



idat

**INSTITUTO DE EDUCACIÓN SUPERIOR PRIVADO
IDAT**

**PROGRAMA DE ESTUDIOS EN DESARROLLO DE
SISTEMA DE INFORMACIÓN**

**DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB
PARA LA ADMINISTRACIÓN DE VENTAS, COMPRAS E
INVENTARIO PARA LA BODEGA JESMAR EN EL DISTRITO
DE JOSÉ LEONARDO ORTIZ – CHICLAYO - LAMBAYEQUE
EN EL PERIODO 2022**

**Trabajo de aplicación profesional para obtener el título Profesional Técnico en
Desarrollo de Sistemas de Información**

**JESMAR MEDALY CONCHA TINOCO
(0000-0002-7653-1604)**

**LIMA – PERÚ
2022**

Dedicatoria

*Este proyecto le dedico a Dios que siempre
ilumina y guía mi camino junto a mis dos
abuelos (Juana y José), que velan por mi desde
el cielo.*

*A mis padres por apoyarme y darme las
fuerzas para salir adelante e inculcarme buenos
valores en mi vida.*

Agradecimiento

Agradezco a Dios por iluminar mi camino y darme la oportunidad de realizar mi tesis y por ayudarme a lograr mis objetivos.

A mi Asesor de proyecto de titulación al Ing.

ANDRES EDUARDO BARBA REBAZA por brindarme la orientación y guía para realizar mi tesis y surgir satisfactoriamente.

ÍNDICE

DEDICATORIA.....	2
AGRADECIMIENTO	3
RESUMEN EJECUTIVO	12
INTRODUCCIÓN... ..	13
CAPITULO I: GENERALIDADES DE LA Empresa... ..	14
Descripción de la Empresa	14
Antecedentes de la Empresa.....	14
Visión Empresarial	15
Misión Empresarial	16
Análisis FODA de la Empresa	16
Organización de la Empresa.....	16
Funciones Principales de la Empresa	17
Organigrama de la Empresa	17
Identificación del Problema y Propuesta de Solución	18
Análisis del Problema de la Empresa	18
Alternativa de Solución de la Empresa	19
Objetivos del Proyecto y Alcances.....	19
Objetivo General del Proyecto	19
Objetivo Específico del Proyecto.....	19
Alcance del Proyecto.....	19
Recursos Actuales de la Empresa	20
Recursos Tecnológicos de la Empresa	20
Recursos Humanos de la Empresa	20
Factibilidad del Proyecto a Desarrollar	20
Factibilidad Técnica del Proyecto	20
Factibilidad Operativa del Proyecto	21
Factibilidad Económica del Proyecto.....	21
CAPITULO II: DIAGNOSTICO DEL PLAN DE NEGOCIO	24
Metodología de Implementación.....	24
Enfoque del Marco del Proyecto	24

Arquitectura del Software	24
¿Qué es un MVC?	24
Administrador del Proyecto	25
Recursos Humanos del Proyecto	25
Matriz de Riesgos	26
Diagrama de Gantt del Proyecto	26
CAPITULO III: DESARROLLO DEL PROYECTO	27
Fase Inicio	27
Visión del proyecto.....	27
Presentación de Proyecto (Kick off)	27
Historias de Usuario	29
Product Backlog	33
Fase de Planificación	35
Historias de Usuario Priorizado	35
Estimación de las Tareas de las Historias de Usuarios.....	36
Estructura de Desglose de Trabajo (EDT)	38
Product Backlog Priorizado y Estimado	38
Sprint Planning.....	40
Descomposición de Tareas (Plantilla Scrum)	43
Fase Implementación	41
Incremento Sprint Backlog...	44
Mantenimiento del Product Backlog.....	45
Daily Scrum Meeting...	45
Fase Revisión Y Retrospectiva	47
Scrum Board.....	47
Validación de los Sprint. Sprint Review	48
Burn Down Chart	48
Creación de la Lista de Impediment Log	51
Retrospectiva del Sprint	52
Pruebas de Funcionalidad. Pruebas de Aceptación. Pruebas de Rendimiento	53

Fase Lanzamiento	54
Actas de Conformidad de los Entregables	54
CAPÍTULO IV: PROGRAMACIÓN...	55
Implementación de la Arquitectura de Software.....	55
Creación de la Base de Datos y Generación del Script	55
Implementación de Librerías y Dependencias	56
Codificación del Front End y Back End.....	56
Listado de Consultas	57
Listado de Reportes.....	58
Procesos Transaccionales.....	58
Listado de Mantenimientos	59
CAPÍTULO V: PRUEBAS DE CALIDAD DEL SOFTWARE	60
Pruebas Unitarias	60
Pruebas Integrales	61
CONCLUSIONES	62
RECOMENDACIONES	62
REFERENCIAS BIBLIOGRÁFICAS	63
ANEXOS	64

ÍNDICE DE TABLAS

TABLA 01: RAZÓN SOCIAL.....	17
TABLA 02: RECURSOS TECNOLÓGICOS PARA EL PROYECTO	20
TABLA 03: RECURSOS HUMANOS DE LA EMPRESA.....	20
TABLA 04: RECURSOS TÉCNICOS PARA EL DESARROLLO DEL PROYECTO	20
TABLA 05: FACTIBILIDAD ECONÓMICA	22
TABLA 06: FLUJO DE PAGO DEL PROYECTO	23
TABLA 07: RECURSOS HUMANOS DEL PROYECTO	25
TABLA 08: HISTORIAS DE USUARIO.....	29
TABLA 09: PRODUCT BACKLOG.....	33
TABLA 10: HISTORIAS DE USUARIOS PRIORIZADOS	34
TABLA 11: ESTIMACIÓN DE LAS TAREAS DE LAS HISTORIAS DE USUARIOS	36
TABLA 12: PRODUCT BACKLOG PRIORIZADO Y ESTIMADO.....	38
TABLA 13: SPRING PLANNING.....	39
TABLA 14: IMPEDIMENT LOG	51
TABLA 15: DATOS DE LE EMPRESA EJECUTORA Y NOMBRE DEL PROYECTO	52
TABLA 16: DATOS DE LA REUNIÓN	52
TABLA 17: FORMATO DE REUNIÓN DE RETROSPECTIVA	52

ÍNDICE DE FIGURAS

FIGURA 01: UBICACIÓN DE LA EMPRESA.....	14
FIGURA 02: IMÁGENES DE LA BODEGA JESMAR.....	15
FIGURA 03: ANÁLISIS FODA DE LA EMPRESA.....	16
FIGURA 04: ORGANIGRAMA DE LA EMPRESA	17
FIGURA 04: FLUJO DE CAJA PROYECTADO	23
FIGURA 05: MARCO DE TRABAJO SCRUM	24
FIGURA 06: MODELO -VISTA -CONTROLADOR	25
FIGURA 07: MATRIZ DE RIESGO	26
FIGURA 08: DIAGRAMA DE GANTT	26
FIGURA 09: PRESENTACIÓN DE PROYECTO (KICK OFF).....	27
FIGURA 10: CONTENIDO DEL PROYECTO (KICK OFF).	27
FIGURA 11: ANTECEDENTES DE LA EMPRESA (KICK OFF).....	27
FIGURA 12: VISIÓN Y MISIÓN DE LA EMPRESA (KICK OFF).	27
FIGURA 13: ESTRUCTURA DE DESGLOSE DE TRABAJO.....	37
FIGURA 14: DESCOMPOSICIÓN DE TAREAS (SPRING 1).....	41
FIGURA 15: DESCOMPOSICIÓN DE TAREAS (SPRING 2).....	41
FIGURA 16: DESCOMPOSICIÓN DE TAREAS (SPRING 3).....	42
FIGURA 17: DESCOMPOSICIÓN DE TAREAS (SPRING 4).....	42
FIGURA 18: DESCOMPOSICIÓN DE TAREAS (SPRING 5).....	43
FIGURA 19: DESCOMPOSICIÓN DE TAREAS (SPRING 6).....	43

FIGURA 20: SPRINT BACKLOG IDENTIFICADO	44
FIGURA 21: SPRINT BACKLOG EN PROCESO Y RESUELTO	44
FIGURA 22: SPRINT BACKLOG RESUELTO	44
FIGURA 23: MANTENIMIENTO DE PRODUCT BACKLOG	45
FIGURA 24: DAILY SCRUM MEETING (SPRINT 01).....	45
FIGURA 25: DAILY SCRUM MEETING (SPRINT 02	46
FIGURA 26: DAILY SCRUM MEETING (SPRINT 03).....	46
FIGURA 27: DAILY SCRUM MEETING (SPRINT 04	46
FIGURA 28: DAILY SCRUM MEETING (SPRINT 05).....	46
FIGURA 29: DAILY SCRUM MEETING (SPRINT 06).....	46
FIGURA 30: SCRUM BOARD	47
FIGURA 30: SPRINT REVIEW	48
FIGURA 31: SCRUM BOARD (SPRINT 01).....	48
FIGURA 32: SCRUM BOARD (SPRINT 02).....	49
FIGURA 33: SCRUM BOARD (SPRINT 03).....	49
FIGURA 34: SCRUM BOARD (SPRINT 04).....	50
FIGURA 35: SCRUM BOARD (SPRINT 05).....	50
FIGURA 36: SCRUM BOARD (SPRINT 06).....	51
FIGURA 37: PRUEBA DE FUNCIONALIDAD DE VENTAS	51
FIGURA 38: PRUEBAS DE FUNCIONALIDAD DE REPORTE	51
FIGURA 39: ACTA DE ACEPTACIÓN DEL PROYECTO.....	54

FIGURA 40: IMPLEMENTACIÓN MODELO -VISTA - CONTROLADOR	55
FIGURA 41: DIAGRAMA DE LA BASE DE DATOS	55
FIGURA 42: LIBRARÍAS Y DEPENDENCIAS DEL SOFTWARE.....	56
FIGURA 43: BACK END DE PROVEEDOR	56
FIGURA 44: FRONT END DE PROVEEDOR	57
FIGURA 45: LISTADO DE CONSULTAS DE PRODUCTOS	57
FIGURA 46: LISTADO DE REPORTES	58
FIGURA 47: PROCESOS TRANSACCIONALES (FRONT END)	58
FIGURA 48: PROCESOS TRANSACCIONALES (BACK END)	59
FIGURA 49: LISTADO DE MANTENIMIENTO (BACK END)	59
FIGURA 50: PRUEBAS UNITARIAS DE VENTAS	60
FIGURA 51: PRUEBAS UNITARIAS DE COMPRAS.....	60
FIGURA 52: PRUEBAS INTEGRALES DE REPORTES DE COMPRAS Y VENTAS SEGUN LA FECHA.	61
FIGURA 53: PRUEBAS INTEGRALES DE INVENTARIO	61
FIGURA 54: CÓDIGO DE LA TABLA PRIVILEGIO	64
FIGURA 55: CÓDIGO DE LA TABLA DE USUARIO	64
FIGURA 56: CÓDIGO DE PROCEDIMIENTOS.....	65
FIGURA 57: CÓDIGO DE DATA PARA OBTENER E INSERTAR INFORMACIÓN	65
FIGURA 58: CÓDIGO DE DATA PARA MODIFICAR Y ELIMINAR DATA.....	66
FIGURA 59: CÓDIGO DE LA TABLA PROVEEDOR	66
FIGURA 60: CÓDIGO DE LA TABLA CATEGORÍA Y SUS PROCEDURES.....	67

FIGURA 61: CÓDIGO DE LA TABLA PRODUCTOS Y SUS PROCEDIMIENTOS.....	67
FIGURA 62: CÓDIGO DE LA TABLA COMPRA Y SUS PROCEDURES	68
FIGURA 63: CÓDIGO DEL PROCEDIMIENTO TOTAL DE COMPRAS.....	68
FIGURA 64: CÓDIGO DEL PROCEDIMIENTO MODIFICAR COMPRAS.....	69
FIGURA 65: CÓDIGO DEL PROCEDIMIENTO ELIMINAR COMPRAS.....	69
FIGURA 66: CÓDIGO DEL PROCEDIMIENTO GENERAR REPORTES DE COMPRAS.....	70
FIGURA 67: CÓDIGO DE LA VENTA Y SUS PROCEDIMIENTOS	70
FIGURA 68: CÓDIGO DE PROCEDIMIENTOS DE GENERAR REPORTE DE VENTAS.....	71
FIGURA 69: CÓDIGO DE PROCEDIMIENTOS DE TOTAL DE REPORTE DE VENTAS.....	71
FIGURA 70: CÓDIGO DE PROCEDIMIENTOS DE MODIFICAR VENTAS	72
FIGURA 71: CÓDIGO DE PROCEDIMIENTOS DE ELIMINAR VENTA.....	72
FIGURA 72: CÓDIGO DE TABLA DE INVENTARIO	73
FIGURA 73: CÓDIGO DE PROCEDIMIENTO OBTENER EL TOTAL DE LAS COMPRAS Y VENTAS	73
FIGURA 74: CÓDIGO DE TRIGGERS COMPRAS	74
FIGURA 75: CÓDIGO DE TRIGGERS ACTUALIZAR COMPRAS	74
FIGURA 76: CÓDIGO DE TRIGGERS INSERTAR DATOS VENTAS	75
FIGURA 77: CÓDIGO DE TRIGGERS ACTUALIZAR Y ELIMINAR VENTAS	75

RESUMEN EJECUTIVO

Este trabajo se realizó para poder controlar los egresos e ingresos de la bodega Jesmar y realizar un inventario de productos para poder controlar el stock de productos. La mayoría de las microempresas cuentan con un conocimiento básico en realizar un inventario y organizarlo por medio de un sistema, es por ello que realizan sus anotaciones en cuadernos, notas, proformas; generando un desorden administrativo ya que requiere de mucho tiempo para organizar un inventario por lo que no registran sus compras y sus ventas, por eso no saben con exactitud cuánto fue las ventas que realizo en un día, cuanto fue su ganancia, sus pérdidas o que producto le falta para satisfacer a sus clientes y también por la falta de asesoramiento en la parte administrativa. El objetivo de este trabajo es documentar el análisis y automatizar los procesos administrativos de la Bodega Jesmar. La solución que realice para la problemática de este proyecto es: un sistema web para controlar el stock de productos y obtener un balance de los egresos e ingresos por día, mes, año. Y así llevar una buena contabilidad ya sea en la bodega Jesmar o en diferentes tipos de microempresas.

Por ello que mi proyecto brindará la satisfacción correspondiente y necesaria para el administrador ya que se va estructurar una data de productos según la información brindada por la microempresa, donde la persona encargada del sistema puede acceder a un login e ingresar mediante su número de DNI y su contraseña para acceder a los módulos de usuario, inventario y administración, en el módulo inventario en las opciones de reportes se obtendrá un control exacto de egresos e ingresos de acuerdo a su fecha correspondiente y también se controlara el stock de productos para una buena contabilidad y calidad administrativa, cabe mencionar que la información será dada en tiempo real y generara los reportes diarios.

INTRODUCCIÓN

En el presente proyecto se propone un sistema web para la administración de ventas, compras e inventario para una empresa dedicada a la venta de Abarrotes y artículos de primera necesidad. El objetivo es mejorar el flujo de información mediante el administrador y el empleado de la empresa para que puedan interactuar con el sistema web. En el capítulo I hablamos todo acerca de la descripción de la empresa, identificar el problema que presenta la empresa y la solución que se brindara con el proyecto, también hablaremos acerca de los objetivos y alcances de la empresa, los recursos de la empresa y la factibilidad que brinda el proyecto. En el Capítulo II hablamos acerca del diagnóstico del plan de negocio, la arquitectura y la administración del proyecto. En el capítulo III hablamos acerca del desarrollo del proyecto que está dividida en cinco fases: Fase de inicio, Fase de Planificación, fase de Implementación, fase de revisión y retrospectiva, por último, la fase de lanzamiento. En el capítulo IV están las pruebas de calidad del Software que se dividen en dos pruebas: pruebas Unitarias y pruebas integrales. Por último, tenemos las conclusiones, recomendaciones, sus referencias bibliográficas y sus Anexos del proyecto. Los alcances que posee este proyecto es brindarles una buena información con respecto a sus reportes de sus egresos e ingresos que realiza la empresa. Y así llevar una buena contabilidad por el administrador.

En el año 2016 cambiaron de local por motivo que querían expandir más la tienda de abarrotes, fue más estabilizada y más surtida de productos de buena calidad. Para hacer el inventario se realizó un pequeño inventario en un Excel, para finalizar el día ya no tengas problemas con sus reportes, en el año 2019 comenzó a bajar las ventas por motivo de la pandemia y por qué los proveedores ya no llegaban al lugar de la tienda para realizar los pedidos. Ya no había un manejo completo en el área de compras por que las compras se comenzaron hacer en los mercados más cercanos y los precios de los productos no eran fijos, un día encontrabas a un precio al siguiente día encontrabas con otro precio y así continua hasta el año 2022. Este proyecto busca sistematizar el control de compras y ventas de los productos mediante un sistema web Administrativo.

Figura 02: Imágenes de la Bodega Jesmar



Visión Empresarial.

Buscamos ser la primera opción de nuestros clientes al momento de adquirir los productos; permitiéndonos crear relaciones duraderas basadas en la confianza, el

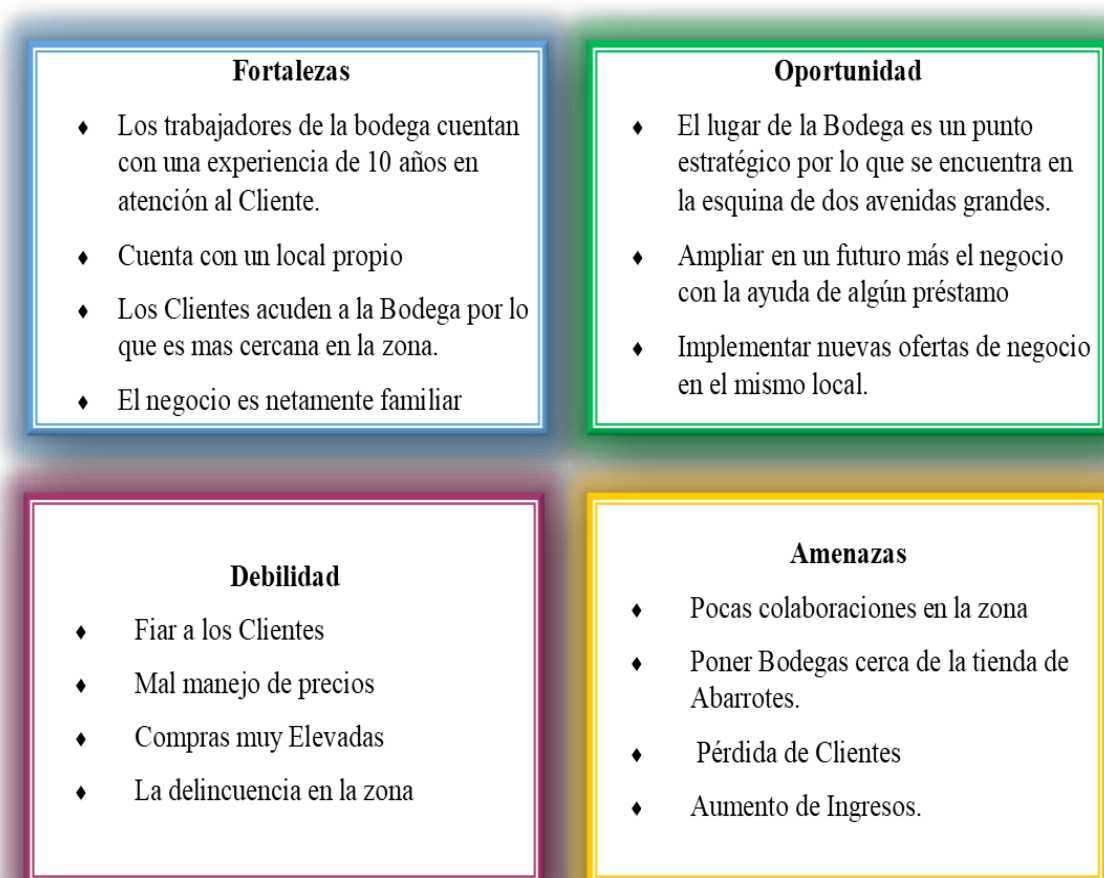
respeto y la honestidad que nos permita crecer todos los días con la ayuda de nuestros clientes.

