

UNIVERSIDAD PRIVADA ANTENOR ORREGO

FACULTAD DE INGENIERÍA

PROGRAMA DE ESTUDIO DE INGENIERÍA INDUSTRIAL



TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO INDUSTRIAL

Desarrollo de un Dashboard como herramienta de monitoreo para optimizar el sistema de aprovisionamiento de la Distribuidora Ferbanz S.A.C

Línea de Investigación: Diseño, Manufactura y Mecanización

Sub Línea de Investigación: Gestión empresarial

Autor:

De Mendiburu Osorio, Luis Felipe

Jurado evaluador:

Presidente : Castillo Torres, Luciano

Secretario : Espinoza Raymundo, Marco Antonio

Vocal : Seminario Vásquez, Ricardo Gerónimo

Asesor:

Ludeña Gutiérrez, Alfredo Lázaro

Código Orcid: <https://orcid.org/0000-0001-5674-5886>

Piura-Perú

2023

Fecha de Sustentación: 17/11/23

El Desarrollo de un Dashboard como herramienta de monitoreo para optimizar el sistema de aprovisionamiento de la Distribuidora Ferbanz S.A.C

INFORME DE ORIGINALIDAD

3%	3%	0%	%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL

ESTUDIANTE

FUENTES PRIMARIAS

 **vsip.info**
Fuente de Internet



Chy
Ing. Alfredo Lázaro Ludeña Gutiérrez
Dr. Ingeniero Industrial
CIP: 38159
ID: 000154095

3%

Excluir citas Activo
Excluir bibliografía Activo

Excluir coincidencias < 3%

Declaración de Originalidad

Yo, **Alfredo Ludeña Gutiérrez** docente del Programa de Estudio de Ingeniería Industrial de la Universidad Privada Antenor Orrego, asesor de la tesis de investigación titulada: "Desarrollo de un Dashboard como herramienta de monitoreo para optimizar el sistema de reaprovisionamiento de la Distribuidora Ferbanz S.A.C", autor Luis Felipe De Mendiburu Osorio, dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 3%. Así lo consigna el reporte de similitud emitido por el software Turnitin el (03/11/2023).
- He revisado con detalle dicho reporte y la tesis y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las normas establecidas por la Universidad.

Lugar y fecha: 17 de Noviembre del 2023

Asesor: Alfredo Ludeña Gutiérrez

DNI: 07557252

ORCID <https://orcid.org/0000-0001-5674-5886>



Luis Felipe De Mendiburu Osorio

DNI: 43076624



UNIVERSIDAD PRIVADA ANTENOR ORREGO

FACULTAD DE INGENIERIA

PROGRAMA DE ESTUDIO DE INGENIERÍA INDUSTRIAL



DESARROLLO DE UN DASHBOARD COMO HERRAMIENTA DE
MONITOREO PARA OPTIMIZAR EL SISTEMA DE
REAPROVISIONAMIENTO DE LA DISTRIBUIDORA FERBANZ S.A.C

APROBADA EN CONTENIDO Y ESTILO POR

.....
.....
Castillo Torres Luciano PRESIDENTE
CIP: 36854

.....
.....
Espinoza Raymundo Marco Antonio
SECRETARIO
CIP:97122

.....
.....
Seminario Vásquez Ricardo Gerónimo
VOCAL
CIP: 98876

.....
.....
Alfredo Lázaro Ludeña Gutiérrez
ASESOR
CIP: 38159

DEDICATORIA

Con mucho cariño, le dedico a mis padres, Felipe De Mendiburu Delgado y Ruby María Osorio Osorio, y mi novia Nubia Elena Romero Moreno; quienes fueron mi apoyo y motivación, creyendo firmemente en mi progreso y capacidad, siempre con mucho amor y confiando plenamente en mi éxito.

De Mendiburu Osorio Luis Felipe

AGRADECIMIENTO

Con mucha gratitud quiero agradecer a mi casa de estudios, está la Universidad Privada Antenor Orrego, quien me abrió las puertas y me dio la gran oportunidad de desarrollar esta hermosa carrera de Ingeniería Industrial, junto a un grupo excelente de docentes como área administrativa, un acogedor ambiente de estudios, y a grandes amigos que sumaron tanto para lograr dar más pasos seguros en cada materia.

De igual manera, quiero agradecer a mi asesor el Dr. Alfredo Ludeña Gutiérrez, quien fue además de un gran apoyo para mi investigación una mano amiga en todo el tiempo de la vida universitaria, siempre impulsando a lograr las metas.

Por último, y con un gran recuerdo quiero agradecer a quien no solo fue un amigo, compañero de estudios, sino un hermano, al Ing. Javier Segundo Silva Vidal que Dios guardé en toda su gloria y a toda su familia que con mucha calidez me extendió una mano muy fraternal, que encuentren la paz en sus corazones.

De Mendiburu Osorio Luis Felipe

RESUMEN

El presente trabajo de investigación tuvo como principal objetivo desarrollar un Dashboard como herramienta de monitoreo para optimizar el sistema de reaprovisionamiento de la DISTRIBUIDORA FERBANZ S.A.C.

Para el desarrollo, se obtuvo información de los datos de demanda de los últimos años, posteriormente después de evaluarlas con el coeficiente de variación se determinó, que el Modelo de revisión Continua o Modelo "Q" era el más propicio, al contar con una demanda variable y reaprovisionamiento siguiendo el patrón de punto de recompra, además se definió la política de inventarios para demanda variable, con tiempos de entrega constantes.

Posteriormente se diseñó el código fuente basado exclusivamente en tecnologías web libres con licencia MIT, como Laravel para el backend, JQuery y Javascript nativo para el lado del cliente todo esto en sus últimas versiones, perfectas para el uso empresarial.

Finalmente, se validó el sistema tomando en cuenta todos los procesos programados, revisando con los usuarios si se cumplían con las especificaciones de la empresa.

Palabras claves: Dashboard, coeficiente de variación, Framework, Backend, punto de recompra, Javascript, JQuery MIT

ABSTRACT

The main objective of this research work was to develop a Dashboard as a monitoring tool to optimize the replenishment system of DISTRIBUIDORA FERBANZ S.A.C.

For the development, information was obtained from the demand data of the last years, later after evaluating them with the coefficient of variation, it was determined that the Continuous Review Model or "Q" Model was the most favorable, having a demand variable and replenishment following the repurchase point pattern, in addition, the inventory policy for variable demand was defined, with constant delivery times.

Subsequently, the source code was designed based exclusively on free web technologies with MIT license, such as Laravel for the backend, JQuery and native Javascript for the client side, all of this in its latest versions, perfect for business use.

Finally, the system was validated taking into account all the programmed processes, checking with the users if they met the company's specifications.

Keywords: Dashboard, coefficient of variation, Framework, Backend, repurchase point, Javascript, JQuery MI

INDICE DE CONTENIDO

I. INTRODUCCIÓN	15
1.1 REALIDAD PROBLEMÁTICA.....	15
1.2 DESCRIPCIÓN DEL PROBLEMA	16
1.3 FORMULACIÓN DEL PROBLEMA.....	16
1.4 JUSTIFICACIÓN	16
1.5 OBJETIVOS	17
2.2.1 OBJETIVO GENERAL	17
2.2.2 OBJETIVOS ESPECÍFICOS	17
II. MARCO DE REFERENCIA	18
2.1 ANTECEDENTES DEL ESTUDIO	18
2.2 MARCO TEÓRICO	19
2.3 MARCO CONCEPTUAL	28
2.4 HIPÓTESIS	36
2.5 VARIABLES E INDICADORES.....	36
TABLA 1. CUADRO DE OPERACIONALIZACIÓN DE VARIABLES	37
III. METODOLOGIA EMPLEADA	38
3.1 TIPO Y NIVEL DE INVESTIGACIÓN	38
3.1.1 TIPO.....	38
3.1.2 NIVEL.....	38
3.2 POBLACIÓN Y MUESTRA	38
3.2.1 POBLACIÓN	38
3.2.2 MUESTRA.....	38
3.3 DISEÑO DE INVESTIGACIÓN	38
3.4 TÉCNICAS E INSTRUMENTOS DE INVESTIGACIÓN	40
TABLA 2. TÉCNICAS E INSTRUMENTOS	40
IV. PRESENTACION DE RESULTADOS	42
V. DISCUSIÓN DE RESULTADOS	90
VI. CONCLUSIONES	92
VII. RECOMENDACIONES	94

REFERENCIAS BIBLIOGRAFICAS	95
ANEXOS	97

ÍNDICE DE TABLAS

Tabla 1. Cuadro de operacionalización de variables	37
Tabla 2. Técnicas e instrumentos.....	40
Tabla 3. Resumen de ventas mensual para el ladrillo king kong de 18 huecos lark del local José Gálvez período 2016 al 2021.	42
Tabla 4. Plan de pruebas para el módulo usuarios	79
Tabla 5. Plan de pruebas para el módulo proveedores	80
Tabla 6. Plan de pruebas para el módulo clientes.....	81
Tabla 7. Plan de pruebas para el módulo historial de demanda.....	82
Tabla 8. Plan de pruebas para el módulo pronóstico de demanda	83
Tabla 9. Plan de pruebas para el módulo de productos	84
Tabla 10. Plan de pruebas para el módulo de revisión continua	85
Tabla 11. Plan de pruebas para el módulo compras	86
Tabla 12. Plan de pruebas para el módulo ventas	87
Tabla 13. Plan de pruebas para el módulo de notificaciones	88
Tabla 14. Diseño de pruebas unitarias	89

ÍNDICE DE FIGURAS

Figura 1.	Base De Datos Bajo El Modelo De Entidad Relación.....	46
Figura 2.	Creación De Tabla Detalle De Compras, Llaves Foráneas A Productos	47
Figura 3.	Modelo Detalle De Compras Relacionado Con La Pertenencia A Una Compra Y A Un Producto	48
Figura 4.	Portal De Inicio Para La Empresa Distribuidora Ferbanz S.A.C.....	49
Figura 5.	Módulo De Inicio De Sesión Basado En Laravel Jetstream.	49
Figura 6.	Vista De Perfil De Usuario Basado En Laravel Jetstream.	50
Figura 7.	Vista De Perfil De Usuario Apartado Gestión De Sesiones Del Navegador.....	51
Figura 8.	Sección De Barra De Navegación (Navbar) Incluye El Sistema De Notificaciones Y Del Perfil De Usuario	52
Figura 9.	Sección De Barra De Lateral (Sidebar)	53
Figura 10.	Diseño De Registro De Proveedor	54
Figura 11.	Diseño De Listado De Proveedores	55
Figura 12.	Diseño De Registro De Producto	56
Figura 13.	Diseño De Listado De Productos	57
Figura 14.	Diseño De Información De Productos	58
Figura 15.	Diseño De Ventana Modal De Actualización De Datos Del Producto.	58
Figura 16.	Diseño De Registro De Compra De Productos	59
Figura 17.	Código De La Función “Change_Status” Responsable De Aumentar El Stock	60
Figura 18.	Código De La Función “Translate” Para Traducir El Estado Del Producto Para La Vista Del Cliente	61
Figura 19.	Código De La Función Change_Status Responsable De Disminuir El Stock	62

