

ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN



CARRERA EN INGENIRÍA EN TIC'S

INGENIERÍA DE SOFTWARE II

NRC: 8517

PROFESOR: EFRAIN RODRIGO FONSECA CARRERA

***INTEGRANTES DEL GRUPO 3: MARLON CEVALLOS,
DYLAN JIMENEZ, VALERY NARANJO, ROBERTO
PALLO Y RÓMULO PARDO.***

FECHA: 18/12/2022

ÍNDICE

INTRODUCCIÓN	2
1. OBJETIVOS	2
2. DESARROLLO	3
2.1 Problemática.....	3
2.1.2 Diseño del software con las interfaces requeridas basadas en el paradigma de reutilización.....	4
2.1.3 Depure la arquitectura del sistema propuesta	5
2.2.4 Actualizaciones del software y Mantenimiento inherente a las mismas.....	6
2.2.5 Pruebas de Caja Blanca y Caja Negra	7
3. CONCLUSIONES.....	10
4 BIBLIOGRAFÍA.....	10

ÍNDICE DE ILUSTRACIONES

ILUSTRACIÓN 1. DIAGRAMA DE CASOS DE USO. FUENTE: AUTORÍA PROPIA	3
ILUSTRACIÓN 2. DIAGRAMAS DE SECUENCIA DE LOS ACTORES. FUENTE: AUTORÍA PROPIA.....	4
ILUSTRACIÓN 3. DIAGRAMA DE SECUENCIA PEDIDOS. FUENTE: AUTORÍA PROPIA	4
ILUSTRACIÓN 4. INTERFAZ DE INICIO DE SESIÓN. FUENTE: AUTORÍA PROPIA	4
ILUSTRACIÓN 5. INTERFAZ REGISTRO DE USUARIO. FUENTE: AUTORÍA PROPIA	5
ILUSTRACIÓN 6. REUTILIZACIÓN DE COMPONENTES	5
ILUSTRACIÓN 7. ARQUITECTURA PROPUESTA. FUENTE: AUTORÍA PROPIA.	6
ILUSTRACIÓN 8. CRONOGRAMA PARA PROPUESTA DE MANTENIMIENTO DEL SOFTWARE. FUENTE: AUTORÍA PROPIA	7
ILUSTRACIÓN 9. CÓDIGO FUENTE REGISTRO DE USUARIOS. FUENTE: AUTORÍA PROPIA	7
ILUSTRACIÓN 10. DIAGRAMA DE FLUJO. FUENTE: AUTORÍA PROPIA	8
ILUSTRACIÓN 11. GRAFO REALIZADO EN BASE AL CÓDIGO. FUENTE: AUTORÍA PROPIA	8
ILUSTRACIÓN 12. TABLA DE VALIDACIÓN DE REGISTROS. FUENTE: AUTORÍA PROPIA	9
ILUSTRACIÓN 13. VALIDACIÓN DE INGRESO DE DATOS. FUENTE: AUTORÍA PROPIA	9
ILUSTRACIÓN 14. VALIDACIÓN DE INGRESO DEL CORREO. FUENTE: AUTORÍA PROPIA.....	9
ILUSTRACIÓN 15. VALIDACIÓN DE REGLA CAMPOS VACÍOS. FUENTE: AUTORÍA PROPIA.....	9
ILUSTRACIÓN 16. VALIDACIÓN DE DATOS MAL TIPIADOS. FUENTE: AUTORÍA PROPIA	10

ÍNDICE DE TABLAS

TABLA 1. TABLA DE PROPUESTA DE MANTENIMIENTO Y ACTUALIZACIÓN.....	6
---	---

INTRODUCCIÓN

Desarrollo de un software para el restaurante "Mis Delicias", en el cual se visualizará los menús que ofrece el mismo. Para una mejor comprensión de lo realizado en el siguiente proyecto, se presentarán el diagrama de uso del sistema a utilizar, diagrama de secuencia, arquitectura, etc.

