Proyecto de Testing

Contents

[Propósito 2](#_Toc136111888)

[Framework 2](#_Toc136111889)

[Base de datos 2](#_Toc136111890)

[Proyecto ASP .net Core 2](#_Toc136111891)

[Retos técnicos 3](#_Toc136111892)

[Creación de facturas 3](#_Toc136111893)

[Búsqueda de facturas 4](#_Toc136111894)

[Criterios de evaluación y restricciones 5](#_Toc136111895)

# Propósito

El propósito de este proyecto es evaluar los conocimientos en campo de la arquitectura estándar de una aplicación web en el Stack de tecnologías de Microsoft.

* FrontEnd
* BackEnd
* Base de datos

# Framework

## Base de datos

* El motor de base de datos es SQL server
* Nombre de la base datos DevLab
* Usuario: developer
* Contraseña: abc123ABC
* Cadena de conexión para la aplicación: Server=DESKTOP-D4KSLA3;Database=LabDev;User Id=developer;Password=abc123ABC;Encrypt=False;
* A screenshot of a computer

  Description automatically generatedDiagrama

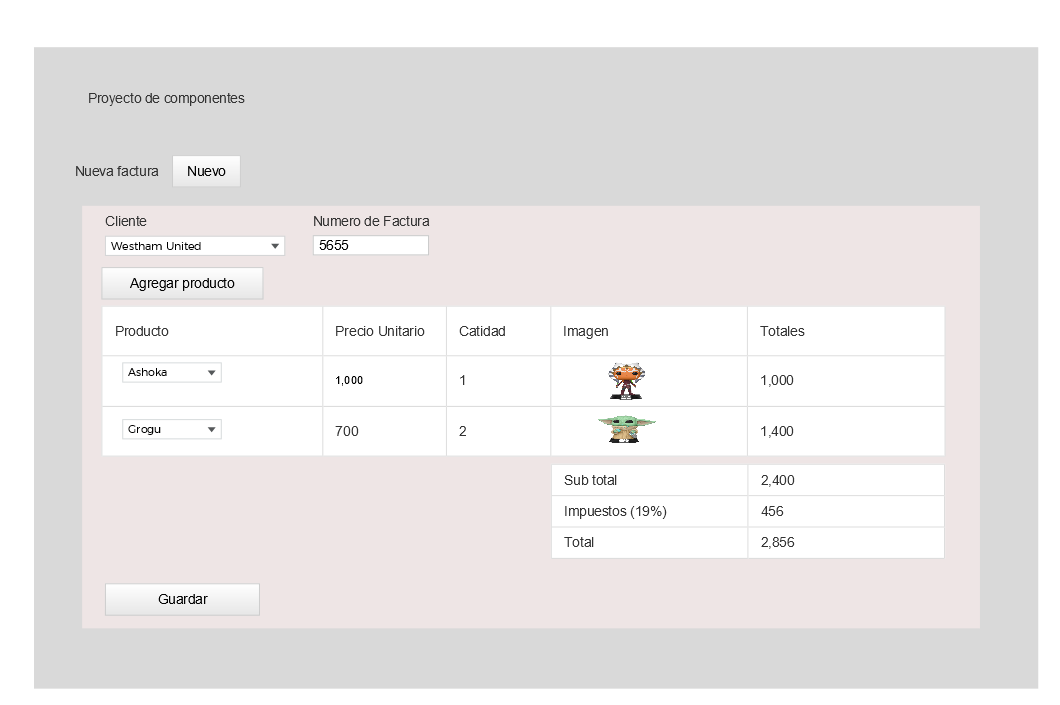
## Proyecto ASP .net Core

* Visual studio code
* SQL Server Managment Studio
* Plantilla básica con setup básico de MVC .Net core