Figura 20. Diseño De Registro De Compra De Productos	63
Figura 21. Diseño De Historial De Ventas	65
Figura 22. Diseño De Ventana Emergente Para Compras, Múltiples Ventanas Abiertas	66
Figura 23. Diseño De Base De Datos Para La Demanda De Productos	67
Figura 24. Selector De Producto En El Apartado De Demanda Promedio Histórica	68
Figura 25. Tabla Demanda Mensual Promedio Histórica Ladrillo King Kong	68
Figura 26. Gráfica De Ventas Demanda Mensual Promedio Histórica Ladrillo King	69
Figura 27. Gráfica De Ventas Demanda Mensual Promedio Histórica Ladrillo King Kong, Descartando Los Años Previos 2016 Al 2019	69
Figura 28. Diseño Del Modelo De Variación Estacional O Cíclica Para El Pronóstico De Demanda	70
Figura 29. Diseño Del Pronóstico De Ventas Totales Por Regresión Lineal	71
Figura 30. Diseño Del Sistema De Inventario De Revisión Continua, Entrada De Parámetros	72
Figura 31. Barra De Progreso Para El Nivel De Existencia Por Producto	73
Figura 32. Panel De Cálculo Del Valor “Z” Para El Nivel De Servic	74
Figura 33. Código Fuente Del Script De La Función “Calcular”	75
Figura 34. Diseño De La Presentación De Resultados Para Los Cálculos Del Modelo.....	76
Figura 35. Diseño Del Sistema De Notificaciones Para La Recompra	77
Figura 36. Micro Peticiones Al Servidor Para Actualizar Las Notificaciones	77
Figura 37. Código Fuente Del Controlador De Notificaciones	78

INDICE DE ANEXOS

Anexo 1. Manual de instalación del framework laravel en windows 10	97
Anexo 2. Manual de instalación de la plantilla admin lte 3.....	100
Anexo 3. Repositorio de la plantilla integrada laravel-adminlte	105
Anexo 4. Ventas detalle local josé gálvez ferbanz s.a.c 2016-2021	106
Anexo 5. Modelo de variación estacional o cíclica en excel	107

I. INTRODUCCIÓN

1.1 Realidad problemática

Desde los inicios de la pandemia del SARS-Cov-2, los sistemas de gestión de inventarios y logísticos se han tenido que reinventar o renovar. Las empresas buscan lograr mantener un manejo más estricto del inventario. El valor de la información en tiempo real cobra mayor importancia en la nueva normalidad.

Dentro del sector de construcción, se está teniendo una recuperación sistemática, desde el 2020 al presente año. El presidente del BCR, Julio Velarde, proyectó que el sector construcción se contraerá 15.6% este año, informó el Diario El Peruano. Y remarcó que crecería 17.4% en el 2021 y 3.8% en el 2022. Por lo que muchas empresas del sector buscarán reposicionarse.

La empresa Distribuidora FERBANZ S.A.C, como otras empresas del rubro de venta al por mayor de materiales de construcción y artículos de ferretería, presenta la característica de adquirir más materiales, con ello tener un nivel alto de existencias para responder a la alta volatilidad de la variación de la demanda.

A su vez, Distribuidora FERBANZ S.A.C realiza un apalancamiento financiero anual para hacer frente a los volúmenes de venta y costos. Lo que se traduce a un sobre financiamiento.

Esto es un problema para la empresa, porque actualmente mantiene un sobredimensionamiento de sus niveles de stock para mantener un nivel de servicio adecuado y una sobre estimación de su pronóstico de inversión, la cual es un costo de oportunidad importante.

1.2 Descripción del problema

Hoy en día, la empresa DISTRIBUIDORA FERBANZ S.A.C pretende optimizar la gestión actual de inventarios. Por lo que, la empresa busca una mejora sostenible con un mejor monitoreo de su nivel de stocks, y para ello se empleará un Dashboard, con un modelo probabilístico que permita mejorar su administración.

1.3 Formulación del problema

¿Cómo desarrollar un Dashboard como herramienta de monitoreo para optimizar el sistema de reaprovisionamiento de la DISTRIBUIDORA FERBANZ S.A.C.?

1.4 Justificación

Teórica.

En el presente trabajo se empleó los conocimientos de Ingeniería Estadística, Gestión de base de datos, Programación cliente servidor y Control de Inventarios para el desarrollo de un Dashboard y con ello tener un control de las variables de entrada para hacer más eficiente el aprovisionamiento del inventario, teniendo como resultado mejoras en los niveles de stock, costos, inventarios de seguridad y niveles óptimos de servicio.

Práctica.

El presente trabajo de investigación representa un gran beneficio para el proceso de reaprovisionamiento de la empresa, empezando por el seguimiento de la demanda como etapa más temprana y a la fluidez del Dashboard para monitoreo constante ahorrando en hardware y software, reduciendo los errores y protegiendo los datos al no depender de almacenamiento interno todo ello mediante una interfaz amigable y fácil de usar.

Metodológica.

El presente trabajo de investigación se basa en la necesidad de saltar de un sistema básico de Excel a un sistema reactivo dinámico y personalizable con potencial de abarcar más áreas de la empresa como área de ventas, compras, pronósticos. Con lo que se utiliza registros históricos en Excel de Ventas y Costos para determinar la política de inventarios la cual se basa el dashboard.

Económica.

El presente trabajo de investigación consigue regular el apalancamiento que ejecuta la Distribuidora FERBANZ S.A.C que lo lleva a sobredimensionamiento de inventarios y con ello logra un mayor control del pronóstico de demanda para la inversión, que representa un coste de oportunidad relevante.

1.5 Objetivos

2.2.1 Objetivo General

Desarrollar un Dashboard como herramienta de monitoreo para optimizar el sistema de reaprovisionamiento de la DISTRIBUIDORA FERBANZ S.A.C.

2.2.2 Objetivos Específicos

- 1.5.2.1** Identificar el tipo de demanda de la empresa DISTRIBUIDORA FERBANZ S.A.C.
- 1.5.2.2** Definir el modelo de inventario más adecuado para la empresa DISTRIBUIDORA FERBANZ S.A.C.
- 1.5.2.3** Establecer la política de inventario para la empresa DISTRIBUIDORA FERBANZ S.A.C.
- 1.5.2.4** Diseñar el código fuente del entorno web de la aplicación para la empresa DISTRIBUIDORA FERBANZ S.A.C.
- 1.5.2.5** Validar el modelo de reposición de inventario bajo la plataforma web para la empresa DISTRIBUIDORA FERBANZ S.A.C.

II. MARCO DE REFERENCIA

2.1 Antecedentes del estudio

Navarrete, D. y Vasquéz, T. (2013) en su tesis “Desarrollo de un sistema de control de inventarios en ABASTESUMINISTROS LTDA” presenta un estudio de diferentes modelos de inventarios con la finalidad de establecer la forma más óptima de generar pedidos según las limitaciones de la empresa.

Ancori, Y. (2020), en su tesis “Propuesta del modelo de gestión de inventario con revisión periódica para la mejora de abastecimiento de mercadería en la DISTRIBUIDORA ESPECERÍAS NELLY”, se plantea como objetivo determinar como el modelo de inventario con revisión periódica mejoraría el abastecimiento de los faltantes y exceso de mercadería. La contribución de esta tesis es una guía para la elaboración de la propuesta para este estudio.

Reyes, V (2020) en su tesis " Diseño de un modelo de control de inventario basado en la metodología sistemas de inventarios de revisión periódica para prever el agotamiento y exceso de existencias de producto terminado en la empresa Guayas BYG." propone como objetivo diseñar un modelo de control para inventarios basada en metodología de inventarios con revisión periódica para prevenir el agotamiento y exceso de existencias. Su aporte es una guía referente para la finalidad del presente estudio.

Gamboa J. (2015) En su tesis “Modelo de gestión de inventario probabilístico de revisión periódica para reducir los costos del inventario de la Curtiembre Ecológica del Norte E.I.R.L.” se propone aplicar un modelo de gestión de inventarios probabilísticos de revisión periódica, para reducir costos de inventario en la empresa Curtiembre Ecológica del Norte E.I.R.L en Trujillo.

Ancori, Y. y (2020), en su tesis “Propuesta del modelo de gestión de inventario con revisión periódica para la mejora de abastecimiento de mercadería en la DISTRIBUIDORA ESPECERÍAS NELLY”, se planteó como objetivo determinar

como el modelo de inventario con revisión periódica mejora el abastecimiento de los faltantes y exceso de mercadería. La contribución de esta tesis es una guía para la elaborar la propuesta de estudio.

Garcia, S. y Moreno M. (2019) en su tesis “Determinación de una política óptima de inventario para la mejora del proceso de ventas en EMPRESA INTERNACIONAL DE INSTRUMENTAL CLIMATOLÓGICO”, se propone determinar una política óptima de inventarios para mejorar el proceso de ventas, determinando el método a aplicar el software más adecuado y determinar los productos a priorizar, con el fin de obtener resultados eficientes.

2.2 Marco teórico

2.2.3 Modelos estocásticos de inventario

Un inventario estocástico o probabilístico muestra un tiempo o demanda de entrega que se desconoce (aleatorio), por ello este tiempo o demanda es expresado mediante una variable aleatoria.

Así mismo, un modelo de inventario probabilístico refiere al uso de una distribución probabilística para detallar el valor de la demanda u otra variable desconocida. Y ese es el gran aporte de la estadística.

