

1.Documento SCRUM

1.1 Estructura y funcionamiento del equipo Scrum

En cuanto a los roles, en primer lugar el product owner es el director de marketing del equipo de fútbol cuya tienda gestionamos. Este aceptará o rechazará nuestros incrementos y seguiremos todas sus directrices. El equipo está formado por Lucero Abregú, Sergio Atienza, Madalina Basca, Jesús Hernández, Andrés Galbán, Cristina Morillo, Alejandro Pareja y Olenka Lazo. El scrum master será Sergio Atienza, que guiará al equipo y servirá como medio de comunicación de este con elementos externos.

Al ser un equipo SCRUM, estamos autodirigidos. Todo el trabajo lo hacemos nosotros, no necesitamos ayuda externa. Utilizaremos el drive para ir subiendo toda la información y avances del proyecto. Nos comunicaremos por discord de manera telemática, y en clase de forma presencial. Cada miembro o miembros tendrán una o más tareas asignadas y nos reunimos semanalmente para comprobar el avance incremental de nuestro proyecto.

1.2. Historias de usuario

(Historias de usuario iniciales-9 marzo)

HU #1 Mostrar Tienda		
Como comprador quiero ver los productos para poder realizar una compra		
Valoración 138	Prioridad 2	Estimación XL
Validación		
1) Comprobar que se muestran todos los productos		
2) Comprobar que se muestran correctamente los productos por página.		
3) Comprobar que el desplazamiento de páginas es correcto.		
4) Comprobar que no podemos salir de los límites de las páginas.		

HU #2 Ordenar Productos por Jugador		
Como fanático de un jugador en concreto quiero comprarme su equipación para poder llevar su equipación		
Valoración 54	Prioridad 18	Estimación M

HU #3 Añadir Producto		
Como vendedor quiero añadir productos para venderlos en la web.		
Valoración 146	Prioridad 1	Estimación S
Validación		
1) Mostramos tienda		
2) Añadimos el producto deseado tanto de la tienda como del inventario.		
3) Volvemos a mostrar la tienda para comprobar que sale esta en ella.		

HU #4 Retirar Producto		
Como vendedor quiero eliminar productos para que no se venda más.		
Valoración 115	Prioridad 5	Estimación XS
Validación		
1) Mostramos tienda		
2) Eliminamos el producto deseado tanto de la tienda como del inventario.		
3) Volvemos a mostrar la tienda para comprobar que no sale esta en ella.		

HU #5 Reabastecer Producto		
Como vendedor quiero comprar más unidades de productos agotados para que haya unidades de nuevo		
Valoración 117	Prioridad 4	Estimación XS
Validación		
1) Mostramos tienda		
2) Cambiamos las unidades del producto seleccionado.		
3) Volvemos a mostrar la tienda para comprobar que el stock del ha cambiado.		

HU #6 Mostrar Inventario		
Como vendedor quiero comprobar cuántas unidades tengo de cada producto para poder saber si tengo que reabastecer.		
Valoración 120	Prioridad 3	Estimación L
Validación		
1) Comprobar si entra al inventario		
2) Comprobar si se muestran todos los productos		
3) Comprobar si se muestran todos los datos		

HU #7 Cambiar Precio Producto		
Como vendedor quiero comprobar cuántas unidades tengo de cada producto para poder saber si tengo que reabastecer.		
Valoración	84	Prioridad 12 Estimación XS
Validación		
1) Comprobamos que el nuevo precio sea positivo		
2) Actualizamos el precio		

HU #8 Reservar Producto		
Como vendedor quiero reservarle un producto a un cliente en específico para impedir que otros usuarios lo compren antes.		
Valoración	57	Prioridad 16 Estimación S

HU #9 Mostrar Ingresos y Productos Vendidos		
Como vendedor quiero visualizar los productos vendidos y ganancias en un intervalo de tiempo para tener información del negocio.		
Valoración	66	Prioridad 15 Estimación M

