

Bidding Trading

Andrés Felipe Miranda Díaz
Julián Alejandro Rojas Daza
Julián David Bermúdez Franco
Nicolas Moreno Roa

Universidad San Buenaventura
Bogotá-DC
2020

Contenido

I.	INTRODUCCIÓN	1
II.	JUSTIFICACIÓN DEL PROYECTO	1
III.	DESCRIPCIÓN DEL PROYECTO	1
A.	ALCANCE	1
1)	TIEMPO	2
2)	COSTOS	2
B.	OBJETIVO GENERAL.....	2
C.	OBJETIVOS ESPECÍFICOS.....	2
IV.	ESPECIFICACIÓN DE REQUERIMIENTOS	2
V.	ROLES	4
A.	MAPA DE PODER	4
VI.	STAKEHOLDERS.....	4
VII.	CASOS DE USO	5
VIII.	CRONOGRAMA	5
IX.	DIAGRAMA DE DESPLIEGUE	6
X.	PATRONES DE DISEÑO	6
A.	PATRONES DE CREACIÓN	6
B.	PATRONES ESTRUCTURALES.....	7
C.	PATRONES DE COMPORTAMIENTO.....	7
D.	PATRÓN SELECCIONADO Y APLICADO AL PROYECTO.....	7
XI.	DIAGRAMA DE ACTIVIDADES	7
XII.	ANEXOS.....	10
A.	CRONOGRAMA DE ACTIVIDADES.....	10

Bidding Trading

Resumen – Este documento consiste en informar la realización del proyecto Bidding Trading con un despliegue de información que especificara las funciones del servicio del proyecto.

Abstrap

This document consists of informing the completion of the Trading Tender project with a display of information that specifies the functions of the project service

Keywords— Bidding Trading (Pagina web de interacción de subastas)

I. INTRODUCCIÓN

Este documento presenta un informe detallado de la aplicabilidad que presenta el proyecto de la página web Bidding Trading, identificando una solución para las subastas online para el área de dispositivos móviles, llevando un control de registro

Bidding Trading, nace para suplir una necesidad proporcionándoles una herramienta útil a los usuarios que sirve como apoyo y referencia en el momento de realizar respectiva subasta.

II. JUSTIFICACIÓN DEL PROYECTO

En la actualidad las personas cambian por lo menos una vez al año del teléfono móvil, por distintas razones como actualizarse en tecnología, daño en el dispositivo, perdida de este mismo entre otras, pero que pasa con los dispositivos que son reemplazados por el nuevo, son almacenados, olvidados, regalados o revendidos como dispositivos usados atreves de portales de venta o como parte de pago en compañías telefónicas que teniendo en cuenta de estas estadísticas han implementado estrategias comerciales en las cuales le ofrecen a los usuarios el cambio de un equipo “antiguo” por una parte del precio del nuevo si está bien cuidado (en perfecto estado), con una extensión del tiempo de sus pagos, buscando así fidelizar y amarrar al usuario hacia estas compañías beneficiándolos únicamente a ellos.

Sin embargo, si comenzamos a enumerar las innumerables posibilidades para nuestros dispositivos existe una que da más beneficios económicos que darlo como parte de pago de otro celular o venderlo por partes.

Teniendo en cuenta todas las veces que se tiene un dispositivo móvil antiguo guardado en el closet, con el display dañado, un modelo viejo que ya no compran en el mercado tradicional. Se crean páginas de venta de productos usados (Mercado Libre, Amazon o simplemente una publicación en Facebook), deseado cambiar nuestro teléfono por alguna remuneración económica y vendiéndolo a un menor precio y a un único postor, esperando la mejor oferta y teniendo que comparar manualmente si la primera oferta que le hicieron fue mejor que la última o que alguna de las doce siguientes a esta misma. Si se piensa en esto se ve la razón por la cual las personas prefieren no buscar la mejor oportunidad económica, encontrar el mejor postor en una venta tradicional es complicado, pero en una subasta no, el producto será vendido a la persona que pueje la mayor cifra.

En una subasta se inicia con el precio mínimo, el mismo precio que daría en una página tradicional de ventas y comienza la puja. Por el teléfono móvil de modelo viejo que ofrecían solo dos personas el mismo precio, en una subasta esas dos personas se enfrentan por el mismo objeto y lo sabes, saben cuánto a ofrecido el otro y que pujar un poco más le daría las partes de ese teléfono móvil del cual se consiguen pocos repuestos por lo viejo de su modelo. Por el celular de gama media que fue cambiado por un modelo de gama alta porque tenía el display dañado, una persona comienza la oferta con el precio mínimo y a alguien le sigue pareciendo barato y vuelve a ofertar, así mismo hacen otras tres personas que, aunque el precio original se allá elevado siguen ofertando. Por eso las subastas son la mejor forma de encontrar al mejor postor de tu teléfono móvil, ya que el precio siempre tiende a elevarse, puedes consultar en cualquier momento quien es el mejor postor en ese momento y cuando se cierre la puja siempre se la llevara la mejor puja.

Existen páginas de ventas en línea como EBAY que ofrece un servicio mixto entre subastas y compras directas, pero paginas exclusivas de subastas son pocas, SUPERBID es el mejor ejemplo de una página exclusiva de subastas y aparte de dicha página son muy pocos. La idea de nuestro proyecto parte de dicha ausencia de páginas de subastas, BIDDING TRADING será una página a nivel Colombia que proporcione un servicio de subastas en línea y así proporcionar una página de calidad en donde se pueda subastar a nivel nacional.