Misión Empresarial.

Somos una Bodega que se encargada de brindar y ofrecer productos de excelente calidad; a través de un buen servicio, el mejor precio y un trato amable por parte de quienes les brinda la atención al cliente.

Análisis FODA de la Empresa.

Figura 03: Análisis FODA de la Empresa



Organización de la Empresa.

Tabla 01: Razón Social

RUC:	10277170359
RAZON SOCIAL:	AA.HH. Javier Castro Cruz N°01 Mz. D- Lt. 01 - José Leonardo Ortiz - Chiclayo - Lambayeque
NOMBRE COMERCIAL:	Bodega “JESMAR”
TIPO DE EMPRESA:	Microempresa
INICIO DE ACTIVIDADES:	01/04/2016
CIU V4:	4630
CIU V3:	51225
ACTIVIDADES COMERCIALES:	Venta Al Por Mayor De Alimentos, Bebidas Y Tabaco

Funciones De Las Principales de la Empresa.

Área De Recepción: Se encarga de registrar, procesar y administrar todas las transacciones financieras de la Bodega.

Área De Almacenamiento: Se encarga de controlar las entradas y salidas de los productos para llevar un buen control de productos.

Área Administrativa: Se encarga de administrar el negocio y analizar nuevas ofertas que se presenta en el mercado para una buena distribución en la Bodega.

Organigrama de la Empresa.

Figura 04: Organigrama de la Empresa



Identificación Del Problema Y Propuesta De Solución.

Análisis Del Problema de la Empresa.

Antecedentes.

La BODEGA JESMAR es una tienda de abarrotes, el cual su proceso de Ventas y Compra lo hacen solo registrando los productos en una hoja de Cuaderno, blocks, hoja bond, lo cual trae problemas, en la atención a los clientes y en las compras a los proveedores. Su local se encuentra en José Leonardo Ortiz y piensan expandirse por lo cual quieren automatizar la Gestión de Venta y Compras. Cuando se genera una venta el encargado en este caso el cajero tiene que realizar el registro manualmente, generando una incomodidad del usuario por el tiempo que tarda en realizar el proceso. Además, debido de la contingencia el aforo del local es reducido y los clientes deben realizar grandes colas. Existen también problemas en el Área de Venta, Administración debido el stock de los productos muchas veces no están actualizados correctamente, ocasionando problemas al área de ventas. Y en administración no tiene el inventario con precios precisos por motivo que no cuentan con un sistema.

Situación de Problemática de la Empresa.

- En el área de almacén no llevan un buen control de stock, no saben con exactitud que producto tiene poco stock.
- Su proceso de Ventas y Compras lo hacen solo registrando los productos en una hoja de Cuaderno, blocks, hoja bond. Por lo cual no pueden realizar un buen registro porque solo anotan lo que recuerdan que han vendido.

Alternativas De Solución de la Empresa.

La solución que realice para la problemática de este proyecto es: un sistema web para controlar el stock de productos y obtener un balance de los egresos e ingresos por día, mes, año. Y así llevar una buena contabilidad ya sea en a la bodega Jesmar o en diferentes tipos de microempresas.

Objetivos Del Proyecto Y Alcances.

Objetivo General del Proyecto.

Facilitar los procesos que realiza el usuario en el área de compras y ventas e inventario de la BODEGA JESMAR para tener una buena administración.

Objetivo Específico del Proyecto.

- Analizar la funcionalidad actual del proceso de compra y ventas de la empresa.
- Desarrollar interfaces para registro de personal, manejo de perfiles administrativos, registro de venta, registro de compra, registro de proveedores, registro de productos.
- Reducir el tiempo de los procesos que realiza el área usuaria y evitar contratiempos, ya que esta estará almacenada en una base de datos confiable que permitirá gestión la información almacenada.

Alcance Del Proyecto.

Desarrollar e implementar un sistema web para la administración de ventas, compras e inventario de la bodega JESMAR.

En el sistema web incluye los siguientes módulos:

- ***Módulo 1:*** Módulo de Usuario
- ***Módulo 2:*** Módulo Administrativo
- ***Módulo 3:*** Módulo de Inventario
- ***Módulo 4:*** Módulo de Reportes

En el sistema no incluye las siguientes funciones

- Módulo para los clientes de la bodega.
- Carrito de compras para el cliente de la bodega.

Recursos Actuales de la Empresa

Recursos Tecnológicos de la Empresa.

Tabla 02: Recursos Tecnológicos para el proyecto

HARDWARE	SOFTWARE	CANTIDAD
COMPUTADORA	Procesador: AMD A8 -7600 Radeon R7 RAM:4GB	1

En conclusión, según los recursos técnicos la empresa posee los recursos requeridos para poder desarrollar el proyecto y así no presentar ningún problema para implementarlo.

Recursos Humanos de la Empresa.

Tabla 03: Recursos Humanos de la empresa

AREA	CANTIDAD
Administrador	1
Cajero	1
Almacén	1

En conclusión, en la bodega solo contamos con tres personas que se encargan de administrarla ya que es netamente familiar.

Factibilidad Del Proyecto A Desarrollar

Factibilidad Técnica del Proyecto.

Tabla 04: Recursos Técnicos para el desarrollo del proyecto

Tipo de Recursos	Nombre de Recursos	Descripción	Cantidad
Recursos Humanos	Área de Desarrollo	Analista	1
		Diseñador de Base de datos y programador	1
		Diseñador	1
Hardware	Laptop	Procesador: Intel(R) Core (TM) i5 8250 U	1
		RAM: 8GB	
Software	Visual Studio Code	Interfaz de desarrollo para codificar.	1
	JDK 8	Kit de desarrollo para programar en Java.	1
	XAMPP	Software, encargado de la gestión de base de datos MySQL.	1
	MySQL	Gestor de base de datos.	1
	Windows 10 pro	Sistema Operativo	1
	Microsoft office	Herramienta para aplicación de la ofimática	1

En conclusión, si fue factible desarrollar el sistema web, porque contábamos con buenas herramientas, Además cuento con la experiencia necesarias para realizar el proyecto y cumplir con la fecha programada con el dueño de la bodega.

Factibilidad Operativa del Proyecto.

Debido a que se conocen los procesos de negocio, se tiene acceso a la información de la empresa, a sus funciones que realizan cada trabajador en cada área de la empresa.

También conto con un presupuesto económico que fue asignado para el proyecto.

En conclusión, si hubo una factibilidad operativa porque contamos con buenas aportaciones de información por parte de la empresa y porque el proyecto si brinda la solución a la problemática que se presentó en la empresa.

Factibilidad Económica del Proyecto.

Tabla 05: Factibilidad Económica

Costos						
Recursos Humanos						
Descripción	Horas (A)	Costo x Hora(B)	Costo Unitario (C=A*B)	Meses (D)	Total (C*D)	
(1) Analista	120	S/ 20	S/ 2 400	4	S/ 9 600	
(1) Programador	120	S/ 15	S/ 1 800	4	S/ 7 200	
(1) Diseñador	120	S/ 15	S/ 1 800	4	S/ 7 200	
				Total	S/ 24 000	
Recursos Tecnológicos						
Tipo de Recursos	Descripción	Horas (A)	Costo x Hora(B)	Costo Unitario (C=A*B)	Meses (D)	Total (C*D)
Hardware	Laptop	120	S/ 1,00	S/ 120	4	S/ 480
				Total	S/ 480	
Tipo de Recursos	Descripción			Costo Unitario (C)	Meses (D)	Total (C*D)
Software	Visual Studio Code (*)			S/ 0	4	S/ 0
	JDK 8			S/ 120	4	S/ 480
	XAMPP (*)			S/ 0	4	S/ 0
	MySQL (*)			S/ 0	4	S/ 0
	Windows 10 pro (*)			S/ 0	4	S/ 0
	Microsoft office (*)			S/ 0	4	S/ 0
				Total	S/ 480	

Nota: En la siguiente información los (*) pertenecen a los softwares que están sin costo.

- (*) => Los softwares cuentan con licencia gratuita por un año

Tabla 06: flujo de pago del proyecto

Flujos de Pago	
Recursos	Costos
Recursos Humanos	S/ 24 000
Recursos Tecnológicos	S/. 960
Total	S/. 24 960

Análisis Costo Beneficio del Proyecto.

A continuación, se realizó el flujo de caja proyectado en un año.

Figura 04: Flujo de caja proyectado

FLUJO DE CAJA PROYECTADO													
PROYECTO:	Sistema de compra y venta e												
COSTO DE OPORTUNIDAD MENSUAL DEL CAPITAL(COK):	12%												
#MES:		1	2	3	4	5	6	7	8	9	10	11	12
MES:	INVERSION DEL PROY.	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SET	OCT	NOV	DIC
INGRESOS (A):		S/ 15.000	S/ 18.000	S/ 25.000	S/ 30.000	S/ 35.000	S/ 40.000	S/ 42.000	S/ 40.000	S/ 48.000	S/ 50.000	S/ 45.000	S/ 55.000
EGRESOS (B):	-S/ 24.960	S/ 7.600	S/ 10.000	S/ 14.000	S/ 18.000	S/ 20.000	S/ 22.000	S/ 28.000	S/ 22.000	S/ 25.000	S/ 22.000	S/ 25.000	S/ 30.000
FLUJO DE CAJA C=(A-B)	-S/ 24.960	S/ 7.400	S/ 8.000	S/ 11.000	S/ 12.000	S/ 15.000	S/ 18.000	S/ 14.000	S/ 18.000	S/ 23.000	S/ 28.000	S/ 20.000	S/ 25.000
VALOR ACTUAL E=(C/(1+D)^#MES):		S/ 6.607,14	S/ 6.377,55	S/ 7.829,58	S/ 7.626,22	S/ 8.511,40	S/ 9.119,36	S/ 6.332,89	S/ 7.269,90	S/ 8.294,03	S/ 9.015,25	S/ 5.749,52	S/ 6.416,88
VALOR ACTUAL NETO (VAN):	S/64.190	--> Representa la ganancia adicional de ejecutar este Proyecto.											
TASA INTERNA DE RETORNO (TIR)	43%												

En conclusión, el Valor Actual neto que nos brindara este proyecto es de S/. 64 190 soles que es la ganancia adicional que se dará al ejecutar este proyecto ya que este proyecto si es rentable porque tenemos un 43% de TIR mayor a un COK que está a un 12% de inversión.

CAPÍTULO II

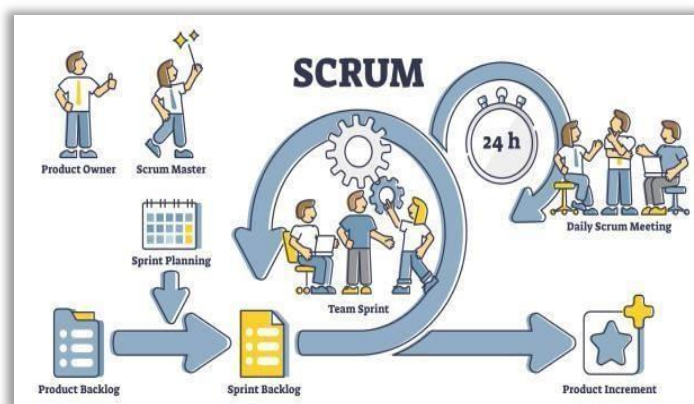
(DIAGNÓSTICO DEL PLAN DE NEGOCIO)

Metodologías de implementación.

Enfoque del Marco de proyecto.

Se aplicó scrum, porque es un marco de trabajo para el desarrollo ágil del proyecto y está expuesto a cambios y revisiones que se realizan por intermedio del cliente y el desarrollador, ya que es un conjunto específico de reglas que se deben seguir mientras se va desarrollando el proyecto. según Ken Schwaber y Jeff Sutherland (1995).

Figura 05: Marco de trabajo Scrum



Arquitectura De Software.

¿Qué es un MVC?

se aplicó (MVC) modelo - vista - controlador que es un patrón de arquitectura de software, que separa los datos y la lógica de un sistema web.

- **En el modelo** se encarga de la base de datos realizando consultas, búsquedas, actualizaciones y eliminaciones de datos.
- **En el controlador** se encarga de recibir órdenes, solicitar datos al modelo y comunicar a la vista.

- **En las vistas** se encarga la representación visual de los datos y todo lo que tenga que ver con la interfaz gráfica del usuario.

Modelo vista controlador (MVC). Servicio de Informática ASP.NET MVC 3 Framework. (s. f.-b).

Universitat d'Alacant Universidad de Alicante. Recuperado 11 de julio de 2022, de

<https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

Figura 06: Modelo -Vista -Controlador



Administración De Proyecto.

Recursos Humanos del Proyecto.

Tabla 07: Recursos Humanos del proyecto

Roles	Funciones	Encargada
Analista	Analizar los procesos administrativos de la empresa	Jesmar Medaly Concha Tinoco
Diseñadora	Diseñar la interfaz gráfica para el usuario	Jesmar Medaly Concha Tinoco
Desarrolladora	Implementar y Desarrollar todas las funciones del sistema web	Jesmar Medaly Concha Tinoco

Matriz de Riesgos.

Figura 07: Matriz de Riesgo

	Ponderado	Probabilidad	Impacto	P x I	Resultado
Alcance	36%	5	3	15	5,4
Tiempo	18%	5	4	20	3,6
Costo	9%	5	2	10	0,9
Calidad	36%	5	1	5	1,8
			Calificación de riesgo		11,7

Diagrama de Gantt del Proyecto.

Figura 08: Diagrama de Gantt

Nº	Actividades	Abril				Mayo				Junio				Julio				Días	Hora Inicio	Hora Fin
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4			
1	Planteamiento de Negocio	■															8	04/04/2022	11/04/2022	
2	Análisis del Problema	■															5	12/04/2022	16/04/2022	
3	Solución del Problema		■														8	17/04/2022	24/04/2022	
4	Elaboración de Historias de Usuario			■													6	25/04/2022	30/04/2022	
5	Planificación de Sprints				■												9	01/05/2022	09/05/2022	
6	Validaciones de Sprint					■											6	10/05/2022	15/05/2022	
7	Elaboración de Entregables						■										8	16/05/2022	23/05/2022	
8	Desarrollo de Software							■									13	24/05/2022	05/06/2022	
9	Implementación de Software								■								15	06/06/2022	20/06/2022	
10	Pruebas Funcionales									■							12	21/06/2022	01/07/2022	
11	Retroalimentación y ajustes del software										■						14	02/07/2022	15/07/2022	
12	Desarrollo de Materiales para sustentación											■					6	16/07/2022	21/07/2022	
13	Entrega del Proyecto													■			8	22/07/2022	29/07/2022	
14	Sustentación del proyecto															■	8	30/07/2022	06/08/2022	

CAPÍTULO III (DESARROLLO DEL PROYECTO)

Fase Inicio.

Visión del Proyecto.

Desarrollar e Implementar un sistema web para la administración de ventas, compras e inventario para la Bodega Jesmar.

Presentación del Lanzamiento del Proyecto (Kick Off).

Figura 09: Presentación de proyecto (Kick Off).

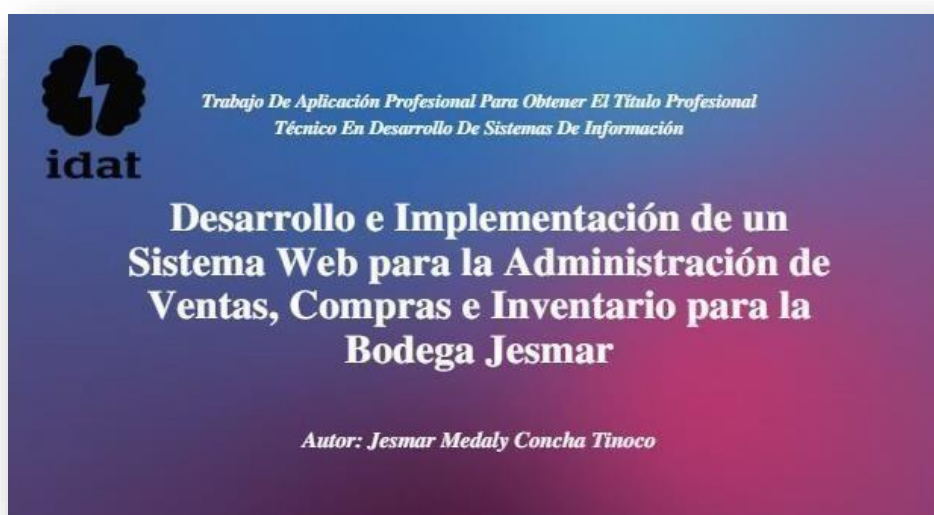


Figura 10: Contenido del proyecto (Kick Off).



Figura 11: Antecedentes de la empresa (Kick Off).



Figura 12: Visión y Misión de la empresa (Kick Off).

Visión Empresarial

Buscamos ser la primera opción de nuestros clientes al momento de adquirir los productos; permitiéndonos crear relaciones duraderas basadas en la confianza, el respeto y la honestidad que nos permita crecer todos los días con la ayuda de nuestros clientes.

Misión Empresarial.

Somos una Bodega que se encargada de brindar y ofrecer productos de excelente calidad; a través de un buen servicio, el mejor precio y un trato amable por parte de quienes les brinda la atención al cliente.

Historias de Usuarios.

Tabla 08: Historias de Usuario

<i>HU</i>	TITULO	DESCRIPCION
<i>HU01</i>	Registro de usuario	Como Administrador quiero registrar a los usuarios para asignarle un privilegio a cada usuario registrado de acuerdo al área que va trabajar.
<i>HU02</i>	Inicio de Sesión	Como Administrador quiero iniciar sesión para acceder al sistema ingresando el DNI del usuario y su contraseña.
<i>HU03</i>	Listar Usuario	Como Administrador quiero tener un listado de Usuarios para llevar el control de los usuarios.
<i>HU04</i>	Actualizar usuario	Como Administrador quiero Actualizar algunos datos del usuario para que tengas sus datos Actualizados o cuando deseen cambiar de privilegios.
<i>HU05</i>	Eliminación de usuario	Como Administrador quiero la eliminación de usuarios para quitar a los usuarios que ya no trabajan en la empresa.
<i>HU06</i>	Lista de Proveedores	Como Administrador quiero tener un listado de proveedores para llevar un control de los proveedores.
<i>HU07</i>	Registrar de Proveedores	Como Administrador quiero registrar a los proveedores para llevar un control en las

		compras y saber cuál es el producto de cada proveedor.
<i>HU08</i>	Actualización de Proveedores	Como Administrador quiero Actualizar los proveedores para llevar un control de datos Actualizados.
<i>HU09</i>	Eliminación de Proveedores	Como Administrador quiero Eliminar los proveedores para llevar un control Actualizado por lo que algunos proveedores ya no aportan a la empresa.
<i>HU10</i>	Listar Categoría	Como Administrador quiero Listar las categorías de los productos para poder tener un control de las categorías que hay en la empresa.
<i>HU11</i>	Registrar Categoría	Como Administrador quiero Registrar las categorías para poder llevar el control de las categorías.
<i>HU12</i>	Actualizar Categoría	Como Administrador quiero Actualizar Categoría para poder llevar un control Actualizado.
<i>HU13</i>	Buscar Categoría	Como Administrador quiero Buscar Categoría para que el administrador pueda buscar más rápido una Categoría.
<i>HU14</i>	Eliminar Categoría	Como Administrador quiero Eliminar Categoría para poder llevar un control cuando la categoría ya no tenga Stock o cuando ya no sea utilizado.