HU #10 Añadir al Carrito		
Como comprador quiero añadir un producto a una lista de productos personales para comprarlos en el futuro.		
Valoración	113	Prioridad 6 Estimación M
Validación		
1) Mostramos el carrito		
2) Añadimos el producto al carrito		
3) Volvemos a mostrar el carrito para ver que se ha añadido correctamente		
4) Probamos que no se sale del rango del array		

HU #11 Eliminar del Carrito		
Como comprador quiero eliminar un productode la lista listos para no comprar.		
Valoración 81	Prioridad 13	Estimación XS
Validación 1) Mostramos el carrito 2) Eliminamos el producto del carrito 3) Volvemos a mostrar el carrito para ver que se ha eliminado correctamente 4) Probamos que no se sale del rango del array		

HU #12 Añadir Método de Pago		
Como cliente quiero añadir una fuente de ingreso para pagar.		
Valoración 84	Prioridad 11	Estimación M
Validación 1) Pulsamos el botón de comprar del carrito 2) Al estar en la ventana de pago rellenamos todos los campos 3) Pulsamos el botón de pago para realizar la acción 4) Comprobamos que las ventanas se cierran correctamente		

HU #13 Añadir Promoción		
Como vendedor quiero añadir una promoción de productos para incentivar ventas.		
Valoración 44	Prioridad 21	Estimación S

HU #14 Introducir Promoción		
Como comprador quiero introducir el código de una promoción de productos para comprarla a menor precio.		
Valoración 44	Prioridad 21	Estimación S

HU #15 Realizar Devolución		
Como comprador quiero devolver un producto para recuperar el dinero.		
Valoración 87	Prioridad 10	Estimación M

HU #16 Buscar un Producto		
Como comprador quiero buscar un producto en la tienda para comprarlo		
Valoración 91	Prioridad 9	Estimación S
Validación		
1) Introducimos el id del producto a buscar		
2) Se busca el producto y se comprueba que lo que aparece es:		
3) Si es nulo se muestra que no se encontró.		
4) Si no es nulo se muestra que producto es.		

HU #17 Publicitar Producto		
Como vendedor quiero dar visualización de mis productos para generar más ingresos.		
Valoración 45	Prioridad 20	Estimación M

HU #18 Iniciar Sesión Usuario Cliente		
Como cliente quiero iniciar sesión en mi cuenta de usuario para poder ver mi historial de compras y guardar mi información para compras futuras.		
Valoración 106	Prioridad 8	Estimación L
Validación		
1) Se rellena el formulario		
2) Se introducen los datos de inicio de sesión		
3) Se muestra de nuevo la tienda		

HU #19 Calificar Producto		
Como vendedor quiero que los usuarios puedan calificar mis productos para demostrar mi fiabilidad.		
Valoración 70	Prioridad 14	Estimación XS

HU #20 Reportar un Error		
Como usuario quiero reportar un error para que sea resuelto y continuar navegando con normalidad.		
Valoración 55	Prioridad 17	Estimación XS

HU #21 Comprar		
Como usuario quiero adquirir un producto para su uso y disfrute		
Valoración 112	Prioridad 7	Estimación M
Validación		
1) Se muestra el carrito		
2) Se rellenan los datos de compra		
3) Volvemos a ver el carrito para comprobar que se ha vaciado		

(Historias de usuario nuevas 2º sprint-24 marzo)

HU #22 Mostrar Carrito		
Como comprador quiero ver mi lista de productos personales para comprarlos o eliminarlos.		
Valoración 80	Prioridad 6	Estimación M
Validación		
1) Comprobamos que se muestran todos los productos correctamente		
2) Comprobamos que se vaya actualizando correctamente		

(Historias de usuario nuevas 3º sprint-7 abril)

