**Práctica: Desarrollo de aplicaciones en .NET**

**Presentado por: Paola Andrea Álvarez Betancourt**

Estudiante matrícula no: 4592108 – 790463

Grupo 11

**Presentado a: Ingeniero Javier Díaz Díaz**

Docente de la asignatura

Asignatura: Plataformas de Desarrollo de Software

Especialización Ingeniería de Software

Fundación Universitaria Internacional de La Rioja



Bogotá, D.C., 01 de julio de 2024

**Contenidos**

[1. Introducción 3](#_Toc169622591)

[2. Objetivo 3](#_Toc169622592)

[3. desarrollo del laboratorio 3](#_Toc169622593)

[3.1. Especificaciones del desarrollo 3](#_Toc169622594)

[3.2. Propósito del negocio 3](#_Toc169622595)

[3.3. Muestras del desarrollo del laboratorio 4](#_Toc169622596)

[4. Conclusiones 6](#_Toc169622597)

**Lista de figuras**

[Figura 1 Diligenciamiento del JFrame respectivo a las opciones de pedido por el usuario. 4](#_Toc169622598)

[Figura 2 Prompt cuando el usuario ha oprimido 'CONFIRMAR' y no todos los datos están diligenciados. 5](#_Toc169622599)

[Figura 3 Aviso de caracteres no admitidos en nombre de medicamento. 5](#_Toc169622600)

[Figura 4 Ejemplo de resultado de resumen de pedido y aviso de pedido enviado. 6](#_Toc169622601)

**Laboratorio: Desarrollo de aplicaciones en Eclipse y NetBeans**

# Introducción

En el desarrollo de contenidos de la asignatura Plataformas de Desarrollo de Software de la Especialización en Ingeniería de Software, UNIR, se encuentra propuesto desde la planeación de su contenido el “Laboratorio: Desarrollo de aplicaciones en Eclipse y NetBeans”, el cual, busca profundizar o incluso generar (muchos alumnos nunca habían utilizado Java o IDE que permitieran trabajar con su interfaz gráfica, incluyéndome) en los estudiantes los conocimientos en el manejo de herramientas como Eclipse y NetBeans.

El ejercicio, consistió en simular un sistema de pedidos de medicamentos de una farmacia. Para ello, durante la sesión 3 de las clases en vivo impartidas por el docente Javier Díaz Díaz, fue explicado con ejemplos planteados por él, el uso de la interfaz gráfica de NetBeans y el manejo de código para interacción de dos JFrames. En la clase, el docente procuró dejar en claro la importancia de separar las responsabilidades y que la lógica de negocio quedara por aparte de los códigos de ajuste de interfaz gráfica, no obstante, se procuró manejar por parte de la autora del presente informe, una clase para manejar la interacción entre dos JFrames y en el principal, manejar las condiciones del llenado de campos.

# Objetivo

Generar conocimientos en el manejo de herramientas como NetBeans para la creación de interfaz gráfica de usuario en lógicas de negocio aplicadas a contextos cotidianos.

# desarrollo del laboratorio

## Especificaciones del desarrollo

A continuación, se dan a conocer al docente las especificaciones del desarrollo:

* **Lenguaje de programación:** C#.
* **IDE**: Visual Studio Code
* **Ubicación del desarrollo:** <https://github.com/PaolaAlvarez27/AvancesPlataformasDesarrollo/tree/main/Actividad2>.
* **Nombre del proyecto:** ‘DrugstoreOrders’.
* **Forms:**
  + Forms/Form1.cs
  + Forms/Form1.Designer.cs
  + Forms/Form2.cs
  + Forms/Form2.Designer.cs
* **Modelo de interacción entre Forms:** Models/PedidoManager.cs.
* **Archivo tipo ‘Main’:** Program.cs.
* **Csproj:** DrugstoreOrders.csproj:

<Project Sdk="Microsoft.NET.Sdk">

<PropertyGroup>

<OutputType>Exe</OutputType>

<TargetFramework>net8.0</TargetFramework>

<ImplicitUsings>enable</ImplicitUsings>

<Nullable>enable</Nullable>

</PropertyGroup>

<PropertyGroup>

<OutputType>WinExe</OutputType>

<TargetFramework>net6.0-windows</TargetFramework>

<UseWindowsForms>true</UseWindowsForms>

</PropertyGroup>

</Project>

El último “Property Group” permitió la generación de los Forms

## Propósito del negocio

De acuerdo con los contenidos de la Actividad estipulada, el propósito de negocio corresponde a simular un sistema de pedidos de medicamentos a un distribuidor, que, para ser realizado, el usuario debe indicar la siguiente información con las siguientes condiciones.

