

Diseño de la Base de Datos y Relaciones

Definir tablas necesarias. Establecer relaciones entre tablas.
Planificar con enfoque en escalabilidad y SEO



¿Por qué usamos MySQL?

Motivos principales:

- **Compatibilidad completa** con Laravel y XAMPP.
- **Amplio soporte** en la industria y en servicios de hosting.
- **Facilidad de uso** para bases de datos relacionales.
- **Alta disponibilidad y rendimiento excelente** en proyectos de tamaño pequeño y mediano.
- **Soporte para relaciones y claves foráneas**, fundamentales en nuestra estructura.

Notas importantes:

- Laravel utiliza **Eloquent ORM** para trabajar de forma **nativa** con MySQL, lo que facilita la creación y manejo de datos.
- Amplia documentación disponible en caso de necesitar ayuda.



Tablas Principales de la Base de Datos

1. Users (Usuarios)
2. Roles (Roles de usuario)
3. Categories (Categorías)
4. Services (Servicios/Productos)
5. Posts (Publicaciones)
6. Subscribers (Suscriptores)
7. Contacts (Mensajes de contacto)



◆ Tabla: users

¿Qué guarda?

Usuarios del sistema.

Campos principales:

- **id**: Identificador único.
- **name**: Nombre completo.
- **email**: Correo electrónico.
- **password**: Contraseña cifrada.
- **role_id**: Relación con la tabla **roles**.
- **timestamps**: Fechas de creación y actualización.

Relación: Cada usuario pertenece a **un rol**.



◆ **Tabla: roles**

- **¿Qué guarda?**
Tipos de usuario: Admin y Editor.
- **Campos principales:**
 - **id**: Identificador único.
 - **name**: Nombre del rol (admin/editor).
 - **description**: Descripción del rol.
- **Relación**: Un rol puede estar asignado a **varios usuarios**



◆ Tabla: categories

- ¿Qué guarda?
Categorías de servicios y publicaciones.
- Campos principales:
 - `id`: Identificador único.
 - `name`: Nombre de la categoría.
 - `slug`: URL amigable (SEO).
 - `description`: Breve descripción.
 - `timestamps`: Fechas de creación y actualización.
- Relación: Una categoría puede tener **muchos servicios** o **muchos posts**.



◆ Tabla: services

- ¿Qué guarda?
Servicios o productos ofrecidos por la empresa.
- Campos principales:
 - `id`: Identificador único.
 - `title`: Nombre del servicio.
 - `slug`: URL amigable (SEO).
 - `description`: Detalle del servicio.
 - `price`: Precio.
 - `image_url`: Imagen representativa.
 - `category_id`: Categoría relacionada.
 - `seo_title`: Título SEO.
 - `seo_description`: Descripción SEO.
 - `timestamps`: Fechas de creación y actualización.
- Relación: Un servicio pertenece a una categoría.



◆ Tabla: posts

¿Qué guarda?

Publicaciones del blog de la empresa.

Campos principales:

- `id`: Identificador único.
- `title`: Título del post.
- `slug`: URL amigable (SEO).
- `content`: Contenido del artículo.
- `image_url`: Imagen destacada.
- `category_id`: Categoría relacionada.
- `seo_title`: Título SEO.
- `seo_description`: Descripción SEO.
- `timestamps`: Fechas de creación y actualización.

Relación: Un post pertenece a **una categoría**.

www.codea.app



◆ **Tabla: subscribers**

¿Qué guarda?

Correos de personas que desean recibir novedades.

Campos principales:

- **id**: Identificador único.
- **email**: Correo del suscriptor.
- **created_at**: Fecha de registro.



◆ **Tabla: contacts**

¿Qué guarda?

Mensajes enviados desde el formulario de contacto.

Campos principales:

- **id**: Identificador único.
- **name**: Nombre de quien envía.
- **email**: Correo de quien envía.
- **message**: Mensaje enviado.
- **created_at**: Fecha de envío.





Relaciones Entre Tablas

users → roles

- Cada **usuario** pertenece a **un rol**.
- Ejemplo: Juan Pérez (user) → Rol: **Admin**.

services → categories

- Cada **servicio** pertenece a **una categoría**.
- Ejemplo: "Desarrollo Web" → Categoría: **Servicios Tecnológicos**.

posts → categories

- Cada **publicación** del blog pertenece a **una categoría**.
- Ejemplo: "Consejos de Marketing" → Categoría: **Marketing**.





Notas Importantes

Integridad referencial:

Siempre usaremos claves foráneas (**foreign keys**) para garantizar que los datos estén correctamente conectados.

SEO desde la base de datos:

Cada servicio y publicación tendrá campos específicos para **optimizar el posicionamiento** (slug, SEO title, SEO description).



cap 05



ROLES DE USUARIO

1. **Instalación de Spatie Laravel Permission**
2. **Configuración de Roles y Permisos**
3. **Pruebas**





Roles

1. Admin

- **Acceso total a todas las funcionalidades del sistema.**

2. Editor

- **Acceso limitado únicamente a:**
 - **Categorías**
 - **Servicios**
 - **Posts**



Permisos

Permiso	Descripción	Rol Admin	Rol Editor
Categorías			
view categories	Ver la lista de categorías	✓	✓
create categories	Crear nuevas categorías	✓	
edit categories	Editar categorías existentes	✓	✓
delete categories	Eliminar categorías existentes	✓	
Servicios			
view services	Ver la lista de servicios	✓	✓
create services	Crear nuevos servicios	✓	
edit services	Editar servicios existentes	✓	✓
delete services	Eliminar servicios existentes	✓	
Posts			
view posts	Ver la lista de posts	✓	✓
create posts	Crear nuevos posts	✓	✓
edit posts	Editar posts existentes	✓	✓
delete posts	Eliminar posts existentes	✓	
publish posts	Publicar/despublicar posts	✓	✓
Usuarios			
view users	Ver la lista de usuarios	✓	
create users	Crear nuevos usuarios	✓	
edit users	Editar información de usuarios ex	✓	
delete users	Eliminar usuarios existentes	✓	
assign roles	Asignar/remover roles a usuarios	✓	
assign permissions	Asignar/remover permisos directo	✓	





Reglas

1. El **Admin** tiene todos los permisos en todos los módulos.
2. El **Editor** tiene **permiso completo solo en Categorías, Servicios y Posts.**
3. El **Editor no puede ver ni gestionar Usuarios, Roles, Permisos ni Ajustes Web.**



FIN

Curso Laravel Página Web Administrable

www.codea.app