HU #23 Actualizar Tienda		
Como vendedor quiero actualizar los productos de mi tienda para tener los nuevos que haya añadido.		
Valoración 50	Prioridad 2	Estimación XS
Validación		
1) Comprobamos que se muestran todos los productos correctamente		
2) Comprobamos que se vaya actualizando correctamente		

(Historias de usuario nuevas 4ºsprint-25 abril)

HU #24 Iniciar Sesión Usuario Vendedor		
Como vendedor quiero iniciar sesión en mi cuenta de usuario para poder ver mi inventario y poder realizar modificaciones en este		
Valoración	106	Prioridad 8 Estimación L
Validación 1) Se rellena el formulario 2) Se introducen los datos de inicio de sesión 3) Se muestra de nuevo la tienda		

HU #25 Crear Perfil Usuario Cliente		
Como cliente quiero crear un perfil de usuario para poder ver los productos en mi carrito y así poder comprarlos		
Valoración	106	Prioridad 8 Estimación L
Validación 1) Se rellena el formulario 2) Se introducen los datos de creación de perfil de usuario 3) Se muestra de nuevo la tienda		

HU #26 Iniciar Sesión Usuario Vendedor		
Como vendedor quiero crear un perfil de usuario para poder tener acceso a mi inventario y poder realizar modificaciones en este		
Valoración	106	Prioridad 8 Estimación L
Validación 1) Se rellena el formulario 2) Se introducen los datos de creación de perfil de usuario 3) Se muestra de nuevo la tienda		

HU #27 Resumen de compras		
Como comprador quiero saber las compras para poder tener constancia acceso a mi inventario y poder realizar modificaciones en este		
Valoración 106	Prioridad 8	Estimación L
Validación 1) Se visualiza el resumen de compras 2) Se realiza una compra en una sesión en concreto 3) Se comprueba que el resumen de compras se haya modificado		

HU no implementadas

HU #2 Ordenar Productos por Jugador		
Como fanático de un jugador en concreto quiero comprarme su equipación para poder llevar su equipación		
Valoración 54	Prioridad 18	Estimación M

HU #13 Añadir Promoción		
Como vendedor quiero añadir una promoción de productos para incentivar ventas.		
Valoración 44	Prioridad 21	Estimación S

HU #14 Introducir Promoción		
Como comprador quiero introducir el código de una promoción de productos para comprarla a menor precio.		
Valoración 44	Prioridad 21	Estimación S

HU #15 Realizar Devolución		
Como comprador quiero devolver un producto para recuperar el dinero.		
Valoración 87	Prioridad 10	Estimación M

HU #19 Calificar Producto		
Como vendedor quiero que los usuarios puedan calificar mis productos para demostrar mi fiabilidad.		
Valoración 70	Prioridad 14	Estimación XS

HU #20 Reportar un Error		
Como usuario quiero reportar un error para que sea resuelto y continuar navegando con normalidad.		
Valoración	55	Prioridad 17 Estimación XS

HU implementadas sin vista

HU #4 Retirar Producto		
Como vendedor quiero eliminar productos para que no se venda más.		
Valoración	115	Prioridad 5 Estimación XS
Validación		
1) Mostramos tienda		
2) Eliminamos el producto deseado tanto de la tienda como del inventario.		
3) Volvemos a mostrar la tienda para comprobar que no sale esta en ella.		

HU #5 Reabastecer Producto		
Como vendedor quiero comprar más unidades de productos agotados para que haya unidades de nuevo		
Valoración	117	Prioridad 4 Estimación XS
Validación		
1) Mostramos tienda		
2) Cambiamos las unidades del producto seleccionado.		
3) Volvemos a mostrar la tienda para comprobar que el stock del ha cambiado.		