<i>HU15</i>	Lista de Producto	Como Administrador quiero un listado de productos para llevar el control de stock de productos y también para realizar las ventas.
<i>HU16</i>	Registro de Producto	Como Administrador quiero registrar los productos para llevar un control de productos.
<i>HU17</i>	Actualizar Producto	Como Administrador quiero Actualizar los productos para llevar un control de datos Actualizados.
<i>HU18</i>	Buscar Producto	Como Administrador de quiero realizar un filtro de productos de acuerdo a los proveedores para obtener una lista por proveedor.
<i>HU19</i>	Eliminar Producto	Como Administrador quiero Eliminar los productos para llevar un control Actualizado por lo que algunos productos están escasos o a veces se cambian de nombre del producto.
<i>HU20</i>	Registro de Compras	Como Administrador quiero registrar las compras para llegar un control de las compras que se realizan por día.
<i>HU21</i>	Lista de Compras	Como Administrador quiero listar las compras para tener un control de las compras según las compras realizadas.
<i>HU22</i>	Buscar por fecha Compras	Como Administrador quiero realizar búsqueda de compras para obtener más rápido las

		compras realizadas de acuerdo a la fecha asignada.
<i>HU23</i>	Eliminar Compras	Como administrador quiero eliminar una compra para que pueda modificar algunas compras mal ingresadas.
<i>HU24</i>	Registro de ventas	Como Administrador quiero registrar las ventas para obtener un reporte de ventas.
<i>HU25</i>	Lista de ventas	Como Administrador quiero listar las ventas para tener un control de ventas según las ventas realizadas.
<i>HU26</i>	Buscar por fecha ventas	Como Administrador quiero realizar búsqueda de ventas para obtener más rápido las ventas realizadas de acuerdo a la fecha asignada.
<i>HU27</i>	Eliminar venta	Como Administrador quiero eliminar la venta para que pueda modificar algunas ventas mal registradas.
<i>HU28</i>	Listado de inventario	Como Administrador quiero visualizar el listado de productos para controlar el stock.
<i>HU29</i>	Listado de Poco Stock	Como Administrador quiero llevar el control de Stock para saber que producto tiene poco Stock.
<i>HU30</i>	Generar Reportes de compra y venta	Como Administrador quiero generar reporte de compra y venta para llevar un control en la caja de la empresa.

Product Backlog.

Tabla 09: Product Backlog

ID	ID_PB	DESCRIPCION	MOSCOW	ESTIMACION
1	PB01	Registro de usuario	M	30
2	PB02	Inicio de Sesión	M	30
3	PB03	Listar Usuario	S	20
4	PB04	Actualizar usuario	M	30
5	PB05	Eliminación de usuario	C	20
6	PB06	Lista de Proveedores	S	40
7	PB07	Registrar de Proveedores	M	50
8	PB08	Actualización de Proveedores	M	20
9	PB09	Eliminación de Proveedores	C	20
10	PB10	Listar Categoría	S	40
11	PB11	Registrar Categoría	M	50
12	PB12	Actualizar Categoría	M	20
13	PB13	Buscar Categoría	S	30
14	PB14	Eliminar Categoría	C	20
15	PB15	Lista de Producto	S	70
16	PB16	Registro de Producto	M	60
17	PB17	Actualizar Producto	M	40
18	PB18	Búsqueda de Producto	S	20

19	PB19	Eliminar Producto	C	20
20	PB20	Registro de Compras	M	80
21	PB21	Lista de Compras	S	50
22	PB22	Búsqueda por fecha Compras	S	60
23	PB23	Eliminar Compras	C	20
24	PB24	Registro de ventas	M	80
25	PB25	Lista de ventas	S	50
26	PB26	Búsqueda por fecha ventas	S	30
27	PB27	Eliminar venta	C	20
28	PB28	Listado de inventario	M	60
29	PB29	Listado de Poco Stock	M	50
30	PB30	Generar Reportes de compra y venta	M	80

Fase Planificación.

Historias de Usuarios Priorizados.

Tabla 10: Historias de Usuarios Priorizados.

ID_PB	DESCRIPCION	MOSCOW
PB01	Registro de usuario	M
PB02	Inicio de Sesión	M
PB03	Listar Usuario	S
PB04	Actualizar usuario	M

PB05	Eliminación de usuario	C
PB06	Lista de Proveedores	S
PB07	Registrar de Proveedores	M
PB08	Actualización de Proveedores	M
PB09	Eliminación de Proveedores	C
PB10	Listar Categoría	S
PB11	Registrar Categoría	M
PB12	Actualizar Categoría	M
PB13	Buscar Categoría	S
PB14	Eliminar Categoría	C
PB15	Lista de Producto	S
PB16	Registro de Producto	M
PB17	Actualizar Producto	M
PB18	Búsqueda de Producto	S
PB19	Eliminar Producto	C
PB20	Registro de Compras	M
PB21	Lista de Compras	S
PB22	Búsqueda por fecha Compras	S
PB23	Eliminar Compras	C
PB24	Registro de ventas	M
PB25	Lista de ventas	S
PB26	Búsqueda por fecha ventas	S
PB27	Eliminar venta	C
PB28	Listado de inventario	M

PB29	Listado de Poco Stock	M
PB30	Generar Reportes de compra y venta	M

Estimación de las Tareas de las Historias de Usuarios.

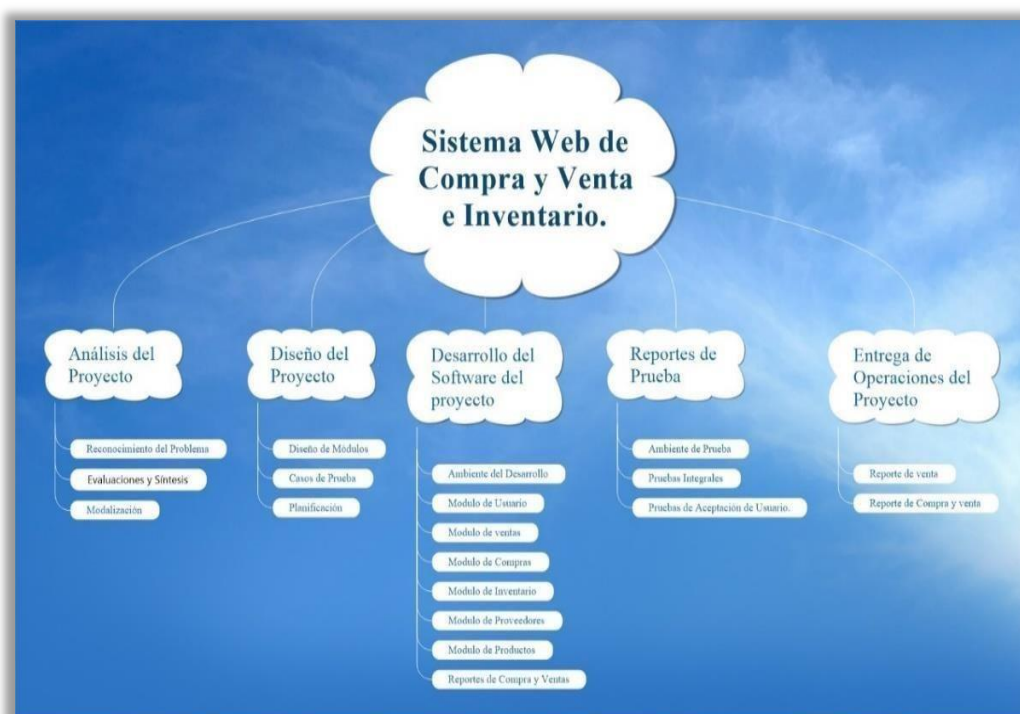
Tabla 11: Estimación de las Tareas de las Historias de Usuarios

ID_PB	DESCRIPCION	ESTIMACION
PB01	Registro de usuario	30
PB02	Inicio de Sesión	30
PB03	Listar Usuario	20
PB04	Actualizar usuario	30
PB05	Eliminación de usuario	20
PB06	Lista de Proveedores	40
PB07	Registrar de Proveedores	50
PB08	Actualización de Proveedores	20
PB09	Eliminación de Proveedores	20
PB10	Listar Categoría	40
PB11	Registrar Categoría	50
PB12	Actualizar Categoría	20
PB13	Buscar Categoría	30
PB14	Eliminar Categoría	20
PB15	Lista de Producto	70
PB16	Registro de Producto	60
PB17	Actualizar Producto	40

PB18	Búsqueda de Producto	20
PB19	Eliminar Producto	20
PB20	Registro de Compras	80
PB21	Lista de Compras	50
PB22	Búsqueda por fecha Compras	60
PB23	Eliminar Compras	20
PB24	Registro de ventas	80
PB25	Lista de ventas	50
PB26	Búsqueda por fecha ventas	30
PB27	Eliminar venta	20
PB28	Listado de inventario	60
PB29	Listado de Poco Stock	50
PB30	Generar Reportes de compra y venta	80

Estructura de Desglose de Trabajo (EDT).

Figura 13: Estructura de Desglose de Trabajo



Product Backlog Priorizado y Estimado.

Tabla 12: Product Backlog Priorizado y Estimado.

ID_PB	DESCRIPCION	MOSCOW	ESTIMACION
PB01	Registro de usuario	M	30
PB02	Inicio de Sesión	M	30
PB03	Listar Usuario	S	20
PB04	Actualizar usuario	M	30
PB05	Eliminación de usuario	C	20
PB06	Lista de Proveedores	S	40
PB07	Registrar de Proveedores	M	50
PB08	Actualización de Proveedores	M	20
PB09	Eliminación de Proveedores	C	20
PB10	Listar Categoría	S	40
PB11	Registrar Categoría	M	50
PB12	Actualizar Categoría	M	20
PB13	Buscar Categoría	S	30
PB14	Eliminar Categoría	C	20
PB15	Lista de Producto	S	70
PB16	Registro de Producto	M	60
PB17	Actualizar Producto	M	40
PB18	Búsqueda de Producto	S	20
PB19	Eliminar Producto	C	20
PB20	Registro de Compras	M	80
PB21	Lista de Compras	S	50

PB22	Búsqueda por fecha Compras	S	60
PB23	Eliminar Compras	C	20
PB24	Registro de ventas	M	80
PB25	Lista de ventas	S	50
PB26	Búsqueda por fecha ventas	S	30
PB27	Eliminar venta	C	20
PB28	Listado de inventario	M	60
PB29	Listado de Poco Stock	M	50
PB30	Generar Reportes de compra y venta	M	80

Sprint Planning.

Tabla 13: Spring Planning

SPRING	ID_PB	HISTORIA DE USUARIO
Spring 1	PB01	Registro de usuario
	PB02	Inicio de Sesión
	PB03	Listar Usuario
	PB04	Actualizar usuario
	PB05	Eliminación de usuario
Spring 2	PB06	Lista de Proveedores
	PB07	Registrar de Proveedores
	PB08	Actualización de Proveedores
	PB09	Eliminación de Proveedores
	PB10	Listar Categoría
Spring 3	PB11	Registrar Categoría

	PB12	Actualizar Categoría
	PB13	Buscar Categoría
	PB14	Eliminar Categoría
	PB15	Lista de Producto
Spring 4	PB16	Registro de Producto
	PB17	Actualizar Producto
	PB18	Buscar Producto
	PB19	Eliminar Producto
	PB20	Registro de Compras
Spring 5	PB21	Lista de Compras
	PB22	Buscar por fecha Compras
	PB23	Eliminar Compras
	PB24	Registro de ventas
	PB25	Lista de ventas
Spring 6	PB26	Buscar por fecha ventas
	PB27	Eliminar venta
	PB28	Listado de inventario
	PB29	Listado de Poco Stock
	PB30	Generar Reportes de compra y venta

Descomposición de Tareas (Plantilla Scrum).

Figura 14: Descomposición de Tareas (Spring 1)

SPRING	PRODUCT BACKLOG	HISTORIA DE USUARIO	TAREAS	TIPO DE TAREA	DESCRIPCION DE TAREAS	VALOR	RESPONSABLE	FECHA DE INICIO	FECHA DE FIN	ESTADO
Spring 1	PB01	Registro de usuario	T1	Análisis	Análisis de Registro de usuario	30	Jesmar Medaly Concha Tinoco	25/04/2022	27/04/2022	Terminado
			T2	Diseño	Diseño de Registro de usuario					
			T3	Programación	Implementación de Registro de usuario					
			T4	Pruebas	Autenticación de Pruebas Registro de usuario					
	PB02	Inicio de Sesión	T1	Análisis	Análisis de Inicio de Sesión	30	Jesmar Medaly Concha Tinoco	28/04/2022	30/04/2022	Terminado
			T2	Diseño	Diseño de Inicio de Sesión					
			T3	Programación	Implementación de Inicio de Sesión					
			T4	Pruebas	Autenticación de Pruebas Inicio de Sesión					
	PB03	Listar Usuario	T1	Análisis	Análisis de Lista Usuario	20	Jesmar Medaly Concha Tinoco	01/05/2022	02/05/2022	Terminado
			T2	Diseño	Diseño de Lista Usuario					
			T3	Programación	Implementación de Lista Usuario					
			T4	Pruebas	Autenticación de Pruebas de Lista Usuario					
	PB04	Actualizar usuario	T1	Análisis	Análisis de Actualización Usuario	30	Jesmar Medaly Concha Tinoco	03/05/2022	05/05/2022	Terminado
			T2	Diseño	Diseño de Actualización Usuario					
			T3	Programación	Implementación de Actualización Usuario					
			T4	Pruebas	Autenticación de Actualización Usuario					
PB05	Eliminación de usuario	T1	Análisis	Análisis de Eliminación de Usuario	20	Jesmar Medaly Concha Tinoco	06/05/2022	07/05/2022	Terminado	
		T2	Diseño	Diseño de Eliminación de Usuario						
		T3	Programación	Implementación de Eliminación de Usuario						
		T4	Pruebas	Autenticación de Eliminación de Usuario						

Figura 15: Descomposición de Tareas (Spring 2)

SPRING	PRODUCT BACKLOG	HISTORIA DE USUARIO	TAREAS	TIPO DE TAREA	DESCRIPCION DE TAREAS	VALOR	RESPONSABLE	FECHA DE INICIO	FECHA DE FIN	ESTADO
Spring 2	PB06	Listar de Proveedores	T1	Análisis	Análisis de Lista de Proveedores	40	Jesmar Medaly Concha Tinoco	08/05/2022	09/05/2022	Terminado
			T2	Diseño	Diseño de Lista de Proveedores					
			T3	Programación	Implantación de Lista de Proveedores					
			T4	Pruebas	Autenticación de Lista Proveedores					
	PB07	Registrar de Proveedores	T1	Análisis	Análisis de Registro de Proveedores	50	Jesmar Medaly Concha Tinoco	10/05/2022	13/05/2022	Terminado
			T2	Diseño	Diseño de Registro de Proveedores					
			T3	Programación	Implementación de Registro de Proveedores					
			T4	Pruebas	Autenticación de Registro de Proveedores					
	PB08	Actualización de Proveedores	T1	Análisis	Análisis de Actualización de Proveedores	20	Jesmar Medaly Concha Tinoco	14/05/2022	16/05/2022	Terminado
			T2	Diseño	Diseño de Actualización de Proveedores					
			T3	Programación	Implementación de Actualización de Proveedores					
			T4	Pruebas	Autenticación de Actualización de Proveedores					
	PB09	Eliminación de Proveedores	T1	Análisis	Análisis de Eliminación de Proveedores	20	Jesmar Medaly Concha Tinoco	17/05/2022	18/05/2022	Terminado
			T2	Diseño	Diseño de Eliminación de Proveedores					
			T3	Programación	Implementación de Eliminación de Proveedores					
			T4	Pruebas	Autenticación de Eliminación de Proveedores					
PB10	Listar Categoría	T1	Análisis	Análisis de Lista Categoría	40	Jesmar Medaly Concha Tinoco	19/05/2022	20/05/2022	Terminado	
		T2	Diseño	Diseño de Lista Categoría						
		T3	Programación	Implementación de Lista Categoría						
		T4	Pruebas	Autenticación de Lista Categoría						

Figura 16: Descomposición de Tareas (Spring 3)

SPRING	PRODUCT BACKLOG	HISTORIA DE USUARIO	TAREAS	TIPO DE TAREA	DESCRIPCION DE TAREAS	VALOR	RESPONSABLE	FECHA DE INICIO	FECHA DE FIN	ESTADO
Spring 3	PB11	Registrar Categoría	T1	Análisis	Análisis de Registro Categoría	50	Jesmar Medaly Concha Tinoco	21/05/2022	24/05/2022	Terminado
			T2	Diseño	Diseño de Registro Categoría					
			T3	Programación	Implementación de Registro Categoría					
			T4	Pruebas	Autenticación de Registro Categoría					
	PB12	Actualizar Categoría	T1	Análisis	Análisis de Actualización de Categoría	20	Jesmar Medaly Concha Tinoco	25/05/2022	27/05/2022	Terminado
			T2	Diseño	Diseño de Actualización Categoría					
			T3	Programación	Implementación de Actualización Categoría					
			T4	Pruebas	Autenticación de Actualización Categoría					
	PB13	Buscar Categoría	T1	Análisis	Análisis de Buscar Categoría	30	Jesmar Medaly Concha Tinoco	28/05/2022	29/05/2022	Terminado
			T2	Diseño	Diseño de Buscar Categoría					
			T3	Programación	Implementación de Buscar Categoría					
			T4	Pruebas	Autenticación de Buscar Categoría					
	PB14	Eliminar Categoría	T1	Análisis	Análisis de Eliminación Categoría	20	Jesmar Medaly Concha Tinoco	30/05/2022	31/05/2022	Terminado
			T2	Diseño	Diseño de Eliminación Categoría					
			T3	Programación	Implementación de Eliminación Categoría					
			T4	Pruebas	Autenticación de Eliminación Categoría					
PB15	Lista de Producto	T1	Análisis	Análisis de Lista de Producto	70	Jesmar Medaly Concha Tinoco	01/06/2022	02/06/2022	Terminado	
		T2	Diseño	Diseño de Lista de Producto						
		T3	Programación	Implementación de Listar de Producto						
		T4	Pruebas	Autenticación de Listar de Producto						

Figura 17: Descomposición de Tareas (Spring 4)

SPRING	PRODUCT BACKLOG	HISTORIA DE USUARIO	TAREAS	TIPO DE TAREA	DESCRIPCION DE TAREAS	VALOR	RESPONSABLE	FECHA DE INICIO	FECHA DE FIN	ESTADO
Spring 4	PB16	Registro de Producto	T1	Análisis	Análisis de Registro de Producto	60	Jesmar Medaly Concha Tinoco	03/06/2022	06/06/2022	Terminado
			T2	Diseño	Diseño de Registro de Producto					
			T3	Programación	Implementación de Registro de Producto					
			T4	Pruebas	Autenticación de Registro de Producto					
	PB17	Actualizar Producto	T1	Análisis	Análisis de Actualización de Producto	40	Jesmar Medaly Concha Tinoco	07/06/2022	09/06/2022	Terminado
			T2	Diseño	Diseño de Actualización de Producto					
			T3	Programación	Implementación de Actualización de Producto					
			T4	Pruebas	Autenticación de Actualización de Producto					
	PB18	Buscar de Producto	T1	Análisis	Análisis de Búsqueda de Producto	20	Jesmar Medaly Concha Tinoco	10/06/2022	11/06/2022	Terminado
			T2	Diseño	Diseño de Búsqueda de Producto					
			T3	Programación	Implementación de Búsqueda de Producto					
			T4	Pruebas	Autenticación de Búsqueda de Producto					
	PB19	Eliminar Producto	T1	Análisis	Análisis de Eliminación de Producto	20	Jesmar Medaly Concha Tinoco	12/06/2022	13/06/2022	Terminado
			T2	Diseño	Diseño de Eliminación de Producto					
			T3	Programación	Implementación de Eliminación de Producto					
			T4	Pruebas	Autenticación de Eliminación de Producto					
	PB20	Registro de Compras	T1	Análisis	Análisis de Registro de Compras	80	Jesmar Medaly Concha Tinoco	14/06/2022	17/06/2022	Terminado
			T2	Diseño	Diseño de Registro de Compras					
			T3	Programación	Implementación de Registro de Compras					
			T4	Pruebas	Autenticación de Registro de Compras					

Figura 18: Descomposición de Tareas (Spring 5)

