

DOCUMENTACIÓN FINAL

Ingeniería del Software II

Víctor Domínguez Martínez

Abel Martínez Monje

Adrián Riesco Valbuena

Contenido

1.	Funcionalidad final	2
2.	SCRUM	3
2.1	Roles.....	3
2.2	Historias de usuario	3
2.3	Product Backlog o Pila de Producto.....	4
2.4	Sprint.....	5
2.4.1	Sprint 1.....	5
2.4.2	Sprint 2.....	7
2.4.3	Sprint 3.....	8
3.	Pruebas	11
3.1	Pruebas de caja blanca	11
3.1.1	Complejidad.....	12
3.1.2	Caminos	12
3.1.3	Casos de prueba	12
3.2	Pruebas de caja negra.....	12
4.	Controlador de versiones	14

1. Funcionalidad final

Los usuarios principales del sistema son dos: cliente (con roles registrado o anónimo) y trabajador (con roles camarero, cocinero, encargado del almacén o administrador).

La aplicación es capaz de gestionar clientes, trabajadores y proveedores. También puede gestionar pedidos, tanto de clientes (anónimos o registrados), al almacén y de proveedores. Otra funcionalidad es que permite a los cocineros realizar solicitudes de nuevos platos al administrador, para que sean incluidos en el sistema.

En cuanto a la seguridad, se ha implementado un control de acceso mediante usuario y contraseña, así como el uso de sesiones para que cada usuario sólo pueda acceder a su conjunto de acciones determinadas. Se utilizan plantillas con un evento para controlar dichas sesiones al cargar la página que permite que ningún usuario sin autorización acceda a ella.

2. SCRUM

Este proyecto se ha realizado en base al marco de trabajo SCRUM.

2.1 Roles

En SCRUM se distinguen tres roles, que hemos interpretado de la siguiente manera:

- Product owner: El papel de dueño del producto o cliente lo hemos realizado tanto poniéndonos en el lugar del cliente, como preguntando a terceras personas acerca de sus preferencias en la aplicación.
- SCRUM master: La función del máster la hemos realizado en conjunto las tres personas del grupo, ya que al estar todos informados acerca de SCRUM y ser un equipo tan pequeño, no creímos necesario el establecer un rol de SCRUM máster diferenciado.
- Equipo de desarrollo: Tal y como indica SCRUM, es multidisciplinar (los tres integrantes conocemos todos los aspectos necesarios para el desarrollo del proyecto) y auto-organizado.

2.2 Historias de usuario

Las historias de usuario las hemos recogido utilizando parcialmente los casos de uso desarrollados en Software I, adaptándolos a su estructura. Para poder mostrarlas correctamente, hemos usado una misma plantilla que recoge lo necesario para implementar con el marco de trabajo SCRUM.

Las historias de usuario responden de manera correcta a las preguntas acerca de quién se beneficia, qué se quiere y cuál es el beneficio.

El id de cada historia se ha conformado de la forma “XX-YY-ZZ”, siendo XX el rol principal (CL para cliente o TR para trabajador), YY el rol secundario: RG cliente registrado, y AN cliente anónimo o ambos; CO cocinero, EN encargado del almacén, CA camarero o AD administrador. Por último, las siglas ZZ indican el número de la historia, comenzando por 01.

Para asignar cada estimación de puntos de historia hemos seguido la técnica de Planning Poker, utilizando cartas de 0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100 puntos. Al ser un grupo de 3 personas, no hemos incluido cartas de interrogación o descanso.

El id de las historias de usuario generadas (los detalles se encuentran en el archivo “Historias de usuario.xls”) y sus diferentes puntos de historia asignados son:

HISTORIAS DE USUARIO		
Historia de usuario	Puntos de historia	Resultado de la votación
CL-AN-01	300 puntos	100 + 100 + 100
CL-AN-02	240 puntos	100 + 100 + 40
CL-CL-01	6 puntos	3 + 2 + 1

