

Test Software Developer

GetechnologiesMx



[**Introducción**](#_z3y29xrun7z0) **2**

[**Herramientas**](#_bm3cssyva824) **2**

[**Ejercicios**](#_lmiarw3se81l) **2**

[**Contactos**](#_esaqbksrn33k) **4**

[Larry Ruiz Delgadillo](#_mri7mxuxtwlw) 4

[Rony Ricardo Díaz Ortega](#_uopoqqat9r9x) 4

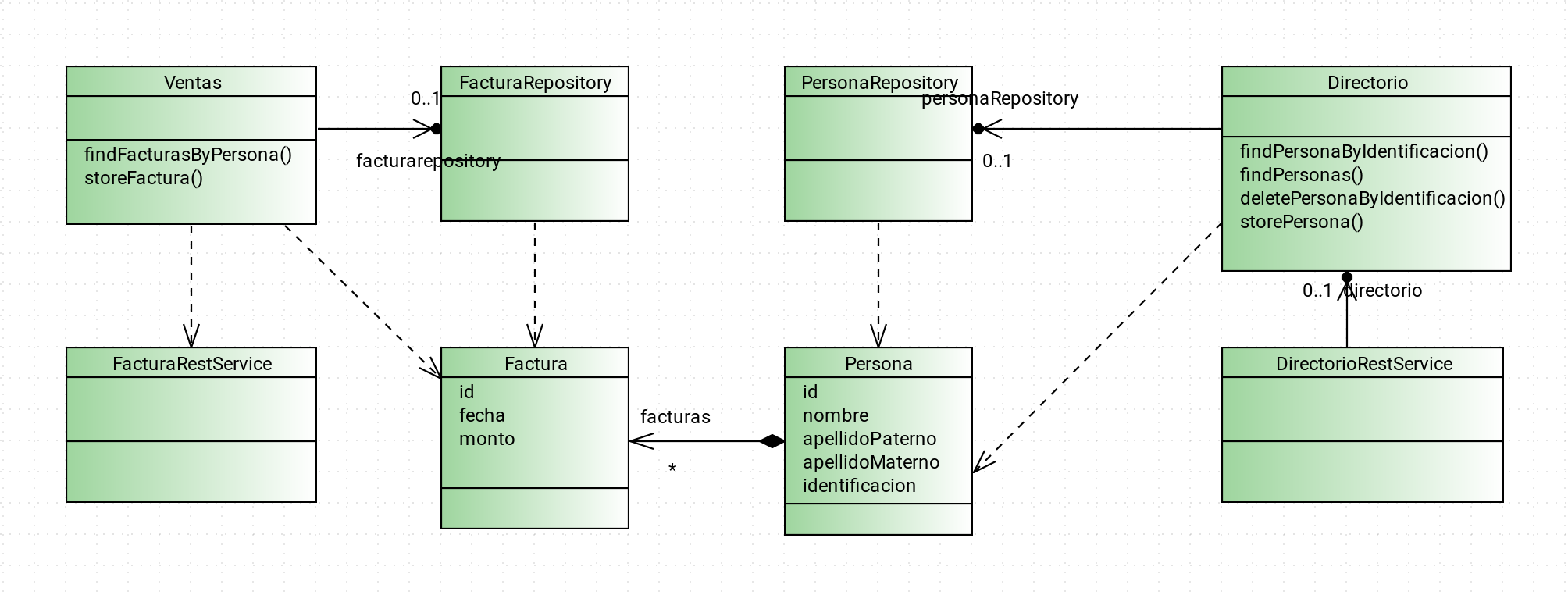
# Introducción

El ejercicio trata de dar evidencia de las capacidades de análisis, diseño, y conocimientos en las tecnologías de interés.

# Herramientas

* Backend - .NET 6 - Repository Design Pattern
* Frontend - WPF - Windows Presentation Framework.
* Postman

# Ejercicios

1. **Crear un proyecto.**
2. **Implementar el siguiente modelo de clases en el proyecto.**

**Hacer referencia al patrón de diseño Repository Design Pattern**

DirectorioRestService: Componente que expone el servicio REST.

Directorio: Servicio de negocio con el objetivo de gestionar la información de personas (almacenar, buscar, borrar), el borrado de una persona elimina todos los datos de una persona incluso su facturacion.

Ventas: Servicio de negocio con el objetivo de gestionar la información de facturacion (almacenar, buscar),

Persona: Clase del modelo de dominio, que representa a una persona.

Factura: Clase del modelo de dominio, que representa las facturas de un persona.

PersonaRepository: Componente encargado ( Repository Design Pattern ) de la interacción del datasource para obtener, reconstruir los datos de las personas.

FacturaRepository: Componente encargado ( Repository Design Pattern ) de la interacción del datasource para obtener, reconstruir los datos de las facturas.

Todos los atributos de la clase Persona son obligatorios para persistir el objeto excepto apellidoMaterno.  
  
Utilizar una base de datos embebida como h2 o la que desees para el ejercicio.

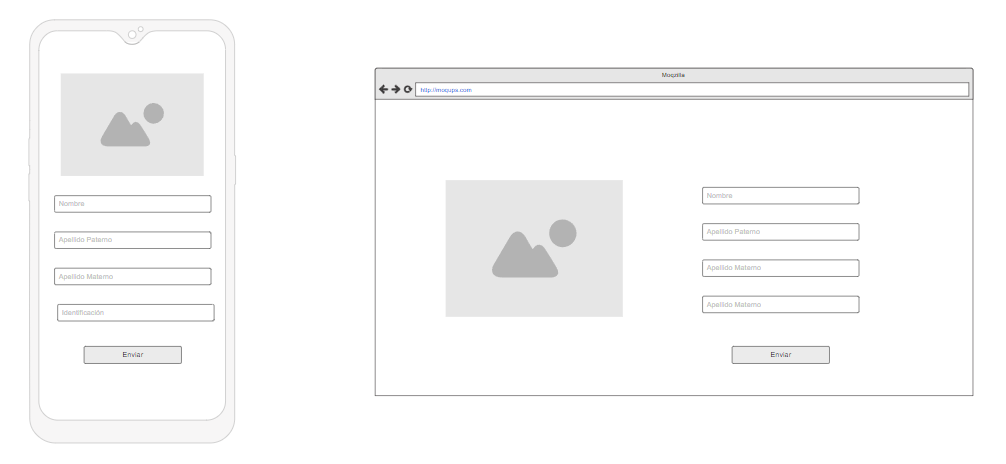
Como plus puedes agregar logger para mostrar en consola la información (elige uno de los métodos).

1. **Realiza un cliente para la API anteriormente creada en el paso 1 y 2 , tomando en cuenta las tecnologías mencionadas en la sección de herramientas. Puedes utilizar el framework de estilos que prefieras o bien, crear tus propios estilos.**

El objetivo es registrar usuarios en el sistema.

Valida los datos que son obligatorios (nombre,ApellidoPaterno,identificación), mostrar mensaje al usuario si no los ha ingresado, de forma reactiva sería excelente.

Puedes darle el diseño desees a tu aplicación, te dejamos un mookup de ejemplo de diseño.



No olvides subir tu código a GitLab o Github y pasarnos el link.

# Contactos

### Larry Ruiz Delgadillo

[lruiz@getechnologiesmx.com](mailto:lruiz@getechnologiesmx.com)

### Rony Ricardo Díaz Ortega

[ricardo.diaz@getechnologiesmx.com](mailto:ricardo.diaz@getechnologiesmx.com)