III. DESCRIPCIÓN DEL PROYECTO

A. ALCANCE

Desarrollar un FRONT END escrito en PHP integrado con Microsoft Azure y MYSQL el FRONT END contara con campos de ingreso y registro de usuarios para usuarios, un módulo de compra y venta de productos, publicación de

productos y una modulo específico para las subastas de los dispositivos móviles.

- Plataforma web desarrollada en PHP usando el Framework Laravell (FRONT END)
- Conexión a base de datos MYSQL
- Hosting de servicios en Microsoft Azure
- Libro de proyectos.
- Libro de plan de proyectos.
- Libro de requerimientos
- Test del aplicativo
- Manuales de usuarios

1) TIEMPO

Finalización: El aplicativo web debe estar completamente finalizado integrando base de datos en Microsoft Azure.

Fecha de finalización: 08 de mayo de 2020

2) COSTOS

En la realización de Bidding Trading se utiliza los recursos gratis de la cuenta academia de Microsoft Azure para generar nuestra página web.

B. OBJETIVO GENERAL

Desarrollar un aplicativo web donde los usuarios pueda subastar los dispositivos móviles.

C. OBJETIVOS ESPECÍFICOS

- Desarrollar una matriz de riesgos que mida los tiempos de entrega y desarrollo de la plataforma web.
- Implementar una metodología ágil para el desarrollo del proyecto
- Desarrollar la Plataforma web en PHP usando el Framework laravell (FRONT END) con campos de ingreso, registro y consulta, publicación de productos y botón de subasta.
- Realizar la base de datos en MYSQL
- Desarrollar la integración con Microsoft Azure

ESTIMACIÓN INICIAL DE RIESGOS

- Que las partes externas influyan en el tiempo de desarrollo del proyecto.
- Enfermedad de alguno de los gerentes y desarrolladores del producto
- Elementos externos no controlables tales como desastres naturales.

REQUERIMIENTOS.

- Aprobación del diseño preliminar.
- Aceptación final de la plataforma.
- Aceptación de cambios en plazos adicionales.

IV. ESPECIFICACIÓN DE REQUERIMIENTOS

IDENTIFICADOR: r1		NOMBRE: Registrar información o datos de una persona	
Tipo: (NECESARIO/DESEABLE) Necesario		REQUERIMIENTO QUE LO UTILIZA O ESPECIALIZA:	¿CRÍTICO? Si
PRIORIDAD DE DESARROLLO: Alta		DOCUMENTOS DE VISUALIZACIÓN ASOCIADOS: no	
ENTRADA: <ul style="list-style-type: none">• Apellidos y Nombres completos• Cedula• Dirección De Residencia• País• Código Postal• Edad• Correo Electrónico• Firma• Contraseña De Usuario• Teléfono Fijo• Celular		SALIDA: Registro correcto de una persona	
DESCRIPCIÓN: Precondición: Completar los campos asignados Descripción: Se debe completar los campos solicitados por la pagina Postcondición: Se realizará el registro de una persona			
MANEJO DE SITUACIONES ANORMALES 1. Persona ya registrada en el sistema (se mostrará en pantalla un mensaje que dirá que la persona ya está registrada en el sistema)			
CRITERIOS DE ACEPTACIÓN Se supondrá por defecto que hay al menos dos criterios de aceptación: 1. Los datos ingresados al sistema en el momento de realizar el registro de una persona son correctos y los indicados y establecidos para llevar a cabo su correcto registro en el sistema y poder realizar sus trámites dentro del mismo.			

Tabla 1 (requerimientos registro de información).

IDENTIFICADOR: R2	NOMBRE: SubastarUn Articulo	
Tipo: (NECESARIO/ DESEABLE) Necesario	REQUERIMIENTO QUE LO UTILIZA O ESPECIALIZA:	¿CRÍTICO? Si
PRIORIDAD DE DESARROLLO: Alta	DOCUMENTOS DE VISUALIZACIÓN ASOCIADOS: No	
ENTRADA: <ul style="list-style-type: none"> Tipo de Objeto Nombre del Producto Descripción Del Producto Características: Procesador Memoria RAM Memoria Interna Memoria Expandible Fecha De Inicio de la subasta Fecha de cierre de la subasta Imagen de producto Precio De Arranque 		SALIDA: <ul style="list-style-type: none"> Genera la creación e interacción de la subasta Publicación del producto
DESCRIPCIÓN: Precondición: Tener una cuenta para ingresar al sistema. Descripción: Permite la creación, publicación e interacción de un producto. Postcondición: Visualización e interacción de la subasta con los demás usuarios del sistema		
MANEJO DE SITUACIONES ANORMALES <ol style="list-style-type: none"> No completar la información requerida para la publicación del producto (Se informaría al usuario que debe completar toda la información requeridas y activar el checkbox) 		
CRITERIOS DE ACEPTACIÓN Los datos ingresados al sistema en el momento de realizar el registro de un producto son correctos y los indicados y establecidos para llevar a cabo la subasta en el sistema y poder realizar sus trámites.		

Tabla 2 (requerimientos subastar artículo).