SPRING	PRODUCT BACKLOG	HISTORIA DE USUARIO	TAREAS	TIPO DE TAREA	DESCRIPCION DE TAREAS	VALOR	RESPONSABLE	FECHA DE INICIO	FECHA DE FIN	ESTADO
Spring 5	PB21	Lista de Compras	T1	Análisis	Análisis de Lista de Compras	50	Jesmar Medaly Concha Tinoco	18/06/2022	19/06/2022	Terminado
			T2	Diseño	Diseño de Lista de Compras					
			T3	Programación	Implementación de Lista de Compras					
			T4	Pruebas	Autenticación de Lista de Compras					
	PB22	Buscar por fecha Compras	T1	Análisis	Análisis de Búsqueda por fecha de Compras	60	Jesmar Medaly Concha Tinoco	20/06/2022	21/06/2022	Terminado
			T2	Diseño	Diseño de Búsqueda por fecha de Compras					
			T3	Programación	Implementación de Búsqueda por fecha de Compras					
			T4	Pruebas	Autenticación de Búsqueda por fecha de Compras					
	PB23	Eliminar Compras	T1	Análisis	Análisis de Eliminación de compras	20	Jesmar Medaly Concha Tinoco	22/06/2022	23/06/2022	Terminado
			T2	Diseño	Diseño de Eliminación de Compras					
			T3	Programación	Implementación de Eliminación de Compras					
			T4	Pruebas	Autenticación de Eliminación de Compras					
	PB24	Registro de ventas	T1	Análisis	Análisis de Registro de Ventas	80	Jesmar Medaly Concha Tinoco	24/06/2022	27/06/2022	Terminado
			T2	Diseño	Diseño de Registro de Ventas					
			T3	Programación	Implementación de Registro de Ventas					
			T4	Pruebas	Autenticación de Registro de Ventas					
PB25	Lista de ventas	T1	Análisis	Análisis de Lista de Ventas	50	Jesmar Medaly Concha Tinoco	28/06/2022	30/06/2022	Terminado	
		T2	Diseño	Diseño de Lista de Ventas						
		T3	Programación	Implementación de Lista de Ventas						
		T4	Pruebas	Autenticación de Lista de Ventas						

Figura 19: Descomposición de Tareas (Spring 6)

SPRING	PRODUCT BACKLOG	HISTORIA DE USUARIO	TAREAS	TIPO DE TAREA	DESCRIPCION DE TAREAS	VALOR	RESPONSABLE	FECHA DE INICIO	FECHA DE FIN	ESTADO
Spring 6	PB26	Buscar por fecha ventas	T1	Análisis	Análisis de Búsqueda por fecha de Ventas	30	Jesmar Medaly Concha Tinoco	01/07/2022	04/07/2022	Terminado
			T2	Diseño	Diseño de Búsqueda por fecha de Ventas					
			T3	Programación	Implementación de Búsqueda por fecha de Ventas					
			T4	Pruebas	Autenticación de Búsqueda por fecha de Ventas					
	PB27	Eliminar venta	T1	Análisis	Análisis de Eliminación de Venta	20	Jesmar Medaly Concha Tinoco	05/07/2022	07/07/2022	Terminado
			T2	Diseño	Diseño de Eliminación de Venta					
			T3	Programación	Implementación de Eliminación de Venta					
			T4	Pruebas	Autenticación de Eliminación de Venta					
	PB28	Listado de inventario	T1	Análisis	Análisis de Listado de Inventario	60	Jesmar Medaly Concha Tinoco	08/07/2022	10/07/2022	Terminado
			T2	Diseño	Diseño de Listado de Inventario					
			T3	Programación	Implementación de Listado de Inventario					
			T4	Pruebas	Autenticación de Listado de Inventario					
	PB29	Listado de Poco Stock	T1	Análisis	Análisis de Listado de Poco Stock	50	Jesmar Medaly Concha Tinoco	11/07/2022	13/07/2022	Terminado
			T2	Diseño	Diseño de Listado de Poco Stock					
			T3	Programación	Implementación de Listado de Poco Stock					
			T4	Pruebas	Autenticación de Listado de Poco Stock					
PB30	Generar Reportes de compra y venta	T1	Análisis	Análisis de Generar Reportes de Compra y Venta	80	Jesmar Medaly Concha Tinoco	14/07/2022	18/07/2022	Terminado	
		T2	Diseño	Diseño de Generar Reportes de Compra y Venta						
		T3	Programación	Implementación de Generar Reportes de Compra y Venta						
		T4	Pruebas	Autenticación de Generar Reportes de Compra y Venta						

Fase Implementación.
Incremento Sprint Backlog.

Figura 20: Sprint Backlog Identificado

IDENTIFICADO						EN PROGRESO	RESUELTO
01 20 Registro de usuario	02 30 Inicio de Sesión	03 20 Listar Usuario	04 30 Actualizar usuario	05 20 Eliminación de usuario	06 40 Lista de Proveedores		
07 30 Registrar de Proveedores	08 20 Actualización de Proveedores	09 20 Eliminación de Proveedores	10 40 Listar Categoría	11 30 Registrar Categoría	12 20 Actualizar Categoría		
13 20 Buscar Categoría	14 20 Eliminar Categoría	15 65 Lista de Producto	16 60 Registro de Producto	17 35 Actualizar Producto	18 20 Búsqueda de Producto		
19 20 Eliminar Producto	20 80 Registro de Compras	21 50 Lista de Compras	22 60 Búsqueda por fecha Compras	23 20 Eliminar Compras	24 80 Registro de ventas		
25 50 Lista de ventas	26 20 Búsqueda por fecha ventas de ventas	27 20 Eliminar venta	28 60 Listado de inventario	29 50 Listado de Poco Stock de inventario	30 80 Generar Reportes de compra y venta		

Figura 21: Sprint Backlog En proceso y Resuelto

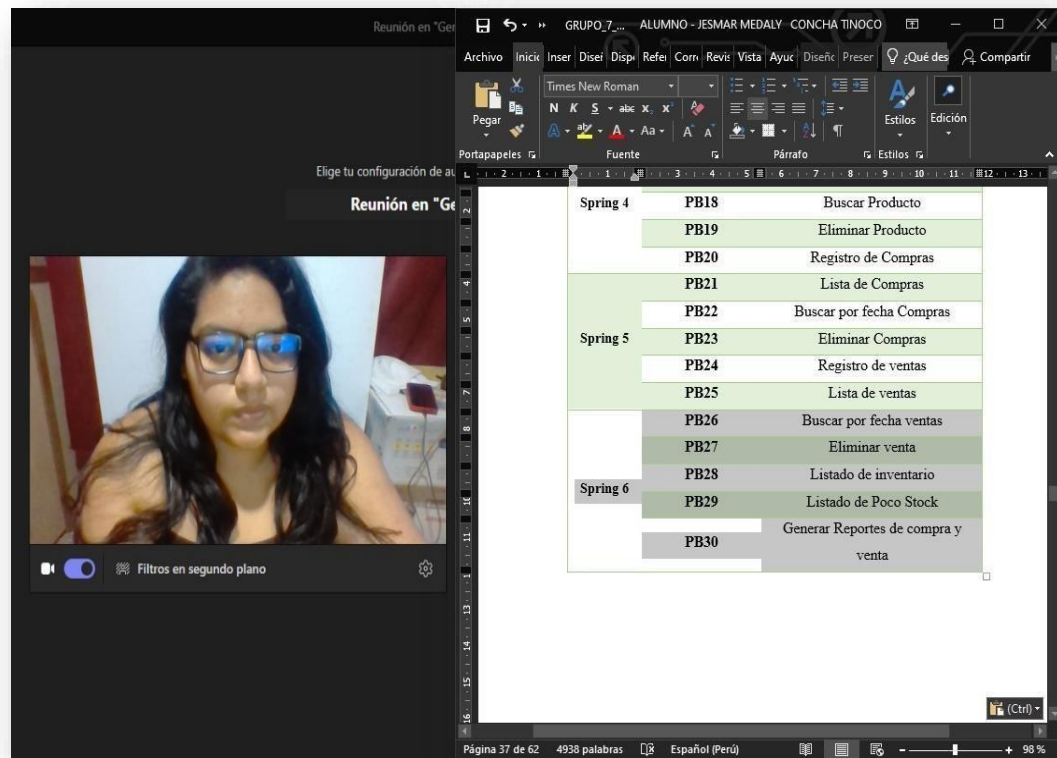
IDENTIFICADO	EN PROGRESO	RESUELTO
	21 50 Lista de Compras	01 20 Registro de usuario
	22 60 Búsqueda por fecha Compras	02 30 Inicio de Sesión
	23 20 Eliminar Compras	03 20 Listar Usuario
	24 80 Registro de ventas	04 30 Actualizar usuario
	25 50 Lista de ventas	05 20 Eliminación de usuario
	26 20 Búsqueda por fecha ventas de ventas	06 40 Lista de Proveedores
	27 20 Eliminar venta	07 30 Registrar de Proveedores
	28 60 Listado de inventario	08 20 Actualización de Proveedores
	29 50 Listado de Poco Stock de inventario	09 20 Eliminación de Proveedores
	30 80 Generar Reportes de compra y venta	10 40 Listar Categoría
		11 30 Registrar Categoría
		12 20 Actualizar Categoría
		13 20 Buscar Categoría
		14 20 Eliminar Categoría
		15 65 Lista de Producto
		16 60 Registro de Producto
		17 35 Actualizar Producto
		18 20 Búsqueda de Producto
		19 20 Eliminar Producto
		20 80 Registro de Compras

Figura 22: Sprint Backlog Resuelto

IDENTIFICADO	EN PROGRESO	RESUELTO
		01 20 Registro de usuario
		02 30 Inicio de Sesión
		03 20 Listar Usuario
		04 30 Actualizar usuario
		05 20 Eliminación de usuario
		06 40 Lista de Proveedores
		07 30 Registrar de Proveedores
		08 20 Actualización de Proveedores
		09 20 Eliminación de Proveedores
		10 40 Listar Categoría
		11 30 Registrar Categoría
		12 20 Actualizar Categoría
		13 20 Buscar Categoría
		14 20 Eliminar Categoría
		15 65 Lista de Producto
		16 60 Registro de Producto
		17 35 Actualizar Producto
		18 20 Búsqueda de Producto
		19 20 Eliminar Producto
		20 80 Registro de Compras
		21 50 Lista de Compras
		22 60 Búsqueda por fecha Compras
		23 20 Eliminar Compras
		24 80 Registro de ventas
		25 50 Lista de ventas
		26 20 Búsqueda por fecha ventas de ventas
		27 20 Eliminar venta
		28 60 Listado de inventario
		29 50 Listado de Poco Stock de inventario
		30 80 Generar Reportes de compra y venta

Mantenimiento del Product Backlog.

Figura 23: Mantenimiento de Product Backlog



Daily Scrum Meeting.

Figura 24: Daily Scrum Meeting (Sprint 01)

Spring	Scrum Team	Meta	¿Qué hiciste ayer?	¿Qué vas hacer hoy?	¿Algún Impedimento?
Spring 1	Jesmar Medaly Concha Tinoco	Analizar, Desarrollar e implementar lo que es el modulo de usuarios para que los usuarios tengas un registro y puedan iniciar sesión en el sistema	Corrigiendo algunos errores con el listado de usuarios en el sistema y Analizando nuevas historias de usuario para avanzar con la implementación del sistema.	Implementando el sistema con las Actualizaciones y Eliminación de Usuarios.	Complicaciones con el internet o por motivo de falta de fluido eléctrico.

Figura 25: Daily Scrum Meeting (Sprint 02)

Spring	Scrum Team	Meta	¿Qué hiciste ayer?	¿Qué vas hacer hoy?	¿Algún Impedimento?
Spring 2	Jesmar Medaly Concha Tinoco	Analizar, Desarrollar e implementar lo que es el modulo de proveedores para que el administrador tengas un registro de proveedores	Corrigiendo algunos errores con el listado de proveedores en el sistema y Analizando nuevas historias de Proveedores.	Implementando el sistema con las Actualizaciones y Eliminación de Proveedores de Producto.	Problemas de Salud

Figura 26: Daily Scrum Meeting (Sprint 03)

Spring	Scrum Team	Meta	¿Qué hiciste ayer?	¿Qué vas hacer hoy?	¿Algún Impedimento?
Spring 3	Jesmar Medaly Concha Tinoco	Analizar, Desarrollar e implementar lo que es el modulo de categoria.	Corrigiendo algunos errores con el listado de productos en el sistema y Analizando nuevas historias de productos.	Implementando el sistema con las Actualizaciones y Eliminación de categorias.	sin impedimentos

Figura 27: Daily Scrum Meeting (Sprint 04)

Spring	Scrum Team	Meta	¿Qué hiciste ayer?	¿Qué vas hacer hoy?	¿Algún Impedimento?
Spring 4	Jesmar Medaly Concha Tinoco	Analizar, Desarrollar e implementar lo que es el modulo de productos para que el administrador tengas un registro de productos.	Corrigiendo algunos errores que se presentan en el modulo de productos y Analizando nuevas historias de usuario para avanzar con la implementación del sistema.	Implementando el sistema analizando el registro de compras de producto.	Falta de energia de luz.

Figura 28: Daily Scrum Meeting (Sprint 05)

Spring	Scrum Team	Meta	¿Qué hiciste ayer?	¿Qué vas hacer hoy?	¿Algún Impedimento?
Spring 5	Jesmar Medaly Concha Tinoco	Analizar, Desarrollar e implementar lo que es el modulo de compras para que el administrador tengas un control de sus compras.	Corrigiendo algunos errores con el listado de compras en el sistema y Analizando nuevas historias de compras para avanzar con la implementación del sistema.	Implementando el sistema con las Actualizaciones y Eliminación de compras.	sin impedimentos

Figura 29: Daily Scrum Meeting (Sprint 06)

Spring	Scrum Team	Meta	¿Qué hiciste ayer?	¿Qué vas hacer hoy?	¿Algún Impedimento?
Spring 6	Jesmar Medaly Concha Tinoco	Analizar, Desarrollar e implementar lo que es el modulo de ventas para que el administrador tengas un registro de venta.	Corrigiendo algunos errores con el listado de ventas en el sistema y Analizando nuevas historias de usuario para avanzar con la implementación del sistema.	Implementando el sistema generando reportes de compra y venta.	Problemas de Salud

Fase Revisión Y Retrospectiva.

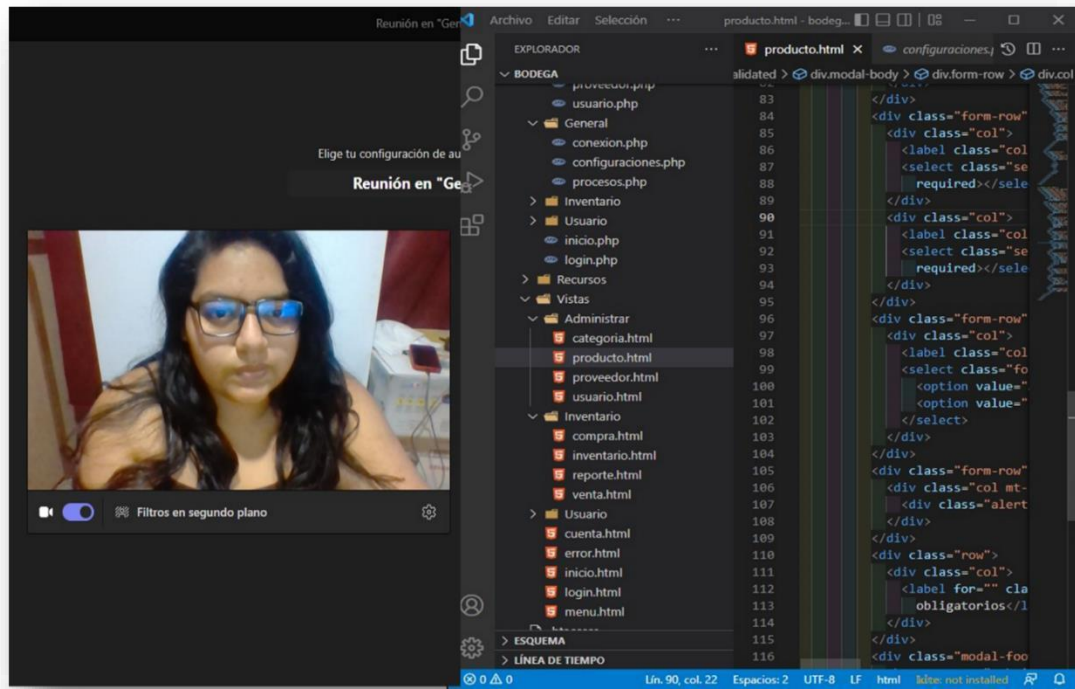
Scrum Board .

Figura 30: Scrum Board

IDENTIFICADO	EN PROGRESO	RESUELTO
<p>20 80 Registro de Compras</p> <p>21 50 Lista de Compras</p> <p>22 60 Búsqueda por fecha Compras</p> <p>23 20 Eliminar Compras</p> <p>25 50 Lista de ventas</p> <p>26 20 Búsqueda por fecha ventas de ventas</p> <p>27 20 Eliminar venta</p> <p>28 60 Listado de inventario</p> <p>29 50 Listado de Poco Stock de inventario</p> <p>30 80 Generar Reportes de compra y venta</p>	<p>11 30 Registrar Categoría</p> <p>12 20 Actualizar Categoría</p> <p>13 20 Buscar Categoría</p> <p>14 20 Eliminar Categoría</p> <p>15 65 Lista de Producto</p> <p>16 60 Registro de Producto</p> <p>17 35 Actualizar Producto</p> <p>18 20 Búsqueda de Producto</p> <p>24 80 Registro de ventas</p> <p>19 20 Eliminar Producto</p>	<p>01 20 Registro de usuario</p> <p>02 30 Inicio de Sesion</p> <p>03 20 Listar Usuario</p> <p>06 40 Lista de Proveedores</p> <p>05 20 Eliminación de usuario</p> <p>04 30 Actualizar usuario</p> <p>09 20 Eliminación de Proveedores</p> <p>08 20 Actualización de Proveedores</p> <p>07 30 Registrar de Proveedores</p> <p>10 40 Listar Categoría</p>

Validación de los Sprint. Sprint Review.

Figura 30: Sprint Review



Burn Down Chart.