HU #11 Eliminar del Carrito		
Como comprador quiero eliminar un productode la lista listos para no comprar.		
Valoración	81	Prioridad 13 Estimación XS
Validación		
1) Mostramos el carrito		
2) Eliminamos el producto del carrito		
3) Volvemos a mostrar el carrito para ver que se ha eliminado correctamente		
4) Probamos que no se sale del rango del array		

HU #16 Buscar un Producto		
Como comprador quiero buscar un producto en la tienda para comprarlo		
Valoración 91	Prioridad 9	Estimación S
Validación		
1) Introducimos el id del producto a buscar		
2) Se busca el producto y se comprueba que lo que aparece es:		
3) Si es nulo se muestra que no se encontró.		
4) Si no es nulo se muestra que producto es.		

1.3. Sprint reviews, Sprint retrospective, Sprint planning y Sprint backlog

Los sprints reviews recogerán los datos reunidos al final del sprint para inspeccionar el Incremento y adaptar el Product Backlog si fuese necesario.

El sprint retrospectivo recogerá planes de mejoras que se implementaran en el siguiente sprint.

El sprint planning recogerá el trabajo a realizar en cada sprint.

En el sprint backlog recogeremos la lista de tareas del product backlog que decidimos realizar en cada sprint.

1.3.1 Primer sprint (1 Marzo-15 Marzo)

1.3.1.1 Sprint Review

Al final del primer sprint, hemos conseguido acabar con todo el trabajo planeado para este. Este trabajo se corresponde con las historias de usuario más importantes según la estimación realizada (mostrar tienda, mostrar inventario, retirar producto y reabastecer producto). También hemos creado el repositorio GitHub e instalado los programas necesarios. También hemos añadido al product backlog todas las historias de usuario iniciales, a las que se podrán añadir (o eliminar alguna) en próximos sprints. En general estamos satisfechos con este primer sprint y la forma de trabajo escogida, por lo que intentaremos extenderla al próximo sprint.

1.3.1.2 Sprint retrospective

En el primer sprint, en el que hemos realizado la organización del proyecto, el diseño de las primeras historias de usuario y la primera codificación del trabajo, valoramos positivamente el uso de métodos de SCRUM realizados. Para empezar, hemos usado un método de planning póker para asignar valoración a las historias de usuario. Para ello, cada persona le asignaba un valor subjetivo y así en conjunto obtenemos un valor más fiable de su valor y por tanto, de su estimación, reduciendo así las diferencias subjetivas que pueda tener cada uno. Esto ha sido muy útil para priorizar las historias de usuario más importantes y empezar a codificarlas. Además, del método de estimación de tallas de camisetas, para hacernos una idea de la dimensión de cada una. Finalmente, con los daily meetings, nos hemos reunido cada pocos días (unas 2 veces por semana) para aclarar las dudas que surjan entre

nosotros y acordar el trabajo próximo a realizar. Por tanto, en el siguiente sprint continuaremos realizando los métodos de SCRUM pues han sido útiles para la realización del trabajo.

1.3.1.3 Sprint planning

Comenzamos con la preparación del entorno de desarrollo, creando carpeta compartida en Drive y creando cuentas personales y repositorio en GitHub. Instalación de todos los programas necesarios para el proyecto.

Creación de historias de usuario más importantes, descripción de estas y ordenación por prioridad.

Codificación de historias de usuario más importantes (4 más importantes) y realización de sus diagramas correspondientes.

1.3.1.4 Sprint backlog

- Creación repositorio GitHub y cuentas personales GitHub
- Instalación programas
- Historia de usuario Mostrar Tienda
- Historia de usuario Retirar producto
- Historia de usuario Reabastecer Producto
- Historia de usuario Mostrar Inventario

1.3.2 Segundo sprint (16 Marzo-29 Marzo)

1.3.2.1 Sprint Review

Al final de este sprint hemos conseguido todos los objetivos propuestos. Además hemos avanzado con el código añadiendo todos los productos que pensamos ofrecer. Se ha añadido el patrón de Singleton y se han implementado las funciones del carrito. Se han ampliado los diagramas de clases y hemos añadido el diagrama de dominio. Hemos adaptado algunos métodos a las nuevas funcionalidades. Consideramos que hemos avanzado bien, al igual que en el sprint anterior. En el siguiente sprint trabajaremos de la misma manera para intentar mantener la línea de trabajo.

