

TAREA 1,2 y 3:

Alumno: Antonio Jiménez Cortés

A) DESCRIPCIÓN DETALLADA DE LA EMPRESA

El nombre de la empresa es Barbería Guaniros y se trata de una empresa que realiza cortes de pelo para caballero y cuidado estético. Además, tiene un servicio de venta de productos de peluquería y de belleza. Esta empresa tiene dos sedes en Granada.

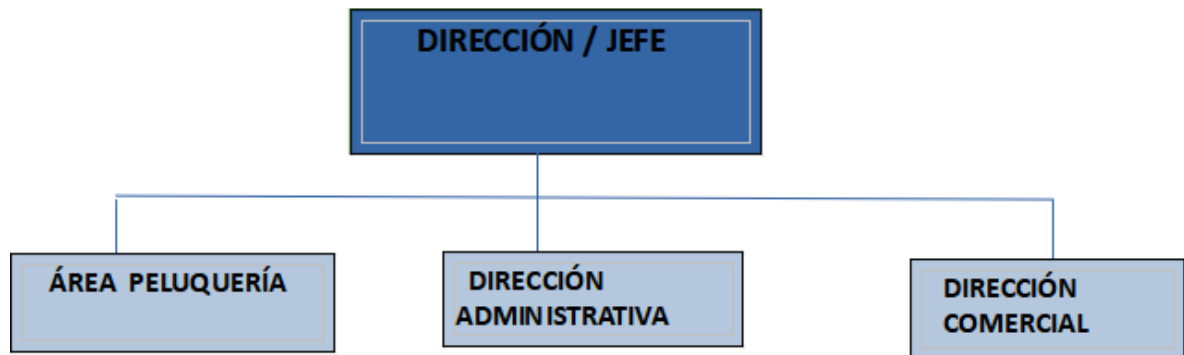
Una **sede** es dedicada al salón de barbería donde se realizan las siguientes actividades:

- Corte de pelo para niños y caballeros. Podrá ser a tijera, a máquina o ambas.
- Lavado de cabello.
- Peinado de cabello.
- Coloración de cabello mediante productos químicos. Podrá ser coloración completa del cabello o solo mechas.
- Afeitado clásico de barba: afeitado completo con toalla caliente y navaja.
- Arreglo de barba: arreglo de barba a máquina o a tijera.

La segunda **sede** está dedicada a la venta de productos de barbería y peluquería. Los productos que se ofertan son los siguientes:

- Productos para cabello: champús, mascarillas, acondicionadores, keratina, laca, cera, gomina, tintes.
- Productos eléctricos: secadores de pelo, planchas, difusores, máquinas de corte.
- Productos para cuidado de barba: navajas, cuchillas, crema de afeitar, aftershave, aceites, bálsamos, crema hidratante.
- Accesorios: peines, cepillos, pinzas, tijeras.

ORGANIGRAMA:



Dirección/Jefe: El jefe se llama José López Jiménez y tiene las siguientes funciones:

- Realiza la planificación diaria.
- Supervisa el área de peluquería, la dirección administrativa y la dirección comercial.
- El jefe también es barbero por lo que realiza servicios de cortes de pelo y arreglos de barba.
- Contratar personal para las distintas áreas.
- Organizar los turnos de los empleados.
- Llevar los registros del personal (salario y horas trabajadas).
- Encargado de contabilizar los ingresos y gastos.
- Gestionar las redes sociales.

Área peluquería: hay dos barberos contratados, Eduardo Ruiz Gómez y Alejandro Ramírez Barranco. Se encuentran en el salón de barbería. Sus funciones son las siguientes:

- Realizar los servicios de cortes de pelo, arreglo de barba y coloración de cabello.
- Realizar los lavados de cabello.
- Asesorar a los clientes sobre el tipo de corte a realizar más adecuado.

-Asesorar a los clientes sobre los productos necesarios para el cuidado de cabello y para el cuidado de barba.

-Mantener una higiene adecuada en todas las herramientas y accesorios.

Dirección administrativa: hay una persona contratada para ello y se llama María Fernández Romero. Se encuentra en el salón de barbería. Sus funciones son:

-Concertar citas previas mediante llamada telefónica.

-Recibir el pago de los clientes.

-Recibir a los clientes a su entrada.

-Gestionar tickets y facturas.

-Dar información a los clientes.

Dirección comercial: hay una persona contratada en esta área y se llama Roberto Torres Ruiz. Se encuentra en la sede donde se venden los productos. Sus funciones son:

-Vender los productos de cara al público.

-Realizar los pedidos y compras que le encomiende el jefe.

-Recibir el pago de los productos.

-Gestionar facturas y contabilidad.

-Dar soporte al cliente sobre los productos que necesita o se adecúan más a su necesidad.

Gestión actual de la información: en ambas sedes posee una TVP (terminales punto de venta) que les permite gestionar todo el proceso compra y venta de productos, y el pago de los clientes en el salón.

B) DESCRIPCIÓN DE LOS RECURSOS DISPONIBLES EN LA EMPRESA

Este negocio posee un componente principal en ambas sedes y se trata de un Terminal Punto de Venta (TPV). Este componente está compuesto por diferentes elementos:

-Software: es el elemento principal y específico para el ámbito de barbería. Este software permite el cobro de pago de clientes, impresión de facturas, visualizar el catálogo de productos y gestionar devoluciones.

-Hardware: incluye ordenador, impresora de tickets, cajón Portamonedas y lector de código de barras.

