3. Desarrollar Local:

```
@terminal npx prisma generate
@terminal yarn dev
```

Resultado: `http://localhost:3000` con **datos reales** de producción

4. Hacer Modificaciones:

- **UI/UX:** Archivos en `app/components/` , `app/(dashboard)/`
- **APIs:** Archivos en `app/api/`
- **Base de datos:** Archivo `app/prisma/schema.prisma`
- **Estilos:** Archivos `.css` y Tailwind

5. Subir Cambios:

```
@terminal git add .
@terminal git commit -m "Descripción del cambio"
@terminal git push origin master
```

6. Cambios Automáticos:

El **preview live** se **actualiza automáticamente** con tus cambios

CREDENCIALES Y ACCESOS

Cuentas del Sistema (Todas con password: `admin123`):

```
Admin: admin@hotelpaseolm.com
Gerente: gerente@hotelpaseolm.com
Recepción: recepcion@hotelpaseolm.com
```

Datos de Conexión Base de Datos:

```
Host: db-a4a563bb9.db001.hosteddb.reai.io
Port: 5432
Database: a4a563bb9
Username: role_a4a563bb9
Password: lMY2RR4lrIvERMLjoTpbC9f9SlRsmxRm
SSL: Required
```

String de Conexión Completa:

```
postgresql://role_a4a563bb9:lMY2RR4lrIvERMLjoTpbC9f9SlRsmxRm@db-a4a563b-
b9.db001.hosteddb.reai.io:5432/a4a563bb9?connect_timeout=15
```

HERRAMIENTAS PARA MODIFICACIONES

En Cursor:

1. **Terminal integrada:** Para comandos
2. **Explorer:** Para navegar archivos
3. **Git integration:** Para control de versiones
4. **AI Chat:** Para ayuda con código

Para Base de Datos:

1. **Prisma Studio** (`npx prisma studio`): Interface visual
2. **pgAdmin** (opcional): Cliente PostgreSQL completo
3. **DBeaver** (opcional): Universal database tool
4. **Cursor AI:** Para generar queries y schemas

Para Testing:

1. **Servidor local:** `yarn dev` → `http://localhost:3000`
2. **Preview live:** `https://hotelpmercedes.abacusai.app`
3. **API testing:** Thunder Client o Postman en `http://localhost:3000/api/`

ESTRUCTURA DEL PROYECTO

Carpetas Principales:

```
app/
├── (dashboard)/      # Páginas principales del sistema
├── api/              # Endpoints REST API
├── components/       # Componentes reutilizables
├── lib/              # Utilidades y configuraciones
├── prisma/           # Esquema y migraciones de DB
├── public/           # Assets estáticos
└── styles/           # Estilos CSS
```

Para Modificar UI:

- **Dashboard:** `app/(dashboard)/page.tsx`
- **Habitaciones:** `app/(dashboard)/rooms/page.tsx`
- **Reservas:** `app/(dashboard)/reservations/page.tsx`
- **Facturación:** `app/(dashboard)/billing/page.tsx`
- **Housekeeping:** `app/(dashboard)/housekeeping/page.tsx`

Para Modificar APIs:

- **Dashboard APIs:** `app/api/dashboard/`
- **Habitaciones APIs:** `app/api/rooms/`
- **Reservas APIs:** `app/api/reservations/`
- **Facturación APIs:** `app/api/billing/`
- **Housekeeping APIs:** `app/api/housekeeping/`



COMANDOS ÚTILES EN CURSOR

Git (Control de Versiones):

@terminal	git status	# Ver cambios
@terminal	git add .	# Agregar todos los cambios
@terminal	git commit -m "mensaje"	# Confirmar cambios
@terminal	git push origin master	# Subir cambios
@terminal	git pull origin master	# Descargar cambios

Package Management:

@terminal	yarn add [package]	# Instalar dependencia
@terminal	yarn remove [package]	# Remover dependencia
@terminal	yarn dev	# Ejecutar en desarrollo
@terminal	yarn build	# Construir para producción

Base de Datos:

@terminal	npx prisma studio	# Abrir interface visual
@terminal	npx prisma generate	# Generar cliente Prisma
@terminal	npx prisma migrate dev	# Aplicar migraciones
@terminal	npx prisma db seed	# Poblar con datos de prueba
@terminal	npx prisma migrate reset	# Resetear base de datos