1. Objetivos

Objetivo General

Proponer y desarrollar un software a partir de un caso práctico que se ha planteado, proponiendo una problemática a ser abordada conociendo las características y los requerimientos generales del sistema, además, de considerar una posible evolución del sistema.

Objetivos Específicos

- Diseñar el software con las interfaces requeridas basándose en el paradigma de reutilización.
- Proponer actualizaciones del software diseñado y realizar su mantenimiento.
- Proponer los respectivos casos de prueba de caja blanca y de caja negra.

2. Desarrollo

2.1 Problemática

El restaurante "Mis Delicias" requiere una aplicación web que le permita visualizar los platillos y menús que ofrece a sus clientes, automatizar el proceso de pedidos y la gestión de clientes en el cual se registrará los datos personales y ubicación del cliente para su entrega a domicilio.

2.1.1 Requerimientos generales del sistema

a) Diagramas de casos de uso

El Diagrama de Caso de Uso ayuda a representar el comportamiento que se espera del sistema o software. En este Diagrama de Caso de Uso se tiene la intervención de dos actores, el primer actor es el cliente y el segundo es el administrador.

El actor cliente puede registrarse, iniciar sesión, visualizar recetas, seleccionar recetas, agregar recetas o menús al carrito de compras, confirmar y generar facturas de pedidos. Así mismo el actor administrador y Propietario es quien se encarga de la gestión de usuarios, gestión de las recetas y gestión de formas de pago respectivamente juntamente con la visualización del reporte de ventas del día.

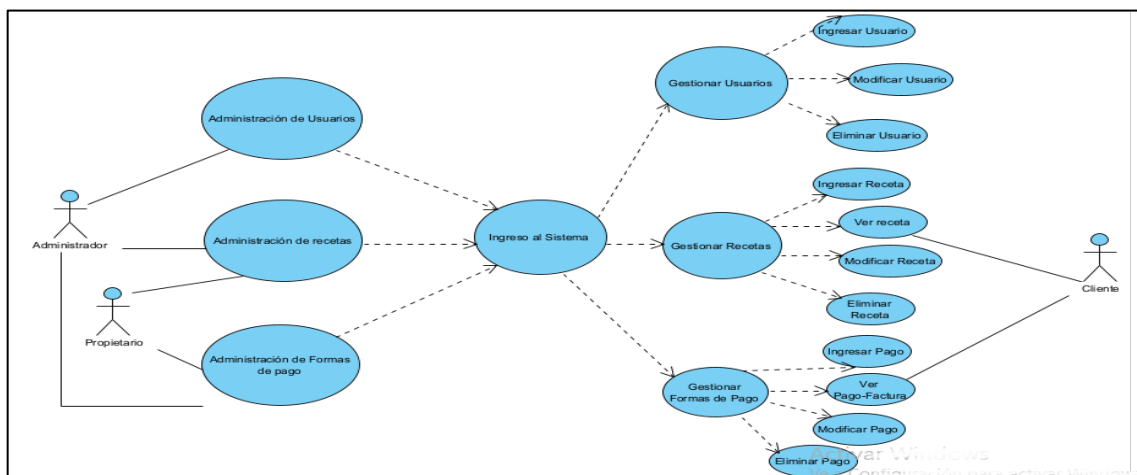


Ilustración 1. Diagrama de casos de uso. Fuente: Autoría propia

b) Diagrama de Secuencia

Muestra la forma en que se comunican o interactúan los objetos del sistema con nuestros actores (Cliente, Propietario y Administrador) a lo largo de una línea de vida, que representa en nuestro caso los pedidos de recetas, menú del día o platos fuertes del Restaurante "Mis Delicias".