Ordenador: tiene un sistema operativo Windows 7, pantalla táctil (15 pulgadas), procesador de núcleos, memoria RAM de 4GB y disco duro de 256GB.

Impresora de Tickets: impresora térmica para las facturas.

Cajón de Portamonedas: cajón donde se guarda el dinero en efectivo.

Lector de código de barras: permite acceder al producto leyendo el código de barras impreso.

Además, este negocio posee *Redes Sociales*, principalmente *Instagram*, donde se muestra contenido de los cortes de pelos y servicios realizados mediante imágenes. También se muestran algunos de los productos más vendidos y más solicitados por los clientes. Nos informa también sobre la dirección de las sedes y los teléfonos de contacto.

C) DESCRIPCIÓN DE LOS REQUISITOS PARA LA APLICACIÓN

La aplicación para este comercio, será una aplicación de escritorio en la que implementaremos un diseño ERP. Esta app constará de diversas secciones que serán: contabilidad, gestión de clientes, almacén productos, gestión de caja y gestión de citas.

En esta aplicación solo el jefe podrá acceder a la sección de contabilidad, donde podrá ver dos tipos de documentos. Un documento con los ingresos mensuales, trimestrales y anuales, y otro documento con gastos mensuales, trimestrales y anuales.

Los trabajadores pertenecientes a la dirección administrativa y comercial tendrán acceso a la gestión de clientes, almacén de productos y gestión de caja.

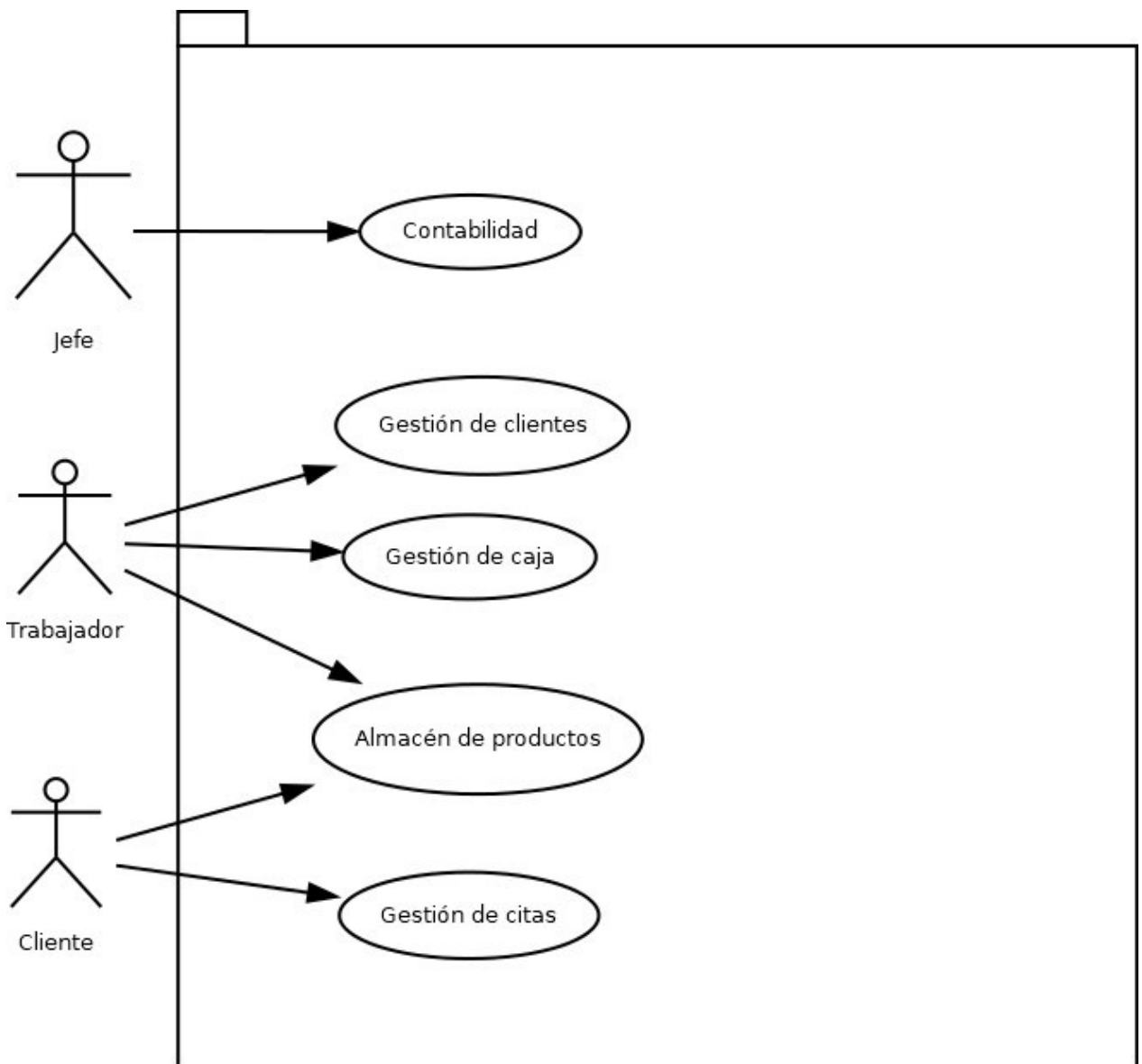
En la sección de clientes el trabajador podrá dar de alta a los clientes. A cada cliente se le asignará un usuario y contraseña.