Figura 31: Scrum Board (Sprint 01)

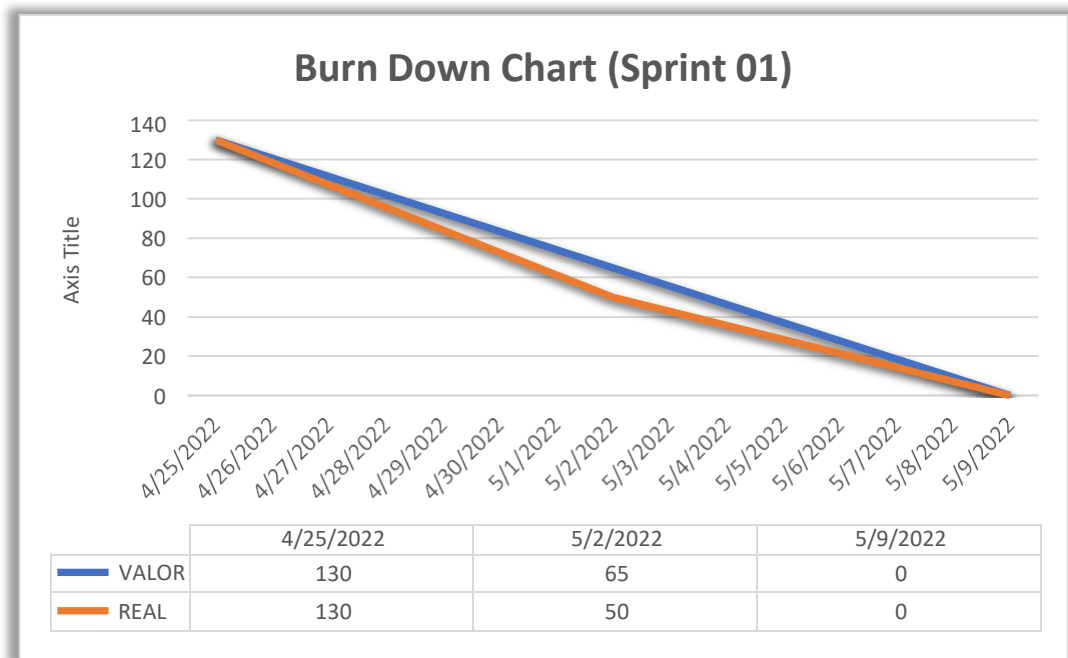


Figura 32: Scrum Board (Sprint 02)

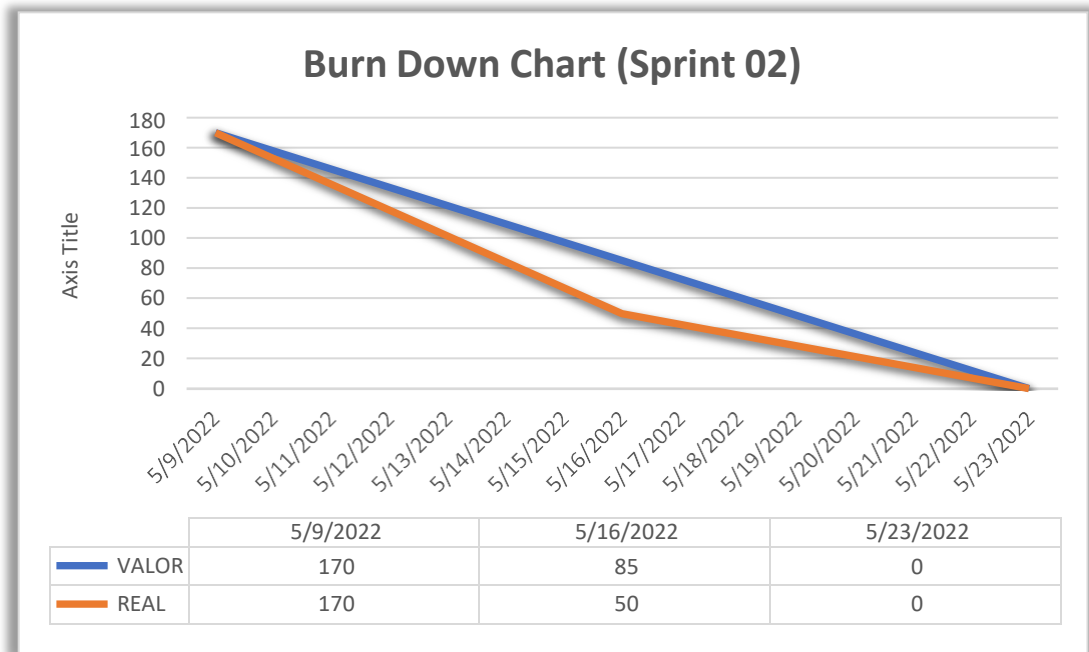


Figura 33: Scrum Board (Sprint 03)

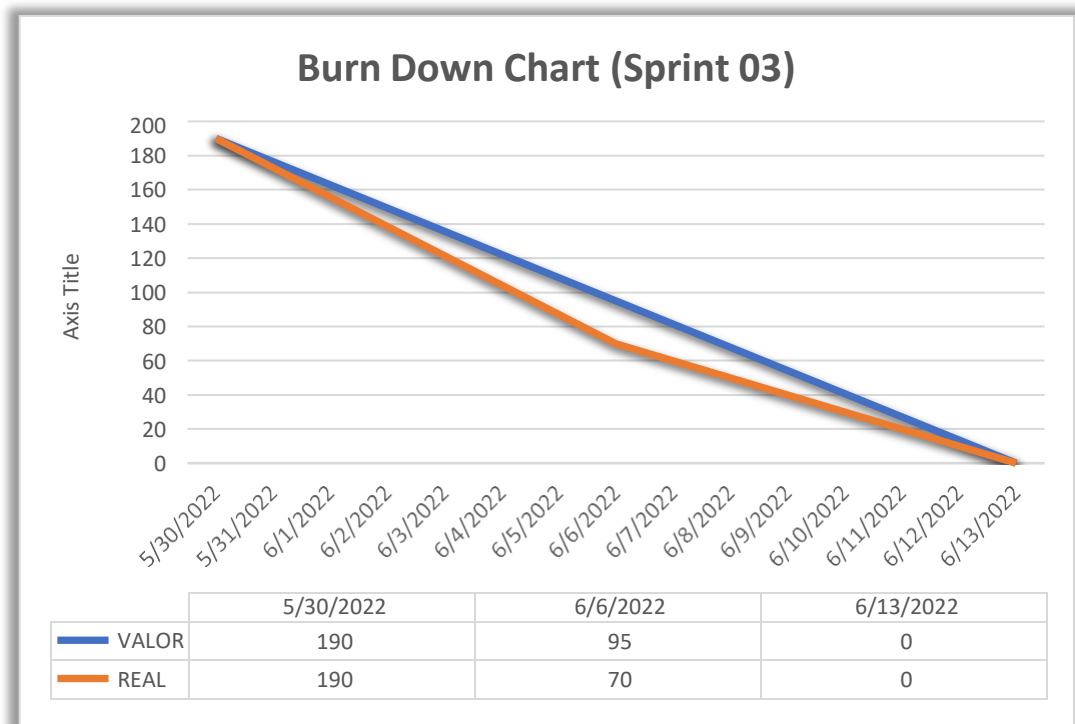


Figura 34: Scrum Board (Sprint 04)

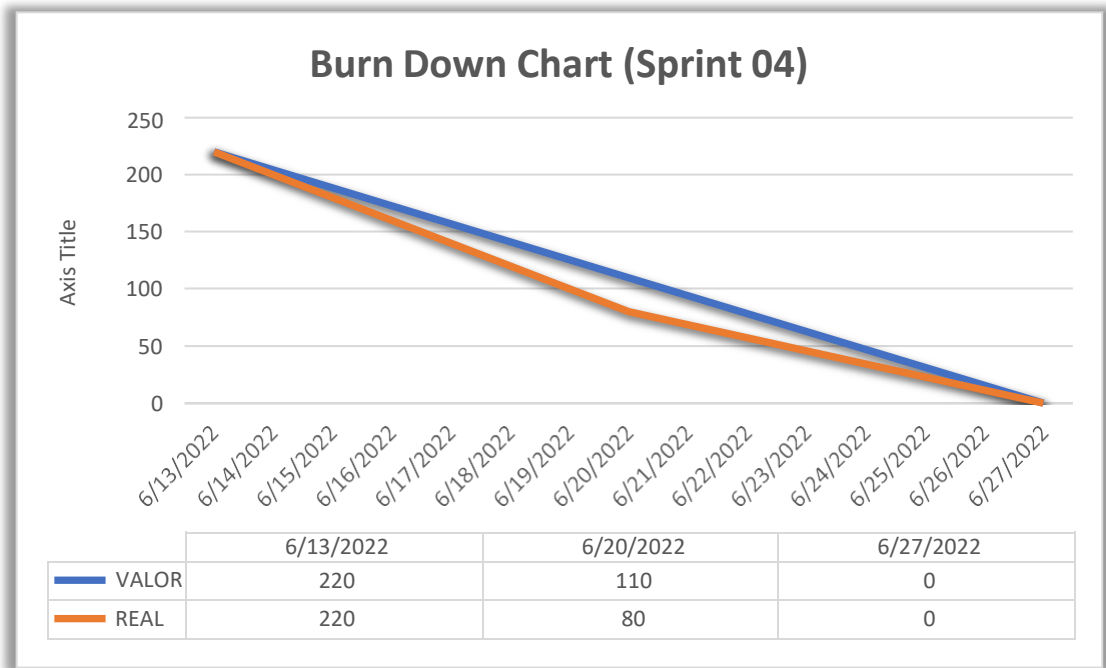


Figura 35: Scrum Board (Sprint 05)

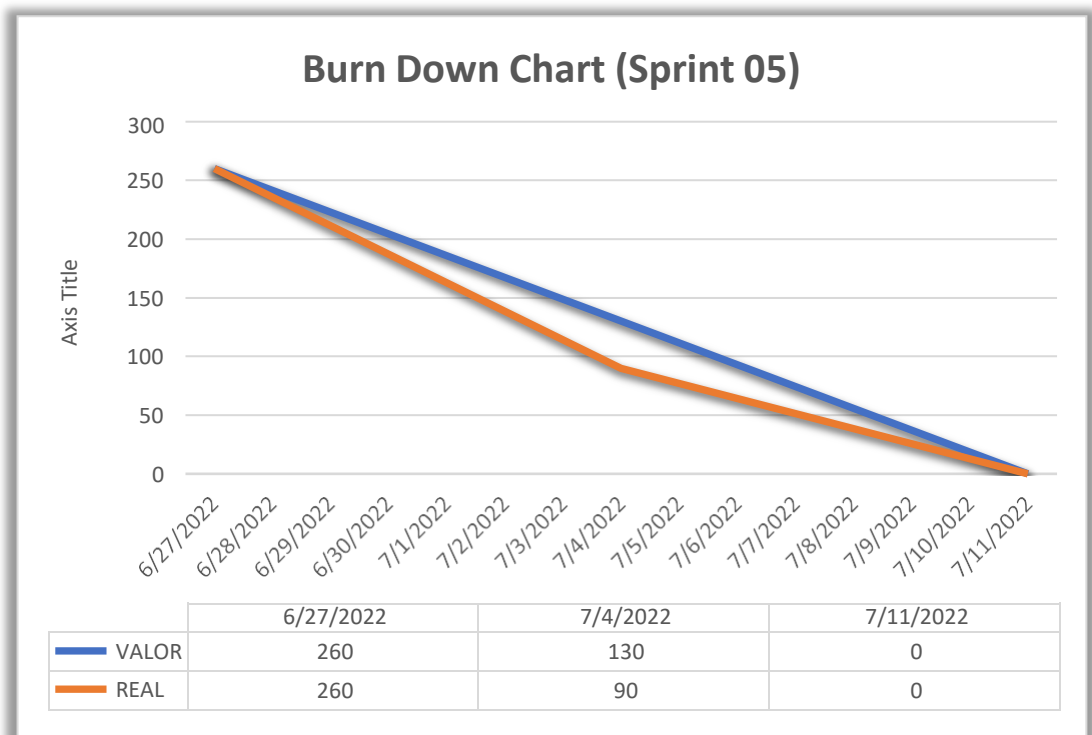
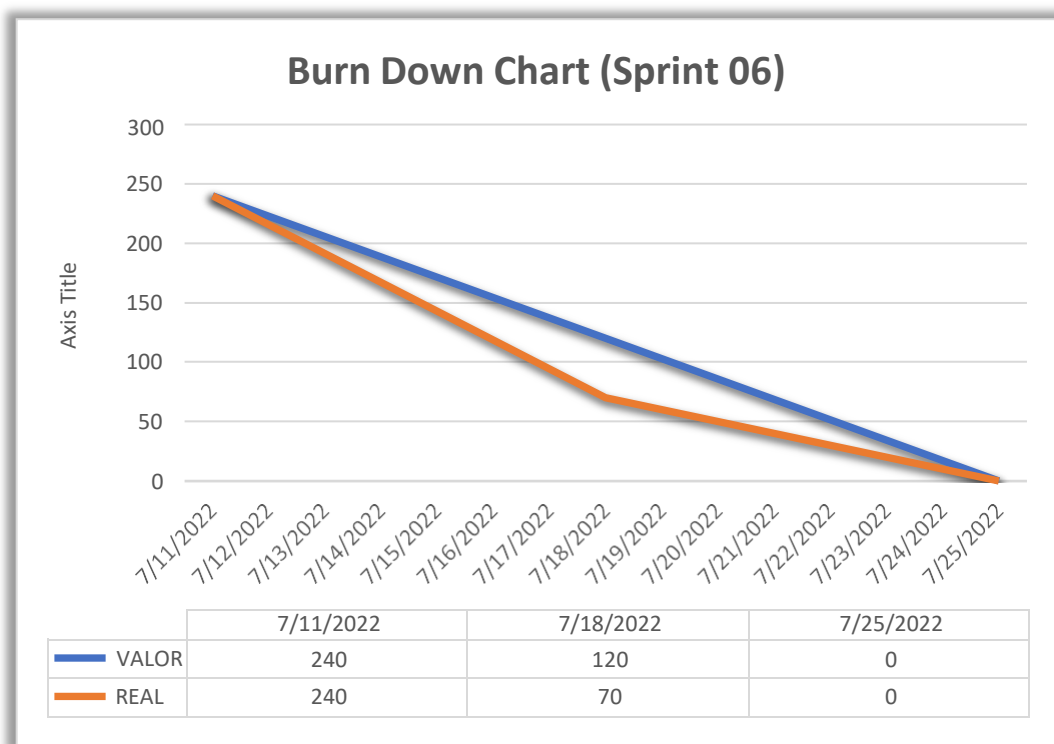


Figura 36: Scrum Board (Sprint 06)**Creación de la Lista de Impediment Log.****Figura 14: Impediment Log**

Ítems	Impediment Log
01	Problemas de conexión a Internet.
02	Perdida de Código fuente por motivo que se formateo la laptop.
03	Problemas con la disponibilidad de tiempo por motivo de trabajo.
04	Problemas de Salud
05	Problemas con el programa XAMPP.
06	Problemas con la Documentación

Retrospectiva del Sprint

Tabla 15: Datos de la empresa ejecutora y nombre del proyecto.

Empresa	Bodega Jesmar
Proyecto	Desarrollo e implementación de un sistema web para la administración de ventas, compras e inventario para la bodega JESMAR en el distrito de José Leonardo Ortiz – Chiclayo - Lambayeque en el periodo 2022

Tabla 16: Datos de la reunión:

Fecha	15/07/2022	
Hora inicio	07:00 pm	
Hora fin	07:30 pm	
Número de iteración	1	
Personas convocadas a la reunión	Nombre(s)	Apellidos
	• Jesmar Medaly	• Concha Tinoco

Tabla 17: Formato de reunión de retrospectiva

¿Qué se hizo bien en la iteración?	¿Qué no se hizo bien en la iteración?	¿Qué mejoras se van a implementar en la siguiente iteración?
<ul style="list-style-type: none"> Las actividades correspondientes en el front-end fueron realizadas con eficiencia, presentando pocos problemas en su desarrollo de los sprints. La documentación de los Sprint ha tenido algunos inconvenientes por motivo de que algunos módulos aún no se avanzaban para poder completar la documentación 	<ul style="list-style-type: none"> A pesar de que las tareas habían sido todos los objetivos de los sprint, principalmente las partes de programación back-end. Esto se debió a diferentes factores como: <ul style="list-style-type: none"> Sucesos imprevistos como cortes de luz a causa del reciente sismo. Problemática gestión de tiempo, pues otra clase de ocupaciones, como lo son estudios y trabajo, se cruzaron en los horarios de algunos integrantes. 	<ul style="list-style-type: none"> Definir mejor plazos y expectativas realistas en base a la disponibilidad de trabajo. Mejorar gestión de horarios con el tema de los avances en los sprint. Definir el plazo de entrega del proyecto de tesis.

Pruebas de Funcionalidad. Pruebas de Aceptación. Pruebas de Rendimiento.

Figura 37: Pruebas de funcionalidad de ventas

The screenshot shows a web application interface on the left and a browser's developer console on the right. The web application has a date input field set to '21/07/2022' and a table with three columns: 'Acciones', 'Fecha', 'Producto', and 'Precio_venta'. The table contains three records:

Acciones	Fecha	Producto	Precio_venta
	2022-07-21 07:33:43	Jabón Bolívar (comercial mabry)	2.50
	2022-07-21 07:35:06	Lejía patito 500ml (comercial mabry)	2.00
	2022-07-21 07:36:18	avena kauer (comercial mabry)	1.50

Below the table, it says 'Total: 3 registros'. The developer console on the right shows a network log for 'ventas.js?v=0.263382' with a response containing an array of objects:

```
[[{"Fecha": "2022-07-21 07:36:18", "Producto": "avena kauer (comercial mabry)", "Precio_venta": 1.5}, {"Fecha": "2022-07-21 07:33:43", "Producto": "Jabón Bolívar (comercial mabry)", "Precio_venta": 2.5}, {"Fecha": "2022-07-21 07:35:06", "Producto": "Lejía patito 500ml (comercial mabry)", "Precio_venta": 2.0}]]
```

Figura 38: Pruebas de funcionalidad de Reporte

The screenshot shows a web application interface on the left and a browser's developer console on the right. The web application has a 'Generar Reporte' button and a preview of a report. The report is titled 'Bodega Jesmar' and includes the following data:

Fecha: 21/07/2022

Ingresos / Compras			
Producto	Precio Compra	Cantidad	Sub Total
avena kauer (comercial mabry)	1.50	20	30.00
Jabón Bolívar (comercial mabry)	2.00	48	96.00
Lejía patito 500ml (comercial mabry)	1.00	12	12.00
lejía patito de 250ml (comercial mabry)	0.90	20	18.00
		100	146.00

Egresos / Ventas			
Producto	Precio Venta	Cantidad	Sub Total
avena kauer (comercial mabry)	1.50	2	3.00
Jabón Bolívar (comercial mabry)	2.50	1	2.50
Lejía patito 500ml (comercial mabry)	2.00	1	2.00
		4	7.50

Resumen			
Total ingresos	Total egresos	Total	
146.00	7.50	138.50	

The developer console on the right shows a network log for 'reporte.php' with a response containing a PDF report. The response headers include:

```

Solicitar URL: http://localhost/bodega/Web/Recursos/Reportes/Reporte%20Inventario%20(2022-07-21_2022-07-21).pdf
Método de la solicitud: GET
Código de estado: 200 OK
Dirección remota: [::1]:80
Política de referencia: strict-origin-when-cross-origin
Encabezados de respuesta: Ver origen
Accept-Ranges: bytes
Cache-Control: no-cache, no-store, must-revalidate
Connection: Keep-Alive
Content-Length: 177266
Content-Type: application/pdf
Date: Thu, 21 Jul 2022 21:43:38 GMT
ETag: W/"2b472-5e4579c485bd3"
Expires: 0
Keep-Alive: timeout=5, max=99
Last-Modified: Thu, 21 Jul 2022 21:43:38 GMT
  
```

Fase Lanzamiento.

Actas de Conformidad de los Entregables.

Figura 39: Acta de Aceptación del proyecto

ACTA DE ACEPTACIÓN DEL PROYECTO

NOMBRE DEL PROYECTO	DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB PARA LA ADMINISTRACIÓN DE VENTAS, COMPRAS E INVENTARIO PARA LA BODEGA JESMAR
NOMBRE DEL CLIENTE	Jesmar Medaly Concha Tinoco
MONTO EJECUTADO (Expresado en número)	S/. 24 960
Fecha:	20 de Julio del 2022

DECLARACIÓN DE LA ACEPTACIÓN FORMAL

Por medio de la presente acta se deja constancia de la finalización y aceptación del proyecto "BODEGA JESMAR" a cargo de la Alumna del Instituto IDAT.

En este punto se da por concluido el proyecto, por lo que habiendo constatado la finalización, entrega y aceptación del sistema web BODEGA JESMAR se certifica el cierre del proyecto, en el cual culmina de manera exitosa.

El proyecto comprendía la entrega de los siguientes entregables:

GESTIÓN DEL PROYECTO:

- Cronograma de Actividades
- Matriz de Riesgo
- Requerimientos Funcionales y no Funcionales
- Estándares de Base de Datos
- Entandares de Desarrollo

DISEÑO:

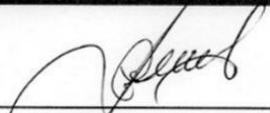
- Modelación del sistema web
- Elaboración del Diseño del Sistema Web
- Diseño de la Base de Datos

DESARROLLO

- Elaboración de la base de datos
- Historias de Usuarios
- Product Backlog
- Interfaz de cada historia de Usuario y Product Backlog.
- Desarrollo de cada historia de Usuario y Product Backlog
- Manual de Usuario
- Información general de sistema web

El proyecto ha sido desarrollado dentro de los tiempos planificados, iniciando el 16 de abril del 2022 y culmino el 20 de julio del 2022.