Tomando esto como punto de inicio, se han establecido diferentes modelos de inventario de probabilidad con multiperiodos que son clasificados por la forma de revisión del inventario, modelos tal cuales su variación será causada por la incertidumbre y no provocar predecibles. (Ingeniosa Empresa, 2018)

2.2.4 Sistema de revisión continua

También se lee conoce como sistema de punto de reorden, sistema de cantidad fija o modelo Q. Los modelos de esta clasificación para dicho sistema tienen la cualidad de realizar un pedido cuando el inventario alcanza un determinado nivel. Esto conlleva a controlar el inventario sobrante de forma regular y muchas veces,

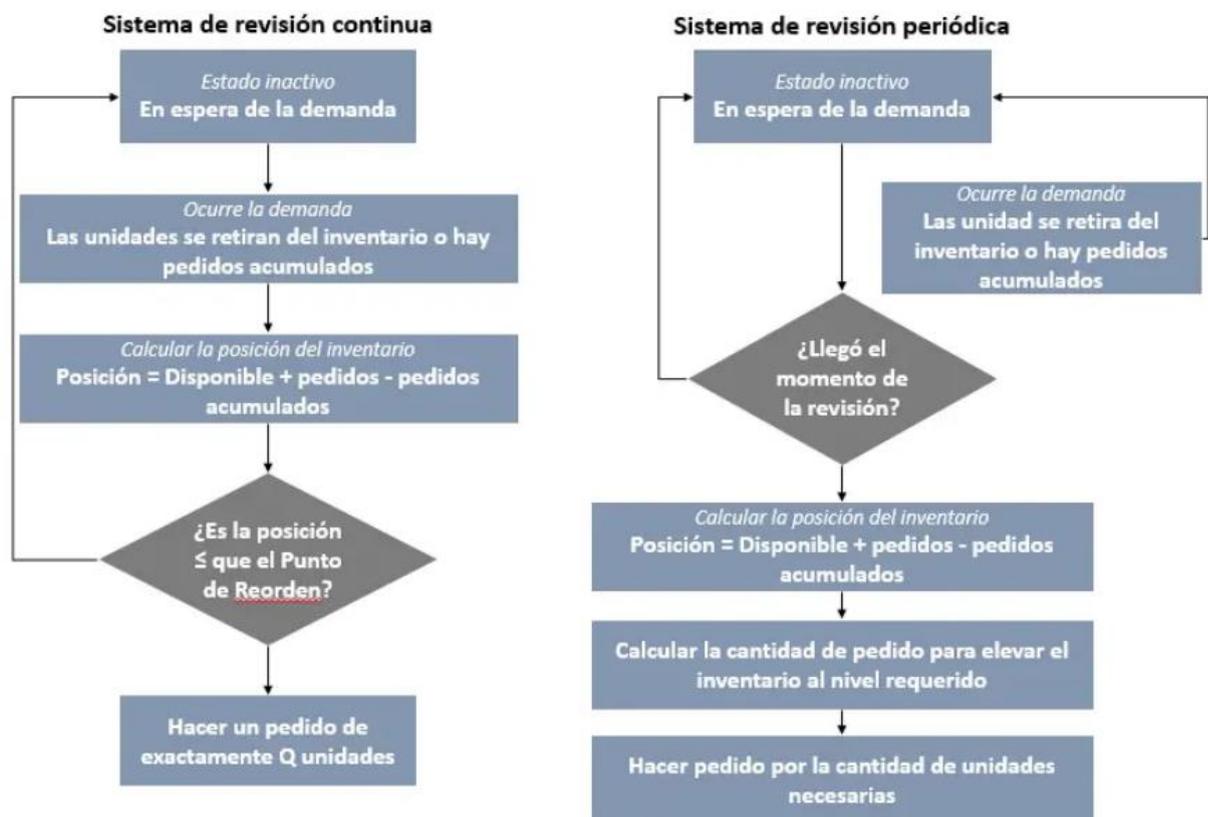
con cada vez que se realizan los retiros de algún artículo para establecer si es necesario emitir un nuevo pedido.

Dicho de otra forma: La cantidad está fijada, el tiempo no variable.

Para los casos de inventarios de revisión continua, tenemos dos variantes del Modelo EOQ: La primera basada en un EOQ con inventario de seguridad para satisfacer la demanda incierta, el segundo es un Modelo EOQ Probabilístico que toma la demanda aleatoria en la formulación. (web, 2020)

2.2.5 Diferencia entre los modelos probabilísticos de inventario

Diagrama de flujo para cada modelo.



Fuente: Chase, 2013

Para el sistema de revisión continua, cuando el inventario es menor o igual al punto de reorden, se necesitará generar un pedido nuevo, de otra forma, los inventarios seguirán consumiéndose. Este control es constante.

Con el sistema de revisión periódica, nos hacemos la pregunta ¿llegó el momento de revisión? Eso con fundamento en una periodicidad determinada en el tiempo. Si es el caso, calculamos cuánto se pedirá y pedimos, si no, el inventario sigue consumiéndose.

2.2.6 Sistema de revisión continua de existencias

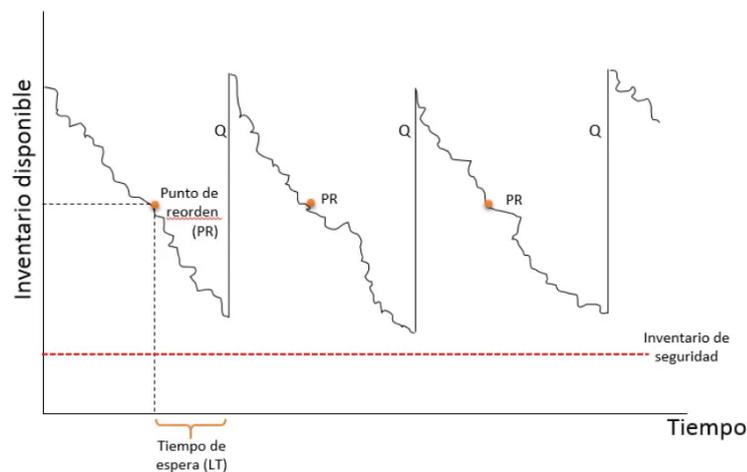
También se le conoce como sistema Q, sistema de cantidad fija de pedido o sistema de punto de pedido. Este tipo de sistema de control de inventario, cada vez que se retira un artículo, se verifica el exceso, es decir, se hace una revisión del nivel de inventario resultante, teniendo en cuenta que:

- Si el stock está por debajo de cierto nivel, se asigna una nueva orden.
- Si el stock resultante no está por debajo del nivel, se seguirá consumiendo.

Los inventarios no cambiarán debido a demanda o tiempos de entrega inciertos, o pueden cambiar. Esta es la razón de ser de un escenario probabilístico.

Entonces, para este se tendrán tres casos:

- La demanda cambia y los tiempos de entrega no cambian.
- La demanda se mantiene constante y los tiempos de entrega varían.
- La demanda variable y cambios en los tiempos para las entregas.



Fuente: (Ingeniosa Empresa, 2018)

No hay certeza sobre la demanda y/o tiempos de entrega. Siempre se realizará un pedido cuando el stock esté en un determinado nivel (punto de pedido), por lo que los tiempos de entrega pueden variar, por lo que se debe mantener un stock seguro.

2.2.7 El punto de reorden

Cuando el número de artículos alcanza el nivel especificado en stock, realizamos un nuevo pedido y lo marcamos como "Q". Este nivel de determinación es lo que llamamos el punto de reposición o el punto de reposicionamiento.

Recuerde que dicha "Q" siempre será fija (se procede a calcularla) y el tiempo entre los valores de "Q" es variable; Si este no es el caso, procesaremos de acuerdo con un patrón determinista donde la demanda y los plazos de entrega se conocen con certeza.

2.2.8 Niveles de stock de seguridad

Este presente escenario donde no se conoce la demanda (escenario de probabilidades), siempre se va contar un stock de seguridad y se revisa permanentemente el nivel de inventario con la finalidad de evitar faltantes.

De esta forma el stock de seguridad es el almacenamiento de las unidades adicionales con el fin de evitar faltantes.

Para poder elegir aquel inventario de seguridad debe buscarse un equilibrio entre el nivel del servicio al cliente y sus costos asociados a mantener el inventario.

Un enfoque de planteamiento es usando modelos de minimización del costo, pero este resulta ser una labor compleja al incorporar la determinación del costo de faltantes y los pedidos aplazados.

Otra posibilidad más frecuente es determinar las políticas para el nivel de servicios para el inventario, y desde ahí hacer el cálculo del inventario de seguridad según la política.

2.2.9 Política de nivel de servicio de inventario

Consiste en definir la probabilidad de que quedarnos fuera de stock durante el periodo entre la emisión de una orden de pedido y su llegada (lead time)

No obstante, la demanda pueda que no tenga un comportamiento uniforme en el tiempo de espera (lead time), o pueda que sí.

En otras palabras, si existe variabilidad en la demanda respecto al promedio no es grande, se puede mantener un inventario de seguridad pequeño. Caso contrario, si esta demanda oscila mucho entre pedidos, conviene mejor tener un nivel de stock de seguridad mayor para hacer evitar faltantes.

Esta variabilidad nos lleva a hablar sobre la distribución de probabilidad, y con ello incluir dos conceptos de gran importancia para la gestión de inventarios: media y varianza.

2.2.10 Cálculo de existencias de seguridad.

La existencia de seguridad estará definida de acuerdo a la fórmula.

$$\text{Inventario de seguridad} = Z\sigma_{LT}$$

Para el siguiente gráfico:

Definamos que la demanda tiene una distribución normal durante el tiempo de entrega. La línea en medio es el promedio. Si se define el nivel de servicio del inventario en 50% (es la probabilidad de que ocurra faltantes) el punto de reorden coincidirá con la línea del medio, de ser el caso, la existencia de seguridad es 0.