IDENTIFICADOR: R3	NOMBRE: Realizar una puja (realizar una oferta)	
Tipo: (NECESARIO/ DESEABLE) Necesario	REQUERIMIENTO QUE LO UTILIZA O ESPECIALIZA:	¿CRÍTICO? Si
PRIORIDAD DE DESARROLLO: Alta	DOCUMENTOS DE VISUALIZACIÓN ASOCIADOS:	
ENTRADA: <ul style="list-style-type: none"> Usuario Para Login Seleccionar El Producto Ofertar En La Subasta Cantidad a pujar en la subasta 		SALIDA: <ul style="list-style-type: none"> Iteración con la subasta. Seleccionar método de pago. Digitar Datos De Pago Registrar dirección de entrega. Pagar.
DESCRIPCIÓN: Precondición: Tener una cuenta creada Descripción: Ofertar por los productos de la subasta a interés teniendo interacción con los de más usuarios logueados. Postcondición: Se dará por finalizada la subasta		
MANEJO DE SITUACIONES ANORMALES <ol style="list-style-type: none"> Campos obligatorios solicitados quedaron sin diligenciar (mensaje que indique los campos que eran obligatorios quedaron sin diligenciar) El usuario ya está registrado en el sistema (mensaje que indique que el usuario ya está registrado en el sistema) 		
CRITERIOS DE ACEPTACIÓN Los datos ingresados del método de pago son correctos y los indicados y establecidos para llevar a cabo su correcto registro de pago en el sistema y poder realizar sus trámites.		

Tabla 3 (requerimientos login).

V. ROLES

NOMBRE	CARGO	DESCRPCIÓN
Andrés Miranda	Gerente de proyectos.	Encargado de presentar el diseño inicial, ajustar alcance, tiempo y entrega final.
Julián Bermúdez	Gestión base de datos	Encargado de presentar el diseño y modelo de la base de datos
Nicolas Moreno	Gestión de diseño	Encargado de presentar el diseño inicial, ajustes dados en el camino
Julián Rojas	Gestión desarrollo	Encargado de presentar el diseño inicial para el desarrollo del aplicativo

Tabla 4 (roles de los integrantes).

A. MAPA DE PODER

El mapa de poder es la descripción de todas las personas que pueden tener influencias sobre el proyecto, para esto se debe determinar su naturaleza, fuerza y dirección.

Al desarrollar el mapa de poder se deben identificar las influencias sobre las personas que buscan apoyar nuestro proyecto, las relaciones que hay entre ellas, lo que ayudará a enfocarse hacia obtener los recursos y el apoyo que se necesitará para la gestión de su desarrollo; En un gran porcentaje de las ocasiones el éxito del proyecto llega a depender de las partes interesadas por el poder de decisión que aportan a sus iniciativas, por lo que la gestión de las mismas es un aspecto importante de la gestión de proyectos.

A continuación, podrá encontrar el mapa de poder de nuestro proyecto en el cual se define el rol y el área a la que pertenece cada uno.



Ilustración 1 (mapa de poder)

VI. STACKEOLDERS

<i>Stackeholders Principales</i>	Descripción
Giovanny Sánchez	Encargo de contactar y realizar las conexiones con los clientes.
Empresa, familia, recursos de la Universidad san buenaventura.	Se verán afectados por la dedicación del tiempo, además de ser inversores externos de recursos físicos, monetarios y no tangibles será importante establecer planes de desarrollo para que no afecte la entrega del producto final.

Tabla 5 (Stackeholders principales).

VII. CASOS DE USO

Un caso de uso es la descripción de una acción o una actividad en un proceso que es realizada por un actor que interactúa directamente con el sistema. Los casos de uso para Bidding Trading son los siguientes:

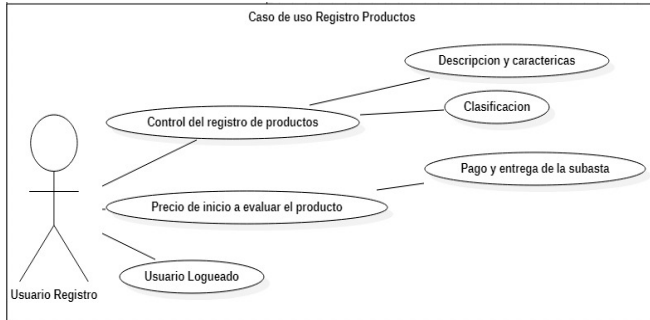


Ilustración 2 (caso de uso registro de productos)

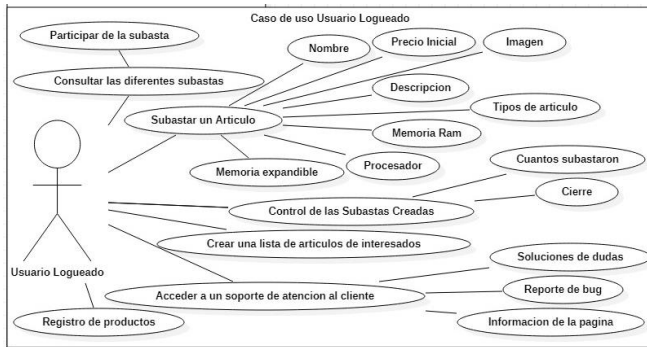


Ilustración 3 (caso de uso usuario logueado)