ACEPTADO POR


Jesús Concha Julca
 ADMINISTRADOR DE LA EMPRESA

CAPÍTULO IV (PROGRAMACIÓN)

Implementación de la Arquitectura de Software.

Figura 40: Implementación Modelo -Vista - Controlador

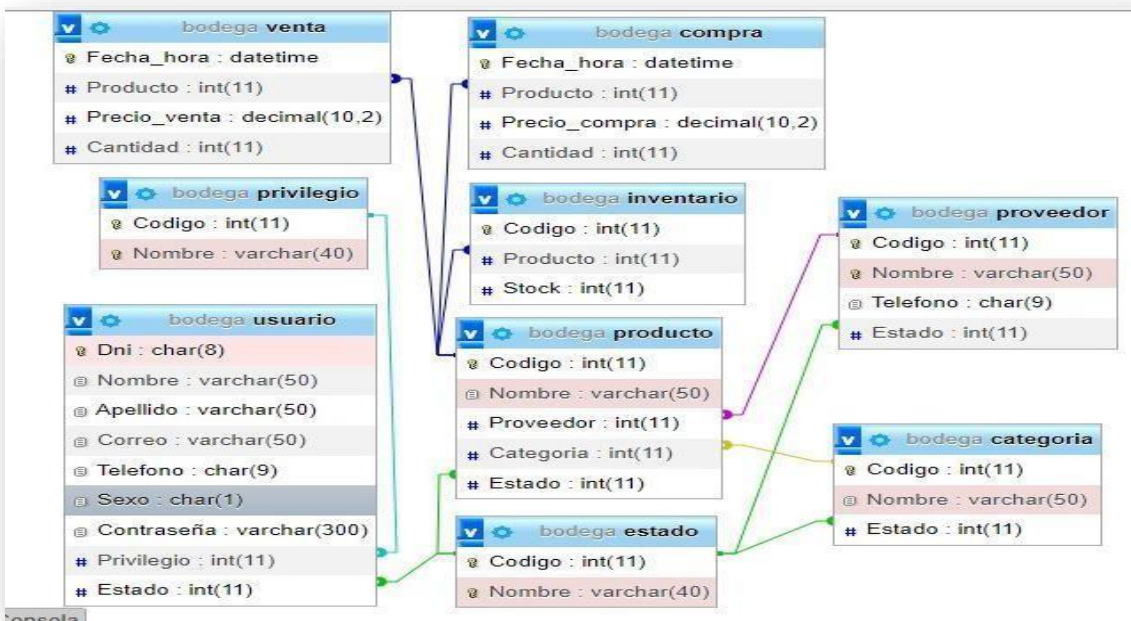
```

1 <!DOCTYPE html>
2 <html lang="es" dir="ltr">
3
4 <head>
5 <title>Bodega Jesmar</title>
6 <link type="text/css" href=".../Recursos/Cdnjs/DataTable/Css/datatables.bootstrap4.css" rel="stylesheet" />
7 <link type="text/css" href=".../Recursos/Cdnjs/DataTable/Css/datatables.bootstrap4.min.css" rel="stylesheet" />
8 <link type="text/css" href=".../Recursos/Cdnjs/FontAwesome/Css/fontAwesome.css" rel="stylesheet" />
9 <link type="text/css" href=".../Recursos/Cdnjs/Fonts/Css/font.css" rel="stylesheet" />
10 <link type="text/css" href=".../Recursos/Cdnjs/Sb-admin-2/Css/Sb-admin-2.min.css" rel="stylesheet">
11 <link type="text/css" href=".../Recursos/Cdnjs/Selectize/Css/selectize.min.css" rel="stylesheet" />
12 <link href=".../Recursos/css/stylo.css" rel="stylesheet" />
13 </head>
14
15 <body id="page-top">
16 <div id="wrapper">
17 <ul class="navbar-nav sidebar sidebar-dark accordion" id="accordionSidebar">
18
19 <a class="sidebar-brand d-flex align-items-center justify-content-center href="javascript:Datos()">
20 <div class="sidebar-brand-icon">
21 
22 </div>
23 <div class="sidebar-brand-text mx-4 text-capitalize">Bodega Jesmar</div>
24 </a>
25 <hr class="sidebar-divider my-0">
26 <li class="nav-item item-list">
27 <a id="Home" class="nav-link" href="javascript:Datos()">
28 <i class="fas fa-home"></i>
29 <span>Inicio</span></a>
30 </li>
31 <hr class="usuario sidebar-divider d-none">
32 <div class="usuario sidebar-heading d-none">
33 Modulos
34 </div>
35 <li class="nav-item item-list usuario d-none" id="Usuario">

```

Creación de la Base de Datos y Generación del Script.

Figura 41: Diagrama de la Base de Datos



Implementación de Librerías y Dependencias.

Figura 42: Librerías y Dependencias del software

The screenshot shows a Visual Studio Code editor window titled 'changelog.htm - bodega - Visual Studio Code'. The Explorer sidebar on the left shows a project structure with folders like 'Web', 'Recursos', 'Cdnjs', 'Bootstrap', 'Crypto', 'DataTable', 'FontAwesome', 'Fonts', 'Jquery', 'Sb-admin-2', 'Selectize', 'SweetAlert', 'Css', 'fpdf', 'doc', 'font', 'makefont', 'FAQ.htm', 'fpdf.css', 'fpdf.php', 'install.txt', 'license.txt', 'Imagenes', 'Reportes', and 'Vistas'. The main editor area displays the content of 'changelog.htm', which is a list of updates and dependencies:

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
5 <title>Changelog</title>
6 <link type="text/css" rel="stylesheet" href="fpdf.css">
7 <style type="text/css">
8 dd {margin:1em 0 1em 1em}
9 </style>
10 </head>
11 <body>
12 <h1>Changelog</h1>
13 <dl>
14 <dt><strong>v1.83</strong> (2021-04-18)</dt>
15 <dd>
16 - Fixed an issue related to annotations.<br>
17 </dd>
18 <dt><strong>v1.82</strong> (2019-12-07)</dt>
19 <dd>
20 - Removed a deprecation notice under PHP 7.4.<br>
21 </dd>
22 <dt><strong>v1.81</strong> (2015-12-20)</dt>
23 <dd>
24 - Added GetPageWidth() and GetPageHeight().<br>
25 - Fixed a bug in SetXY().<br>
26 </dd>
27 <dt><strong>v1.8</strong> (2015-11-29)</dt>
28 <dd>
29 - PHP 5.1.0 or higher is now required.<br>
30 - The MakeFont utility now subsets fonts, which can greatly reduce font sizes.<br>
31 - Added ToUnicode CMaps to improve text extraction.<br>
32 - Added a parameter to AddPage() to rotate the page.<br>
33 - Added a parameter to SetY() to indicate whether the x position is relative to the top or bottom of the page.<br>
34 - Added a parameter to Output() to specify the encoding of the content.<br>
35 - The Error() method now throws an exception.<br>

```

The status bar at the bottom indicates 'Ln. 1, col. 1 Espacios: 4 UTF-8 LF html'.

Codificación del Front End y Back End.

Figura 43: Back End de Proveedor

The screenshot shows a Visual Studio Code editor window titled 'proveedor.html - bodega - Visual Studio Code'. The Explorer sidebar on the left shows a project structure with folders like 'BODEGA', 'Controladores', 'Modelos', 'Administrador', 'Recursos', 'Vistas', and 'Administrar'. The main editor area displays the content of 'proveedor.html', which is an HTML form for adding a new provider:

```

48 <!-- Modal para CRUD -->
49 <div class="modal fade" id="modal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
50 <div class="modal-dialog">
51 <div class="modal-content">
52 <div class="modal-header">
53 <h5 class="modal-title"></h5>
54 <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
55   aria-hidden="true">&times;</span></button>
56 </div>
57 <div class="modal-body">
58 <div class="form-row">
59 <div class="col">
60 <input class="form-control" type="hidden" id="Cod" name="Cod" placeholder="" onkeypress="val
61   pattern="[0-9]+" readonly />
62 </div>
63 </div>
64 <div class="form-row">
65 <div class="col">
66 <label for="" class="col-form-label">Nombre: <span class="text-danger">*</span></label>
67 <input type="text" class="form-control text-capitalize" id="Nom" name="Nom" maxlength="50"
68   placeholder="Ingrese nombre" onkeyup="minuscula(this);" pattern="[a-zA-ZñÁéíóúÀÉÍÓÚ ]{1,50}"
69   required />
70 </div>
71 </div>
72 <div class="form-row">
73 <div class="col">
74 <label for="" class="col-form-label">Teléfono:</label>
75 <input type="text" id="Tel" name="Tel" maxlength="9" class="form-control" placeholder="Ingrese
76   pattern="[0-9]{9}" onkeypress="validar(event)" />
77 </div>
78 </div>
79 <div class="col">
80 <label class="col-form-label" class="label">Estado: <span class="text-danger">*</span></label>
81 <select class="form-control" id="Est" name="Est" pattern="[a-zA-ZñÁéíóúÀÉÍÓÚ ]{1,50}" requi
82

```

The status bar at the bottom indicates 'Ln. 1, col. 1 Espacios: 2 UTF-8 LF html'.

Figura 44: Front End de Proveedor

Proveedores

Búsqueda

Nombre

+
✖
🗑

Acciones	Codigo	Nombre	Telefono	Estado
	1	distribuciones angeles	074586532	Activo
	2	negocios made	074562563	Activo
	3	distribuciones paquito	074589656	Activo
	4	tailoy	516193050	Activo
	5	comercial melisa	745869365	Activo
	6	viteri fernandez mario enrique	745868965	Activo
	7	comercial mabry	972630085	Activo

Listado de Consultas

Figura 45: Listado de Consultas de Productos

Productos

Búsqueda

Nombre Proveedor Categoria

Seleccione una opción

Seleccione una opción

+
✖
🗑

Acciones	Codigo	Nombre	Proveedor	Categoria	Estado
	1	cereales azul	distribuciones angeles	cereales	Activo
	2	Jabón Bolívar	comercial mabry	jabones	Activo
	3	Lejía patito 500ml	comercial mabry	Legía	Activo
	4	lejía patito de 250ml	comercial mabry	Legía	Activo
	5	avena kauker	comercial mabry	avenas	Activo
	6	ayudín 500ml	comercial mabry	materiales de cocina	Activo
	7	filete primor	comercial mabry	lacteos	Activo

Listado de Reportes

Figura 46: Listado de Reportes

Reporte de inventario

Fecha Inicio: * 21/07/2022 ✓ Fecha Fin: * 21/07/2022 ✓

* Campos obligatorios

Generar Reporte

Reporte 1 / 1 68%

Bodega Jesmar
 RUC: 10277170159
 DIRECCIÓN: Av. Villa Hermosa Mz. D - Lt.
 AA.HBL Javier Castro Cruz N° 01
 Reporte del 21/07/2022

Fecha: 21/07/2022

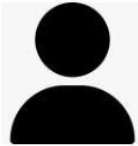
Ingresos / Compras			
Producto	Precio Compra	Cantidad	Sub Total
avena kauker (comercial mabry)	1.00	20	20,00
Jabón Bolívar (comercial mabry)	2.00	48	96,00
Lejía patito 500ml (comercial mabry)	1.00	12	12,00
lejía patito de 250ml (comercial mabry)	0.90	20	18,00
		100	146,00

Egresos / Ventas			
Producto	Precio Venta	Cantidad	Sub Total
avena kauker (comercial mabry)	1.50	2	3,00
Jabón Bolívar (comercial mabry)	2.50	1	2,50
Lejía patito 500ml (comercial mabry)	1.50	1	1,50


Procesos Transaccionales.

Figura 47: Procesos Transaccionales (Front End)

Login



Dni:
 ✓

Contraseña:
 ✓ 

Recordarme

Ingresar

Síguenos en: [f](#) [t](#) [@](#)

Copyright @ 2022 - Bodega Jesmar

Figura 48: Procesos Transaccionales (Back End)

```

1 <div class="card shadow mx-1">
2 <div class="card-header">
3 <h6 class="m-0 text-negrita text-primary text-center">Login</h6>
4 </div>
5 <div class="card-body">
6 <form id="form" class="col-xl-4 col-md-5 col-sm-12 form mx-auto was-validated" autocomplete="off">
7 <div class="form-row">
8 <div class="col text-center">
9 
10 </div>
11 </div>
12 <div class="form-row">
13 <div class="col">
14 <label class="col-form-label">Dni:</label>
15 <input class="form-control" type="text" minlength="8" maxlength="8" id="Dni" name="Dni"
16 placeholder="Ingrese Dni" pattern="[0-9]+" onkeypress='validar(event)' required />
17 </div>
18 </div>
19 <div class="form-row">
20 <div class="col">
21 <label class="col-form-label">Contraseña:</label>
22 <div class="input-group">
23 <input class="form-control" type="password" minlength="8" maxlength="40" id="Con" name="Con"
24 placeholder="Ingrese Contraseña" pattern="[0-9a-zA-Z]+" required />
25 <span class="input-group-append">
26 <button class="btn-primary btn-input-group border bt-con" type="button">
27 <i class="fas fa-eye icon-con"></i>
28 </button>
29 </span>
30 </div>
31 </div>
32 </div>
33 <div class="form-row">
34 <div class="col">
35 <input type="checkbox" class="col-form-label" id="Chb" /><label class="ml-2">Recordarme</label>

```

Listado de Mantenimientos

Figura 49: Listado de Mantenimiento (Back End)

```

30 try {
31 $resultado = $GLOBALS["conexion"]->prepare($consulta);
32 $resultado->execute();
33 $data = $resultado->fetchAll(PDO::FETCH_ASSOC);
34 print json_encode($data, JSON_UNESCAPED_UNICODE);
35 return json_encode($data, JSON_UNESCAPED_UNICODE);
36 } catch (Exception $e) {
37 Procedimientos::Mensaje_Error($e->getMessage());
38 }
39 }
40
41 //procesos para obtener el privilegio del usuario que inicie sesión;
42 public static function Privilegio(string $dni)
43 {
44 try {
45 $privilegio = "Usuario";
46 $resultado = $GLOBALS["conexion"]->prepare("SELECT p.Nombre as 'Privilegio' FROM usuario u IN
47 $resultado->execute();
48 $data = $resultado->fetchAll(PDO::FETCH_ASSOC);
49 $data = json_encode($data, JSON_UNESCAPED_UNICODE);
50 $array = json_decode($data, true);
51 foreach ($array as $value) {
52 $privilegio = $value['Privilegio'];
53 }
54 return $privilegio;
55 } catch (Exception $e) {
56 Procedimientos::Mensaje_Error($e->getMessage());
57 }
58 }
59
60 public static function ObtenerCodigo(string $consulta)
61 {
62 try {
63 $codigo = "1";
64 $resultado = $GLOBALS["conexion"]->prepare($consulta);

```

CAPÍTULO V (PRUEBAS DE CALIDAD DEL SOFTWARE)

Pruebas Unitarias

Figura 50: Pruebas Unitarias de Ventas

The screenshot shows a web browser displaying the 'Ventas' page. The page includes a search bar with the date '21/07/2022' and a table of sales records. The table has columns for 'Acciones', 'Fecha', 'Producto', and 'Precio_venta'. The code editor on the right shows the JavaScript logic for the 'ventas.js' file, including functions for adding, editing, deleting, and viewing sales records.

Acciones	Fecha	Producto	Precio_venta
	2022-07-21 07:33:43	Jabón Bolívar (comercial mabry)	2.50
	2022-07-21 07:35:06	Lejía patito 500ml (comercial mabry)	2.00

```

51
52 /***** Agregar *****/
53 $(document).on("click", ".btAgregar" + Procesos.ObtenerVista(
54   opcion = 2;
55   Procesos.MostrarModal("#modal", "Nuevo Registro");
56 });
57
58 /***** Editar *****/
59 $(document).on("click", ".btEditar" + Procesos.ObtenerVista(f
60   opcion = 3;
61   var fila = $(this).closest("tr");
62   Procesos.MostrarDatosForm("#modal", fila);
63 });
64
65 /***** Submit *****/
66 $('#form').submit(function (e) {
67   e.preventDefault();
68   Procesos.Registrar(opcion);
69 });
70
71 /***** Eliminar *****/
72 $(document).on("click", ".btBorrar" + Procesos.ObtenerVista(f
73   var fila = $(this).closest("tr");
74   Procesos.Eliminar(fila);
75 });
76
77 /***** Ver *****/
78 $(document).on("click", ".btVer" + Procesos.ObtenerVista(fals
79   var fila = $(this).closest("tr");
80   var fecha = Procesos.Separar(Procesos.ObtenerValueTabla(fil
81   Procesos.MostrarModal("#modalReporte", "Reporte Producto");
82   var producto = Procesos.ObtenerValueTabla(fila, 2);
83   Procesos.setValue("#UserReporte", Procesos.getDni());
84   Procesos.setValue("#productoReporte", producto);
85   Procesos.setValue("#fechaReporte", fecha);
  
```

Figura 51: Pruebas Unitarias de Compras

The screenshot shows a web browser displaying the 'Compras' page. The page includes a search bar with the date '21/07/2022' and a table of purchase records. The table has columns for 'Acciones', 'Fecha', 'Producto', and 'Precio_compra'. The code editor on the right shows the JavaScript logic for the 'compras.js' file, including functions for adding, editing, deleting, and viewing purchase records.

Acciones	Fecha	Producto	Precio_compra
	2022-07-21 07:27:23	Jabón Bolívar (comercial mabry)	2.00
	2022-07-21 07:30:02	lejía patito de 250ml (comercial mabry)	0.90
	2022-07-21 07:31:28	Lejía patito 500ml (comercial mabry)	1.00