1.3.2.2 Sprint retrospective

En el segundo sprint, hemos arreglado en grupo los errores del sprint 1, ya indicados en el sprint planning, y hemos realizado una planificación conjunta para las tareas a realizar, recogido en ese mismo apartado de sprint planning. Continuamos usando la estimación del sprint anterior para continuar realizando las historias de usuario más prioritarias, añadiendo algunas nuevas, recogidas en el product backlog, realizando meetings grupales cada varios días para las posibles dudas o problemas. Por último, realizamos los diagramas de clases y dominio de la aplicación. Continuaremos en el tercer sprint con la misma metodología, aprovechando para intentar aplicar interfaz gráfica a las historias de usuario ya creadas, además de continuar con la realización de estas.

.

1.3.2.3 Sprint planning

En este segundo sprint, realizaremos las correspondientes correcciones del sprint 1 (eliminar diagramas repetidos en Documento UML Diseño, modificación GitHub), realización de historias de usuario más prioritarias, realización de modelo de dominio y diagrama de clases de la aplicación e implementación de patrones de diseño en las clases ya implementadas.

1.3.2.4 Sprint backlog

- Patrón Singleton en Clase Tienda (22/marzo)
- Historia de Usuario Añadir al Carrito (22/marzo)
- Historia de usuario Eliminar del carrito (24/ marzo)
- Historia de usuario Mostrar carrito (24/ marzo)
- Creación de clases de productos (29/marzo)

1.3.3 Tercer sprint (30 Marzo-19 Abril)

1.3.3.1 Sprint Review

Al final de este sprint hemos conseguido todos los objetivos propuestos. Finalmente, para conseguir tener el código más compacto y unificar los arrays de Tienda e Inventario hemos usado el array de la clase Tienda de forma que se vaya actualizando según los productos que haya en el array de Inventario que se encuentren “activos”. Hemos separado las clases en paquetes siguiendo el modelo-vista-controlador. Hemos creado las bases de datos inventario, usuarios y ventas aplicando el patrón DAO. Hemos empezado a implementar la interfaz gráfica, creando la ventana principal y las ventanas de usuario y de carrito. También hemos añadido observadores para su futuro uso en los siguientes sprints. Hemos avanzado con las historias de usuario añadiendo la clase de usuario.

1.3.3.2 Sprint retrospective

En este sprint hemos realizado uso del sistema Github de control de versiones, para realizar un mejor trabajo en equipo. Continuaremos usándolo en los sprints restantes, ya que ha mejorado enormemente la coordinación y el control del trabajo realizado. Además, hemos pasado a complementar este documento con un product backlog más visual, en Github, útil y llamativo visualmente. Valoramos positivamente estos cambios en el modelo de trabajo y continuaremos aplicándolos en el resto del proyecto.

.

1.3.3.3 Sprint planning

En el tercer sprint avanzaremos en el código, introduciendo una clase que contenga un array donde añadiremos todos los productos y corregiremos los métodos para conseguir una implementación más compacta. Avanzaremos con las historias de usuario, implementaremos más patrones de diseño y empezaremos a trabajar con la interfaz, intentando acabar o dejar avanzada la implementación del Modelo-Vista-Controlador. Además, realizaremos la implementación de la base de datos, en XML.