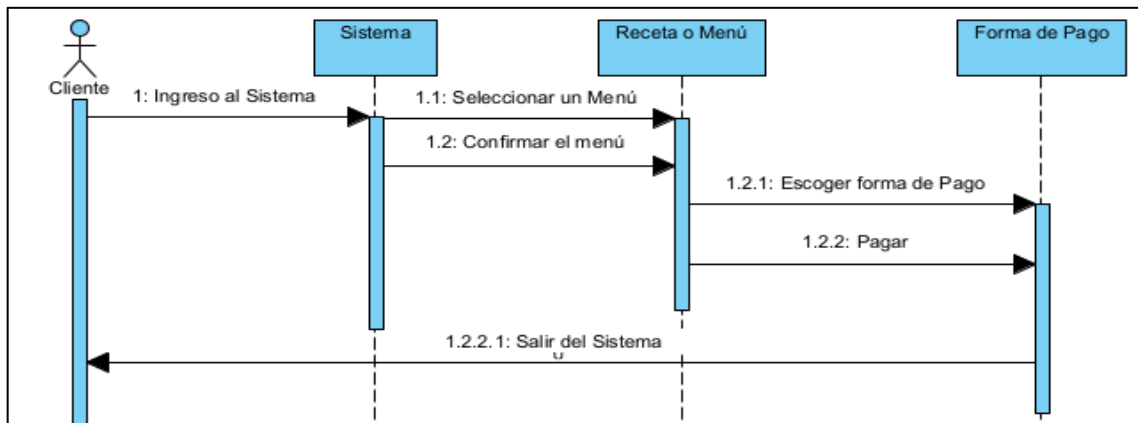


Ilustración 2. Diagramas de secuencia de los actores. Fuente: Autoría propia

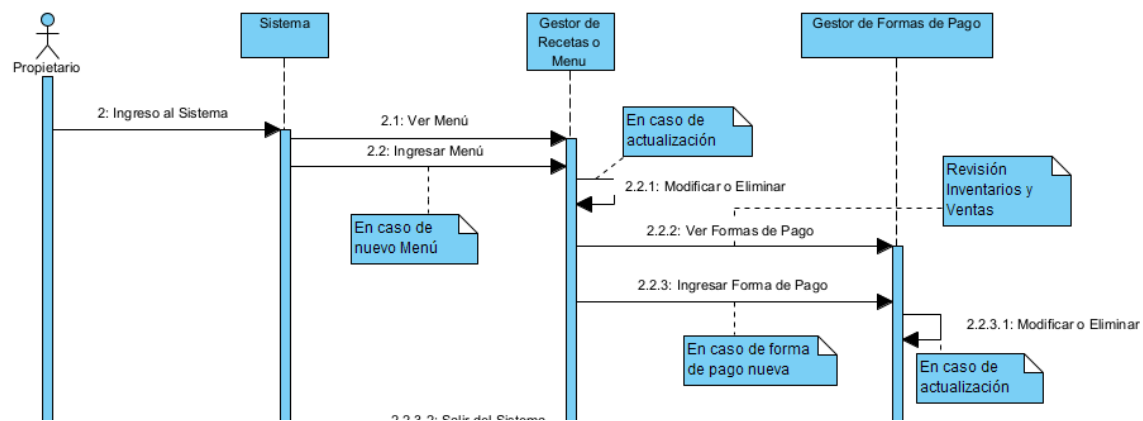


Ilustración 3. Diagrama de secuencia pedidos. Fuente: Autoría propia

c) Marco de Trabajo-Historias de Usuario

Se enfoca en la perspectiva general de lo que se va a realizar en el sistema en función del usuario final del restaurante “Mis Delicias”. Con el propósito de generar un elemento de trabajo o pequeñas descripciones de los requerimientos de los clientes.

Historias de Usuario.

La presentación del Marco de Trabajo se puede visualizar en el siguiente link:

[MATRIZ_MARCO_TRABAJO_HU_MISDELICIAS.xlsx](#)

2.1.2 Diseño del software con las interfaces requeridas basadas en el paradigma de reutilización

Interfaz de inicio de sesión

Ilustración 4. Interfaz de inicio de sesión. Fuente Autoría propia

Interfaz de registro de usuario

Mis Delicias
¿Ya tienes una cuenta?
Inicia sesión para entrar en la página

Registrarse

Nombre Completo
Correo Electronico
Dirección Entrega
Usuario
Contraseña

Ilustración 5. Interfaz registro de usuario. Fuente: Autoría propia

En nuestras interfaces, reutilizamos algunos de sus componentes del formulario de registro en el formulario de iniciar sesión como por ejemplo sus textbox, buttons, placeholder, textos o colores, también se reutilizan ciertas validaciones como los campos requeridos o formatos.