El punto de reorden está determinado por la demanda promedio más el stock de seguridad, si se liquida el stock de seguridad tenemos:

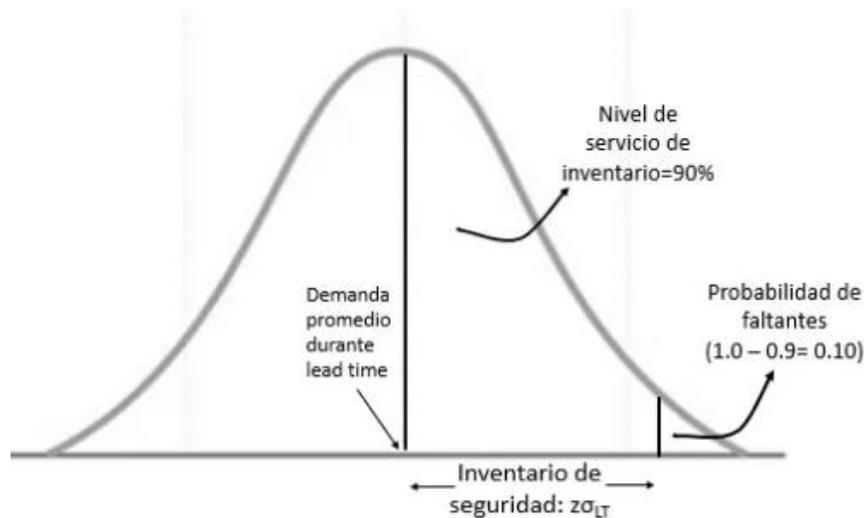
$$\text{Punto de reorden} = \text{Demanda promedio} + \text{inventario de seguridad}$$

$$\text{Punto de reorden} - \text{Demanda promedio} = \text{inventario de seguridad}$$

Por lo tanto, si desea un nivel de servicio un 50 % más alto, la puntuación de reorden tendrá que ser más alta que la demanda promedio durante el tiempo de entrega, esto incluirá un punto de reorden que esté fuera de la línea central de la distribución gaussiana o campana.

Si, por ejemplo, si quisiera tener un valor del 90% (probabilidad de tener faltantes en un 10%)

Gráfica de la campana de Gauss, Distribución de la demanda



Fuente: (Ingeniosa Empresa, 2018)

De esta forma, cuanto mayor sea el valor de "Z", mayor será el nivel de servicio de inventario y por lo tanto mayor será el inventario de seguridad. Por otro lado, si "Z" es cero, no habrá existencia de seguridad, porque el reabastecimiento es igual a la demanda y habrá escasez el 50% del tiempo.

2.2.11 Sistema de demanda Q variable, con tiempo de entrega permanente

Desde este punto considerando la demanda variable, se necesita hacer una estimación de la demanda durante el tiempo de entrega (lead time).

Si se conoce la demanda promedio con su desviación estándar durante un período (t). De esta forma, la demanda media con su desviación estándar es constante para cada tiempo (t).

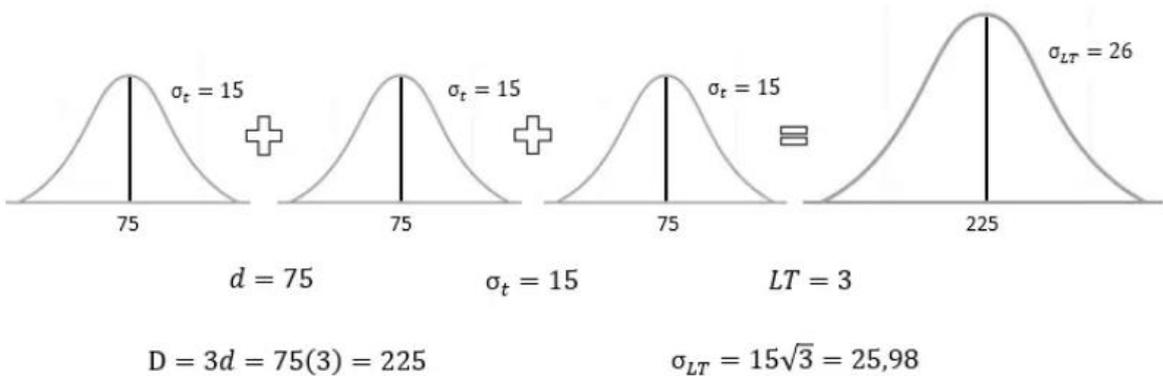
Piense en los tiempos de entrega como un múltiplo o una fracción del tiempo de entrega. Por ejemplo, t = 1 semana y LT = 3 semanas.

De esta forma, en el tiempo LT, la demanda se puede presentar como la suma de las necesidades en cada período t. Si t = 1 semana y LT = 3 semanas, la demanda que denotaremos por D para esta LT será d + d + d, también la asociamos con d * LT. Se parecerá a la varianza de la distribución de la demanda a lo largo del tiempo hasta la entrega (LT), como una fórmula:

$$\sigma_t^2 + \sigma_t^2 + \sigma_t^2 + \dots = \sigma_t^2 LT$$

Y su forma gráfica será:

Gráfica de la desviación estándar del tiempo de entrega



Fuente: (Krajewski, 2008)

Con esta gráfica, se aprecia como la demanda se desarrolla en el tiempo de entrega (LT) parte de una distribución individual de demandas semanales. Con $d=75$, $\sigma_{LT}=15$ y $LT=3$ semanas, se tiene una demanda media a lo largo del tiempo de espera de 225 unidades ($3d=75*3$) con una desviación estándar de $15\sqrt{3}=25.98$.

Ordenado por pasos tendríamos:

Paso 1: Calcular "Z" para el nivel de atención requerido por ejemplo 95%, para este caso el valor "Z" para 90% es de 1,28.

Paso 2: Calcular el punto de recompra y el stock de seguridad

$$\begin{aligned} \text{Inventario de seguridad} &= Z\sigma_{LT} \\ \text{Punto de reorden} &= \bar{d}LT + Z\sigma_{LT} \end{aligned}$$

Paso 3: Antes de calcular el costo total del sistema Q de inventario, se necesita la cantidad óptima de pedido EOQ. Se debe determinar la demanda anual, una vez hecho esto se podrá realizar el cálculo del costo total.

Q*: cantidad óptima de pedido

H: costo de mantener por un año el inventario

S: costo de hacer la orden

D: demanda anual

d: demanda semanal

D = d*52 semanas

Cantidad óptima de pedido

$$Q^* = \sqrt{\frac{2DS}{H}}$$

Costo Total

$$C = \frac{Q}{2}H + \frac{D}{Q}S + H * Z\sigma_t$$

Nótese que la fórmula del costo anual es parecida al modelo EOQ clásico, lo que varía es aquella parte referida al costo anual por mantenimiento. Para su cálculo se considera que el inventario de seguridad esté disponible todo el tiempo. Aun así, habiendo algunas veces mayor demanda que la promedio durante el tiempo de entrega, o a veces menor.

Por ello, al transcurrir de un año, se supondrá que el inventario de seguridad estará disponible.

2.2.12 Sistema de demanda Q constante, pero tiempo de entrega variable.

Para este caso se tiene precisión de la demanda, pero el tiempo entre pedidos, varia.

El cálculo para el punto de reorden estaría dado por:

$$\text{Punto de reorden} = \bar{d}LT + Z\sigma_{LT} * d$$

Para el caso de las demás variables como el costo total, tendrían la misma formulación para su cálculo que el modelo anterior.

2.2.13 Sistema de demanda y tiempo de entrega variables.

Este es el caso más frecuente que se presenta en las empresas, porque no tenemos la certeza tanto de la demanda como el tiempo de espera.

Al ser más frecuente este caso, se asistirá por medio de la simulación por computadora.

Los cálculos de las variables estarían dados por las siguientes expresiones matemáticas:

$$\text{Inventario de seguridad} = Z \sqrt{\sigma_d^2 \bar{L}\bar{T} * \bar{d}^2 \sigma_{LT}^2}$$

$$\text{Punto de reorden} = \bar{d}LT + Z \sqrt{\sigma_d^2 \bar{L}\bar{T} * \bar{d}^2 \sigma_{LT}^2}$$

2.3 Marco Conceptual

Aplicación web

Una aplicación web es aquella aplicación que es accedida por la web mediante una red como internet o intranet. En general, también incluye a aquellos programas informáticos que son ejecutados por un navegador o codificado por algún tipo de lenguaje compatible con navegadores (como lo son JavaScript, HTML o PHP); confiando en el navegador web que renderice la aplicación. (Alegsa, 2018)

Framework

Un Framework es aquella estructura previa que puede aprovecharse para el desarrollo de un proyecto.

El Framework es aquella forma de plantilla, un tipo de esquema conceptual, que facilita el desarrollo de tareas, ya que normalmente se necesita complementarlo

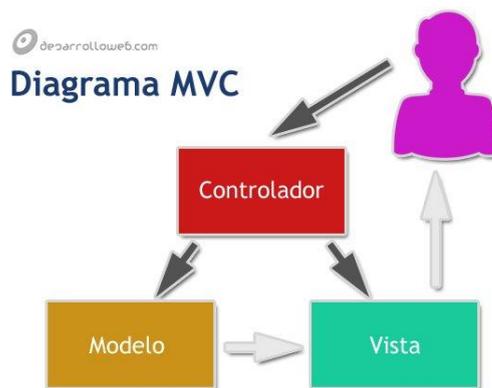
con aquello que se desea realizar.

Estas estructuras favorecen la entrega de proyecto en menor tiempo y con un código más limpio, con una eficacia ya comprobada.

Gracias a ellos, los programadores logran complementar y/o modificar la estructura básica para entregar un software o aplicación que cumpla con los objetivos necesarios. (Muenta, 2020)

Modelo, Vista Controlador (MVC)

El modelo Vista controlador MVC es una propuesta para arquitectura de software empleado para separar el código para distintas responsabilidades, generando diferentes capas que se encargan de realizar una tarea muy específica, lo que facilita enormemente al desarrollo. (Alvarez, 2020)



Fuente: Desarrolloweb.com

Laravel

Laravel es un framework de PHP para ayuda al desarrollo web de aplicaciones escritas en dicho lenguaje de programación. Este framework, nos permite realizar muchas cosas al desarrollar una aplicación, por medio de sus sistemas de paquetería de tipo MVC (Modelo-Vista-Controlador), nos agiliza enormemente el desarrollo web puesto que tiene muchas funciones ya predefinidas, cómo instanciar clases y métodos listos para usar en diferentes partes de nuestra aplicación sin necesidad de escribirlo o repetirlo varias veces y ahorrando mucho tiempo y código innecesario.