En la sección gestión de caja el trabajador podrá calcular el precio de servicio y/o productos solicitados. En el área de peluquería se marcará cada uno de los servicios prestados y se hará un cómputo global de la cantidad que debe abonar el cliente. La aplicación generará una factura que el cliente podrá pagar en efectivo o con tarjeta. En el área comercial se realizará el mismo proceso que en la peluquería, pero hará un cómputo global de los productos solicitados.

En la sección almacén de productos el trabajador podrá añadir nuevos productos al stock y/o retirar productos no deseados del stock.

El cliente tendrá acceso a la sección de gestión de citas introduciendo el usuario y contraseña facilitado por el negocio. Podrá visualizar las citas disponibles, donde elegirá el día y la hora. Después deberá confirmar la cita. Una vez confirmada la cita podrá cancelar la cita o cambiar por otro día u otra hora. El cliente puede acceder a la sección de productos, pero solo podrá visualizar el stock disponible en tienda. Para comprar algún producto deberá acudir físicamente a la sede.

D) DIAGRAMA DE CASOS DE USO



E) DESCRIPCIÓN DE LOS ACTORES

[Jefe]

Actor	[Jefe]	Identificador: DIR-01
Descripción	El jefe es el dueño de la empresa y supervisa a todos los trabajadores.	
Características	El jefe realiza la planificación diaria.	
Relación	Se relaciona con los trabajadores para darles instrucciones que deben realizarse en su trabajo y se relaciona con los clientes si estos desean hablar con él explícitamente.	
Referencias	Interviene en Gestión de la contabilidad.	

[Trabajador]

Actor	[Trabajador]	Identificador: TR-02
Descripción	El trabajador o trabajadores, son los empleados contratados por el jefe.	
Características	Los trabajadores pueden pertenecer al área administrativa o la dirección comercial.	
Relación	Se relaciona con los clientes para recomendarle productos y calcular el precio de los productos y servicios.	
Referencias	Interviene en Gestión de clientes, Gestión de caja y Almacén de productos.	

[Cliente]

Actor	[Cliente]	Identificador: CL-03
Descripción	Cliente cliente que compra productos y/o se realiza un corte de cabello.	
Características	El cliente acudirá al almacén para comprar productos o acudirá al salón para realizarse un corte.	
Relación	Se relaciona con los trabajadores para realizar los pagos de productos solicitados y/o pagar los gastos de los servicios del salón.	
Referencias	Interviene en Almacén de productos y en Gestión de citas.	

F) ESPECIFICACIÓN DE CASOS DE USO

[Contabilidad]

Caso de Uso	[Contabilidad]	Identificador: CU-001
Actores	Interviene el actor Jefe	
Tipo	Caso de uso primario	
Secuencia de pasos	1- Acceder al sistema con usuario y contraseña. Si no permite iniciar sesión, ir a Excepción #1. 2- Acceder a la gestión de contabilidad. 3-Selección de documentos: 3.1-Documentos de ingresos: 3.1.1- Ver o editar ingresos mensuales 3.1.2-Ver o editar ingresos trimestrales 3.1.3-Ver o editar ingresos anuales 3.2-Documentos de gastos: 3.2.1- Ver o editar gastos mensuales 3.2.2-Ver o editar gastos trimestrales 3.2.2-Ver o editar gastos anuales	
Excepción	1-Solicitar recuperación de contraseña mediante correo electrónico.	
Precondición	Debe tener una cuenta de administrador facilitada por el desarrollador del software.	
Postcondición	Ninguna.	

[Gestión de clientes]

Caso de Uso	[Gestión de clientes]	Identificador: CU-002
Actores	Interviene el actor Trabajador.	
Tipo	Caso de uso primario	
Secuencia de pasos	1- Acceder al sistema con usuario y contraseña. Si no permite iniciar sesión, ir a Excepción #1. 2- Acceder a la gestión de clientes. 3- Dar de alta a cliente 4- Rellenar datos personales: nombre, apellido,teléfono,dirección. 5-Adjudicar un usuario y contraseña para cada cliente.	
Excepción	1-Solicitar recuperación de contraseña mediante correo electrónico.	
Precondición	El trabajador tener una cuenta personal creada anteriormente facilitada por el Jefe.	
Postcondición	Ninguna.	

[Gestión de caja]

Caso de Uso	[Gestión de caja]	Identificador: CU-003
Actores	Interviene el actor Trabajador.	
Tipo	Caso de uso primario	
Secuencia de pasos	1- Acceder al sistema con usuario y contraseña. Si no permite iniciar sesión, ir a Excepción #1. 2- Acceder a la gestión de caja. 3- Realizar el cómputo de todos los productos y/o servicios de barbería prestados. 4- Generar factura para el cliente. 5- Imprimir factura. Si el cliente no desea factura, ir a Excepción #2. 6-Solicitar pago en efectivo o pago con tarjeta bancaria.	
Excepción	1-Solicitar recuperación de contraseña mediante correo electrónico. 2- Elige la opción no imprimir y seleccionar la opción de guardar factura en el sistema .	
Precondición	El trabajador debe tener una cuenta personal creada anteriormente facilitada por el Jefe. El cliente debe haber solicitado algún producto para su compra o debe haber consumido algún servicio en la barbería para realizar el pago.	