```

51
52 /***** Agregar *****/
53 $(document).on("click", ".btAgregar" + Procesos.ObtenerVista(
54   opcion = 2;
55   Procesos.MostrarModal("#modal", "Nuevo Registro");
56 });
57
58 /***** Editar *****/
59 $(document).on("click", ".btEditar" + Procesos.ObtenerVista(f
60   opcion = 3;
61   var fila = $(this).closest("tr");
62   Procesos.MostrarDatosForm("#modal", fila);
63 });
64
65 /***** Submit *****/
66 $('#form').submit(function (e) {
67   e.preventDefault();
68   Procesos.Registrar(opcion);
69 });
70
71 /***** Eliminar *****/
72 $(document).on("click", ".btBorrar" + Procesos.ObtenerVista(f
73   var fila = $(this).closest("tr");
74   Procesos.Eliminar(fila);
75 });
76
77 /***** Ver *****/
78 $(document).on("click", ".btVer" + Procesos.ObtenerVista(fals
79   var fila = $(this).closest("tr");
80   var fecha = Procesos.Separar(Procesos.ObtenerValueTabla(f
81   Procesos.MostrarModal("#modalReporte", "Reporte Producto"
82   var producto = Procesos.ObtenerValueTabla(fila, 2);
83   Procesos.setValue("#UserReporte", Procesos.getDni());
84   Procesos.setValue("#productoReporte", producto);
85   Procesos.setValue("#fechaReporte", fecha);
86   $("#form2").submit();
  
```

Pruebas Integrales

Figura 52: Pruebas Integrales de Reportes de compras y Ventas según la fecha.

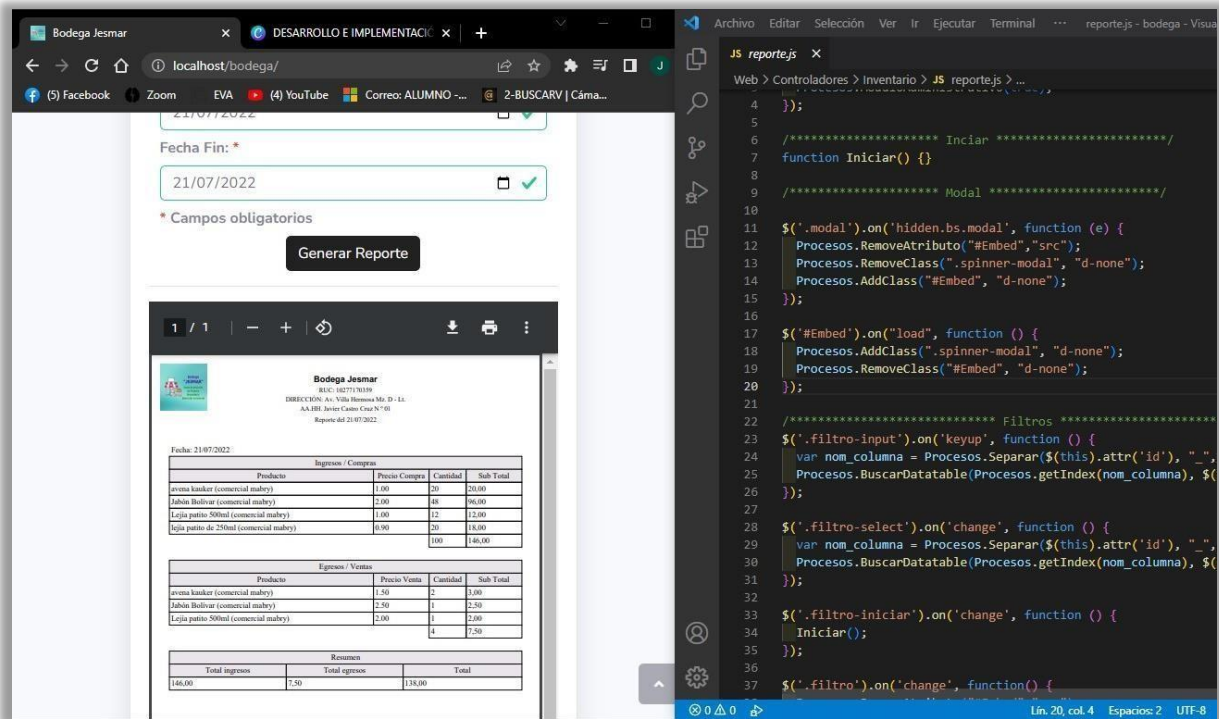
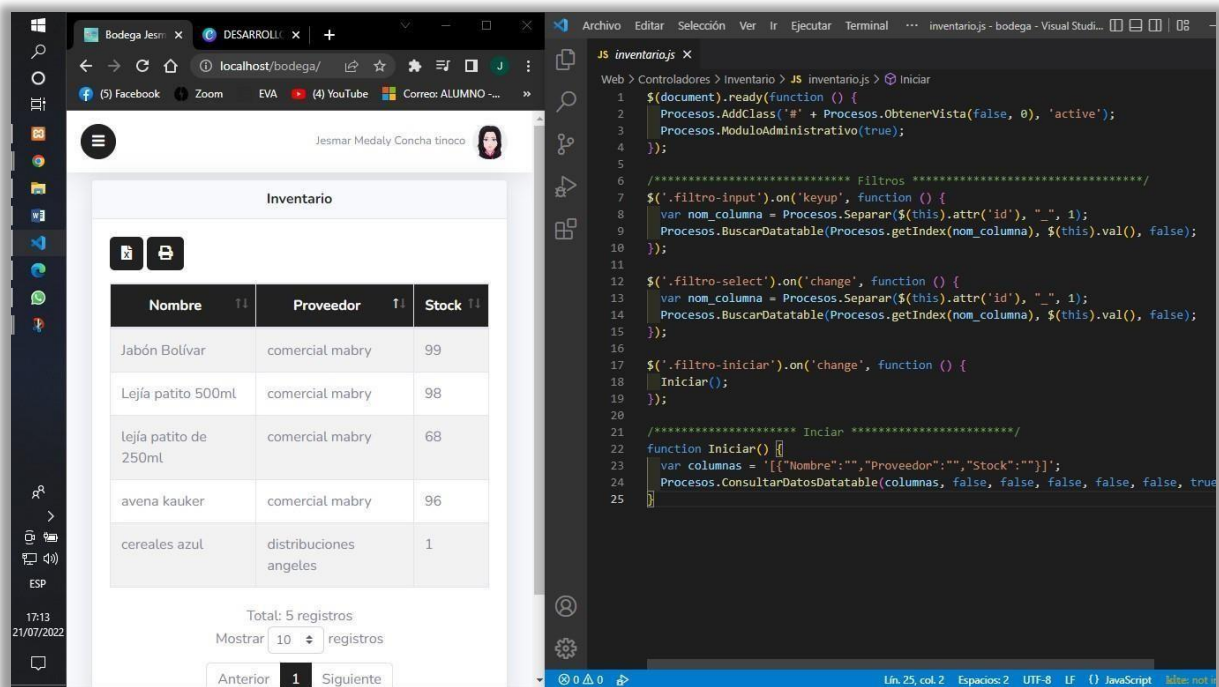


Figura 53: Pruebas Integrales de Inventario



Conclusiones

- A través de este trabajo de investigación, se ha logrado comprobar la hipótesis planteada al inicio del proyecto que nos ayudara a controlar las entradas y salidas de productos de la empresa.
- El manejo de los módulos se realiza de una manera operativa y dinámica por el cliente, ya que está adaptada al marco de trabajo que se realizó en el proceso del proyecto.
- Por lo tanto, utilizar el sistema web nos ahorraría tiempo y ayudaría a organizarnos como empresa para poder brindar un servicio de calidad.

Recomendaciones

- Debemos tener más horas de asesoramiento para realizar nuestra tesis de acuerdo al marco teórico actualmente.
- Convenimos tener conocimiento de toda la problemática de la empresa para poder realizar un buen trabajo.
- Debemos trabajar de una manera operativa con el cliente para poder tener más acceso a su problemática de la empresa y así brindarles un buen resultado como proyecto.

Referencias Bibliográficas

(S/f-a). Recuperado el 30 de mayo de 2022, de

http://file:///C:/Users/Windows%2010/Downloads/TESIS/GESTION_DE_RIESGOS_AEBR_JUE24AGO2017.pdf

(S/f-b). Recuperado el 15 de junio de 2022, de

<http://file:///C:/Users/Windows%2010/Downloads/TESIS/Manual%20para%20la%20presentaci%C3%B3n%20de%20proyectos%20profesionales%20-%20Computaci%C3%B3n%20e%20Inform%C3%A1tica%20-%20SIDET.pdf>

(S/f-c). Recuperado el 27 de junio de 2022, de

<http://file:///C:/Users/Windows%2010/Downloads/TESIS/MATERIAL%20LECTURA/SCRUMstudy-SBOK-Guide-3rd-edition-spanish.pdf>

(S/f-d). Recuperado el 7 de julio de 2022, de

http://file:///C:/Users/Windows%2010/Downloads/TESIS/INTRODUCCION%20AL%20ANALISIS%20DEL%20NEGOCIO_ABARBA_MIE25NOV2020.pdf

(S/e). Servicio de Informática. Universidad de Alicante. (s/f). *Modelo vista*

controlador (MVC). Recuperado el 12 de julio de 2022, de

<https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

(S/f). Scrummanager.net. Recuperado el 12 de julio de 2022, de

https://www.scrummanager.net/files/sm_proyecto.pdf

Anexos

Código Fuente de la Base de Datos bodega.

Figura 54: Código de la tabla privilegio

```

CREATE DATABASE Bodega;

use Bodega;

/***** Privilegio *****/
CREATE TABLE privilegio (
  Codigo int(11) NOT NULL AUTO_INCREMENT,
  Nombre varchar(40) NOT NULL,
  PRIMARY KEY (Codigo),
  UNIQUE KEY Nombre (Nombre)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

INSERT INTO privilegio VALUES (NULL,'Administrador'),(NULL,'Empleado');

/***** Estados *****/
CREATE TABLE estado (
  Codigo int(11) NOT NULL AUTO_INCREMENT,
  Nombre varchar(40) NOT NULL,
  PRIMARY KEY (Codigo),
  UNIQUE KEY Nombre (Nombre)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

INSERT INTO estado VALUES (NULL,'Activo'),(NULL,'Inactivo');

```

Figura 55: Código de la tabla Usuario

```

/***** Usuarios *****/
CREATE TABLE usuario (
  Dni char(8) NOT NULL,
  Nombre varchar(50) NOT NULL,
  Apellido varchar(50) NOT NULL,
  Correo varchar(50) NOT NULL,
  Telefono char(9) DEFAULT NULL,
  Sexo char(1) NOT NULL,
  Contraseña varchar(300) NOT NULL,
  Privilegio int(11) NOT NULL,
  Estado int(11) NOT NULL,
  PRIMARY KEY (Dni),
  KEY fk_usu_pri (Privilegio),
  KEY fk_usu_est (Estado),
  CONSTRAINT fk_usu_pri FOREIGN KEY (Privilegio) REFERENCES privilegio (Codigo),
  CONSTRAINT fk_usu_est FOREIGN KEY (Estado) REFERENCES estado (Codigo)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

INSERT INTO `usuario` VALUES ('70083473','Jesmar Medaly','Concha tinoco','','','F',
'$2y$10$hTghRDg1Uy323To1bpvDSeFU1vX./ueJ7CuC3it.xu.ip11TPoSLu',1,1);/*jesmarcita*/

```

Figura 56: Código de Procedimientos

```

/***** Procedimientos *****/
DROP PROCEDURE IF EXISTS Usuarios;
DELIMITER ;;
CREATE PROCEDURE Usuarios(in Dni varchar(8))
BEGIN
IF (Dni = "") THEN
    SELECT u.Dni,u.Nombre,u.Apellido,u.Correo,u.Telefono,u.Sexo,p.Nombre as 'Privilegio',e.Nombre as
    'Estado' FROM usuario u INNER JOIN privilegio p on u.Privilegio = p.Codigo INNER JOIN estado e
    WHERE e.Codigo = u.Estado;
ELSE
    SELECT u.Dni,u.Nombre,u.Apellido,u.Correo,u.Telefono,u.Sexo,p.Nombre as 'Privilegio',
    e.Nombre as 'Estado' FROM usuario u INNER JOIN privilegio p on u.Privilegio = p.Codigo
    INNER JOIN estado e on e.Codigo = u.Estado WHERE u.Dni=Dni;
END IF;
END ;;
DELIMITER ;

```

Figura 57: Código de data para obtener e insertar información

```

/***** DATA DATA *****/
DROP PROCEDURE IF EXISTS Obtener_Data;
DELIMITER ;;
CREATE PROCEDURE Obtener_Data(in tabla varchar(40))
BEGIN
DECLARE Nom_base VARCHAR(50);
SET Nom_base = CONCAT("bodega.",tabla);
SET @consulta = CONCAT("SELECT * FROM ",Nom_base);
PREPARE stmt FROM @consulta;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;

DROP PROCEDURE IF EXISTS Insertar_Data;
DELIMITER ;;
CREATE PROCEDURE Insertar_Data(
in Tabla varchar(40),
in Datos varchar(300))
BEGIN
DECLARE Nom_base VARCHAR(50);
SET Nom_base = CONCAT("bodega.",Tabla);
SET @consulta = CONCAT("INSERT INTO ",Nom_base," VALUES (",Datos,")");
PREPARE stmt FROM @consulta;
EXECUTE stmt;
SELECT ROW_COUNT() INTO @Row_Count;
IF @Row_Count <> 0 THEN
    SELECT 'registro';
ELSE
    SELECT '!Error!, los datos no se registraron';
END IF;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;

```

Figura 58: Código de data para modificar y eliminar data

```

DROP PROCEDURE IF EXISTS Modificar_Data;
DELIMITER ;;
CREATE PROCEDURE Modificar_Data(
in Tabla varchar(40),
in Datos varchar(300))
BEGIN
DECLARE Nom_base VARCHAR(50);
SET Nom_base = CONCAT("bodega.",Tabla);
SET @consulta = CONCAT("UPDATE ",Nom_base," SET ", Datos);
PREPARE stmt FROM @consulta;
EXECUTE stmt;
SELECT 'actualizo';
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;

DROP PROCEDURE IF EXISTS Eliminar_Data;
DELIMITER ;;
CREATE PROCEDURE Eliminar_Data(
in Cod varchar(200),
in tabla varchar(40),
in Columna varchar(20))
BEGIN
DECLARE Nom_base VARCHAR(50);
SET Nom_base = CONCAT("bodega.",tabla);
SET @consulta = CONCAT("DELETE FROM ",Nom_base," WHERE ",Columna,"='",Cod,"'");
PREPARE stmt FROM @consulta;
EXECUTE stmt;
SELECT ROW_COUNT() INTO @Row_Count;
IF @Row_Count <> 0 THEN
    SELECT 'elimino';
ELSE
    SELECT '!Error!, los datos no se eliminaron';
END IF;
DEALLOCATE PREPARE stmt;
END ;;
DELIMITER ;

```

Figura 59: Código de la tabla Proveedor

```

/***** Proveedor *****/
CREATE TABLE proveedor (
Codigo int(11) NOT NULL AUTO_INCREMENT,
Nombre varchar(50) NOT NULL,
Telefono char(9) DEFAULT NULL,
Estado int(11) NOT NULL,
PRIMARY KEY (Codigo),
UNIQUE KEY Nombre (Nombre),
KEY fk_prov_est (Estado),
CONSTRAINT fk_prov_est FOREIGN KEY (Estado) REFERENCES estado (Codigo)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

DROP PROCEDURE IF EXISTS Proveedores;
DELIMITER ;;
CREATE PROCEDURE Proveedores(in estado int)
BEGIN
IF (estado = 1) THEN
    SELECT p.Codigo,p.Nombre,p.Telefono,e.Nombre as 'Estado' FROM Proveedor p INNER JOIN estado e on
    e.Codigo = p.Estado;
ELSE
    SELECT p.Codigo,p.Nombre,p.Telefono,e.Nombre as 'Estado' FROM Proveedor p INNER JOIN estado e on
    e.Codigo = p.Estado WHERE p.Estado NOT IN ('2');
END IF;

END ;;
DELIMITER ;

```

Figura 60: Código de la tabla Categoría y sus procedures

```

/***** Categorías *****/
CREATE TABLE categoria (
  Codigo int(11) NOT NULL AUTO_INCREMENT,
  Nombre varchar(50) NOT NULL,
  Estado int(11) NOT NULL,
  PRIMARY KEY (Codigo),
  CONSTRAINT fk_cat_est FOREIGN KEY (Estado) REFERENCES estado (Codigo)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

DROP PROCEDURE IF EXISTS Categorías;
DELIMITER ;;
CREATE PROCEDURE Categorías(in estado int)
BEGIN
  IF (estado = 1) THEN
    SELECT c.Codigo,c.Nombre,e.Nombre as 'Estado' FROM categoria c inner JOIN Estado e on e.Codigo = c.Estado;
  ELSE
    SELECT c.Codigo,c.Nombre,e.Nombre as 'Estado' FROM categoria c inner JOIN Estado e on
    e.Codigo = c.Estado WHERE c.Estado NOT IN ('2');
  END IF;
END ;;
DELIMITER ;

DROP PROCEDURE IF EXISTS Categorías;
DELIMITER ;;
CREATE PROCEDURE Categorías(in estado int)
BEGIN
  IF (estado = 1) THEN
    SELECT c.Codigo,c.Nombre,e.Nombre as 'Estado' FROM categoria c inner JOIN Estado e on e.Codigo = c.Estado;
  ELSE
    SELECT c.Codigo,c.Nombre,e.Nombre as 'Estado' FROM categoria c inner JOIN Estado e on
    e.Codigo = c.Estado WHERE c.Estado NOT IN ('2');
  END IF;
END ;;
DELIMITER ;

```

Figura 61: Código de la tabla Productos y sus procedimientos

```

/***** Productos *****/
CREATE TABLE producto (
  Codigo int(11) NOT NULL AUTO_INCREMENT,
  Nombre varchar(50) NOT NULL,
  Proveedor int(11) NOT NULL,
  Categoría int(11) NOT NULL,
  Estado int(11) NOT NULL,
  PRIMARY KEY (Codigo),
  CONSTRAINT fk_prod_prov FOREIGN KEY (Proveedor) REFERENCES proveedor (Codigo),
  CONSTRAINT fk_cat_prod FOREIGN KEY (Categoría) REFERENCES categoría (Codigo),
  CONSTRAINT fk_prod_est FOREIGN KEY (Estado) REFERENCES estado (Codigo)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/***** Procedimientos *****/

DROP PROCEDURE IF EXISTS Productos;
DELIMITER ;;
CREATE PROCEDURE Productos(in estado int)
BEGIN
  IF (estado = 1) THEN
    SELECT p.Codigo,p.Nombre,pv.Nombre as 'Proveedor',c.Nombre as 'Categoría',e.Nombre as 'Estado'
    FROM producto p INNER JOIN Proveedor pv on pv.Codigo = p.Proveedor INNER JOIN estado e on
    e.Codigo = p.Estado INNER JOIN categoría c on c.Codigo = p.Categoría;
  ELSE
    SELECT p.Codigo,p.Nombre,pv.Nombre as 'Proveedor',c.Nombre as 'Categoría',e.Nombre as 'Estado'
    FROM producto p INNER JOIN Proveedor pv on pv.Codigo = p.Proveedor INNER JOIN estado e on
    e.Codigo = p.Estado INNER JOIN categoría c on c.Codigo = p.Categoría WHERE p.Estado NOT IN ('2');
  END IF;
END ;;
DELIMITER ;

```

Figura 62: Código de la tabla Compra y sus procedimientos

```

/***** Compra / entrada (productos) *****/
CREATE TABLE compra (
  Fecha_hora datetime NOT NULL,
  Producto int(11) NOT NULL,
  Precio_compra decimal(10,2) NOT NULL,
  Cantidad int(11) NOT NULL,
  PRIMARY KEY (Fecha_hora),
  KEY fk_com_prod (Producto),
  CONSTRAINT fk_com_prod FOREIGN KEY (Producto) REFERENCES producto (Codigo)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/***** Procedimientos *****/

-- Obtiene todos las compras (entradas)
DROP PROCEDURE IF EXISTS Compras;
DELIMITER ;;
CREATE PROCEDURE Compras(in fecha date)
BEGIN
  IF (fecha = "") THEN
    SELECT c.Fecha_hora as 'Fecha',CONCAT(p.Nombre,' (' ,prov.Nombre,')') as 'Producto',c.Precio_compra,
    SUM(c.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(c.cantidad * c.Precio_compra),0),2,'de_DE') as
    'Sub Total' FROM compra c INNER JOIN Producto p on p.Codigo = c.Producto INNER JOIN proveedor prov
    on prov.Codigo = p.Proveedor WHERE DATE(c.Fecha_hora) = DATE(NOW()) GROUP BY 2,3;
  ELSE
    SELECT c.Fecha_hora as 'Fecha',CONCAT(p.Nombre,' (' ,prov.Nombre,')') as 'Producto',c.Precio_compra,
    SUM(c.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(c.cantidad * c.Precio_compra),0),2,'de_DE') as
    'Sub Total' FROM compra c INNER JOIN Producto p on p.Codigo = c.Producto INNER JOIN proveedor prov
    on prov.Codigo = p.Proveedor WHERE DATE(c.Fecha_hora) = fecha GROUP BY 2, 3;
  END IF;
END ;;
DELIMITER ;

```

Figura 63: Código del procedimiento total de compras

```

-- Obtiene el total de compras
DROP PROCEDURE IF EXISTS Total_compras;
DELIMITER ;;
CREATE PROCEDURE Total_compras(in fecha date)
BEGIN
  IF (fecha = "") THEN
    SELECT SUM(c.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(c.Precio_compra * c.Cantidad),0),2,'de_DE')
    as 'Total' FROM compra c WHERE DATE(c.Fecha_hora) = DATE(now());
  ELSE
    SELECT SUM(c.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(c.Precio_compra * c.Cantidad),0),2,'de_DE')
    as 'Total' FROM compra c WHERE DATE(c.Fecha_hora) = fecha;
  END IF;
END ;;
DELIMITER ;

```

Figura 64: Código del procedimiento modificar compras