Laravel contiene diferentes herramientas y/o utilidades con diversas finalidades:

- Su motor de plantillas, denominado **Blade**, da diversas posibilidades para generar páginas visualmente potentes y eficaces, capaz de emplear sus propias variables y reutilizarlas.
- **Eloquent ORM**, resulta ser muy intuitivo para realizar consultas sobre objetos en formato en PHP.
- En **seguridad**, ofrece un alto nivel de fuerza con mecanismos de tipo hash y salt para encriptar mediante librería tales como, **BCrypt**, que también lo usa Zend Framework.
- **Artisan**, aquel sistema de instrucciones que ofrece gran potencia y grandes ventajas y posibilidades a los desarrolladores, para generar controladores, entidades o actualizaciones a la base de datos, entre otras opciones más.
- **Librerías y modularidad**. Laravel además de sus librerías cuenta con ayuda de Symfony que es, otro MVC de los más empleados en la actualidad con una gran comunidad que hace que su desarrollo evolucione significativamente.
- **Base de datos con migraciones**. Permite actualizar y generar migraciones a la base de datos desde el inicio del desarrollo y hacer cambios al código conforme se disponga sin la necesidad de eliminarla y volverla a generar, con esto el riesgo de perder datos serán mínimos. Y, además, gracias a su **Schema Builder** hace que no necesite usar el SQL, porque cuenta con un sistema intuitivo en código PHP para hacerlo más sencillo. (Vera, 2021)

Composer

Composer es un administrador de paquetes de PHP que brinda un estándar para gestionar, descargar e instalar librerías y dependencias. Al igual que NPM en Node.js y Bundler en Ruby, Composer es la solución perfecta cuando se trabaja en proyectos de alta complejidad que dependen de muchas fuentes de instalación. En vez de descargar cada una de las dependencias manualmente, Composer lo hace automáticamente por nosotros, por medio de comandos precisos. (Yair, 2019)

Livewire

Livewire es un framework para Laravel que permite generar interfaces dinámicas de forma sencilla, sin desvincularse de la comodidad de Blade. (Palacios, 2021)

Jetstream

Es un scaffolding diseñado exclusivamente para framework en PHP. Jetstream ofrece una plataforma de trabajo prediseñada para iniciar el desarrollo de aplicaciones con Laravel. (Chávez, 2021)

Alguna de las funciones que trae Jetstream son las siguientes:

- Verificación por correo electrónico
- Autenticación de dos factores
- Administrador de sesiones
- Soporte de API
- Gestión de equipos
- Etc.

ADMIN LTE 3

Es un panel de administración de bootstrap creado por el estudio Almsaeed. Es una solución de código abierto basada en un diseño modular que permite una fácil construcción y personalización. El planteamiento es que exista entre sus componentes un complemento o un widget que se utiliza para crear la interfaz de usuario tanto en el frontend como en el backend. AdminLTE se puede descargar en una versión prediseñada, con el código compilado para cargarlo directamente en producción, o como una descarga ZIP con el código fuente en caso de que el desarrollador quiera hacer algunos cambios preliminares en la plantilla. (Andyfg, 2021)

Back-end

Es lo que se trabaja con los procesos requeridos para que la plataforma web tenga un correcto funcionamiento. Estos procedimientos o funciones están ocultos, pero son altamente importantes para el correcto funcionamiento de la plataforma. Ciertas acciones que gestiona el backend se conectan con la base de la información o se comunican con el servidor de alojamiento hosting. (Rodríguez, 2020)

Front-end

El frontend es la parte del desarrollo web dedicada al front de un sitio web, en decir al diseño de un sitio web, desde la estructura del sitio hasta estilos como colores, fondos, tamaños, animaciones y efectos.

Es esa aquella parte de la página con la que interactúan los usuarios, es todo la codificación que se ejecuta en el navegador por el usuario, lo que viene siendo la aplicación cliente, es decir, todo lo que el visitante visualiza y experimenta directamente.

En el área del front-end se trabaja con lenguajes generalmente del lado del cliente, como:

- **HTML5** (lenguaje de hipertexto con marcado, utiliza etiquetas que estructuran y organizan contenido web).
- **CSS** (se encarga del formato y diseño visual de las páginas web escritas en html) darle forma estructural con estilo al sitio web.
- **Javascript** (un lenguaje de programación, rápido y seguro para programar centros de datos, consolas, teléfonos móviles o Internet) para darle dinamismo a entorno de desarrollo web.

HTML y CSS son lenguajes de estructura y estilo. Javascript es un lenguaje de programación. (Hernández, 2021)

XAMPP

XAMPP es una distribución de Apache que contiene varios softwares gratuitos. El nombre es un acrónimo conformado por las iniciales de los programas que lo incluyen:

- **Linux:** es el sistema operativo el cual se instalará en nuestra aplicación. Su diferenciación con Windows, es que Linux es una distribución gratuita y segura, no requiere de licencia y provee de un alto rendimiento.
- **Apache:** el servidor web de código abierto, es la aplicación que utilizada a nivel mundial para la distribución de contenido en internet vía web. Las aplicaciones de servidor son brindadas como software gratuito por la fundación Apache.
- **MySQL / MariaDB:** XAMPP posee uno de los sistemas de administración de bases de datos más relacionales y conocidas del mundo. Al combinarse con el servidor web Apache y el lenguaje de programación PHP, MySQL se utilizaba para almacenar los datos para los servicios web. En las versiones más recientes de XAMPP, la base de datos ha sido reemplazada por MariaDB.
- **PHP:** es un lenguaje de programación de código del lado del servidor que permite la creación de aplicaciones o páginas web altamente dinámicas. Es

una plataforma con independencia y tiene compatibilidad con varias bases de datos.

- **Perl:** este lenguaje de programación usado en la gestión de sistemas, el desarrollo web y programación para redes. También permite realizar programación de aplicaciones web dinámicas. (Garcia, 2020)

Hardware

Se denomina Hardware a todo aquel componente sea físico interno de un ordenador, en otras palabras, la parte tangible de un equipo, como por ejemplo los discos duro, memorias, la CPU, el microprocesador, el cableado, el ratón y un largo etc. Por otro lado, al referirnos a Hardware también se hace alusión a componentes periféricos, o fuera del equipo como lo son una impresora o escáner por citas un ejemplo.

La terminología de Hardware no solo se utiliza en la informática sino también para la tecnología en términos generales, como lo son algunos dispositivos externos, máquinas, electrónica, teléfonos móviles, cámaras fotográficas, filmadoras, extensiones en general.

El Hardware se puede clasificar en dos formas:

Hardware básico: a este pertenecen los componentes requeridos para lograr la funcionalidad mínima en un equipo informático. Como por ejemplo puede ser un teclado y una pantalla.

Hardware complementario: Sus funcionalidades principales es específica, En otras palabras, no son estrictamente requeridos para el funcionamiento del equipo, Un ejemplo son los auriculares o un mouse. (Mrhardware, 2018)

Software

El software es una colección de instrucciones que debe seguir un computador, como lo son aquellas instrucciones que tiene que hacer y cómo llevarlas a cabo.

Su concepto abarca aquello que es intangible en el computador, como lo son, los programas, los sistemas operativos, por ejemplo. (Buzón, 2020)

Dashboard

Un Dashboard es un mecanismo de administración de la información que visualiza, analiza y muestra de forma visual los indicadores de desempeño clave (KPI), con sus métricas y datos elementales para establecer un seguimiento del estado de una empresa, campaña, departamento o algún proceso en específico.

Se puede referir a un Dashboard como la forma de "resumen" que recaba datos diferentes mediante fuentes en un solo lugar y los muestra de forma digerible para que lo más elemental salte a la vista. Entre algunas características que debe poseer un centro de control como este son:

- **Personalizado.** Un panel administrativo debe tener exclusivamente KPI que sean importantes. Para enfocarlo, se puede pensar en las interrogantes principales a las que queremos responder. Como, cuáles son las principales fuentes de tráfico web, cómo está funcionando nuestros filtros ventas o cuáles son los productos que nos generan más beneficios.
- **Visual.** La razón de un Dashboard es que podamos visualizar la información que buscamos fácilmente. Para ello, los datos se muestran en forma de gráficos o tablas y debemos contar con indicadores ágiles a través de claves de color, flechas hacia arriba o abajo o cifras destacadas, por ejemplo.
- **Práctico.** La función principal de un Dashboard siempre será orientar las acciones. Por lo tanto, debe agilizar la información necesitada para que

se pueda saber cuáles son los próximos pasos a seguir para mejorar los resultados.

- **En tiempo real.** Al día de hoy, las acciones como en marketing digital evolucionan rápidamente y aprovechar el momento indicado es esencial. Por eso, la información siempre debe estar actualizada al instante en todas las fuentes y visualizarse en el Dashboard en tiempo real. (Ortiz, 2021)

2.4 Hipótesis

Con el desarrollo de un Dashboard como herramienta de monitoreo para optimizar el sistema de reaprovisionamiento de la DISTRIBUIDORA FERBANZ S.A.C se consigue mejorar el nivel de servicio, de stock y reducir costo, con la posibilidad integrarlo a otras áreas de la empresa.

2.5 Variables e indicadores

2.5.1 Variable independiente: Dashboard (panel administrativo web)

2.5.2 Variable dependiente: Optimizar el proceso

2.5.3. Operacionalización de las variables

Tabla 1. Cuadro de operacionalización de variables

Variable		Definición conceptual	Definición operacional	Dimensiones	Indicadores	Escalad de medición
Independiente	Sistema de automatización	Sistema web para control de reaprovisionamiento (DASHBOARD)	Conjunto de operaciones con mínima inclusión del ser humano	Automatización	Tiempo de respuesta	segundos
Dependiente	Optimizar el proceso de reaprovisionamiento	Mejorar el uso y gestión de la información.	Conjunto de operaciones de ingreso de datos de la demanda	Ventas	Promedio	unidades/semana
				Desviación estándar de las ventas	Desviación estándar	unidades
				Tiempo transcurrido desde la colocación de la orden y la recepción de la cantidad ordenada	Tiempo de entrega	semanas
				Demanda estimada anual	Producto de demanda promedio	Unidades/año
			Conjunto de operaciones de ingreso de datos de los costos	Costo de realizar una orden de pedido	Costo de ordenar	Soles
				Costos de resguardar el stock durante el tiempo previo a su venta.	Costo de mantener	Soles
				Total de costos relacionados con el proceso	Costo Total	Soles
			Conjunto de operaciones de cálculos automatizados	nivel de existencias señalado para reabastecimiento	Punto de reorden	unidades
				stock extra en el almacén	Inventario de seguridad	unidades
				Cantidad óptima para generar pedido	Unidades por pedido	unidades
				Cantidad de pedidos	Número de órdenes al año	ordenes

III. METODOLOGIA EMPLEADA

3.1 tipo y nivel de investigación

3.1.1 Tipo

Según su finalidad es un tipo de investigación aplicada debido a que se hace uso de los conocimientos teóricos de la gestión de inventarios y de desarrollo web con la finalidad de migrar y mejorar el sistema en la empresa.