CL-CL-02	100 puntos	40 + 40 + 20
TR-CA-01	46 puntos	20 + 13 + 13
TR-CO-01	300 puntos	100 + 100 + 100
TR-CO-02	180 puntos	100 + 40 + 40
TR-CO-03	9 puntos	5 + 2 + 2
TR-EN-01	93 puntos	40 + 40 + 13
TR-EN-02	100 puntos	40 + 40 + 20
TR-EN-03	80 puntos	40 + 20 + 20
TR-AD-01	60 puntos	20 + 20 + 20
TR-AD-02	120 puntos	40 + 40 + 40
TR-AD-03	100 puntos	40 + 40 + 20
TR-AD-04	3 puntos	1 + 1 + 1

2.3 Product Backlog o Pila de Producto

Tras asignar los puntos de historia a las historias obtenidas, la pila de producto resultante, ordenada por puntos de historia y añadiendo la importancia relativa al dueño del producto, ha sido la siguiente:

PRODUCT BACKLOG			
ID historia	Nombre	Importancia relativa al dueño	Puntos de historia
CL-AN-01	Cliente hacer pedidos	Alta	300 puntos
TR-CO-01	Cocinero ver pedidos	Alta	300 puntos
CL-AN-02	Cliente anónimo registrarse	Alta	240 puntos
TR-CO-02	Cocinero añadir pedidos almacén	Media	180 puntos
TR-AD-02	Admin gestionar trabajadores	Alta	120 puntos
CL-CL-02	Cliente darse de baja	Media	100 puntos
TR-EN-02	Encargado consultar inventario	Media	100 puntos
TR-AD-03	Admin ver pedidos proveedores	Alta	100 puntos
TR-EN-01	Encargado ver pedidos cocina	Alta	93 puntos
TR-EN-03	Encargado elaborar pedidos	Media	80 puntos
TR-AD-01	Admin gestionar proveedores	Alta	60 puntos
TR-CA-01	Camarero ver pedidos	Alta	46 puntos
TR-CO-03	Cocinero introducir nuevos platos	Baja	9 puntos
CL-CL-01	Cliente ver ofertas	Media	6 puntos
TR-AD-04	Admin ver nuevos platos	Baja	3 puntos

Respecto a la forma de probar cada historia, todas se prueban de manera muy sencilla y no requiere de ningún método específico o fuera de ámbito. Los diferentes pasos se encuentran detallados en el apartado correspondiente a cada Sprint.

A partir de esta pila de producto se han extraído las diferentes pilas para cada uno de los Sprint.

2.4 Sprint

Para la realización del proyecto se ha dividido en tres Sprints, cada uno de ellos de cinco días de duración.

2.4.1 Sprint 1

2.4.1.1 Descripción del Sprint

El primer Sprint se ha desarrollado entre los días 23/05/2018 y 27/05/2018, ambos incluidos, teniendo una duración total de cinco días.

El objetivo de este Sprint ha sido el de elaborar unas funcionalidades base y permitir la elaboración de los primeros pedidos.

Las tareas generales realizadas durante este Sprint, antes del comienzo de la realización de cada historia, han sido:

- ❖ Crear base de datos
- ❖ Añadir unos elementos generales a la base de datos (productos y usuarios)
- ❖ Crear la interfaz base

2.4.1.2 Sprint Backlog

La pila del primer Sprint es la siguiente:

SPRINT 1 BACKLOG		
ID Historia	Nombre	Tareas
CL-AN-01	Cliente hacer pedidos	<ul style="list-style-type: none">❖ Distinguir productos, menús y pedidos❖ Realizar una prueba de creación de pedido
CL-AN-02	Cliente anónimo registrarse	<ul style="list-style-type: none">❖ Habilitar la opción de registro si no ha hecho log in❖ Comprobar el registro correcto
TR-EN-02	Encargado consultar inventario	<ul style="list-style-type: none">❖ Realizar una organización de cada producto en categorías y subcategorías❖ Comprobar la pertenencia de cada producto a una subcategoría
TR-EN-03	Encargado elaborar pedidos	<ul style="list-style-type: none">❖ Crear un pedido y comprobar la correcta relación entre productos y proveedor
TR-AD-03	Admin ver pedidos proveedores	<ul style="list-style-type: none">❖ Comprobar que los pedidos realizados se

		muestran correctamente ❖ Validar un pedido y comprobar que desaparece de la lista
--	--	---