```

-- Modificar Compra
DROP PROCEDURE IF EXISTS Modificar_compra;
DELIMITER ;;
CREATE PROCEDURE Modificar_compra(
in Fecha datetime,
in Prod varchar(50),
in Prov varchar(50),
in Precio decimal(10,2),
in Cant int(11))
BEGIN
Declare CodProducto int;
Declare CodProveedor int;
Declare Cant_compra int;
Declare Cant_venta int;
Declare Cantd int;
set CodProveedor = (SELECT p.Codigo FROM Proveedor p WHERE p.Nombre=Prov);
set CodProducto = (SELECT p.Codigo FROM Producto p WHERE p.Nombre=Prod and p.Proveedor=CodProveedor);
set Cant_compra = (SELECT IF(SUM(c.Cantidad) IS NULL,0,SUM(c.Cantidad)) FROM compra c where
DATE(c.Fecha_hora) = DATE(NOW()));
set Cant_venta = (SELECT IF(SUM(v.Cantidad) IS NULL,0,SUM(v.Cantidad)) FROM venta v where
DATE(v.Fecha_hora) = DATE(NOW()) and v.Fecha_hora not in (Fecha));
set Cant_venta = (SELECT SUM(Cant_venta + Cant));
set Cantd = (SELECT SUM(Cant_compra - Cant_venta));
IF (Cantd < 0) THEN
  SELECT '!Error!, no tiene suficiente stock para realizar la venta';
ELSE
  UPDATE compra c SET c.Cantidad=Cant WHERE c.Fecha_hora=Fecha;
  DELETE FROM compra WHERE DATE.Fecha_hora=DATE(Fecha) and Producto=CodProducto and
Precio_compra=Precio and Fecha_hora NOT IN (Fecha);
END IF;
SELECT 'actualizo';
END ;;
DELIMITER ;

```

Figura 65: Código del procedimiento eliminar compras

```

-- Eliminar Compra
DROP PROCEDURE IF EXISTS Eliminar_compra;
DELIMITER ;;
CREATE PROCEDURE Eliminar_compra(in Fecha datetime)
BEGIN
Declare CodProducto int;
DECLARE Precio decimal(10,2);
set CodProducto = (SELECT c.Producto FROM compra c WHERE c.Fecha_hora=Fecha);
set Precio = (SELECT c.Precio_compra FROM compra c WHERE c.Fecha_hora=Fecha);
DELETE FROM compra WHERE DATE.Fecha_hora=DATE(Fecha) and Producto=CodProducto and Precio_compra=Precio;
IF ROW_COUNT() <> 0 THEN
  SELECT 'elimino';
ELSE
  SELECT '!Error!, los datos no se eliminaron';
END IF;
END ;;
DELIMITER ;

```

Figura 66: Código del procedimiento generar reportes de compras

```

DROP PROCEDURE IF EXISTS Compras_reporte;
DELIMITER $$
CREATE PROCEDURE Compras_reporte(
in fecha date,
in producto varchar(50),
in proveedor varchar(50))
BEGIN
DECLARE CodProducto int;
set CodProducto = (SELECT p.Codigo FROM producto p INNER JOIN proveedor pv on p.Proveedor = pv.Codigo
WHERE p.Nombre=producto and pv.Nombre=proveedor);
SELECT v.Fecha_hora as 'Fecha',CONCAT(p.Nombre,' (' ,prov.Nombre,')') as 'Producto',v.Precio_compra,
v.Cantidad as 'Cantidad',FORMAT(COALESCE(SUM(v.cantidad * v.Precio_compra),0),2,'de_DE') as 'Sub Total'
FROM compra v INNER JOIN Producto p on p.Codigo = v.Producto INNER JOIN proveedor prov on
prov.Codigo = p.Proveedor WHERE DATE(v.Fecha_hora) = fecha and v.Producto=CodProducto GROUP BY v.Fecha_hora;
END$$
DELIMITER ;

DROP PROCEDURE IF EXISTS Total_reporte_compra;
DELIMITER ;;
CREATE PROCEDURE Total_reporte_compra(
in fecha date,
in producto varchar(50),
in proveedor varchar(50))
BEGIN
DECLARE CodProducto int;
set CodProducto = (SELECT p.Codigo FROM producto p INNER JOIN proveedor pv on p.Proveedor = pv.Codigo
WHERE p.Nombre=producto and pv.Nombre=proveedor);
SELECT SUM(v.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(v.Precio_compra * v.Cantidad),0),2,'de_DE') as
'Total' FROM Compra v WHERE DATE(v.Fecha_hora) = fecha and Producto=CodProducto;
END ;;
DELIMITER ;

```

Figura 67: Código de la venta y sus procedimientos

```

/***** Venta / salida (productos) *****/
CREATE TABLE venta (
  Fecha_hora datetime NOT NULL,
  Producto int(11) NOT NULL,
  Precio_venta decimal(10,2) NOT NULL,
  Cantidad int(11) NOT NULL,
  PRIMARY KEY (Fecha_hora),
  KEY fk_venta_prod (Producto),
  CONSTRAINT fk_venta_prod FOREIGN KEY (Producto) REFERENCES producto (Codigo)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/***** Procedimientos *****/

-- Obtiene todas las ventas del día
DROP PROCEDURE IF EXISTS Ventas;
DELIMITER $$
CREATE PROCEDURE Ventas(in fecha date)
BEGIN
IF (fecha = "") THEN
SELECT v.Fecha_hora as 'Fecha',CONCAT(p.Nombre,' (' ,prov.Nombre,')') as 'Producto',v.Precio_venta,
SUM(v.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(v.cantidad * v.Precio_venta),0),2,'de_DE') as
'Sub Total' FROM venta v INNER JOIN Producto p on p.Codigo = v.Producto INNER JOIN proveedor prov
on prov.Codigo = p.Proveedor WHERE DATE(v.Fecha_hora) = DATE(NOW()) GROUP BY 2,3;
ELSE
SELECT v.Fecha_hora as 'Fecha',CONCAT(p.Nombre,' (' ,prov.Nombre,')') as 'Producto',v.Precio_venta,
SUM(v.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(v.cantidad * v.Precio_venta),0),2,'de_DE') as
'Sub Total' FROM venta v INNER JOIN Producto p on p.Codigo = v.Producto INNER JOIN proveedor prov
on prov.Codigo = p.Proveedor WHERE DATE(v.Fecha_hora) = fecha GROUP BY 2,3;
END IF;
END$$
DELIMITER ;

```

Figura 68: Código de procedimientos de generar reporte de ventas

```

DROP PROCEDURE IF EXISTS Ventas_reporte;
DELIMITER $$
CREATE PROCEDURE Ventas_reporte(
in fecha date,
in producto varchar(50),
in proveedor varchar(50))
BEGIN
DECLARE CodProducto int;
set CodProducto = (SELECT p.Codigo FROM producto p INNER JOIN proveedor pv on p.Proveedor = pv.Codigo
WHERE p.Nombre=producto and pv.Nombre=proveedor);
SELECT v.Fecha_hora as 'Fecha',CONCAT(p.Nombre,' (' ,prov.Nombre,')') as 'Producto',v.Precio_venta,v.Cantidad
as 'Cantidad',FORMAT(COALESCE(SUM(v.cantidad * v.Precio_venta),0),2,'de_DE') as 'Sub Total' FROM venta v
INNER JOIN Producto p on p.Codigo = v.Producto INNER JOIN proveedor prov on prov.Codigo = p.Proveedor
WHERE DATE(v.Fecha_hora) = fecha and v.Producto=CodProducto GROUP BY v.Fecha_hora;
END$$
DELIMITER ;

-- Obtiene el total de ventas
DROP PROCEDURE IF EXISTS Total_ventas;
DELIMITER ;;
CREATE PROCEDURE Total_ventas(in fecha date)
BEGIN
IF (fecha = "") THEN
SELECT SUM(v.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(v.Precio_venta * v.Cantidad),0),2,'de_DE') as
'Total' FROM Venta v WHERE DATE(v.Fecha_hora) = DATE(now());
ELSE
SELECT SUM(v.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(v.Precio_venta * v.Cantidad),0),2,'de_DE') as
'Total' FROM Venta v WHERE DATE(v.Fecha_hora) = fecha;
END IF;
END ;;
DELIMITER ;

```

Figura 69: Código de procedimientos de total de reporte de ventas

```

DROP PROCEDURE IF EXISTS Total_reporte_venta;
DELIMITER ;;
CREATE PROCEDURE Total_reporte_venta(
in fecha date,
in producto varchar(50),
in proveedor varchar(50))
BEGIN
DECLARE CodProducto int;
set CodProducto = (SELECT p.Codigo FROM producto p INNER JOIN proveedor pv on p.Proveedor = pv.Codigo
WHERE p.Nombre=producto and pv.Nombre=proveedor);
SELECT SUM(v.Cantidad) as 'Cantidad',FORMAT(COALESCE(SUM(v.Precio_venta * v.Cantidad),0),2,'de_DE') as
'Total' FROM Venta v WHERE DATE(v.Fecha_hora) = fecha and Producto=CodProducto;
END ;;
DELIMITER ;

```

Figura 70: Código de procedimientos de modificar ventas

```

-- Modificar Venta
DROP PROCEDURE IF EXISTS Modificar_venta;
DELIMITER ;;
CREATE PROCEDURE Modificar_venta(
in Fecha datetime,
in Prod varchar(50),
in Prov varchar(50),
in Precio decimal(10,2),
in Cant int(11))
BEGIN
Declare CodProducto int;
Declare CodProveedor int;
Declare Cant_compra int;
Declare Cant_venta int;
Declare Cantd int;
set CodProveedor = (SELECT p.Codigo FROM Proveedor p WHERE p.Nombre=Prov);
set CodProducto = (SELECT p.Codigo FROM Producto p WHERE p.Nombre=Prod and p.Proveedor=CodProveedor);
set Cant_compra = (SELECT IF(SUM(c.Cantidad) IS NULL,0,SUM(c.Cantidad)) FROM compra c where
DATE(c.Fecha_hora) = DATE(NOW()));
set Cant_venta = (SELECT IF(SUM(v.Cantidad) IS NULL,0,SUM(v.Cantidad)) FROM venta v where
DATE(v.Fecha_hora) = DATE(NOW()) and v.Fecha_hora not in (Fecha));
set Cant_venta = (SELECT SUM(Cant_venta + Cant));
set Cantd = (SELECT SUM(Cant_compra - Cant_venta));
IF (Cantd < 0) THEN
SELECT '!Error!, no tiene suficiente stock para realizar la venta';
ELSE
UPDATE venta v SET v.Cantidad=Cant WHERE v.Fecha_hora=Fecha;
DELETE FROM venta WHERE DATE.Fecha_hora=DATE(Fecha) and Producto=CodProducto and
Precio_venta=Precio and Fecha_hora NOT IN (Fecha);
END IF;
SELECT 'actualizo';
END ;;
DELIMITER ;

```

Figura 71: Código de procedimientos de eliminar venta

```

-- Eliminar venta
DROP PROCEDURE IF EXISTS Eliminar_venta;
DELIMITER ;;
CREATE PROCEDURE Eliminar_venta(in Fecha datetime)
BEGIN
Declare CodProducto int;
DECLARE Precio decimal(10,2);
set CodProducto = (SELECT v.Producto FROM venta v WHERE v.Fecha_hora=Fecha);
set Precio = (SELECT v.Precio_venta FROM venta v WHERE v.Fecha_hora=Fecha);
DELETE FROM venta WHERE DATE.Fecha_hora=DATE(Fecha) and Producto=CodProducto and Precio_venta=Precio;
IF ROW_COUNT() <> 0 THEN
SELECT 'elimino';
ELSE
SELECT '!Error!, los datos no se eliminaron';
END IF;
END ;;
DELIMITER ;

```

Figura 72: Código de tabla de inventario

```

/***** Inventario *****/
CREATE TABLE inventario (
  Codigo int(11) NOT NULL AUTO_INCREMENT,
  Producto int(11) NOT NULL,
  Stock int(11) NOT NULL,
  PRIMARY KEY (Codigo),
  KEY fk_inv_prod (Producto),
  CONSTRAINT fk_inv_prod FOREIGN KEY (Producto) REFERENCES producto (Codigo)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Obtiene todos stock de productos
DROP PROCEDURE IF EXISTS Inventario;
DELIMITER ;;
CREATE PROCEDURE Inventario()
BEGIN
SELECT p.Nombre,prov.Nombre as 'Proveedor',i.Stock FROM inventario i INNER JOIN
Producto p on i.Producto = p.Codigo INNER JOIN proveedor prov on p.Proveedor = prov.Codigo
WHERE i.Stock NOT IN ('0');
END ;;
DELIMITER ;

```

Figura 73: Código de procedimiento obtener el total de las compras y ventas

```

-- Obtiene el total
DROP PROCEDURE IF EXISTS Total;
DELIMITER $$
CREATE PROCEDURE Total(in fecha date)
BEGIN
Declare ingresos int;
Declare egresos int;
set ingresos = (SELECT IFNULL(SUM(c.Precio_compra * c.Cantidad),0) FROM compra c WHERE
DATE(c.Fecha_hora) = fecha);
set egresos = (SELECT IFNULL(SUM(v.Precio_venta * v.Cantidad),0) FROM venta v WHERE
DATE(v.Fecha_hora) = fecha);
SELECT FORMAT(COALESCE(SUM(ingresos - egresos),0),2,'de_DE') as 'Total';
END$$
DELIMITER ;

-- obtiene el resumen de las compras, ventas, ganancias, total de empleados
DROP PROCEDURE IF EXISTS Resumen;
DELIMITER $$
CREATE PROCEDURE Resumen()
BEGIN
Declare ingresos int;
Declare egresos int;
Declare empleados int;
set ingresos = (SELECT IFNULL(SUM(c.Precio_compra * c.Cantidad),0) FROM compra c WHERE
DATE(c.Fecha_hora) = DATE(NOW()));
set egresos = (SELECT IFNULL(SUM(v.Precio_venta * v.Cantidad),0) FROM venta v WHERE
DATE(v.Fecha_hora) = DATE(NOW()));
set empleados = (SELECT COUNT(*) FROM usuario u WHERE u.Privilegio = '2');
SELECT FORMAT(COALESCE(ingresos,0),2,'de_DE') as 'Ingresos',FORMAT(COALESCE(egresos,0),2,'de_DE') as 'Egresos',
FORMAT(COALESCE(SUM(ingresos - egresos),0),2,'de_DE') as 'Total',empleados as 'Empleados';
END$$
DELIMITER ;

```

Figura 74: Código de Triggers Compras

```

/***** Triggers (Compras) *****/
DROP TRIGGER IF EXISTS Insertar_inv_compra;
DELIMITER ;;
CREATE TRIGGER Insertar_inv_compra AFTER INSERT ON Compra FOR EACH ROW
BEGIN
DECLARE Cant int;
DECLARE Cont int;
DECLARE Cant_compra int;
DECLARE Cant_venta int;
set Cont = (SELECT COUNT(*) FROM inventario i where i.Producto=NEW.Producto);
set Cant_compra = (SELECT IF(SUM(c.Cantidad) IS NULL,0,SUM(c.Cantidad)) FROM compra c where
DATE(c.Fecha_hora) = DATE(NOW()));
set Cant_venta = (SELECT IF(SUM(v.Cantidad) IS NULL,0,SUM(v.Cantidad)) FROM venta v where
DATE(v.Fecha_hora) = DATE(NOW()));
set Cant = (SELECT SUM(Cant_compra - Cant_venta));
IF (Cont = 0) THEN
    INSERT INTO inventario VALUES(NULL,NEW.Producto,Cant);
ELSE
    UPDATE inventario i SET i.Stock=Cant WHERE i.Producto=NEW.Producto;
END IF;
END ;;

```

Figura 75: Código de Triggers Actualizar Compras

```

DROP TRIGGER IF EXISTS Actualizar_inv_compra;
DELIMITER ;;
CREATE TRIGGER Actualizar_inv_compra AFTER UPDATE ON Compra FOR EACH ROW
BEGIN
DECLARE Cant int;
DECLARE Cant_compra int;
DECLARE Cant_venta int;
set Cant_compra = (SELECT IF(SUM(c.Cantidad) IS NULL,0,SUM(c.Cantidad)) FROM compra c where
DATE(c.Fecha_hora) = DATE(NOW()));
set Cant_venta = (SELECT IF(SUM(v.Cantidad) IS NULL,0,SUM(v.Cantidad)) FROM venta v where
DATE(v.Fecha_hora) = DATE(NOW()));
set Cant = (SELECT SUM(Cant_compra - Cant_venta));
UPDATE inventario i SET i.Stock=Cant WHERE i.Producto=old.Producto;
END ;;

DROP TRIGGER IF EXISTS Eliminar_inv_venta;
DELIMITER ;;
CREATE TRIGGER Eliminar_inv_venta AFTER DELETE ON Venta FOR EACH ROW
BEGIN
Declare Cant int;
set Cant = (SELECT SUM(i.Stock + old.Cantidad) FROM inventario i WHERE i.Producto=old.Producto);
UPDATE inventario i SET i.Stock=Cant WHERE i.Producto=old.Producto;
END ;;

```

Figura 76: Código de Triggers Insertar datos Ventas

```

/***** Triggers (Ventas) *****/
DROP TRIGGER IF EXISTS Insertar_inv_venta;
DELIMITER ;;
CREATE TRIGGER Insertar_inv_venta AFTER INSERT ON Venta FOR EACH ROW
BEGIN
DECLARE Cant int;
DECLARE Cont int;
DECLARE Cant_compra int;
DECLARE Cant_venta int;
set Cant = (SELECT COUNT(*) FROM inventario i where i.Producto=NEW.Producto);
set Cant_compra = (SELECT IF(SUM(c.Cantidad) IS NULL,0,SUM(c.Cantidad)) FROM compra c where
DATE(c.Fecha_hora) = DATE(NOW()));
set Cant_venta = (SELECT IF(SUM(v.Cantidad) IS NULL,0,SUM(v.Cantidad)) FROM venta v where
DATE(v.Fecha_hora) = DATE(NOW()));
set Cant = (SELECT SUM(Cant_compra - Cant_venta));
IF (Cant = 0) THEN
INSERT INTO inventario VALUES(NULL,NEW.Producto,Cant);
ELSE
UPDATE inventario i SET i.Stock=Cant WHERE i.Producto=NEW.Producto;
END IF;
END ;;

```

Figura 77: Código de Triggers Actualizar y eliminar Ventas

```

DROP TRIGGER IF EXISTS Actualizar_inv_venta;
DELIMITER ;;
CREATE TRIGGER Actualizar_inv_venta AFTER UPDATE ON Venta FOR EACH ROW
BEGIN
DECLARE Cant int;
DECLARE Cant_compra int;
DECLARE Cant_venta int;
set Cant_compra = (SELECT IF(SUM(c.Cantidad) IS NULL,0,SUM(c.Cantidad)) FROM compra c where
DATE(c.Fecha_hora) = DATE(NOW()));
set Cant_venta = (SELECT IF(SUM(v.Cantidad) IS NULL,0,SUM(v.Cantidad)) FROM venta v where
DATE(v.Fecha_hora) = DATE(NOW()));
set Cant = (SELECT SUM(Cant_compra - Cant_venta));
UPDATE inventario i SET i.Stock=Cant WHERE i.Producto=old.Producto;
END ;;

DROP TRIGGER IF EXISTS Eliminar_inv_compra;
DELIMITER ;;
CREATE TRIGGER Eliminar_inv_compra AFTER DELETE ON Compra FOR EACH ROW
BEGIN
DECLARE Cant int;
DECLARE Cont int;
DECLARE Cant_compra int;
DECLARE Cant_venta int;
set Cant = (SELECT COUNT(*) FROM inventario i where i.Producto=old.Producto);
set Cant_compra = (SELECT IF(SUM(c.Cantidad) IS NULL,0,SUM(c.Cantidad)) FROM compra c where
DATE(c.Fecha_hora) = DATE(NOW()));
set Cant_venta = (SELECT IF(SUM(v.Cantidad) IS NULL,0,SUM(v.Cantidad)) FROM venta v where
DATE(v.Fecha_hora) = DATE(NOW()));
set Cant = (SELECT SUM(Cant_compra - Cant_venta));
IF (Cant = 0) THEN
INSERT INTO inventario VALUES(NULL,old.Producto,Cant);
ELSE
UPDATE inventario i SET i.Stock=Cant WHERE i.Producto=old.Producto;
END IF;
END ;;

```