[Almacén de productos]

Caso de Uso	[Almacén de productos]	Identificador: CU-004
Actores	Interviene el actor Trabajador y el actor Cliente.	
Tipo	Caso de uso primario	
Secuencia de pasos	1-Dependiendo del actor que intervenga: 1.1-Si eres Trabajador: 1.1.1-Acceder al sistema con usuario y contraseña. Si no permite iniciar sesión, ir a Excepción #1. 1.1.2- Acceder a la opción Almacén de productos. 1.1.3-Dos opciones a elegir: 1.1.3.1-Añadir producto al stock:nombre,tipo de producto y precio. 1.1.3.2-Retirar producto del stock. 1.2-Si eres Cliente: 1.2.1-Acceder al sistema con usuario y contraseña. Si no eres un cliente que esté registrado previamente en el sistema, ir a Excepción #2. 1.2.2-Acceder a la opción Almacén de productos. 1.2.3-Visualizar stock de productos disponibles.	
Excepción	1-Solicitar recuperación de contraseña mediante correo electrónico. 2-Solicitar al trabajador que lo registre en alta de clientes.	

Precondición	<p>El trabajador debe tener una cuenta personal creada anteriormente facilitada por el Jefe.</p> <p>El cliente debe estar registrado en el sistema previamente, con un usuario y contraseña adjudicados.</p>
--------------	--

[Gestión de citas]

Caso de Uso	[Gestión de citas]	Identificador: CU-005
Actores	Interviene el actor Cliente.	
Tipo	Caso de uso primario	
Secuencia de pasos	<p>1-Acceder al sistema con usuario y contraseña. Si no eres un cliente que esté registrado previamente en el sistema, ir a Excepción #1.</p> <p>2- Acceder a la Gestión de citas.</p> <p>3-Ver citas disponibles (día y hora). Si no desea continuar, ir a Excepción #2</p> <p>4-Elegir cita en un día y hora concreto.</p> <p>5-Confirmar cita. Si no desea la cita asignada, ir a Excepción #3.</p>	
Excepción	<p>1-Solicitar al trabajador que lo registre en alta de clientes.</p> <p>2-Cancelar operación y será redirigido a la interfaz principal de la aplicación.</p> <p>3-Elegir la opción cancelar cita previa.</p>	
Precondición	El cliente debe estar registrado en el sistema previamente, con un usuario y contraseña adjudicados.	

Tarea 3: Prototipo y manual de usuario:

En el siguiente apartado se mostrará el diseño de un prototipo de la aplicación mediante una interfaz gráfica y junto a ésta un manual de instrucciones de uso para los usuarios.

-¿Cómo acceder a la aplicación?

Primero deberá acceder a la aplicación Barbería Guaniros y hacer click 2 veces en el icono de la aplicación que se encuentra en el escritorio del ordenador y se abrirá la aplicación.

- 1)Gestión de clientes:

Dar de alta a nuevo cliente:	
Nombre	
Apellidos	
Fecha nacimiento	
DNI	
Dirección	
Localidad	
Provincia	
Código postal	
Teléfono	
Correo electrónico	

Una vez abierta la aplicación, nos aparecerá la pantalla principal que nos muestra directamente el apartado de gestión de clientes. A la izquierda de la interfaz de la aplicación tenemos los distintos apartados como gestión de clientes, gestión de caja, contabilidad, almacén de productos y gestión de citas.

1.1) Buscar cliente:

Dentro del apartado de clientes podremos buscar clientes mediante usuario, nombre o DNI y así comprobar que están dados de alta.

1.2) Dar de alta a un nuevo cliente:

Si no son clientes, se clicará en la opción “Dar de alta a nuevo cliente” y se procederá a rellenar un formulario con los datos personales del nuevo cliente. Se obtendrá un usuario o código, y una contraseña propios para cada cliente.

- 2) Gestión de caja:



BARBERÍA GUANIROS









Gestión de clientes



Gestión de caja



Contabilidad



Almacén de productos



Gestión de citas

Crear factura:

Cliente	
Fecha	

Servicio o producto	Precio
Corte cabello	...€
Arreglo de barba	...€
Cera de fijación	...€
Total:	...€

 Guardar factura
  Imprimir factura

 Listado de facturas

2.1) Crear factura nueva:

En el apartado “Gestión de caja” para crear una factura primero debemos introducir el código de usuario del cliente y la fecha actual. Una vez introducidos esos datos, haremos click sobre “Crear factura”, donde iremos indicando los distintos servicios prestados y/o productos solicitados con su correspondiente importe que debe abonar. Una vez creada la factura se procederá a guardar la factura clicando sobre el icono “Guardar factura” y el cliente podrá solicitar factura impresa si lo desea. Para imprimir la factura se clicará sobre el icono “Imprimir factura”

2.2) Listado de facturas:

En este apartado también tenemos la opción de ver el listado de facturas guardadas, por si algún cliente deseara obtener o revisar la factura de otros días anteriores.

- 3) Contabilidad:




BARBERÍA GUANIROS






Gestión de clientes



Gestión de caja




Contabilidad



Almacén de productos




Gestión de citas



Ingresos:

Año:

-Ingresos mensuales → Elegir mes



Concepto	Importe
Servicios	...€
Productos	...€
Total:	...€

-Ingresos trimestrales → Elegir trimestre



Concepto	Importe
Servicios	...€
Productos	...€
Total:	...€



3.1) Ingresos:

En el apartado “Contabilidad” para visualizar los ingresos haremos click sobre la opción “Ingresos”. Dentro de ingresos, podremos visualizar varias opciones.