1.3.3.4 Sprint backlog

- Historia de usuario Cambiar precio producto (5/abril)
- Historia de usuario Reservar producto (5/abril)
- Base de datos (7/abril)
- Paquetes modelo - vista - controlador (11/abril)
- Creación de ventana principal, carrito y ventana de usuario (11/abril)
- Diagrama de clases (12/abril)
- Historia de usuario Buscar producto (19/abril)
- Creación de clase usuario(19/abril)

1.3.4 Cuarto sprint (20 Abril-3 Mayo)

1.3.4.1 Sprint Review

Al final de este sprint hemos conseguido la mayoría de los objetivos propuestos. Finalmente hemos decidido eliminar la clase ListaUsuarios ya que no la usábamos, y la hemos sustituido por una nueva clase Sesion, la cual se inicializa con un usuario por defecto que se puede cambiar luego al iniciar sesión o al registrarse. También, hemos terminado de implementar el patrón DAO para el manejo de las bases de datos, así como los patrones Observer (UsuariosObserver, InventarioObserver). Además, acabamos la implementación del modelo-vista-controlador, para ello hemos terminado de implementar la interfaz gráfica añadiendo diferentes pantallas para iniciar sesión de los usuarios (cliente y vendedor) y que estos también se puedan registrar, ventanas para realizar la compra (carrito y pago), otras para el inventario para que el vendedor pueda añadir o modificar el precio de sus productos, así como que pueda ver un resumen de los productos que ha vendido, y por último una ventana principal en la que se encuentran todos los productos y que mediante el uso de botones poder ir moviéndose por las diferentes pantallas hasta ver todos los productos que están añadidos a la tienda.

1.3.4.2 Sprint retrospective

En este último sprint hemos seguido haciendo uso de la aplicación Github para el control de versiones, realizando así un mejor trabajo en grupos. No se han conseguido implementar todas las HU propuestas al principio y quizás ha sido por falta de comunicación en el equipo. La cantidad de trabajo realizada por cada miembro no ha sido proporcional.

1.3.4.3 Sprint planning

En este último sprint terminaremos el proyecto, para ello debemos terminar de implementar el modelo-vista-controlador, centrándonos sobre todo en acabar la interfaz gráfica, también, debemos terminar de implementar los dos patrones que nos faltan (DAO y Observers), así como realizar los cambios en la parte del modelo y controlador que sean necesarios para que el proyecto funcione correctamente. Intentaremos añadir el mayor número de historias de usuario que nos faltan por implementar. Por último, tenemos que terminar los dos documentos (SCRUM y UML) haciendo los diagramas correspondientes.

1.3.4.4 Sprint backlog

- Historia de usuario Crear cuenta de usuario Cliente (26/abril)
- Historia de usuario Crear cuenta de usuario Vendedor(26/abril)
- Historia de usuario Iniciar sesión de usuario Cliente (26/abril)
- Historia de usuario Iniciar sesión de usuario Vendedor(26/abril)
- Patrones DAO (26/abril)
- Patrón UsuariosObserver (29/abril)
- Historia de usuario Añadir método de pago (29/abril)
- Historia de usuario Comprar (30/abril)
- Patrón InventarioObserver (2/mayo)

1.4. Product backlog

El product backlog recoge la lista de tareas a realizar durante todo el proyecto. El product owner se encargará de su gestión.

1.4.1 Tareas Pendientes

- Historia de usuario Ordenar Productos Por jugador (3/mayo)
- Historia de usuario Cambiar precio producto (3/mayo)
- Historia de usuario Publicitar producto (3/mayo)
- Historia de usuario Calificar producto (3/mayo)
- Historia de usuario Reportar error (3/mayo)

1.4.2 Tareas Terminadas

- Creación repositorio GitHub y cuentas personales GitHub
- Instalación programas
- Historia de usuario Mostrar Tienda
- Historia de usuario Mostrar Inventario-
- Patrón Singleton en Clase Tienda (22/marzo)
- Historia de Usuario Añadir al Carrito (22/marzo)
- Historia de usuario Mostrar carrito (24/ marzo)
- Creación de clases de productos (29/marzo)
- Historia de usuario Reservar producto (5/abril)
- Base de datos (7/abril)
- Paquetes modelo - vista - controlador (11/abril)
- Creación de ventana principal, carrito y ventana de usuario (11/abril)
- Diagrama de clases (12/abril)
- Historia de usuario Buscar producto (19/abril)
- Creación de clase usuario(19/abril)
- Historia de usuario Crear cuenta de usuario Cliente (26/abril)
- Historia de usuario Crear cuenta de usuario Vendedor(26/abril)
- Historia de usuario Iniciar sesión de usuario Cliente (26/abril)
- Historia de usuario Iniciar sesión de usuario Vendedor(26/abril)
- Patrones DAO (26/abril)
- Patrón UsuariosObserver (29/abril)