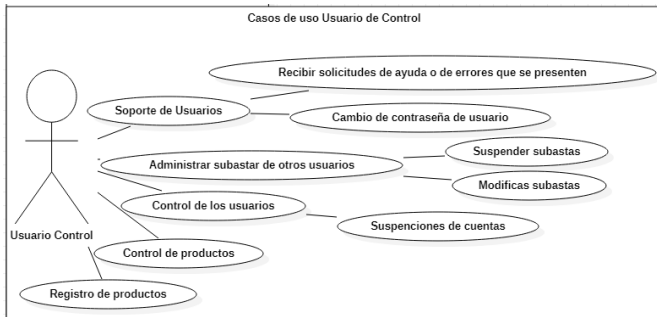


Ilustración 4 (caso de uso usuario de control)

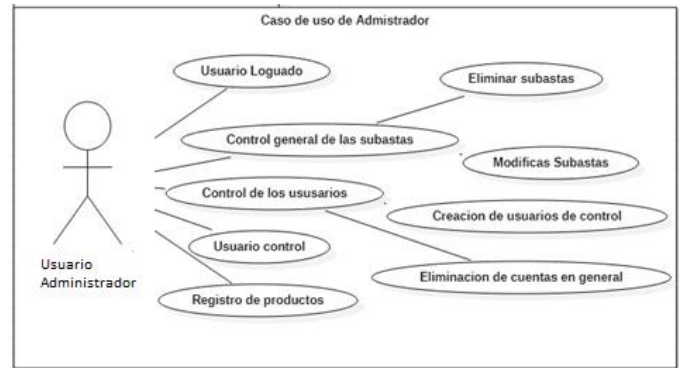


Ilustración 5 (caso de uso Administrador)

VIII. CRONOGRAMA

Una versión resumida del cronograma del proyecto, el cronograma completo se encuentra en el anexo A.

NOMBRE		DURACION
1	Diseño	21,5 días
2	Creación de diseño de casos de uso.	9 días
3	Creación de diseño MARVEL APP para front de presentación de productos.	1 día
4	Creación de diseño MARVEL APP para front de registro de productos.	1 día
5	Creación de diseño MARVEL APP para front de modificación de productos	1 día
6	Creación de diseño MARVEL APP para front de registro de usuarios	1 día
7	Creación de diseño MARVEL APP para front de gestión de pagos	1 día
8	Creación de diseño de estructura relacional de base de datos	2 días
9	Definición de estructura de seguridad de datos - Transaccional.	7 días
10	Aproximación a norma PCI-DSS.	1 día
11	Definición de arquitectura de infraestructura de aplicación	0,5 días
12	Definición de estructura de web Server (Tipo de servidor, puertos, certificados, etc).	0,5 días
13	Definición de estructura de Frontend (Responsive, ad-works, etc).	0,5 días
14	Definición de estructura de Backend (Lógica de negocio, caos de uso, seguridad, etc).	0,5 días
15	Definición de estructura de manipulación de la información.	0,5 días
16	Definición de almacenamiento de información.	1 día
17	Definición de consumo de información.	1 día
18	Definición de respaldo de la información.	1 día

Tabla 6 (cronograma etapa de inicio y planificación).

NOMBRE		DURACION
19	Implementación	17,625 días
20	Creación de base de datos relacional.	1 día
21	Implementación de casos de uso.	1 día
22	Compra y adquisición de servicio de cloud.	0,25 días
23	Despliegue de contenedor de base de datos mysql.	0,125 días
24	Despliegue de contenedor de web server.	0,125 días
25	Despliegue de contenedor para webservices.	0,125 días
26	Creación de elementos de lógica de negocio Backend.	2 días
27	Creación de elementos de lógica de negocio frontend.	2 días
28	Creación de front de presentación de productos.	1 día
29	Creación de front de registro de productos.	1 día
30	Creación de front de modificación de productos	1 día
31	Creación de front de registro de usuarios	1 día
32	Creación de front de gestión de pagos	1 día
33	Integración con pasarela de pagos.	2 días
34	Implementación de procesos de seguridad	2 días
35	Implementación de procesos de almacenamiento.	1 día
36	Implementación de procesos de respaldo de información.	1 día
37	Procesos de PCI-DSS	2 días

Tabla 7 (cronograma etapa de implementación y ejecución).

NOMBRE		DURACION
38	Pruebas y Calidad	12,5 días
39	Pruebas funcionales base de datos.	0,5 días
40	Pruebas funcionales servidor web.	0,5 días
41	Pruebas funcionales de webServices	0,5 días
42	Testing software Casos de uso backend.	2 días
43	Testing software Casos de uso frontend.	2 días
44	Pruebas integración con pasarela de pagos.	2 días
45	Pruebas de carga de seguridad.	1 día
46	Pruebas de almacenamiento.	1 día
47	Pruebas de respaldo de información.	1 día
48	Simple compliance PCI-DSS.	2 días

Tabla 8 (cronograma etapa de supervisión y control).