2.4.1.3 Actas de las reuniones de planificación y revisión del Sprint

Reunión de planificación del Sprint:

- ❖ Fecha: 23/05/2018 – 10:00 am.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 45 minutos.
- ❖ Objetivo: Decidir cómo comenzar a desarrollar el proyecto. Establecer las tareas genéricas necesarias para poder empezar a trabajar en las historias de usuario. Escoger asimismo las historias que se van a implementar.
- ❖ Resultado: Se decidió que el paso principal era crear la base de datos y añadirle unos elementos genéricos, así como la creación del modelo básico de interfaz. Se escogieron también las historias de usuario más necesarias, de importancia alta y media, teniendo en cuenta sus puntos de historia.

Reunión de revisión del Sprint:

- ❖ Fecha: 27/05/2018 – 20:00.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 45 minutos.
- ❖ Objetivo: Establecer los objetivos cumplidos y no cumplidos a lo largo del Sprint.
- ❖ Resultado: Se han cumplido todas las tareas acordadas en la reunión de planificación del Sprint.

2.4.1.4 Acta de la reunión de retrospectiva del Sprint

Reunión de retrospectiva del Sprint:

- ❖ Fecha: 27/05/2018 – 20:45.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 15 minutos.
- ❖ Objetivo: Recoger las impresiones generales del Sprint de cara a mejorar el siguiente.
- ❖ Resultado: Estamos bastante contentos con la desenvolvura del equipo a lo largo del Sprint, siendo la primera vez que trabajamos con este método. Creemos que hemos realizado una buena elección a la hora de escoger las tareas a realizar.

2.4.2 Sprint 2

2.4.2.1 Descripción del Sprint

El segundo Sprint se ha desarrollado entre los días 28/05/2018 y 01/06/2018, ambos incluidos, teniendo una duración total de cinco días.

El objetivo de este Sprint ha sido el de seguir implementando las funciones principales del proyecto.

2.4.2.2 Sprint Backlog

La pila del segundo Sprint es la siguiente:

SPRINT 2 BACKLOG		
ID Historia	Nombre	Tareas
TR-CO-01	Cocinero ver pedidos	<ul style="list-style-type: none">❖ Realizar un pedido desde la perspectiva del cliente❖ Comprobar la correcta recepción, lectura y validación del mismo por el rol de cocinero
TR-CO-02	Cocinero añadir pedidos almacén	<ul style="list-style-type: none">❖ Crear un pedido al almacén, comprobando su correcto guardado
TR-AD-02	Admin gestionar trabajadores	<ul style="list-style-type: none">❖ Crear, modificar y eliminar un trabajador para comprobar el correcto funcionamiento
TR-AD-01	Admin gestionar proveedores	<ul style="list-style-type: none">❖ Crear, modificar y eliminar un proveedor para comprobar el correcto funcionamiento
TR-EN-01	Encargado ver pedidos cocina	<ul style="list-style-type: none">❖ Realizar un pedido desde la perspectiva del cocinero (ya hecho en este Sprint)❖ Comprobar la correcta visualización y validación del pedido por el rol de encargado

2.4.2.3 Actas de las reuniones de planificación y revisión del Sprint

Reunión de planificación del Sprint:

- ❖ Fecha: 28/05/2018 – 16:00 am.

- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 45 minutos.
- ❖ Objetivo: Decidir las siguientes historias de usuario que se van a implementar y cómo hacerlo.
- ❖ Resultado: Se escogieron las historias de mayor importancia y que más afectan al funcionamiento del proyecto, teniendo en cuenta sus puntos de historia, y las tareas necesarias para su desarrollo.

Reunión de revisión del Sprint:

- ❖ Fecha: 01/06/2018 – 17:00.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 45 minutos.
- ❖ Objetivo: Establecer los objetivos cumplidos y no cumplidos a lo largo del Sprint.
- ❖ Resultado: Se han cumplido todas las tareas acordadas en la reunión de planificación del Sprint.