**Primer Form:**

* **Nombre del medicamento:** introducido en cuadro de texto editable (JTextField); solo puede contener caracteres alfanuméricos.
* **Tipo de medicamento:** Entre analgésico, analéptico, anestésico, antiácido, antidepresivo y antibióticos, introducidos en un menú desplegable (JComboBox).
* **Cantidad de medicamento:** introducido en cuadro de texto editable; solo puede contener números enteros positivos.
* **Nombre de distribuidor farmacéutico:** introducido con botón de opción o radio (JRadioButton); puede ser entre : Cofarma, Empsephar y Cemefar.
* **Sucursal de la farmacia:** entre principal y secundaria, elegida mediante checkbox (JCheckBox).
* **Botones:** ‘Borrar‘ y ‘confirmar’ (ambos JButton), para eliminar o continuar con el proceso, respectivamente.

**Segundo Form,** el cual, surge con la interacción del primero al pulsarse el ya mencionado botón de ‘Confirmar’ y tiene como criterios mínimos:

* Texto con el medicamento a pedir: «X unidades del T M» donde X es el número de unidades, T es el tipo de medicamento y M el nombre el medicamento (JLabel).
* Texto con la dirección de la farmacia a donde enviarlo: «Para la farmacia situada en D» donde D es la dirección de la farmacia (JLabel); la farmacia Principal cuya dirección es Calle de la Rosa n. 28 y para la secundaria es Calle Alcazabilla n. 3.
* Dos botones (ambos JButton) uno para cancelar el pedido, que cierra la ventana y otro para enviar el pedido que simula un mensaje «Pedido enviado».

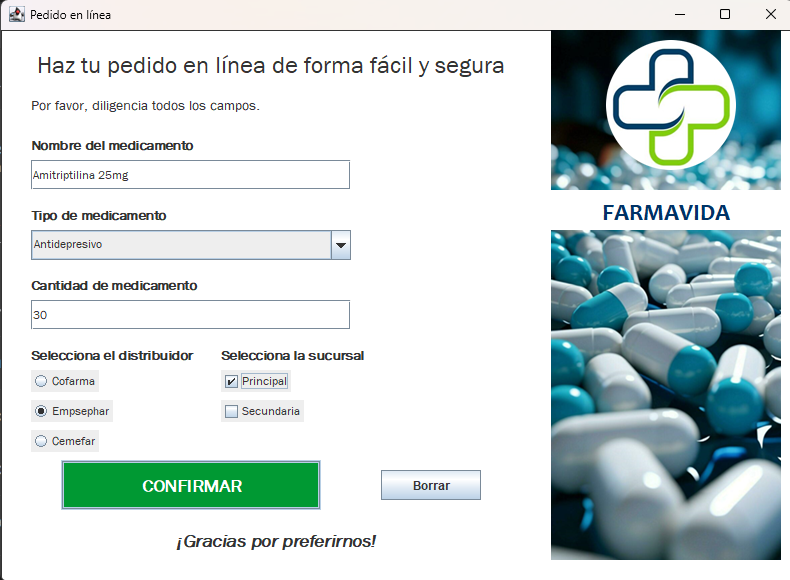
## Muestras del desarrollo del laboratorio

Con respecto a las condiciones nombradas en la sección anterior, se hizo el desarrollo para que fueran incluidas. A continuación, se observan ejemplos de los casos.

* **JFrame 1: ‘OpcionesPedido.java’**

A continuación, se observan algunos ejemplos de interacción con este JFrame principal:

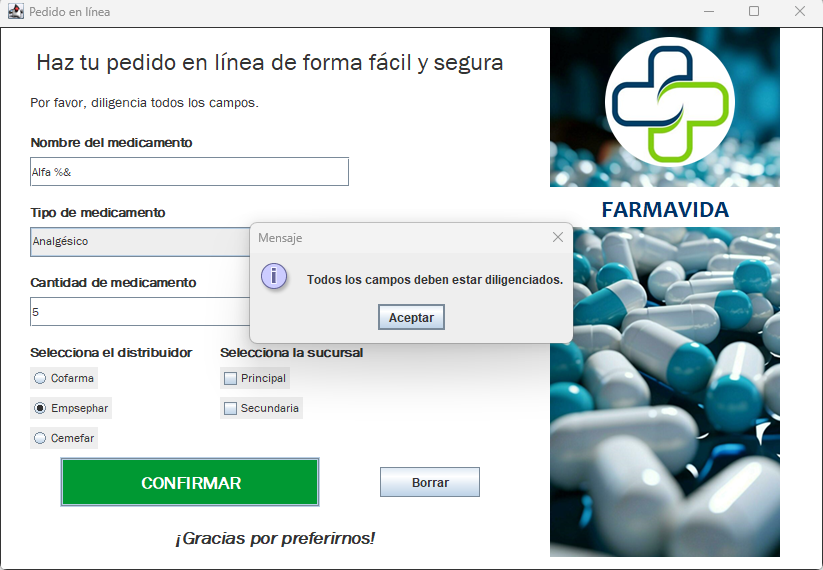
Figura 1 Diligenciamiento del JFrame respectivo a las opciones de pedido por el usuario.