IX. DIAGRAMA DE DESPLIEGUE

Un diagrama de despliegue es un modelo UML que tiene como función modelar físicamente los nodos de un software, Bidding Trading tiene el siguiente modelo:

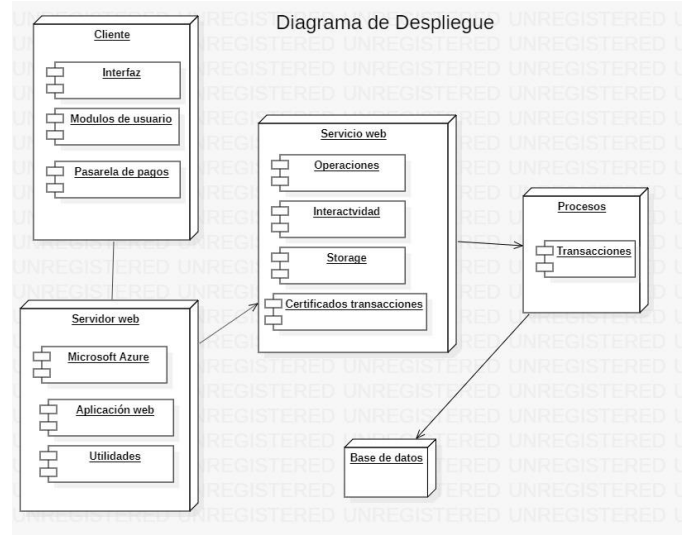


Ilustración 6 (Diagrama De Despliegue)

X. PATRONES DE DISEÑO

Los patrones de diseño son comunes en el desarrollo de software y el diseño de interfaces. Sirven como una solución a problemas de diseño y ayudan a evitarlos si se aplican a tiempo. Los patrones se dividen en tres clases:

- Patrones de creación: utilizados para crear y configurar de clases y objetos.
- Patrones de comportamiento: se centran en la interacción entre asociaciones de clases y objetos definiendo cómo se comunican entre sí.
- Patrones estructurales: su objetivo es desacoplar las interfaces e implementar clases y objetos. Crean grupos de objetos.

Así mismo cada clase tiene su tipo de patrón:

A. PATRONES DE CREACIÓN

- Abstract Factory: Nos provee una interfaz que delega la creación de un conjunto de objetos relacionados sin necesidad de especificar en ningún momento cuáles son las implementaciones concretas.
- Factory Method: Expone un método de creación, delegando en las subclases la implementación de este método.
- Builder: Separa la creación de un objeto complejo de su estructura, de tal forma que el mismo proceso de construcción nos puede servir para crear representaciones diferentes.

- Singleton: limita a uno el número de instancias posibles de una clase en nuestro programa, y proporciona un acceso global al mismo.
- Prototype: Permite la creación de objetos basados en «plantillas». Un nuevo objeto se crea a partir de la clonación de otro objeto.

B. PATRONES ESTRUCTURALES

- Adapter: Permite a dos clases con diferentes interfaces trabajar entre ellas, a través de un objeto intermedio con el que se comunican e interactúan.
- Bridge: Desacopla una abstracción de su implementación, para que las dos puedan evolucionar de forma independiente.
- Composite: Facilita la creación de estructuras de objetos en árbol, donde todos los elementos emplean una misma interfaz. Cada uno de ellos puede a su vez contener un listado de esos objetos, o ser el último de esa rama.
- Decorator: Permite añadir funcionalidad extra a un objeto (de forma dinámica o estática) sin modificar el comportamiento del resto de objetos del mismo tipo.
- Facade: Una facade (o fachada) es un objeto que crea una interfaz simplificada para tratar con otra parte del código más compleja, de tal forma que simplifica y aísla su uso. Un ejemplo podría ser crear una fachada para tratar con una clase de una librería externa.
- Flyweight: Una gran cantidad de objetos comparte un mismo objeto con propiedades comunes con el fin de ahorrar memoria.
- Proxy: Es una clase que funciona como interfaz hacia cualquier otra cosa: una conexión a Internet, un archivo en disco o cualquier otro recurso que sea costoso o imposible de duplicar.

C. PATRONES DE COMPORTAMIENTO

- Command: Son objetos que encapsulan una acción y los parámetros que necesitan para ejecutarse.
- Chain of responsibility: se evita acoplar al emisor y receptor de una petición dando la posibilidad a varios receptores de consumirlo. Cada receptor tiene la opción de consumir esa petición o pasárselo al siguiente dentro de la cadena.
- Interpreter: Define una representación para una gramática, así como el mecanismo para evaluarla. El árbol de sintaxis del lenguaje se suele modelar mediante el patrón Composite.

- Iterator: Se utiliza para poder movernos por los elementos de un conjunto de forma secuencial sin necesidad de exponer su implementación específica.
- Mediator: Objeto que encapsula a otro conjunto de objetos interactúan y se comunican entre sí.
- Memento: Este patrón otorga la capacidad de restaurar un objeto a un estado anterior
- Observer: Los objetos son capaces de suscribirse a una serie de eventos que otro objetivo va a emitir, y serán avisados cuando esto ocurra.
- State: Permite modificar la forma en que un objeto se comporta en tiempo de ejecución, basándose en su estado interno.
- Strategy: Permite la selección del algoritmo que ejecuta cierta acción en tiempo de ejecución.
- Template Method: Especifica el esqueleto de un algoritmo, permitiendo a las subclases definir cómo implementan el comportamiento real.
- Visitor: Permite separar el algoritmo de la estructura de datos que se utilizará para ejecutarlo. De esta forma se pueden añadir nuevas operaciones a estas estructuras sin necesidad de modificarlas.

D. PATRÓN SELECCIONADO Y APLICADO AL PROYECTO

Patrón de comportamiento: State

Permite modificar la forma en que un objeto se comporta en tiempo de ejecución, basándose en su estado interno.

Esto se debe principalmente a que al realizar la postura de productos para la subasta (objeto), se le adicionan varios atributos entre ellos el estado (State), el cual será sobrescrito (@override) de acuerdo a la condición en la que se encuentre durante el proceso de subasta.

