

Sistema de Onboarding de Tenants - Implementación Completada

Resumen

Se ha implementado exitosamente un sistema completo de onboarding para que nuevas empresas puedan registrarse como tenants en Rental Management, crear su subdomain, y configurar su negocio inicial.

Archivos Creados

1. API Routes

`/app/api/tenants/check-subdomain/route.ts`

- **Endpoint:** GET `/api/tenants/check-subdomain?subdomain=xxx`
- **Funcionalidad:** Validación en tiempo real de disponibilidad de subdomain
- **Validaciones:**
 - Formato: solo letras minúsculas, números y guiones
 - Longitud: mínimo 3, máximo 63 caracteres
 - Subdominios reservados (www, api, admin, etc.)
 - Disponibilidad en base de datos

`/app/api/tenants/create/route.ts`

- **Endpoint:** POST `/api/tenants/create`
- **Funcionalidad:** Crear tenant con usuario admin en transacción atómica
- **Validación:** Schema Zod con todos los campos del wizard
- **Proceso:**
 1. Valida subdomain único
 2. Verifica email único
 3. Hashea contraseña con bcryptjs
 4. Crea tenant y usuario OWNER en transacción
 5. Retorna tenant_id y subdomain

`/app/api/tenants/update/route.ts`

- **Endpoint:** PUT `/api/tenants/update` (actualizar), GET `/api/tenants/update` (obtener datos)
- **Funcionalidad:** Actualizar configuración del tenant
- **Permisos:** Solo accesible para roles OWNER y ADMIN
- **Características:**
 - Subdomain es inmutable
 - Actualiza nombre, ubicación, tipos de negocio
 - Gestiona logo y colores del tema
 - Opción de publicar en Marbella4Rent

2. Páginas del Cliente

`/app/onboarding/page.tsx`

- **Ruta:** `/onboarding`
- **Funcionalidad:** Wizard de registro multi-paso
- **Características:**
- **Paso 1 - Información Básica:**
 - Nombre de empresa
 - Nombre del admin
 - Email del admin
 - Contraseña
 - Subdomain deseado (con validación en tiempo real)
- **Paso 2 - Configuración del Negocio:**
 - Ubicación
 - Tipo de negocio (VEHICLE/PROPERTY/BOAT/EXPERIENCE/SCOOTER/EQUIPMENT)
- **Paso 3 - Personalización:**
 - Logo URL (opcional)
 - Color primario
 - Color secundario
 - Checkbox “Publicar en Marbella4Rent” (visible solo si ubicación incluye “Marbella”)
- **Tecnologías:**
 - react-hook-form con validación zod
 - Stepper visual mostrando progreso
 - Navegación entre pasos (siguiente, anterior, finalizar)
 - Validación en cada paso antes de avanzar

`/app/settings/page.tsx y /app/settings/TenantSettingsForm.tsx`

- **Ruta:** `/settings`
- **Funcionalidad:** Página de configuración del tenant
- **Permisos:** Solo OWNER y ADMIN
- **Características:**
 - Ver subdomain (read-only)
 - Editar nombre de empresa
 - Editar ubicación
 - Seleccionar múltiples tipos de negocio
 - Subir/cambiar logo
 - Cambiar colores del tema
 - Checkbox “Publicar en Marbella4Rent” (solo si ubicación incluye “Marbella”)

`/app/page.tsx`

- **Ruta:** `/` (landing page)
- **Funcionalidad:** Página pública de inicio
- **Secciones:**
- **Hero:** Título principal + CTAs (Get Started, Learn More)

- **Features:** 6 características principales con iconos
- **Business Types:** Grid con los 6 tipos de negocio soportados
- **CTA Section:** Call-to-action final
- **Footer:** Links a productos y empresa
- **Navegación:**
 - Botón “Get Started” → /onboarding
 - Botón “Sign In” → /login

3. Middleware y Seed

`middleware.ts` (Actualizado)

- **Cambios:**
 - Agregadas rutas públicas: /, /onboarding, /api/tenants/*
 - Redirige usuarios autenticados de /onboarding a /dashboard
 - Permite acceso público a la landing page

`prisma/seed.ts` (Actualizado)

- **Nuevos Tenants:**
 1. **Scooters Madrid**
 - Subdomain: scooters-madrid
 - Ubicación: Madrid
 - Tipos: VEHICLE_RENTAL, SCOOTER_RENTAL
 - Usuario: admin@scooters-madrid.com / password123 (OWNER)
 - 2 items de ejemplo: Electric Scooter Pro, City Scooter Classic
 1. **Boats Marbella**
 - Subdomain: boats-marbella
 - Ubicación: Marbella
 - Tipo: BOAT_RENTAL
 - Usuario: info@boats-marbella.com / password123 (OWNER)
 - 3 items de ejemplo: Luxury Yacht, Speedboat, Sailboat
 - Publicado en Marbella4Rent