2.4.2.4 Acta de la reunión de retrospectiva del Sprint

Reunión de retrospectiva del Sprint:

- ❖ Fecha: 01/06/2018 – 17:45.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 15 minutos.
- ❖ Objetivo: Recoger las impresiones generales del Sprint de cara a mejorar el siguiente.
- ❖ Resultado: Estamos de acuerdo en que hemos hecho un buen trabajo y que la elección de tareas ha sido la adecuada.

2.4.3 Sprint 3

2.4.3.1 Descripción del Sprint

El tercer y último Sprint se ha desarrollado entre los días 02/06/2018 y 06/06/2018, ambos incluidos, teniendo una duración de cinco días.

El objetivo de este Sprint ha sido el de finalizar la implementación de las historias de usuario restantes, generalmente más sencillas y de menos importancia que las anteriores.

2.4.3.2 Sprint Backlog

La pila del tercer Sprint es la siguiente:

SPRINT 3 BACKLOG		
ID Historia	Nombre	Tareas

CL-CL-02	Cliente darse de baja	<ul style="list-style-type: none"> ❖ Realizar la baja desde un cliente logeado en el sistema ❖ Comprobar la correcta eliminación de sus datos
TR-CA-01	Camarero ver pedidos	<ul style="list-style-type: none"> ❖ Crear un pedido desde la perspectiva de cliente ❖ Comprobar que los datos visualizados por el camarero se corresponden con los del pedido realizado
TR-CO-03	Cocinero introducir nuevos platos	<ul style="list-style-type: none"> ❖ Solicitar la inclusión de un plato en el sistema (ingredientes, elaboración, etc)
CL-CL-01	Cliente ver ofertas	<ul style="list-style-type: none"> ❖ Acceder como un cliente y ver sus ofertas personalizadas
TR-AD-04	Admin ver nuevos platos	<ul style="list-style-type: none"> ❖ Realizar una solicitud de plato desde la perspectiva del cocinero (ya realizado en este Sprint) ❖ Comprobar que el administrador puede ver la solicitud, modificarlo en caso de ser necesario, y validarla

2.4.3.3 Actas de las reuniones de planificación y revisión del Sprint

Reunión de planificación del Sprint:

- ❖ Fecha: 02/06/2018 – 12:00 am.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 45 minutos.
- ❖ Objetivo: Decidir las siguientes historias de usuario que se van a implementar, cómo desarrollarlas, y las pruebas que se van a realizar.
- ❖ Resultado: Se escogieron las historias restantes, siendo en general las de menor puntos de historia, permitiéndonos así hacer las pruebas necesarias sin problema. También se acordaron las tareas necesarias para la comprobación de las historias.

Reunión de revisión del Sprint:

- ❖ Fecha: 06/06/2018 – 20:00.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 45 minutos.
- ❖ Objetivo: Establecer los objetivos cumplidos y no cumplidos a lo largo del Sprint.
- ❖ Resultado: Se han cumplido todas las tareas acordadas en la reunión de planificación del Sprint. La realización de las pruebas ha concluido con éxito.

2.4.3.4 Acta de la reunión de retrospectiva del Sprint

Reunión de retrospectiva del Sprint:

- ❖ Fecha: 06/06/2018 – 20:45.
- ❖ Miembros del equipo presentes: Abel, Víctor y Adrián.
- ❖ Duración: 15 minutos.
- ❖ Objetivo: Recoger las impresiones generales del Sprint y de la realización del proyecto.
- ❖ Resultado: Estamos contentos con el resultado del proyecto. Hemos cumplido todos los Sprint sin ningún problema, respetando las fechas y tareas de cada uno de ellos.

3. Pruebas

Para la comprobación del proyecto hemos realizado tanto pruebas de caja blanca como de caja negra.