Para visualizar los *ingresos mensuales*, elegiremos el año deseado y el mes concreto. A continuación nos mostrará un documento con los distintos ingresos divididos en distintos apartados (servicios prestados, productos vendidos) con su respectivo importe y el total de todos los ingresos mensuales.

Para visualizar los *ingresos trimestrales*, se hará al igual que los ingresos mensuales pero clicando sobre la opción “Ingresos trimestrales” y eligiendo el trimestre deseado, donde el primer trimestre será (Enero, Febrero, Marzo), el segundo trimestre (Abril, Mayo, Junio), el tercero (Julio, Agosto, Septiembre) y el cuarto (Octubre, Noviembre, Diciembre) .

3.2) Gastos:

En el apartado “Contabilidad” para visualizar los gastos haremos click sobre la opción “Gastos”.

Dentro de gastos, podremos visualizar varias opciones.

Para visualizar los *gastos mensuales*, elegiremos el año deseado y el mes concreto. A continuación nos mostrará un documento con los distintos gastos divididos en distintos apartados (alquiler del local, productos y material comprado, pago de autónomo) con su respectivo importe pagado y el total de todos los gastos mensuales.

Para visualizar los *gastos trimestrales*, se hará al igual que los gastos mensuales pero clicando sobre la opción “Gastos trimestrales” y eligiendo el trimestre deseado, donde el primer trimestre será (Enero, Febrero, Marzo), el segundo trimestre (Abril, Mayo, Junio), el tercero (Julio, Agosto, Septiembre) y el cuarto (Octubre, Noviembre, Diciembre) .

- 4) Almacén de productos:



4.1) Productos disponibles:

En el apartado “Almacén de productos” para visualizar los productos disponibles en el stock de la tienda seleccionaremos la opción “Productos disponibles”. Luego se deberá elegir entre cuatro categorías, las cuales son: productos para el cabello, productos eléctricos, productos para el cuidado de la barba y accesorios. Dentro de cada categoría se encontrarán productos específicos de la misma con su correspondiente precio. El cliente solo podrá visualizar el stock disponible, ya que para comprar los productos deseados deberá dirigirse a la sede de la barbería donde se venden los productos.

4.2) Añadir nuevo producto al stock:

En el apartado “Almacén de productos” para añadir un nuevo producto al stock disponible de la tienda seleccionaremos la opción “Añadir producto al stock”. Se deberán introducir las distintas características del nuevo producto, las cuales son: código del producto, nombre del producto, categoría del producto y precio. Una vez rellenadas todas las características se procederá a añadir el producto y ya será disponible en el stock de la tienda.

4.3) Retirar producto del stock:

En el apartado “Almacén de productos” para retirar un producto actual del stock, seleccionaremos la opción “Retirar producto del stock”. Se procederá a buscar el producto mediante el nombre de producto o código del producto, seleccionamos el producto deseado y se hace click sobre la opción “eliminar producto”.

- 5) Gestión de citas:

The screenshot displays the main interface of the 'BARBERÍA GUANIROS' software. The title bar at the top features the shop's name, a barber icon, and standard window controls. A vertical sidebar on the left contains five menu items: 'Gestión de clientes' (with a group of people icon), 'Gestión de caja' (with a cash register icon), 'Contabilidad' (with a document and coins icon), 'Almacén de productos' (with a warehouse icon), and 'Gestión de citas' (with a calendar icon). The main area is titled 'Inicio de sesión:' and contains two input fields for 'Usuario del cliente' and 'Contraseña', followed by an 'Aceptar' button. Below this, the 'Solicitar cita previa:' section includes a 'Ver citas disponibles' link, an arrow pointing to 'Elegir día y hora', and a 'Confirmar' button. At the bottom of this section are two options: 'Cancelar cita' (with a calendar icon) and 'Cambiar cita' (with a circular arrow icon). A large black arrow points from the 'Aceptar' button to the 'Confirmar' button.

5.1) Inicio de sesión para clientes:

Los clientes para poder acceder al apartado “Gestión de clientes” deben iniciar sesión introduciendo sus datos de usuario y contraseña. Si los datos son correctos tendrán acceso a este apartado y podrán solicitar cita previa.

5.2) Solicitar cita previa:

En el apartado “Gestión de citas”, una vez el cliente haya iniciado sesión, para solicitar una cita previa seleccionará la opción “Solicitar cita previa” y posteriormente “Ver citas disponibles” donde deberá elegir el día y hora deseada siempre y cuando esté disponible. Una vez elegido el día y la hora deberá confirmar la operación.

Una vez confirmada la cita, se puede cancelar la cita o cambiar por otra hora y/o día disponibles.

Tarea 5: Probando, probando... :