- Historia de usuario Añadir método de pago (29/abril)
- Historia de usuario Comprar (30/abril)
- Patrón InventarioObserver (2/mayo)

1.4.3 Tareas Terminadas (Sin interfaz)

- Historia de usuario Retirar producto (13/ marzo)
- Historia de usuario Buscar producto (9/ marzo)
- Historia de usuario Eliminar del carrito (24/ marzo)
- Historia de usuario Reabastecer Producto (24/marzo)
- Historia de usuario Cambiar precio producto (5/abril)

1.4.4 Modificaciones

- Definición de productos como clases abstractas (22/marzo)
- Patrón Singleton en Clase Tienda (22/marzo)
- Creación de paquetes modelo - vista - controlador (11/ abril)
- Creación de interfaz gráfica (11/ abril)

ISSUES GitHub:

<input type="checkbox"/>	<input checked="" type="radio"/>	HU Comprar	finalizado	#19 opened 2 days ago by luceroabregu	
<input type="checkbox"/>	<input checked="" type="radio"/>	Boton resumen compras	finalizado	#18 opened 5 days ago by mbasca	
<input type="checkbox"/>	<input checked="" type="radio"/>	Acabar pantalla tienda	finalizado	#17 opened 5 days ago by mbasca	
<input type="checkbox"/>	<input checked="" type="radio"/>	Formularios para añadir producto Conjunto, Botas, Guantes	finalizado	#16 opened 5 days ago by mbasca	
<input type="checkbox"/>	<input checked="" type="radio"/>	Añadir productos al inventario	finalizado	#15 opened 5 days ago by mbasca	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Ordenar productos por jugador	por empezar	#14 opened 6 days ago by luceroabregu	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Crear cuenta de usuario	finalizado	#13 opened 13 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Publicitar un producto	por empezar	#12 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Buscar un producto	finalizado por empezar	#11 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Realizar devolución	por empezar	#10 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Añadir método de pago	finalizado	#9 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Añadir promoción	por empezar	#8 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Mostrar ingresos y productos vendidos	por empezar	#7 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Reservar producto	finalizado	#6 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	HU: Cambiar precio producto	finalizado	#5 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	Implementar Base de Datos	finalizado	#4 opened 27 days ago by alexpareja	
<input type="checkbox"/>	<input checked="" type="radio"/>	Implementar la interfaz	finalizado	#3 opened 27 days ago by alexpareja	

1.5. Descripción del trabajo realizado por cada miembro del grupo

-Sergio

Sprint 1:

Realización de Historia de usuario Mostrar tienda y diagrama secuencia.

Sprint 2:

Realización de Historias de usuario Mostrar carrito, añadir y eliminar del carrito y diagramas Patrón Singleton Mostrar tienda.

Sprint 3:

Diseño de Bases de Datos, codificación del modelo y patrones DAO y Observer.

Sprint 4:

Terminar patrones DAO, crear patrón UsuariosObserver, implementar Historias de usuario Crear cuenta de usuario Cliente, Crear cuenta de usuario Vendedor, Iniciar Sesión usuario Cliente e Iniciar Sesión usuario Vendedor (clase Sesión). Diagramas de clases y secuencia de los patrones DAO y UsuariosObserver y de las historias de usuario implementadas.

Actualizar diagramas de clase y secuencia usando modelo-vista-controlador para las historias de usuario que se habían implementado en anteriores sprints.

-Cristina

Sprint 1:

Realización de Historia de usuario Mostrar tienda y diagrama secuencia.