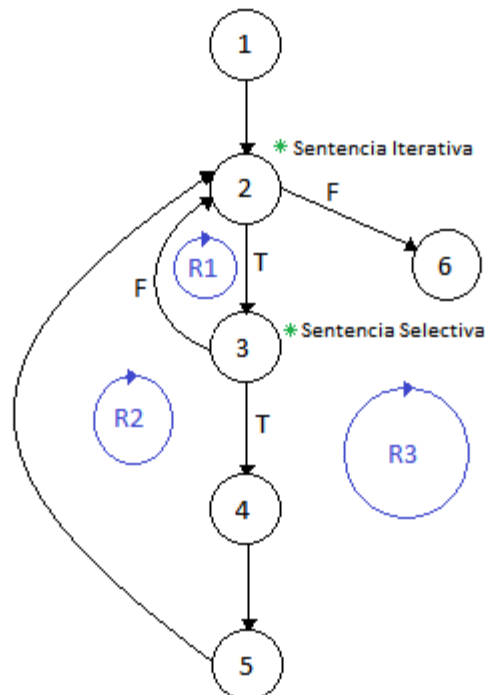
3.1 Pruebas de caja blanca

Las pruebas de caja blanca se centran en la estructura o visión interna del programa, conociendo su funcionamiento. Las pruebas de caja blanca cuentan con dos técnicas para poder realizarlas: Prueba del camino o ruta básico, y prueba de la estructura de control.

Relativo a la prueba del camino básico, hemos analizado el siguiente fragmento de código:

```
175 public int calcularPrecioPedidoCliente(){
176     int precioFinal = 0;
177     int i = 0;
178
179     for(i = 0; i < listaPlatos.size(); i++){
180         if (Integer.valueOf(listaCantidad.get(i)) >= 0){
181             platos = platosEJB.find(listaPlatos.get(i));
182             precioFinal += platos.getPrecio()*Integer.valueOf(listaCantidad.get(i));
183         }
184     }
185
186     return precioFinal;
187 }
```

Resultando en el siguiente grafo del flujo de control:



Una vez hecho esto, creamos las sentencias de prueba relativas a estos casos.

3.1.1 Complejidad

$$V(G) = R = 3$$

$$V(G) = \text{aristas} - \text{nodos} + 2 = 7 - 6 + 2 = 3$$

$$V(G) = \text{nodos predicados} + 1 = 2 + 1 = 3$$

3.1.2 Caminos

- Camino 1: 1 – 2 – 6
- Camino 2: 1 – 2 – 3 – 2 – 6
- Camino 3: 1 – 2 – 3 – 4 – 5 – 2 – 6

Subrayados y en negritas las aristas nuevas que se añaden en cada camino.

3.1.3 Casos de prueba

- Camino 1:
 - listaPlatos.size() = 0
 - return precioFinal = 0;
- Camino 2:
 - listaPrecios.size() = 1
 - listaCantidad.get(i) = 0
 - return precioFinal = 0;
- Camino 3:
 - listaPrecios.size() = 1
 - listaCantidad.get(i) = 1
 - return precioFinal = 35;

3.2 Pruebas de caja negra

Las pruebas de caja negra se centran en la funcionalidad del programa, conociendo la función específica para la que se diseñó. Las técnicas de caja negra son la partición de equivalencia y los análisis de valores límite.

Relativo a la partición de equivalencia, las pruebas que hemos realizado han analizado la misma parte que las pruebas de caja blanca.

- Lógico: el valor introducido en el campo cantidad tiene que ser un número entero (clase válida). Todo lo que no sea un entero será una clase no válida.
 - Clases válidas: Número entero.
 - Clases no válidas: Cualquier otro dato (Double).
- Rango: El valor de cantidad debe de ser mayor que 0. Los valores menores o iguales a 0 será una clase no válida. Rango (0, inf)

- Valores límite: -0.0001, 0 (casos para 0 y menor que 0)
 - Casos válidos: 0
 - Casos no válidos: -0.0001

4. Controlador de versiones

Para la realización del proyecto hemos utilizado el controlador de versiones GitHub, ya que es con el que más familiarizados estamos y uno de los más usados a nivel global.

Principalmente, hemos creado una rama sobre la rama master para cada Split (ramas split1, split2 y split3), al final del cuál hemos unido dicha rama con la principal después de comprobar su funcionamiento. Asimismo, para cada rama de Split hemos creado otras tres ramas basadas en ella para poder trabajar los tres miembros del grupo de manera independiente.

La primera figura es la vista completa del árbol git, mientras que la segunda es el detalle concreto del primer sprint.