Fuente: Elaboración propia.

Cuando el usuario digita con caracteres o valores no aceptados alguno de los campos y adicional a eso oprime ‘CONFIRMAR’, el prompt obtenido tiene como prioridad el de diligenciamiento de todos los campos, como se observa en la Figura 2.

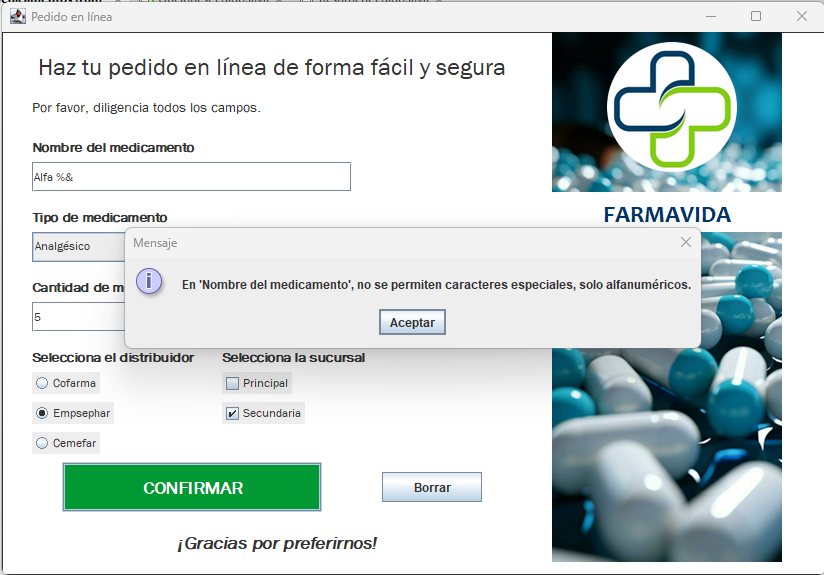
Figura 2 Prompt cuando el usuario ha oprimido 'CONFIRMAR' y no todos los datos están diligenciados.



Fuente: Elaboración propia.

Cuando no es el caso, sino que se han llenado o seleccionado todos los campos, surge el aviso de caracteres no admitidos, como se muestra en la Figura 3

Figura 3 Aviso de caracteres no admitidos en nombre de medicamento.

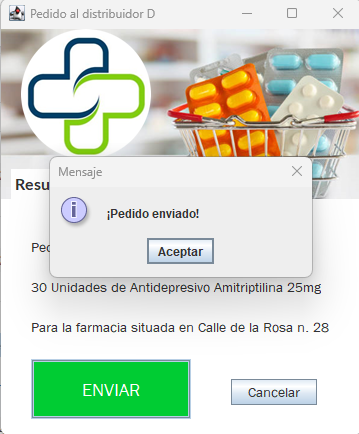


Fuente: Elaboración propia.

* **JFrame 2: ‘ResumenPedido.java’**

La interacción con este JFrame, tal como se describió en la sección ‘3.2 Propósito de negocio’ es resultado del anterior JFrame 1. Como ejemplo, se muestra la Figura 4 en la que es posible observar que se encuentran los mismos datos de la Figura 1; también, se puede ver que se ejecuta el prompt de “¡Pedido enviado!”

Figura 4 Ejemplo de resultado de resumen de pedido y aviso de pedido enviado.



Fuente: Elaboración propia.

Esta interacción es posible gracias a la Clase ‘**PedidoManager.java**’, que se puede evidenciar dentro del proyecto.

# Conclusiones

Se generaron conocimientos básicos sobre el uso de la herramienta NetBeans en el caso de generación de interfaces gráficas de forma adecuada a través del ejercicio planteado en el presente laboratorio, no obstante, es importante tener en cuenta que el trabajo a través de Java por parte de la autora del presente informe debe continuar para adquirir la destreza necesaria para evitar grandes periodos de tiempo en proponer las soluciones de negocio.