3.1.2 Nivel

Es descriptivo por lo que busca determinar la variabilidad y comportamiento de los parámetros de gestión de inventarios con un diseño no experimental pues no se busca manipular las variables de estudios

3.2 Población y Muestra

3.2.1 Población

Esta conformado por los 3 productos de la línea principal de rotación del almacén de la DISTRIBUIDORA FERBANZ S.A.C

3.2.2 Muestra

La muestra esta conformada por los datos de ventas históricos del 2016 al 2021 de los 3 principales productos de la empresa.

3.3 Diseño de Investigación

- Para la identificación del tipo de demanda de la empresa DISTRIBUIDORA FERBANZ S.A.C, se usan los datos históricos de las ventas de los productos estrella.

- Para definir el modelo de inventario más adecuado para la empresa DISTRIBUIDORA FERBANZ S.A.C, se calculan los promedios por meses con sus respectivas desviaciones estándar para el cálculo del coeficiente de variabilidad.
- Para establecer la política de inventario para la empresa DISTRIBUIDORA FERBANZ S.A.C. Se analizan la variabilidad de la demanda y la política actual de la empresa como su gestión y manejo de inventarios.
- Para el diseño del código fuente del entorno web de la aplicación para la empresa DISTRIBUIDORA FERBANZ S.A.C. se realiza a través del editor de código VISUAL STUDIO CODE, usando el Framework de PHP Laravel.
- Para la validación del modelo de reposición de inventarió bajo la plataforma web para la empresa DISTRIBUIDORA FERBANZ S.A.C, se usa como fase preliminar de pruebas el servidor local XAMPP y para la fase final se subirá a un servidor WEB con todas las migraciones de la base de datos como el código fuente y credenciales de uso.

3.4 Técnicas e instrumentos de investigación

Tabla 2. Técnicas e instrumentos

Técnicas	Instrumento	Información o fuentes	Principales ventajas	Principales desventajas
Entrevista	Guía de entrevista	Operadores del sistema de ventas y logística	Permite profundizar los aspectos importantes	Solo aplica a un número de informantes limitados
Observación de campo no experimental	Guía de observación	Informante: primera persona, el investigador	Contacto directo del investigador	Aplicación limitada a aspectos fijos
	Reportes	Reportes de ventas distribuidora FERBANZ S.A.C	Permite profundizar aspectos importantes del proceso	La información sobre las variables es limitada
Revisión de la documentación del framework	Guía web	Página oficial de Laravel.	Contiene todas las funcionalidades con ejemplos.	Se requiere conocimientos básicos de su sintaxis.
Revisión de la documentación de la plantilla ADMIN LTE 3	Guía web	Página de Jeroen Noten en Github.	Permite conocer todos los parámetros de instalación e implementación	Se requiere un conocimiento básico del framework Laravel.
Revisión de los recursos gratuitos en investigación estadística.	Calculadora web	Página de Social Science Statistics	Contiene codificación y programación de operadores estadísticos avanzados.	Se requiere un nivel intermedio de javascript

Nota: elaboración realizada por el autor.

3.5 Procesamiento y análisis de datos

3.5.1 Sistema de web, desarrollo del dashboard:

El desarrollo del dashboard estará Basado en el framework Laravel, con herramientas robustas del sistema de plantillas Blade, y tomando como base la plantilla ADMIN LTE3 con licencia de libre uso y distribución con la facilidad de ser usada de manera comercial, se instalará y configurará el entorno de trabajo de manera local usando el editor de código "Visual Studio Code", facilita enormemente el desarrollo de la plataforma web, mediante sintaxis propias del framework Laravel como lenguajes propios de la web "HTML5" , "PHP", "Javascript", " Bootstrap", se contruye la lógica y programación de las vistas para el usuario, también a su vez cuenta con el gestor de paquetes "Composer" para agilizar funciones en "PHP" , y se incorpora el sistema "Livewire" para la gestión de sesiones y usuarios, ya que este brinda un nivel de seguridad robusta como el de doble autenticación y seguimiento de dispositivos usados por usuario. El dashboard es totalmente "Responsive" de esta forma pueda ser usado también en dispositivos móviles.

3.5.2 Análisis de datos

Una vez recabado toda la información por los instrumentos de recolección y análisis estadístico y descriptivo de los resultados tanto de los datos históricos como los presentes, se establece según la investigación realizada emplear la prueba paramétrica Z, aprovechando el teorema central del límite al usar los promedios como valor principal, con ello podremos basarnos en la distribución normal estándar para los caculos, de esta forma hacer uso del modelo de Revisión Continua Con Demanda Probabilística, el cual se encarga de describir las relaciones entre las variables de entrada del proceso de reaprovisionamiento como los KPI y pronósticos de inventarios.

IV. PRESENTACION DE RESULTADOS

4.1 En relación con el objetivo específico 1

Basado en las entrevistas con el administrador de la empresa DISTRIBUIDORA FERBANZ S.A.C. se describe que tienen identificado de manera objetiva el desarrollo la demanda de sus principales productos, por lo que el análisis ABC de productos pasa a ser agendado para un desarrollo posterior complementario como modulo auxiliar del sistema web. EL diagnóstico se basó en confirmar las afirmaciones de la administración en que existe una demanda altamente variable.

Según Hamdy Taha en su obra Investigación de operaciones, menciona que si la dispersión alrededor de la media genera altos valores demuestra alta incertidumbre en la media.

Se toma como referencia un umbral de datos de ventas del producto estrella Ladrillo King Kong de 18 huecos LARK para el periodo correspondiente de 2016 al 2021, se obtuvo un reporte de detalle de ventas y se analizó la demanda.

Tabla 3. Resumen de ventas mensual para el Ladrillo King Kong de 18 huecos LARK del local José Gálvez período 2016 al 2021.

Año	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2016									1100	1555	1130	933
2017	941	2000			1035	981	1775	1671	1728	1057	1625	1007
2018	2540	1265	1628	1442	1550	1500				1050		1266
2019	1316	1673	2000	1537	2108	1428	3000	2277	1805	1448	1420	1278
2020	2599	3288	2043	1866						1940	1984	1253
2021	3500	1945	1975	1553	1721	1547	1661	2483	1621	1538	1520	1260
CV (%)	47.74%	37.3%	10%	11.52%	27.76%	19.06%	34.6%	19.69%	20.34%	23.61%	20.26%	13.2%
Des.Std	1040.33	758.83	191.08	184.3	445.11	259.98	742.35	422.1	318.08	338	311.12	153.96
Promedio	2179	2034	1912	1600	1604	1364	2145	2144	1564	1431	1536	1166

Fuente: <https://tesis.demendiburuo.com/demanda/1>

En la tabla, se aprecia lo siguiente:

- Las ventas promedio del ladrillo King Kong 18 huecos, mantienen un comportamiento entre los meses de enero, febrero y marzo para luego disminuir los siguientes 3 meses, sin embargo, la dispersión de datos entre años es mucho más grande, reflejando valores altos en el Coeficiente de variación (CV).
- Superando valores del 20% para el coeficiente de variabilidad en gran parte de los meses, tomando en cuenta también la existencia de datos faltantes o perdidos por la empresa lo que cual incrementa la incertidumbre, con ello se concluye que se está frente a una demanda probabilística con una ligera estacionalidad.

Con esta información es suficiente para implementar un modelo de inventario más adecuado para la demanda.

4.2 En relación con el objetivo específico 2:

Identificada el tipo de demanda de la empresa DISTRIBUIDORA FERBANZ S.A.C. correspondiente a una demanda Probabilística e independiente.

Según la literatura se está frente a dos posibles modelos de inventarios para demandas Probabilísticas, **de revisión continua y revisión periódica.**

Según las entrevistas al administrador de la empresa DISTRIBUIDORA FERBANZ S.A.C. sobre el reaprovisionamiento, indica que la empresa realiza recompra de unidades bajas en stock teniendo en cuando un nivel objetivo de recompra porcentual aproximado según el almacén, por lo que se descarta el uso de un modelo de recompra o reorden por intervalos fijos de tiempo. Según los reportes de compras en Excel y facturas, designado por el administrador.

Con ello se llega a la conclusión que la empresa trabaja con reaprovisionamiento con un punto de recompra estimado, por lo que se propuso trabajar con un sistema de revisión continua, como modelo de gestión de inventarios para sus productos.

4.3 En relación con el objetivo específico 3

Definido la demanda de tipo Probabilística y el modelo de gestión de inventario propuesto de revisión continua o sistema Q. Se determina la política de gestión de inventarios tomando en cuenta las reuniones con el personal de compra y venta como el administrador de la empresa DISTRIBUIDORA FERBANZ S.A.C.

Durante las entrevistas, y revisando los reportes de compras y facturas donde detalla las fechas de compra y de entrega del producto. Se establece que los proveedores tienen un tiempo de entrega de las compras constante, pues llegan siempre en las fechas y horas pactadas.

Por lo que se concluye que el modelo de revisión continua posee demanda variable y tiempo de entrega constante.

La empresa pretende tener la facilidad de definir las políticas de nivel de servicio para los diferentes productos con el fin de mantener niveles de stock de seguridad propicios según sus objetivos, para ello se incorpora un elemento dinámico para que según se edite el valor del % de nivel de servicio los parámetros del modelo se recalculen.

Con la finalidad de lograr una revisión continua del inventario se incorpora un sistema POS (point of sale) o punto de venta para controlar las entradas y salidas del inventario.