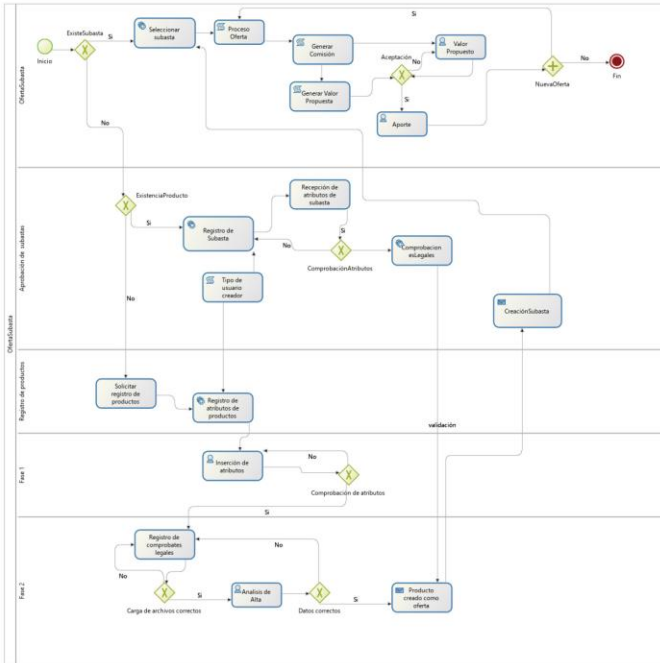
Una vez el producto entra en la fase de adquisición, este objeto se re-instancia como una venta, y a su vez hereda atributos de su estado anterior.

XI. DIAGRAMA DE ACTIVIDADES

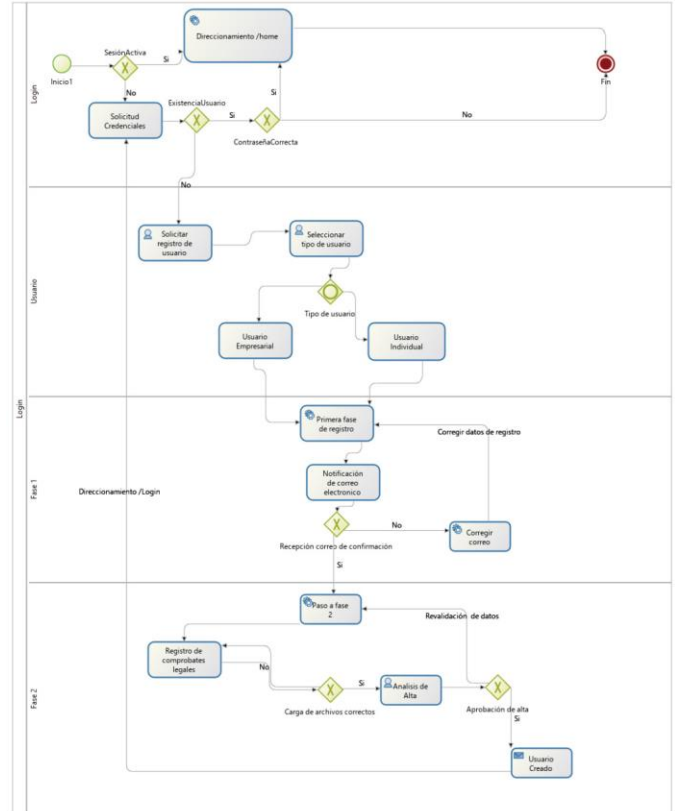
En este diagrama se muestran las actividades, procesos y funciones del sistema. Así como el flujo y las interacciones entre los procesos.

Se utilizó la herramienta BONITASOFT para crear los siguientes diagramas:

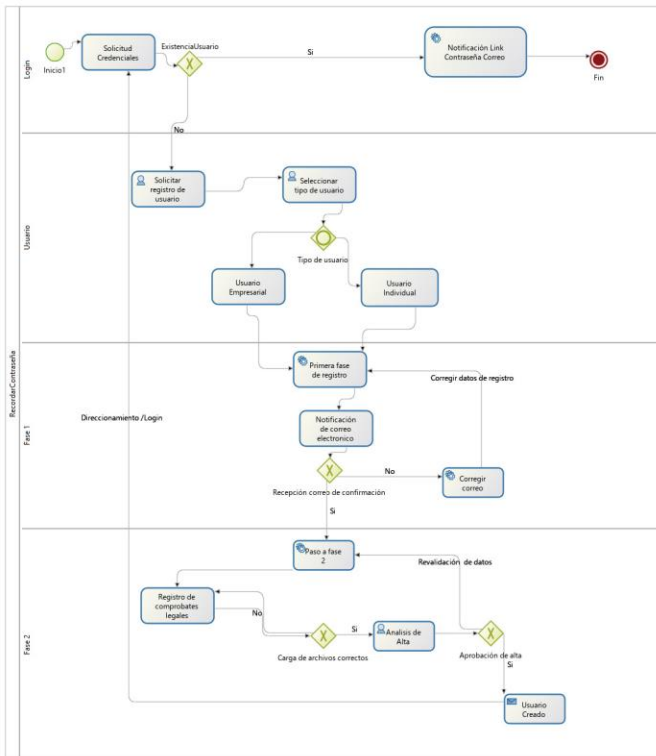
➤ Diagrama de oferta en subasta:



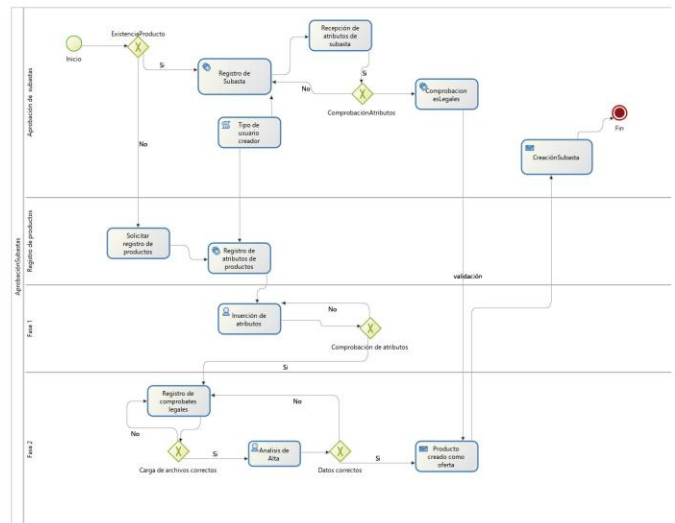
➤ Diagrama de login:



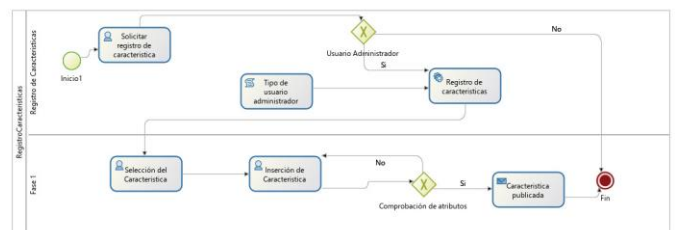
➤ Diagrama de recordar contraseña:



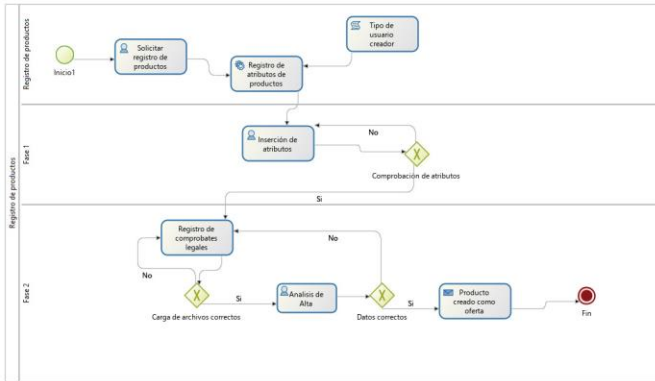
➤ Diagrama de aprobación de una subasta:



➤ Diagrama de registro de características:



➤ Diagrama de registro de productos:



XII. ANEXOS

A. CRONOGRAMA DE ACTIVIDADES

NOMBRE		DURACION	INICIO	TERMINADO	NOMBRES DEL RECURSO
1	Diseño	21,5 días	10/02/20 08:00 AM	10/03/20 01:00 PM	
2	Creación de diseño de casos de uso.	9 días	10/02/20 08:00 AM	20/02/20 05:00 PM	Julián Rojas Daza
3	Creación de diseño MARVEL APP para front de presentación de productos.	1 día	10/02/20 05:00 PM	11/02/20 05:00 PM	Nicolas Moreno Roa
4	Creación de diseño MARVEL APP para front de registro de productos.	1 día	11/02/20 05:00 PM	12/02/20 05:00 PM	Nicolas Moreno Roa
5	Creación de diseño MARVEL APP para front de modificación de productos	1 día	12/02/20 05:00 PM	13/02/20 05:00 PM	Nicolas Moreno Roa
6	Creación de diseño MARVEL APP para front de registro de usuarios	1 día	13/02/20 05:00 PM	14/02/20 05:00 PM	Nicolas Moreno Roa
7	Creación de diseño MARVEL APP para front de gestión de pagos	1 día	14/02/20 05:00 PM	17/02/20 05:00 PM	Nicolas Moreno Roa
8	Creación de diseño de estructura relacional de base de datos	2 días	21/02/20 08:00 AM	24/02/20 05:00 PM	Julián Bermúdez Franco
9	Definición de estructura de seguridad de datos - Transaccional.	7 días	25/02/20 08:00 AM	4/03/20 05:00 PM	Julián Bermúdez Franco
10	Aproximación a norma PCI-DSS.	1 día	20/02/20 01:00 PM	21/02/20 01:00 PM	Andrés Miranda Díaz
11	Definición de arquitectura de infraestructura de aplicación	0,5 días	5/03/20 08:00 AM	5/03/20 01:00 PM	Andrés Miranda Díaz
12	Definición de estructura de webServer (Tipo de servidor, puertos, certificados, etc).	0,5 días	5/03/20 01:00 PM	5/03/20 05:00 PM	Andrés Miranda Díaz
13	Definición de estructura de Frontend (Responsive, ad-works, etc).	0,5 días	6/03/20 08:00 AM	6/03/20 01:00 PM	Julián Rojas Daza
14	Definición de estructura de Backend (Lógica de negocio, caos de uso, seguridad, etc).	0,5 días	6/03/20 01:00 PM	6/03/20 05:00 PM	Julián Bermúdez Franco
15	Definición de estructura de manipulación de la información.	0,5 días	9/03/20 08:00 AM	9/03/20 01:00 PM	Julián Bermúdez Franco
16	Definición de almacenamiento de información.	1 día	9/03/20 01:00 PM	10/03/20 01:00 PM	Julián Rojas Daza
17	Definición de consumo de información.	1 día	26/02/20 05:00 PM	27/02/20 05:00 PM	Julián Rojas Daza
18	Definición de respaldo de la información.	1 día	27/02/20 05:00 PM	28/02/20 05:00 PM	Andrés Miranda Díaz
19	Implementación	17,625 días	3/03/20 05:00 PM	27/03/20 02:00 PM	
20	Creación de base de datos relacional.	1 día	10/03/20 01:00 PM	11/03/20 01:00 PM	Julián Bermúdez Franco
21	Implementación de casos de uso.	1 día	11/03/20 01:00 PM	12/03/20 01:00 PM	Julián Bermúdez Franco
22	Compra y adquisición de servicio de cloud.	0,25 días	3/03/20 05:00 PM	4/03/20 10:00 AM	Andrés Miranda Díaz
23	Despliegue de contenedor de base de datos mysql.	0,125 días	12/03/20 01:00 PM	12/03/20 02:00 PM	Julián Bermúdez Franco
24	Despliegue de contenedor de web server.	0,125 días	12/03/20 02:00 PM	12/03/20 03:00 PM	Nicolas Moreno Roa
25	Despliegue de contenedor para webservices.	0,125 días	12/03/20 03:00 PM	12/03/20 04:00 PM	Nicolas Moreno Roa
26	Creación de elementos de lógica de negocio Backend.	2 días	4/03/20 02:00 PM	6/03/20 02:00 PM	Julián Bermúdez Franco
27	Creación de elementos de lógica de negocio frontend.	2 días	12/03/20 04:00 PM	16/03/20 04:00 PM	Nicolas Moreno Roa
28	Creación de front de presentación de productos.	1 día	10/03/20 02:00 PM	11/03/20 02:00 PM	Julián Rojas Daza
29	Creación de front de registro de productos.	1 día	11/03/20 02:00 PM	12/03/20 02:00 PM	Julián Rojas Daza
30	Creación de front de modificación de productos	1 día	12/03/20 02:00 PM	13/03/20 02:00 PM	Julián Rojas Daza
31	Creación de front de registro de	1 día	13/03/20 02:00 PM	16/03/20 02:00 PM	Julián Rojas Daza