Sprint 2:

Realización de Historias de usuario Mostrar carrito, añadir y eliminar del carrito y diagramas Patrón Singleton Mostrar tienda.

Sprint 3:

Diseño de Bases de Datos, codificación del modelo y patrones DAO y Observer.

Sprint 4:

Terminar patrones DAO, crear patrón UsuariosObserver, implementar Historias de usuario Crear cuenta de usuario Cliente, Crear cuenta de usuario Vendedor, Iniciar Sesión usuario Cliente e Iniciar Sesión usuario Vendedor (clase Sesion). Diagramas de clases y secuencia de los patrones DAO y UsuariosObserver y de las historias de usuario implementadas. Actualizar diagramas de clase y secuencia usando modelo-vista-controlador para las historias de usuario que se habían implementado en anteriores sprints.

-Jesús

Sprint 1:

Realización de Historia de usuario Retirar producto e Historia de usuario Reabastecer Producto, diagrama de secuencias y de clase.

Sprint 2:

Creación de clases Pantalon, PantalonChandal, PantalonCorto

Sprint 3:

HU buscar producto, con su diagrama de secuencias, y modificación de reabastecer. Creación clase usuario.

Sprint 4:

Realización del formulario de los guantes para la interfaz. Explicación de parte de la arquitectura. Crear test PantalonChandal.

-Lucero

Sprint 1:

Realización de Historia de usuario Retirar producto e Historia de usuario Reabastecer Producto, diagrama de secuencias y de clase.

Sprint 2:

Creación de clases Pantalon, PantalonChandal, PantalonCorto

Sprint 3:

HU buscar producto y modificación de reabastecer. Creación clase usuario.

Sprint 4:

Crear clase Usuario. HU Comprar implementada en la interfaz, Añadir método de pago implementada en la interfaz, Realizar comprobante compras implementada en la interfaz. Crear test PantalonChandal.

-Olenka

Realización de Diagrama de clase y modelo de dominio

Sprint 3: Realización de diagramas de caso de uso.

Sprint 4: Actualización diagrama de clases general del proyecto

-Madalina

Realización de Historia de usuario Mostrar inventario. Realización del diagrama de secuencias.

Sprint 2:

Creación de clases Conjunto, ConjuntoLocal, ConjuntoVisitante y realización de diagramas de clase.

Sprint 3: Ventana principal interfaz gráfica, ventana usuario, ventana carrito.

Sprint 4: Ventana inventario con botones, HU añadir producto (conjuntos y camiseta, pantalones y botas) y HU cambiar precio, HU mostrar inventario actualizado, y el patrón Observador inventarioObserver, junto a sus diagramas de secuencias y de clases. Corrección de errores. Comentario Código.

-Alejandro

Realización de Historia de usuario Mostrar inventario. Realización del diagrama de secuencias.

Sprint 2:

Creación de clases Conjunto, ConjuntoLocal, ConjuntoVisitante y realización de diagrama de clases.

Sprint 3: Creación paquetes modelo-vista-controlador. Ventana principal interfaz gráfica, ventana usuario, ventana carrito.

Sprint 4: Ventana inventario, HU mostrar inventario actualizado y el patrón Observador inventarioObserver, junto a sus diagramas de secuencias y de clases. Implementación de la HU Mostrar Carrito. Finalización interfaz pantalla tienda. Corrección de errores. Limpieza de código. Comentario Código. Implementación TestBotas con JUnit.

-Andrés

Realización de Historia de usuario Mostrar inventario. Realización del diagrama de secuencias.

Sprint 2:

Creación de clase Guantes y realización del diagrama de clases.

Sprint 3: HU reservar producto y cambiar precio producto

Sprint 4: Implementación de la HU reservar producto en la interfaz

-Tareas grupales

Actividad grupal estimación planning póker

Realización de Documento Scrum y Diseño

Instalación programas y creación de repositorio GitHub