Mantenimiento:

@terminal	yarn lint	# Linter para errores
@terminal	yarn type-check	# Verificar tipos TypeScript
@terminal	rm -rf node_modules && yarn	# Reinstalar dependencias
@terminal	git clean -fd	# Limpiar archivos no trackeados

QUICK START (Todo en uno)

Para empezar inmediatamente en Cursor:

```
@terminal git clone https://github.com/mfconsultoresdev/hotel-pms-paseo-las-mercedes-chatllm.git && cd hotel-pms-paseo-las-mercedes-chatllm/app && yarn install && echo 'DATABASE_URL="postgresql://role_a4a563bb9:LMY2RR4lrIvERMLjoTpbC9f9S1RsmxRm@db-a4a563bb9.db001.hosteddb.reai.io:5432/a4a563bb9?connect_timeout=15"' > .env && echo 'NEXTAUTH_URL="http://localhost:3000"' >> .env && echo 'NEXTAUTH_SECRET="K0oRFBbRS19HsDXg8zVe0DyK4W7KYBfY"' >> .env && npx prisma generate && yarn dev
```

✓ Resultado:

- Proyecto clonado ✓
- Dependencias instaladas ✓
- Variables configuradas ✓
- Base de datos conectada ✓
- Sistema ejecutándose en `http://localhost:3000` ✓

RESUMEN DE OPCIONES

RECOMENDADO: Desarrollo Local con DB Live

- **Link:** `https://github.com/mfconsultoresdev/hotel-pms-paseo-las-mercedes-chatllm.git`
- **Ventajas:** Desarrollo local + datos reales + cambios automáticos en preview
- **Ideal para:** Modificaciones continuas y desarrollo

ALTERNATIVO: Solo Visualización

- **Link:** `https://hotelpmercedes.abacusai.app`
- **Ventajas:** Sistema funcionando inmediatamente
- **Ideal para:** Demos, testing funcional, referencias

AVANZADO: Acceso Directo a DB

- **Tools:** pgAdmin, DBeaver, Prisma Studio
- **Ventajas:** Control total de la base de datos
- **Ideal para:** Análisis de datos, reports, migraciones

SOPORTE Y TROUBLESHOOTING

Problemas Comunes:

1. “git no reconocido” → Instalar Git
2. “yarn no encontrado” → `npm install -g yarn`
3. “Puerto ocupado” → `yarn dev --port 3001`
4. “Error de conexión DB” → Verificar firewall/VPN
5. “Prisma errors” → `npx prisma generate`

Comandos de Emergencia:

```
@terminal rm -rf node_modules && yarn install      # Reinstalar todo
@terminal git reset --hard origin/master          # Resetear cambios locales
@terminal npx prisma migrate reset                 # Resetear base de datos
@terminal yarn build && yarn start                 # Probar build de producción
```

NEXT STEPS DESPUÉS DE CLONAR

1. Explorar el Código:

- Ver estructura en Cursor Explorer
- Revisar components principales
- Entender flujo de APIs

2. Hacer Cambios de Prueba:

- Modificar un color en CSS
- Cambiar un texto en UI
- Agregar un console.log

3. Probar el Workflow:

- Hacer commit y push
- Verificar que el preview se actualiza
- Confirmar que todo funciona

4. Desarrollo Serio:

- Crear branch para features nuevos
- Implementar cambios requeridos
- Hacer testing completo antes de merge

VALOR DEL SISTEMA

Lo que tienes acceso a modificar:


- **Sistema PMS completo** de \$50,000+ valor comercial
- **10 módulos** de gestión hotelera enterprise
- **Base de datos** con esquema completo y datos reales
- **APIs REST** para integración con otros sistemas
- **UI moderna** con components reutilizables
- **Sistema de autenticación** multi-rol
- **Dashboard ejecutivo** con métricas en tiempo real

Modificaciones Posibles:

- **Agregar nuevos módulos** (ej: Spa, Restaurante, Eventos)
- **Customizar UI/UX** para marca específica
- **Integrar APIs externas** (pagos, SMS, email)

- **Crear reportes personalizados**
 - **Modificar workflows** de negocio
 - **Agregar funcionalidades móviles**
-

 **¡Tienes acceso completo a un sistema PMS enterprise listo para usar y modificar!**

 Toda la documentación y credenciales están en este archivo