	usuarios				
32	Creación de front de gestión de pagos	1 día	<u>16/03/20 02:00 PM</u>	<u>17/03/20 02:00 PM</u>	Julián Rojas Daza
33	Integración con pasarela de pagos.	2 días	<u>17/03/20 02:00 PM</u>	<u>19/03/20 02:00 PM</u>	Andrés Miranda Díaz
34	Implementación de procesos de seguridad	2 días	<u>19/03/20 02:00 PM</u>	<u>23/03/20 02:00 PM</u>	Andrés Miranda Díaz
35	Implementación de procesos de almacenamiento.	1 día	<u>23/03/20 02:00 PM</u>	<u>24/03/20 02:00 PM</u>	Andrés Miranda Díaz
36	Implementación de procesos de respaldo de información.	1 día	<u>24/03/20 02:00 PM</u>	<u>25/03/20 02:00 PM</u>	Julián Bermúdez Franco
37	Procesos de PCI-DSS	2 días	<u>25/03/20 02:00 PM</u>	<u>27/03/20 02:00 PM</u>	Andrés Miranda Díaz
38	Pruebas y Calidad	12,5 días	<u>27/03/20 02:00 PM</u>	<u>15/04/20 09:00 AM</u>	
39	Pruebas funcionales base de datos.	0,5 días	<u>27/03/20 02:00 PM</u>	<u>30/03/20 09:00 AM</u>	Andrés Miranda Díaz
40	Pruebas funcionales servidor web.	0,5 días	<u>30/03/20 09:00 AM</u>	<u>30/03/20 02:00 PM</u>	Andrés Miranda Díaz
41	Pruebas funcionales de webServices	0,5 días	<u>30/03/20 02:00 PM</u>	<u>31/03/20 09:00 AM</u>	Julián Rojas Daza
42	Testing software Casos de uso backend.	2 días	<u>31/03/20 09:00 AM</u>	<u>2/04/20 09:00 AM</u>	Julián Bermúdez Franco
43	Testing software Casos de uso frontend.	2 días	<u>2/04/20 09:00 AM</u>	<u>6/04/20 09:00 AM</u>	Julián Rojas Daza
44	Pruebas integración con pasarela de pagos.	2 días	<u>6/04/20 09:00 AM</u>	<u>8/04/20 09:00 AM</u>	Andrés Miranda Díaz
45	Pruebas de carga de seguridad.	1 día	<u>8/04/20 09:00 AM</u>	<u>9/04/20 09:00 AM</u>	Nicolas Moreno Roa
46	Pruebas de almacenamiento.	1 día	<u>9/04/20 09:00 AM</u>	<u>10/04/20 09:00 AM</u>	Nicolas Moreno Roa
47	Pruebas de respaldo de información.	1 día	<u>10/04/20 09:00 AM</u>	<u>13/04/20 09:00 AM</u>	Julián Bermúdez Franco
48	Simple compliance PCI-DSS.	2 días	<u>13/04/20 09:00 AM</u>	<u>15/04/20 09:00 AM</u>	Andrés Miranda Díaz