4.4 En relación con el objetivo específico 4

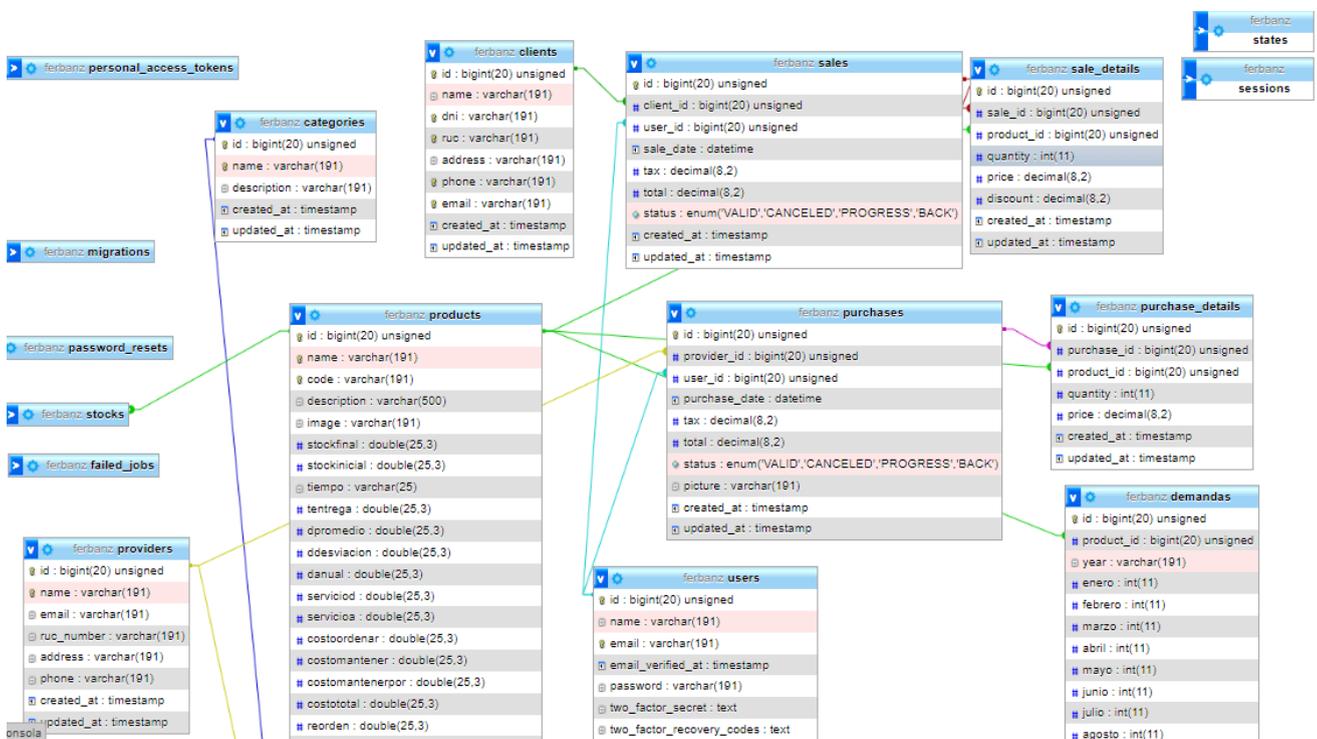
Para facilitar el acceso al desarrollo del código fuente del Dashboard, se divide en 4 etapas de desarrollo:

- 4.4.1** Diseño de la base de datos.
- 4.4.2** Sistema de autenticación.
- 4.4.3** Diseño de panel administrativo.
- 4.4.4** Diseño y programación de las funciones de lado y servidor y el cliente.

4.4.1 Diseño de base de datos

La entrevista con el personal y la gerencia de la empresa permitie establecer la relación entre los principales elementos del sistema ventas y compras con ello desarrollar el diagrama de la base de datos, desarrollado en phpMyAdmin versión 5.1.1, en los cuales se encuentra tablas como: Ventas, Compras, Detalle de compras, detalle de ventas, usuarios, clientes, proveedores, Productos, categoría, demanda, entre otros.

Figura 1. Base de datos bajo el modelo de entidad relación



Fuente: PhyMyAdmin, base de datos tesis, pestaña diseñador

La generación de las relaciones entre tablas se desarrolla en el framework Laravel v.8 a través de su sistema de migración y creación de modelos, con ello se obtiene las relaciones y llaves foráneas basándose en algunos comandos.

Figura 2. Creación de tabla detalle de compras, llaves foráneas a productos

```
web.php 2022_02_24_224621_create_purchase_details_table.php X
database > migrations > 2022_02_24_224621_create_purchase_details_table.php > CreatePurchaseDetailsTable
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  class CreatePurchaseDetailsTable extends Migration
8  {
9
10     public function up()
11     {
12         Schema::create('purchase_details', function (Blueprint $table) {
13             $table->id();
14
15             $table->unsignedBigInteger('purchase_id');
16             $table->foreign('purchase_id')->references('id')->on('purchases');
17             $table->unsignedBigInteger('product_id');
18             $table->foreign('product_id')->references('id')->on('products');
19
20             $table->integer('quantity');
21             $table->decimal('price');
22
23             $table->timestamps();
24         });
25     }
26
27     public function down()
28     {
29         Schema::dropIfExists('purchase_details');
30     }
31 }
```

Fuente: Visual Studio Code, carpeta database/migration

El tipo de relación entre las tablas que representan los modelos para el sistema en Laravel se denota en forma de funciones en su respectivo archivo de modelo.

La relación uno a muchos, para el caso mostrado es una compra que tiene muchos detalles de compra y un detalle de compra le pertenece a una compra.

Figura 3. Modelo detalle de compras relacionado con la pertenencia a una compra y a un producto.

```
app > Models > PurchaseDetails.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class PurchaseDetails extends Model
9  {
10     protected $fillable=[
11         'shopping_id',
12         'product_id',
13         'quantity',
14         'price',
15     ];
16     public function purchase() {
17         return $this ->belongsTo(Purchase::class);
18     }
19     public function product() {
20         return $this ->belongsTo(Product::class);
21     }
22 }
23
```

Fuente: Visual Studio Code, carpeta app/Models

Se desarrolla cada modelo para cada tabla especificando el tipo de relación que existe entre ellas para hacer luego uso de las peticiones a la base de datos para llamar información como almacenar o editar información nueva.

4.4.2 Sistema de autenticación

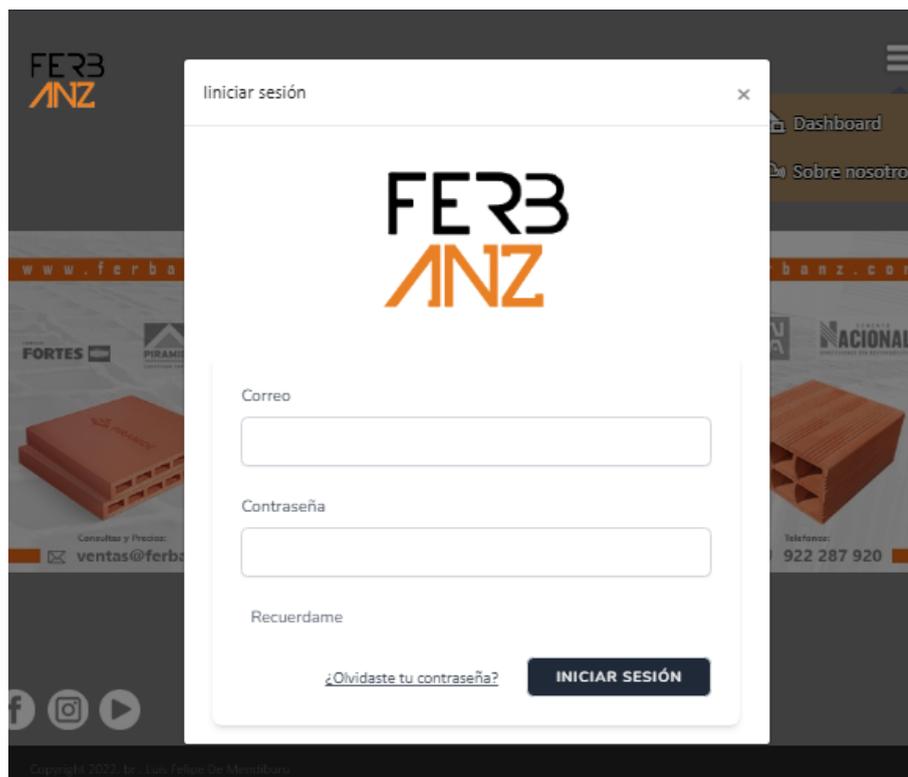
Ya que el sistema es usado para almacenar y gestionar información muy confidencial se implementa el sistema de autenticación robusto basado en el paquete Larave Jetstream en su opción de Livewire, este sistema es editado e implementando dentro de Dashboard empezado desde el sistema de autenticación.

Figura 4. Portal de inicio para la empresa DISTRIBUIDORA FERBANZ S.A.C.



Fuente: <https://tesis.demendiburuo.com/>

Figura 5. Módulo de inicio de sesión basado en Laravel Jetstream.

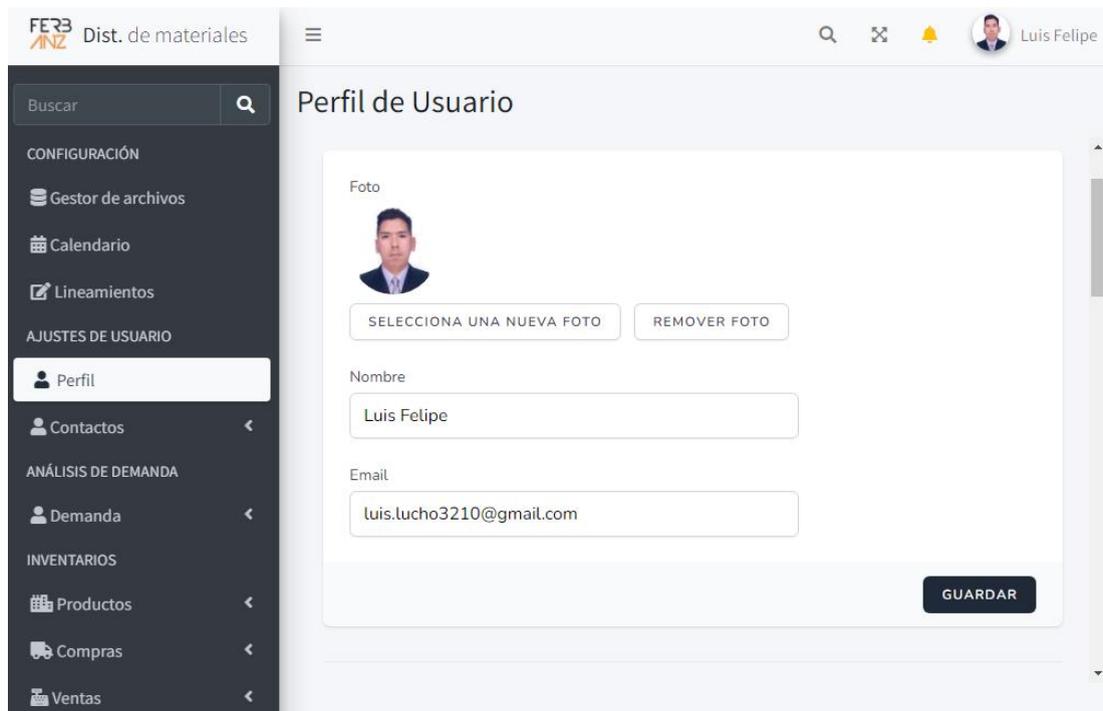


Fuente: <https://tesis.demendiburuo.com/login>

Basado en código HTML y CSS se desarrolla una interfaz de inicio, este se le asigna dentro de las rutas principales, en el archivo “*web.php*”, donde se aloja todas las rutas con sus respectivos controladores que se encargan de las funcionalidades a lo largo del desarrollo.