# Retos técnicos

## Creación de facturas

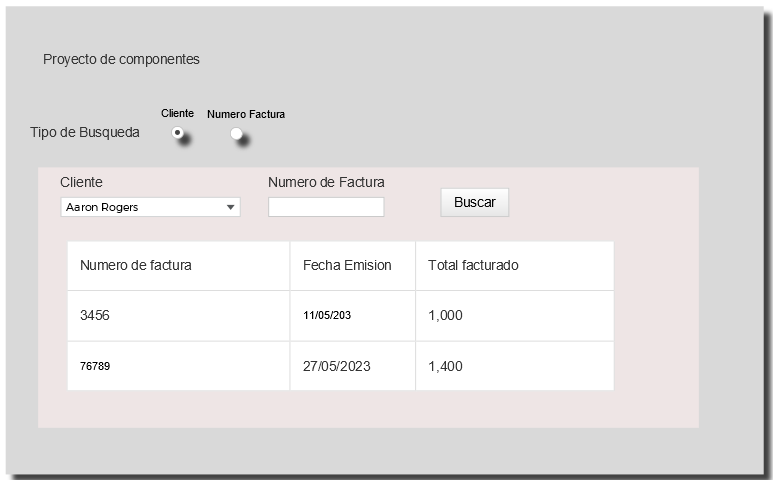
Creación de pantalla básica para creación de “facturas” fiscales.

* Entidades
  + Clientes
  + Encabezado de la factura
  + Productos y precios
* Diseño guía
* Diseño guía
* Reglas de negocio (Funcionalidad)
  + Clic en el botón “nuevo”
    - Limpia los datos de la pantalla
    - Deja la pantalla “lista” para la captura de nueva factura
  + Combo Cliente se nutre con la entidad de “clientes”
  + Numero de Factura es numero y se captura manual, debe ser único
  + Clic en el botón “agregar producto”
    - Agrega una línea de “detalle” a la factura con los datos para capturar de un producto
  + El combo de producto
    - Es nutrido por la entidad de producto
    - Al evento de selección de un producto las siguientes columnas se pueblan automáticamente
      * Precio unitario
      * Imagen
    - Al evento de cambio de valor en la columna cantidad
      * Columna de totales es calculada automáticamente
        + precioUnitario X cantidad = Total del producto
      * El Subtotal es calculado automáticamente
        + Suma de la columna de total de los detalles en la factura
      * Los “impuestos” son calculados automáticamente
        + 19% del campo Subtotal
      * El campo total es calculado automáticamente
        + Subtotal + Impuestos
  + Clic en el botón “guardar”
    - La información es enviada para ser guardada en la base de datos según el diagrama entidad - relación.

## Búsqueda de facturas

Creación de pantalla básica para búsqueda de “facturas” fiscales.

* Diseño guía



* Reglas de negocio (Funcionalidad)
  + Al evento de selección el tipo de búsqueda
    - Se deberá deshabilitar el control no usado en esa búsqueda
      * Ejemplo, si se selecciona “cliente”, el campo de “Numero de factura no podrá ser utilizado.
  + Al evento Clic en el botón “buscar”
    - Se realizará la consulta a base de datos mostrar la información en grid con las columnas indicadas en el diseño guía.

# Criterios de evaluación y restricciones

1. Cumplimento con los requerimientos funcionales indicados en el documento
2. Conceptos de programación orientada objetos.
   1. Front end
   2. Backend
3. Frontend
   1. Puntos de evaluación
      1. Manejo de DOM
      2. Hilos
      3. Tipos de “Request”
      4. Eventos
      5. Principios Solid
   2. Sin restricciones en librerías/frameworks de frontend
4. Backend
   1. Objetos
   2. Estructura de clases
   3. Principios SOLID
   4. Tipos de llamada a la base de datos
   5. Restricciones
      1. Esta prohibido el uso de entity Framework
5. Base de datos
   1. Puntos a evaluación
      1. Uso de objetos de la base de datos
      2. Tolerancia de fallos
      3. Transacciones
      4. Manejo de recursos del servidor
   2. Restricciones
      1. Todas las operaciones de la base de datos serán encapsuladas en store procedures(Consultas, updates, insert, etc.).