Flujo de Usuario Completo

Nuevo Tenant (Primera vez)

1. **Landing Page** (/)
 - Usuario ve la página de inicio
 - Click en “Get Started” o “Empezar ahora”
2. **Onboarding Wizard** (/onboarding)
 - **Paso 1:** Ingresa información básica + subdomain
 - Sistema valida subdomain en tiempo real
 - **Paso 2:** Configura ubicación y tipo de negocio
 - **Paso 3:** Personaliza logo y colores
 - Click en “Complete Setup”

3. Creación del Tenant

- API crea tenant y usuario admin
- Redirige a /login con mensaje de éxito

4. Login

- Usuario inicia sesión con credenciales creadas
- Redirige a /dashboard

Tenant Existente (Configuración)

1. Dashboard → Settings

- Solo OWNER y ADMIN pueden acceder
- Edita información del tenant
- Guarda cambios

Testing

Ejecutar Seed Script

```
cd /home/ubuntu/rental_management
npm run db:seed
```

Credenciales de Prueba

Demo Tenant (subdomain: demo)

- owner@demo.com / password123 (OWNER)
- admin@demo.com / password123 (ADMIN)
- operator@demo.com / password123 (OPERATOR)

Test Tenant (subdomain: test)

- owner@test.com / password123 (OWNER)

Scooters Madrid (subdomain: scooters-madrid)

- admin@scooters-madrid.com / password123 (OWNER)
- 2 scooters disponibles

Boats Marbella (subdomain: boats-marbella)

- info@boats-marbella.com / password123 (OWNER)
- 3 boats disponibles
- Publicado en Marbella4Rent

Testing Local con Subdominios

1. Añadir a /etc/hosts (o C:\Windows\System32\drivers\etc\hosts en Windows):

```
127.0.0.1 demo.localhost test.localhost scooters-madrid.localhost boats-marbella.loc-
alhost
```

1. Iniciar servidor:

```
npm run dev
```

1. Visitar:

- <http://localhost:3000> → Landing page
 - <http://localhost:3000/onboarding> → Wizard de registro
 - <http://demo.localhost:3000> → Login de tenant demo
 - <http://scooters-madrid.localhost:3000> → Login de Scooters Madrid
 - <http://boats-marbella.localhost:3000> → Login de Boats Marbella
-

Checklist de Implementación

- [x] API route para validar subdomain
 - [x] API route para crear tenant
 - [x] API route para actualizar tenant
 - [x] Wizard de onboarding multi-paso
 - [x] Página de configuración de tenant
 - [x] Página de landing/home
 - [x] Actualizar middleware para rutas públicas
 - [x] Actualizar seed script con tenants de prueba
 - [x] Validación con Zod en todos los formularios
 - [x] Integración con react-hook-form
 - [x] Diseño responsive con Tailwind CSS
 - [x] Control de permisos (OWNER/ADMIN)
 - [x] Transacciones atómicas en creación de tenant
 - [x] Compilación exitosa sin errores
-

Características Destacadas

UX/UI

- Stepper visual en wizard de onboarding
- Validación en tiempo real de subdomain
- Feedback inmediato en formularios
- Diseño responsive mobile-first
- Gradientes modernos y animaciones suaves

Seguridad

- Contraseñas hasheadas con bcryptjs
- Validación de permisos en API routes
- Transacciones atómicas para consistencia
- Subdomain inmutable después de creación

Personalización

- Logo personalizado (preparado para AWS S3)

- Colores del tema personalizables
 - Integración con Marbella4Rent (condicional)
 - Múltiples tipos de negocio soportados
-

Próximos Pasos Sugeridos

1. **Integración AWS S3:** Implementar upload de logos a S3
 2. **Email de verificación:** Enviar email de confirmación después del registro
 3. **Plan de suscripción:** Añadir selección de plan (Free/Pro/Enterprise)
 4. **Tour guiado:** Implementar tour inicial después del onboarding
 5. **Integración Stripe:** Añadir pagos para planes premium
 6. **Dashboard de administración:** Panel para ver todos los tenants (SUPER_ADMIN)
-

Notas Técnicas

- **Database:** PostgreSQL con Prisma ORM
 - **Autenticación:** NextAuth.js con JWT
 - **Validación:** Zod schemas
 - **Formularios:** react-hook-form
 - **Styling:** Tailwind CSS
 - **Multi-tenancy:** Header-based tenant isolation
-

Archivos Relacionados

- `types/index.ts` - Tipos TypeScript
 - `lib/tenant.ts` - Utilidades de tenant
 - `lib/auth.ts` - Utilidades de autenticación
 - `prisma/schema.prisma` - Schema de base de datos
 - `app/api/auth/[...nextauth]/route.ts` - Configuración NextAuth
-

Implementación completada exitosamente ✨

Commit: `d5a93c7` - “Implementar sistema completo de onboarding de tenants”