Adicional, podemos ver cómo reutilizamos una sola página para tener dos funciones (iniciar sesión y registro de usuario).

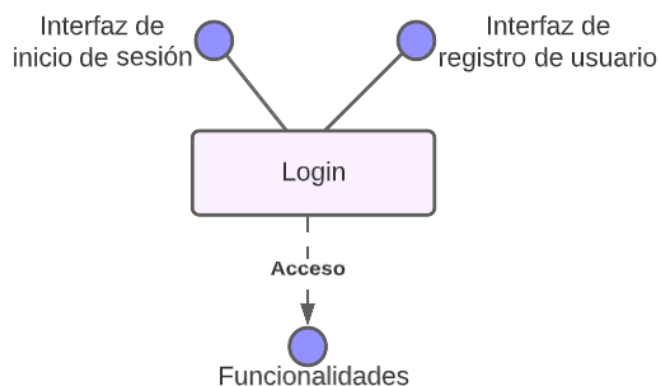


Ilustración 6. Reutilización de componentes

2.1.3 Depure la arquitectura del sistema propuesta

La arquitectura consta de:

Base de Datos

- My SQL: mismo que es un sistema de gestión de bases de datos relacional. [3]

Servidor

- php: es un lenguaje de código abierto adecuado para el desarrollo web. [4]

Cliente

- HTML: es el responsable de darle estructura a la información dentro del sitio.
- CSS: es donde se agrega diseño como: colores, fuentes, sombras, etc.
- JS: se encarga del componente interactivo como: animaciones o respuestas, cuando se ejecuta una acción. [5]

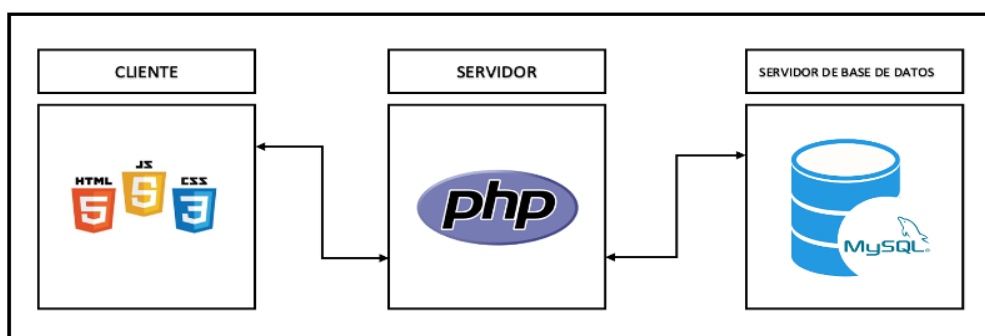


Ilustración 7. Arquitectura propuesta. Fuente: Autoría propia.

2.2.4 Actualizaciones del software y Mantenimiento inherente a las mismas

Las actualizaciones son parte del soporte del software, por tanto, se debe realizar actualizaciones como de seguridad para detectar alguna vulnerabilidad y no exponerse a un gran riesgo o sencillamente para realizar mejoras al software, así como el rendimiento (corrección de errores de funcionamiento) y su utilidad.

Por otro lado, el mantenimiento del software es una de las actividades más importantes y comunes a realizar después de que un software empieza a ser utilizado, generalmente se ejecuta un mantenimiento no solo para modificar y corregir, sino que permite que se conozca el estado, las condiciones de funcionamiento además que se puede aprovechar para mejorar el rendimiento del programa o adaptarlo a otras áreas de ser necesario.

Con lo dicho anteriormente y de acuerdo a los requerimientos generales y específicos que solicita el cliente se propone los siguientes actualizaciones y mantenimiento.