Parte 1: Implementación en java

```
1 package usuario;
2
3 import java.util.Scanner;
4 import java.util.ArrayList;
5 import java.util.List;
6
7
8 public class Usuario {
9
10     public static void main(String[] args) {
11
12         Scanner sc = new Scanner(System.in);
13         String nombre;
14         String apellido;
15         String year_nacimiento;
16         String nombre_usuario;
17
18         List<String> lista_usuarios = new ArrayList<>();
19         lista_usuarios.add("antoniojimenez1997");
20
21         System.out.println("Introduce el nombre:");
22         nombre = sc.nextLine();
23
24         System.out.println("Introduce el apellido:");
25         apellido = sc.nextLine();
26
27         System.out.println("Introduce el año de nacimiento:");
28         year_nacimiento = sc.nextLine();
29
30         nombre_usuario = nombre_user(nombre,apellido,year_nacimiento);
31
32         if (lista_usuarios.contains(nombre_usuario)) {
33             while (lista_usuarios.contains(nombre_usuario)) {
34                 System.out.println("El usuario ya existe, vuélvelo a intentar introduciendo
35                 datos distintos");
36                 System.out.println("Introduce el nombre:");
37                 nombre = sc.nextLine();
38
39                 System.out.println("Introduce el apellido:");
40                 apellido = sc.nextLine();
41
42                 System.out.println("Introduce el año de nacimiento:");
43                 year_nacimiento = sc.nextLine();
44
45                 nombre_usuario = nombre_user(nombre,apellido,year_nacimiento);
46             }
47         }
48
49         lista_usuarios.add(nombre_usuario);
50         System.out.println(lista_usuarios);
51     }
52
53     public static String nombre_user(String nombre, String apellido, String year_nacimiento)
54     {
55         return nombre + apellido + year_nacimiento;
56     }
57 }
58
59
60 }
```


En esta parte he implementado un método para crear el nombre de usuario mediante su nombre, apellido y año de nacimiento como parámetros.

Parte 2: Pruebas

Caja negra:

The screenshot shows an IDE with two tabs: 'Usuario.java' and 'usuarioTest.java'. The 'usuarioTest.java' tab is active, displaying the following code:

```

1 package usuario;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5
6
7 class usuarioTest {
8
9     @Test
10    public void testUsuarioCorrecto() {
11
12        String resultado = Usuario.nombre_user("antonio", "jimenez", "1997");
13        String esperado = "antoniojimenez1997";
14        assertEquals(esperado, resultado);
15    }
16
17    @Test
18    public void testUsuarioIncorrecto() {
19
20        String resultado = Usuario.nombre_user("antonio", "jimenez", "1997");
21        String esperado = "antonioojimenez1997";
22        assertEquals(esperado, resultado);
23    }
24
25 }

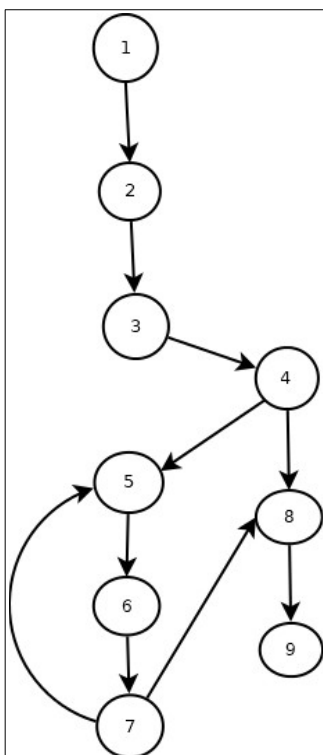
```

On the left, the 'Package Explorer' shows the project structure with 'usuarioTest' containing two test methods: 'testUsuarioIncorrecto()' and 'testUsuarioCorrecto()'. The 'JUnit' tab shows the test results: 'testUsuarioIncorrecto()' failed (0.065 s) and 'testUsuarioCorrecto()' passed (0.006 s). The total run time was 0.265 seconds.

Las pruebas realizadas mediante Junir están bien formuladas ya que el testUsuarioCorrecto() es correcto y el testUsuarioIncorrecto() aparece como fallo.

Caja blanca:

1) Grafo del algoritmo:



```

List<String> lista_usuarios = new ArrayList<>();
lista_usuarios.add("antoniojimenez1997");

System.out.println("Introduce el nombre:");
nombre = sc.nextLine();

System.out.println("Introduce el apellido:");
apellido = sc.nextLine();

System.out.println("Introduce el año de nacimiento:");
year_nacimiento = sc.nextLine();

nombre_usuario = nombre_user(nombre,apellido,year_nacimiento);

if (lista_usuarios.contains(nombre_usuario)) {
    while (lista_usuarios.contains(nombre_usuario)) {
        System.out.println("El usuario ya existe, vuélvelo a intentar introduciendo datos");
        System.out.println("Introduce el nombre:");
        nombre = sc.nextLine();

        System.out.println("Introduce el apellido:");
        apellido = sc.nextLine();

        System.out.println("Introduce el año de nacimiento:");
        year_nacimiento = sc.nextLine();

        nombre_usuario = nombre_user(nombre,apellido,year_nacimiento);
    }
}

lista_usuarios.add(nombre_usuario);
System.out.println(lista_usuarios);

public static String nombre_user(String nombre, String apellido, String year_nacimiento) {
    return nombre + apellido + year_nacimiento;
}

```

The code is annotated with numbered green circles (1-9) that correspond to the nodes in the flowchart. Node 1 is at the start of the main method. Node 2 is after the first input prompt. Node 3 is after the second input prompt. Node 4 is after the third input prompt. Node 5 is at the start of the loop. Node 6 is after the loop body. Node 7 is at the end of the loop. Node 8 is after the loop ends. Node 9 is at the end of the main method.