Tabla 1. Tabla de propuesta de mantenimiento y actualización

Aplicación Web Restaurante “MIS DELICIAS		
Propuesta de mantenimiento y actualización	Fecha:	Versión: 1.0
ACTUALIZACIÓN	MANTENIMIENTO	ESTADO
Implementar requerimientos generales al sistema	Mostrar el Menú con los servicios de alimentos que ofrece el restaurante para que el usuario podrá realizar su elección	En proceso
Implementar requerimientos adicionales al sistema	Crear y activar los botones de compra para que el usuario pueda agregar el menú favorito	En proceso
Implementar reglas de negocio para el funcionamiento de compra en línea	Obtener el precio total de los platos seleccionados del menú	En proceso

Además de contar un plan de mantenimiento en el cual se van a detallar las actividades de realizar y fechas de entrega.

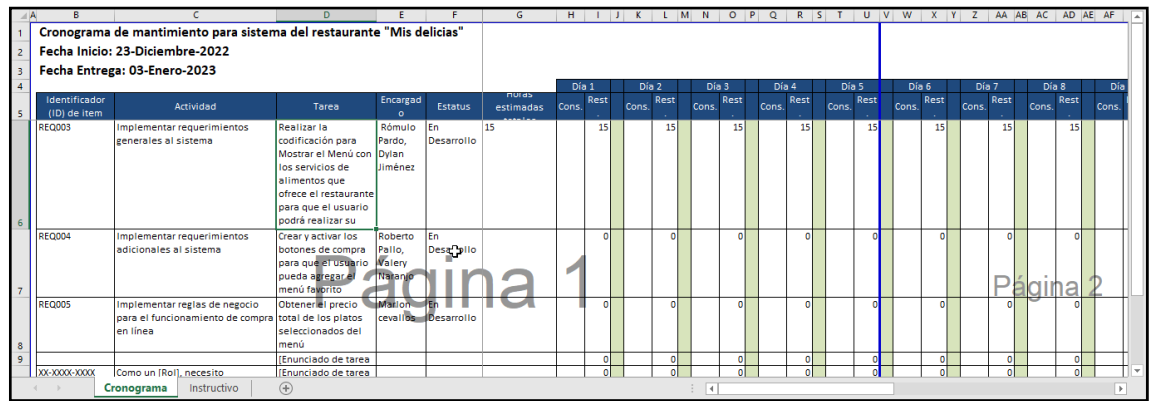


Ilustración 8. Cronograma para propuesta de mantenimiento del software. Fuente: Autoría propia

2.2.5 Pruebas de Caja Blanca y Caja Negra

Pruebas de Caja Blanca

Registro usuarios
Código Fuente

```
include 'conexion_db.php';

$nombre completo = $_POST['nombre'];
$correo = $_POST['correo'];
$direccion = $_POST['direccion'];
$usuario = $_POST['usuario'];
$contraseña = $_POST['contraseña'];
$encriptacion = hash('sha512', $contraseña);

$query = "INSERT INTO usuarios (Nombre, Correo, Direccion, Usuario, Contraseña)
VALUES ('$nombre completo', '$correo', '$direccion', '$usuario', '$contraseña')";

$verificar_usuario = mysqli_query($conexion, "SELECT * FROM usuarios WHERE Correo = '$correo' OR
Usuario = '$usuario'");
if(mysqli_num_rows($verificar_usuario) > 0){
    echo '
    <script>
        alert("El usuario o correo ya existen. Intente registrarse con otros datos.");
        window.location = "../index.php";
    </script>
    ';
    exit();
}
$execute = mysqli_query($conexion, $query);
if($execute){
    echo '
    <script>
        alert("Usuario registrado exitosamente.");
        window.location = "../index.php";
    </script>
    ';
}
else{
    echo '
    <script>
        alert("Hubo un problema, intentar nuevamente.");
        window.location = "../index.php";
    </script>
    ';
}
```

Ilustración 9. Código Fuente registro de usuarios. Fuente: Autoría propia

Diagrama de flujo

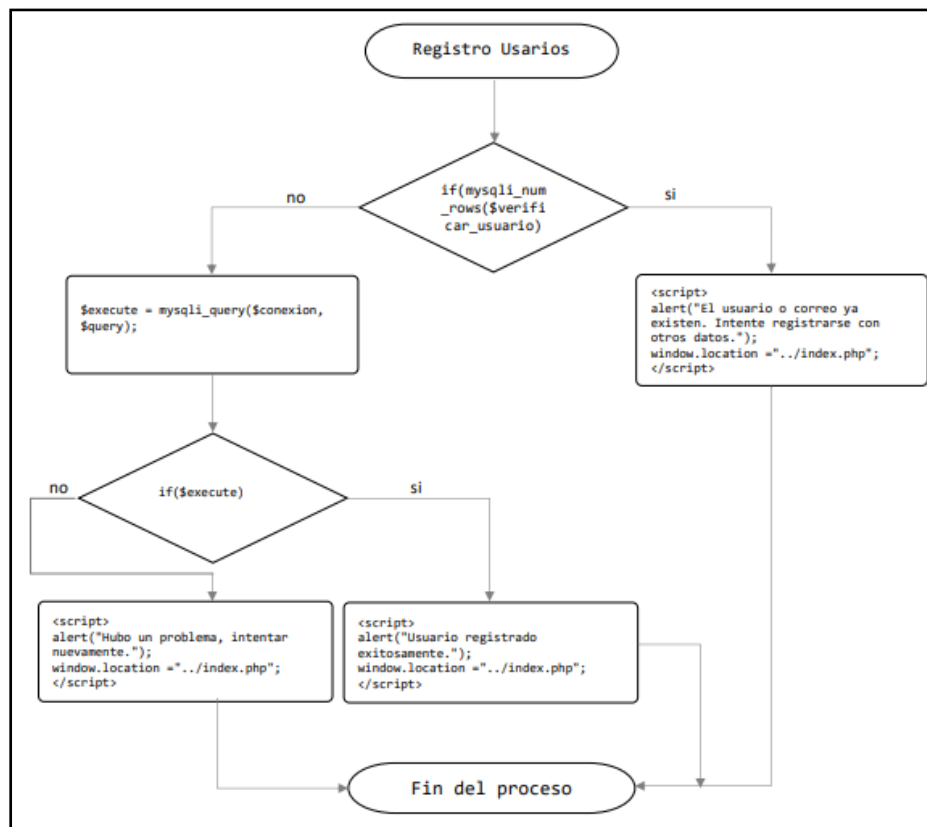


Ilustración 10. Diagrama de flujo. Fuente: Autoría propia

Grafo

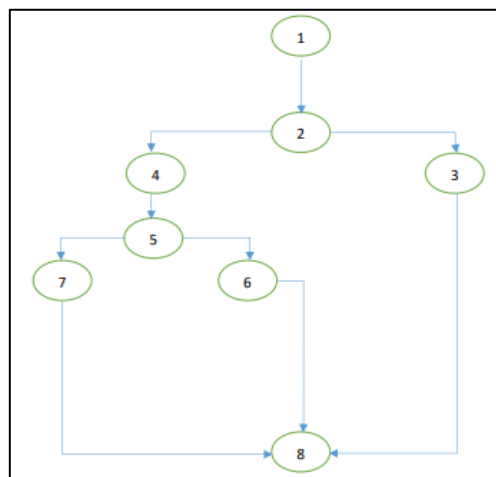


Ilustración 11. Grafo realizado en base al código. Fuente: Autoría propia

Rutas

R1: 1,2,3,8

R2: 1,2,4,5,6,8

R3: 1,2,4,5,7,8

Complejidad Ciclomática

Se puede calcular de las siguientes formas:

- $V(G) = \text{número de nodos predicaados(decisiones)} + 1 = 3$
- $V(G) = A - N + 2 = 9 - 8 + 2 = 3$

DONDE:

P: Número de nodos predicado

A: Número de aristas

N: Número de nodos.