2) Caminos básicos del grafo:

Camino 1: 1-2-3-4-8-9

Camino 2: 1-2-3-4-5-6-7-8-9

Camino 3: 1-2-3-4-5-6-7-5-6-7-8-9

3) Casos de prueba:

Camino 1: nombre = juan, apellido = ruiz, año = 1999

Camino 2:

Primer intento: nombre = antonio, apellido = jimenez, año = 1997

Segundo intento: nombre = luis apellido = montero, año = 1994

Camino3:

Primer intento: nombre = antonio, apellido = jimenez, año = 1997

Segundo intento: nombre = antonio, apellido = jimenez, año = 1997

Tercer intento: nombre = marta, apellido = rico, año = 1996

4) Complejidad ciclomática:

$V(G) = \text{aristas} - \text{nodos} + 2 = 10 - 9 + 2$

$V(G) = 3$

Parte 3: SCV

Rama master:

```
jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Usuario.java

nothing added to commit but untracked files present (use "git add" to track)

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git add Usuario.java

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git commit -m "Añadida la implementación de la clase en java"
[master (root-commit) 673db5f] Añadida la implementación de la clase en java
1 file changed, 60 insertions(+)
create mode 100644 Usuario.java

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Rama pruebas:

```
jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git branch pruebas

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git checkout pruebas
Switched to branch 'pruebas'

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (pruebas)
$ mkdir cajaBlanca

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (pruebas)
$ mkdir cajaNegra
```

```
jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (pruebas)
$ git status
On branch pruebas
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    cajaBlanca/
    cajaNegra/

nothing added to commit but untracked files present (use "git add" to track)

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (pruebas)
$ git add cajaBlanca/prueba.pdf cajaNegra/usuarioTest.java

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (pruebas)
$ git status
On branch pruebas
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   cajaBlanca/prueba.pdf
    new file:   cajaNegra/usuarioTest.java

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (pruebas)
$ git commit -m "Añadido de pruebas de caja blanca y caja negra"
[pruebas 35e5448] Añadido de pruebas de caja blanca y caja negra
 2 files changed, 25 insertions(+)
 create mode 100644 cajaBlanca/prueba.pdf
 create mode 100644 cajaNegra/usuarioTest.java
```

Tarea 6: Documentando:

Parte 1) Documentando

- Clase implementada con comentarios:

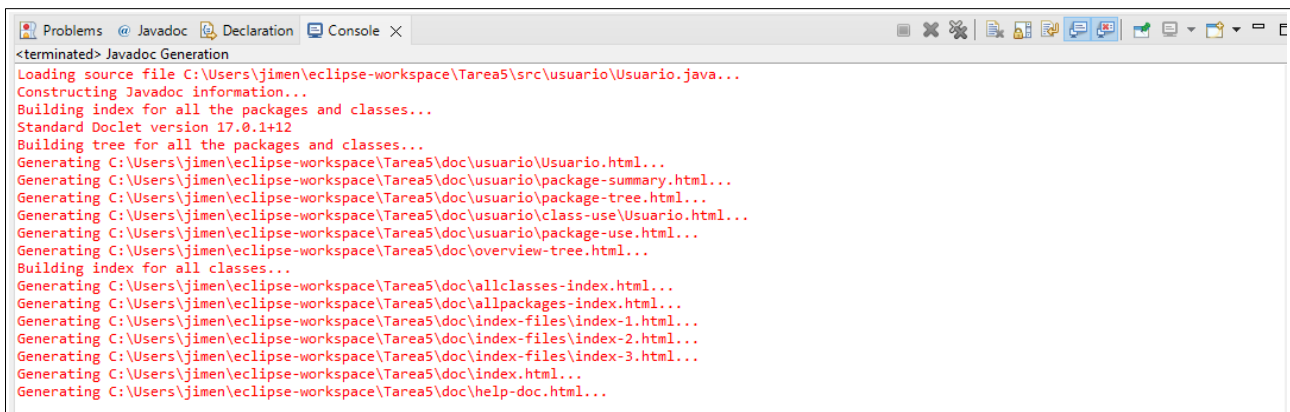
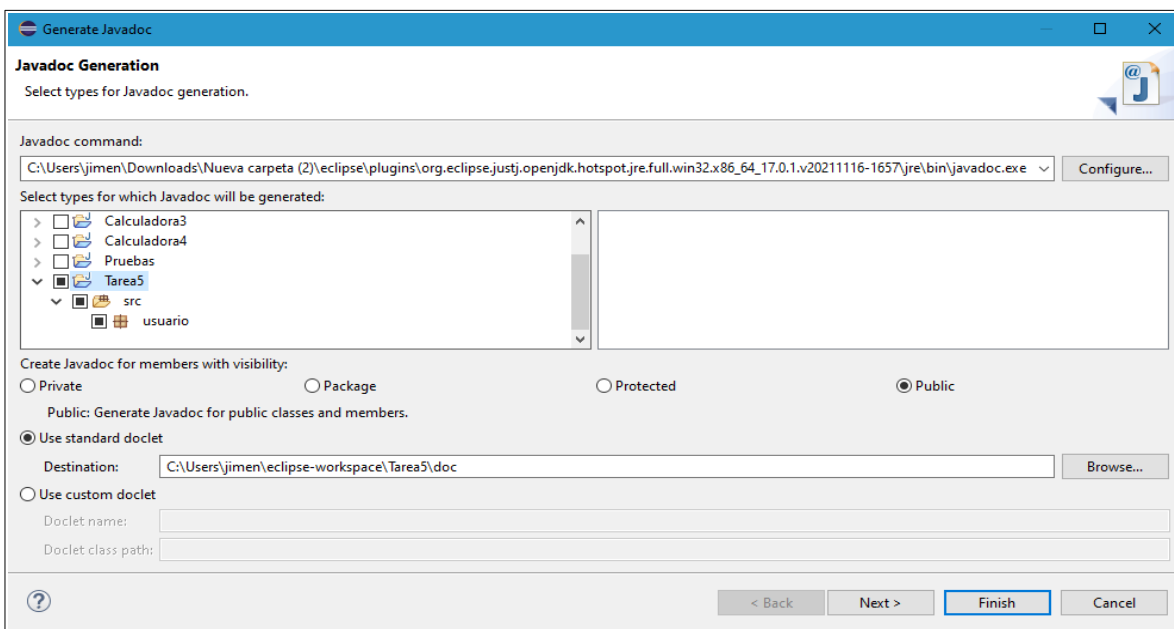
```
1 package usuario;
2
3 import java.util.Scanner;
4
5 /**
6  * Clase que nos permitirá realizar operaciones para crear el nombre de usuario de los
7  * clientes.
8  * Esta clase contiene los atributos de cliente.
9  * @author Antonio Jiménez Cortés
10  * @version 1.1
11  */
12 public class Usuario {
13
14     /**
15      * Algoritmo principal que permite ejecutar el código del programa.
16      * @param args parámetros de entrada. Es un array de Strings que debe aparecer
17      * obligatoriamente como argumento del método main en un programa Java.
18      */
19     public static void main(String[] args) {
20
21         Scanner sc = new Scanner(System.in);
22         String nombre;
23         String apellido;
24         String year_nacimiento;
25         String nombre_usuario;
26
27         List<String> lista_usuarios = new ArrayList<>();
28         lista_usuarios.add("antoniojimenez1997");
29
30         System.out.println("Introduce el nombre:");
31         nombre = sc.nextLine();
32
33         System.out.println("Introduce el apellido:");
34         apellido = sc.nextLine();
35
36         System.out.println("Introduce el año de nacimiento:");
37         year_nacimiento = sc.nextLine();
38
39         nombre_usuario = nombre_user(nombre,apellido,year_nacimiento);
40
41         if (lista_usuarios.contains(nombre_usuario)) {
42             while (lista_usuarios.contains(nombre_usuario)) {
43                 System.out.println("El usuario ya existe, vuélvelo a intentar introduciendo
44                 datos distintos");
45                 System.out.println("Introduce el nombre:");
46                 nombre = sc.nextLine();
47                 System.out.println("Introduce el apellido:");
48                 apellido = sc.nextLine();
49                 System.out.println("Introduce el año de nacimiento:");
50                 year_nacimiento = sc.nextLine();
51                 nombre_usuario = nombre_user(nombre,apellido,year_nacimiento);
52             }
53         }
54         lista_usuarios.add(nombre_usuario);
55         System.out.println(lista_usuarios);
56     }
57 }
58
59
60
61 }
```

```

62
63  /**
64   * Realiza la operación de creación de usuario recibiendo distintos parámetros.
65   * @param nombre Parámetro que indicará el nombre del cliente.
66   * @param apellido Parámetro que indicará el apellido del cliente.
67   * @param year_nacimiento Parámetro que indicará el año de nacimiento del cliente.
68   * @return Devuelve un nombre de usuario que se creado concatenando los distintos
        parámetros recibidos.
69   */
70   public static String nombre_user(String nombre, String apellido, String year_nacimiento)
71   {
72       return nombre + apellido + year_nacimiento;
73   }
74
75
76 }

```

- Generación del HTML con JavaDOC:



Parte 2) SCV

Rama master:

Implementación de la clase en java con comentarios y marcas de JavaDOC.

```
jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   Usuario.java

no changes added to commit (use "git add" and/or "git commit -a")

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git add Usuario.java

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git commit -m "Añadida la implementación de la clase en java con comentarios"
[master 0900317] Añadida la implementación de la clase en java con comentarios
1 file changed, 17 insertions(+), 1 deletion(-)

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 3 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 756 bytes | 756.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:charro97/tarea5.git
   673db5f..0900317  master -> master
```

Rama Documentando:

Contiene todo el contenido html generado con JavaDOC.

```
jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (Documentando)
$ git status
On branch Documentando
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        doc/

nothing added to commit but untracked files present (use "git add" to track)

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (Documentando)
$ git add doc/
```



```
jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (Documentando)
$ git commit -m "Añadido contenido generado con javadoc"
[Documentando 5a42f7f] Añadido contenido generado con javadoc
45 files changed, 2601 insertions(+)
create mode 100644 doc/allclasses-index.html
create mode 100644 doc/allpackages-index.html
create mode 100644 doc/element-list
create mode 100644 doc/help-doc.html
create mode 100644 doc/index-files/index-1.html
create mode 100644 doc/index-files/index-2.html
create mode 100644 doc/index-files/index-3.html
create mode 100644 doc/index.html
```

```
jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (Documentando)
$ git status
On branch Documentando
nothing to commit, working tree clean

jimen@LAPTOP-A6FPBSDM MINGW64 ~/Documents/Tarea5Proyecto (Documentando)
$ git push origin Documentando
Enumerating objects: 58, done.
Counting objects: 100% (58/58), done.
Delta compression using up to 3 threads
Compressing objects: 100% (48/48), done.
Writing objects: 100% (57/57), 93.99 KiB | 1.36 MiB/s, done.
Total 57 (delta 14), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (14/14), done.
remote:
remote: Create a pull request for 'Documentando' on GitHub by visiting:
remote:   https://github.com/charro97/tarea5/pull/new/Documentando
remote:
To github.com:charro97/tarea5.git
 * [new branch]      Documentando -> Documentando
```