Prueba de Caja Negra

Validación de registro de nuevo usuario

VARIABLE	CLASE DE EQUIVALENCIA	ESTADO	REPRESENTANTE
REGISTRO (validación de los campos de nuevo usuario)	EC1: x = ingreso de datos correcto	Válido	User
	EC2: x = correcta digitación de correo	Válido	marlon@gmail.com
	EC3: x = campos vacíos	No válido	NULL
	EC4: x = correo incorrecto x ¢ @	No válido	dylan13gmail.com

Ilustración 12. Tabla de validación de registros. Fuente: Autoría propia

Correcto ingreso de datos

The screenshot shows a registration form titled 'Registrarse' on the right and a login section titled 'Mis Delicias' on the left. The login section asks '¿Ya tienes una cuenta?' and has an 'Iniciar Sesión' button. The registration section has four input fields: 'Nombre Completo' (filled with 'dylan'), 'Correo Electrónico' (filled with 'dylanjim@gmail.com'), 'Dirección Entrega' (filled with 'magdalena'), and 'Usuario' (filled with 'dylan'). The 'Contraseña' field is masked with dots. A red 'Registrarse' button is at the bottom right.

Ilustración 13. Validación de Ingreso de datos. Fuente: Autoría propia

Ingreso correcto del correo

This screenshot is similar to the previous one but highlights the email field. The 'Correo Electrónico' field is filled with 'marlon@gmail.com'. The other fields are filled with 'marlon', 'quito', and 'marlon'. The 'Registrarse' button is visible at the bottom right.

Ilustración 14. Validación de ingreso del correo. Fuente: Autoría propia

Registro sin ingreso de datos

The screenshot shows the registration form with all input fields empty: 'Nombre Completo', 'Correo Electrónico', 'Dirección Entrega', 'Usuario', and 'Contraseña'. A yellow tooltip with an exclamation mark and the text 'Completa este campo' is visible over the 'Correo Electrónico' field. The 'Registrarse' button is at the bottom right.

Ilustración 15. Validación de regla campos vacíos. Fuente: Autoría propia

Registro incorrecto de usuario (correo mal tipeado)



Ilustración 16. Validación de datos mal tipados. Fuente: Autoría propia

Link al video de YouTube

<https://youtu.be/B7o4xcRp29c>

3. Conclusiones

- En el presente documento se realiza el diseño del software para la problemática propuesta, con lo cual se da cumplimiento al objetivo planteado. Lo más importante en este apartado es que se logró diferenciar cada etapa del desarrollo del software con ello se destaca la importancia que tiene cada parte considerando que las partes que el mantenimiento y actualización son las etapas más relevantes en vista que es aquí en donde se empieza a ver los posibles fallos o prevenir la aparición de errores.
- En etapas previas del desarrollo de un sistema o aplicativo podemos ver que resulta de gran ayuda realizar los diagramas UML como Casos de Uso y/o Secuencia que permite tener una visión general hacia dónde debemos dirigirnos y qué personajes o actores representan un punto fuerte en la empresa.
- El desarrollo del proyecto permite entender la importancia de las pruebas de software, dentro de ello la prueba de caja blanca, permitió realizar un análisis del diseño realizado, y conocer la estructura interna del mismo, por otro lado, la prueba de caja negra permite conocer posibles errores en la interfaz.

4 Bibliografía

- [1] La importancia de un buen mantenimiento del software de tu empresa. (2020, octubre 7). *Grupo SIM*. <https://gruposim.eu/blog/la-importancia-de-un-buen-mantenimiento-del-software-de-tu-empresa/>
- [2] Alfonso. (2019, abril 5). *Actualizaciones de software: qué son, para qué sirven, cuándo instalarlas*. idearius. <https://www.idearius.com/es/blog/actualizaciones-de-software-que-son-para-que-sirven-cuando-instalarlas/>
- [3] (2022). *MySQL*. Recuperado de <https://www.computerweekly.com/es/definicion/MySQL>
- [4] (2022). *¿Qué es PHP?*. Recuperado de <https://www.php.net/manual/es/intro-whatis.php>
- [5] Vega. F. *¿Qué es HTML/CSS/JS?*. Recuperado de <https://platzi.com/clases/1050-basico-programacion/5104-que-es-htmlcssjs/>