La plataforma tiene su apartado de perfil de usuario donde se puede editar la información, e inclusive, revisar donde se inició sesión el sistema anteriormente y además hacer uso; si se necesitase de la doble autenticación y con ello dar confianza y seguridad a la empresa.

Figura 6. Vista de perfil de usuario basado en Laravel Jetstream.



Fuente: <https://tesis.demendiburo.com/page/perfil>

Figura 7. Vista de perfil de usuario apartado gestión de sesiones del navegador.



Fuente: <https://tesis.demendiburuo.com/page/perfil>

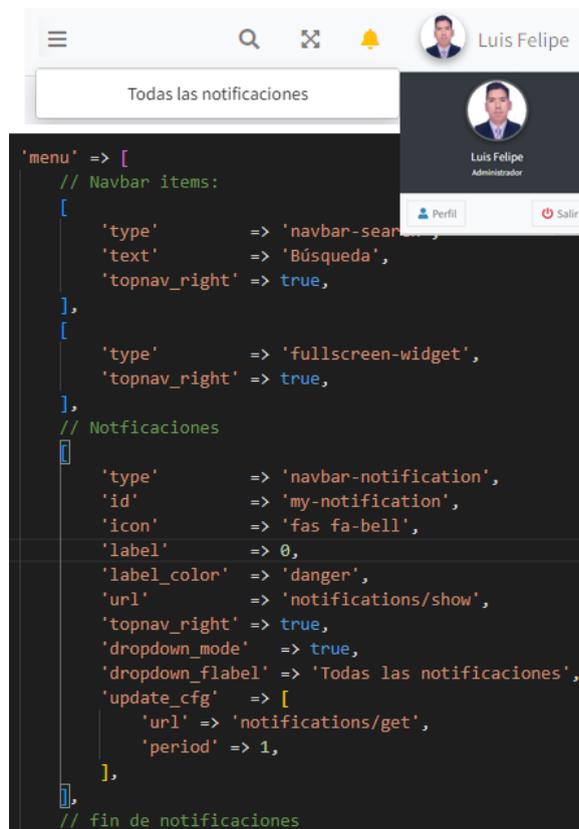
4.4.3 Diseño del panel administrativo

El Dashboard se desarrolla basado en la plantilla AdminLTE v3.1.0. que contiene los estilos los iconos perfectamente editables y reutilizables, y múltiples herramientas para formularios barras de progresos gráficas etc.

El Dashboard se divide en 3 secciones:

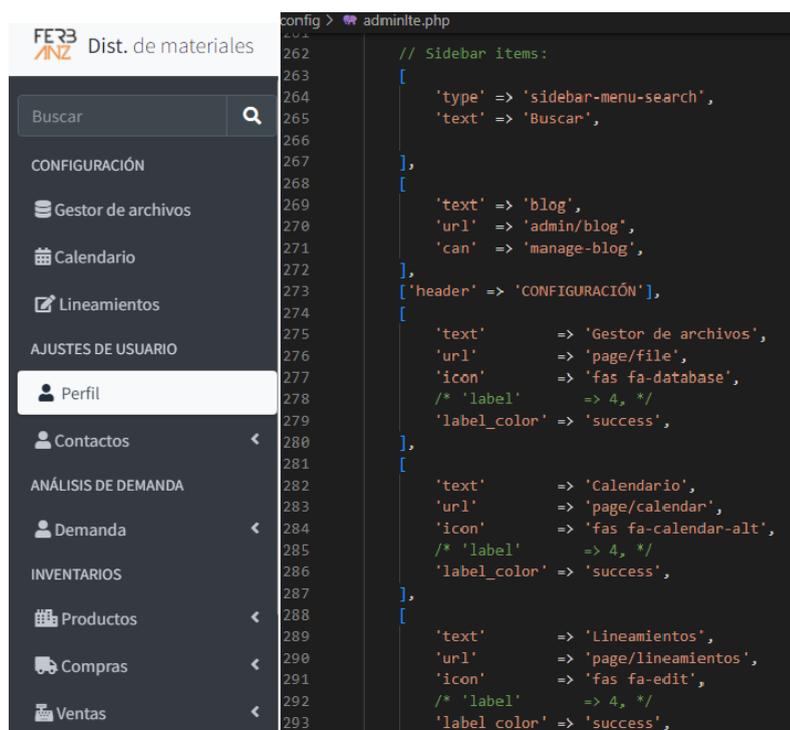
- **Barra de navegación (Navbar):** sección donde se ubica la campana de notificaciones y el panel de usuario.
- **Barra Lateral (Sidebar):** sección ubicada al lado del contenido para listar y organizar todos los elementos como barra de navegación.
- **Contenido:** sección restante donde se muestra el contenido respectivo para cada vista (página)

Figura 8. Sección de barra de navegación (Navbar) incluye el sistema de notificaciones y del perfil de usuario



Fuente: Visual Studio Code, carpeta config/adminlte.php

Figura 9. Sección de barra de lateral (Sidebar)



Fuente: Visual Studio Code, carpeta config/adminlte.php

4.4.4 Diseño y programación de las funciones de lado y servidor y el cliente.

Se trabaja los siguientes módulos para lograr un control de inventarios para revisión continua:

- **Sistema POS (Point of Sale)**, se define y programa un sistema de punto de ventas, para mantener registros en la base de datos, para ello se determina diseños para Proveedores, Clientes, Compras, Productos, Ventas.
- **Proveedores:** Se establece una interfaz de registro de proveedores con los datos (nombre, correo electrónico, número de RUC, y número de contacto)

Figura 10. Diseño de registro de proveedor

The image shows a web application interface for registering a provider. The header includes the logo 'FERB ANZ Dist. de materiales' and the user's name 'Luis Felipe'. The main content area is titled 'Proveedores' and contains a form with the following fields:

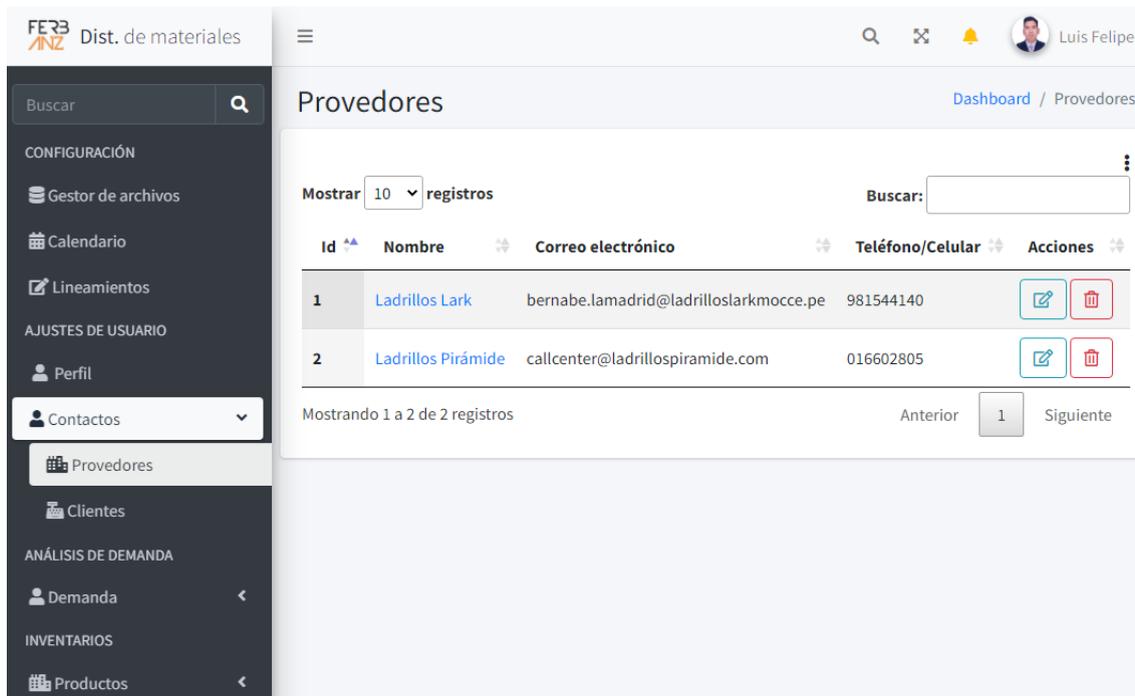
- Registro de proveedores**
- Nombre**: A text input field.
- Correo electrónico**: A text input field containing 'ejemplo@gmail.com'.
- Número de Ruc**: A text input field.
- Dirección**: A text input field.
- Número de Contacto**: A text input field.

At the bottom of the form are two buttons: 'Registrar' (blue) and 'Cancelar' (grey). A sidebar menu on the left lists various navigation options such as 'Gestor de archivos', 'Calendario', 'Lineamientos', 'AJUSTES DE USUARIO', 'Perfil', 'Contactos', 'ANÁLISIS DE DEMANDA', 'Demanda', 'INVENTARIOS', 'Productos', 'Compras', 'Ventas', 'ETIQUETAS', and 'Optimize'.

Fuente: <https://tesis.demendiburuo.com/providers/create>

A su vez, todo registro se visualiza en el apartado de Contacto >> Proveedores, donde se ubica una lista de los mismos con la finalidad de tener acceso a las funciones de editar y borrar.

Figura 11. Diseño de Listado de proveedores



Fuente: <https://tesis.demendiburuo.com/providers>

- **Clientes:** De la misma manera se establece una interface para registro y listado de clientes con las mismas características de proveedores.
- **Productos:** En el ítem de productos se diseña una interface para registrar nuevos productos donde adicionalmente podemos asignar datos como: foto del producto, el nombre del proveedor, precio de venta, el nombre, código de barras, categoría a la que pertenece, el stock, en caso que sea un producto ya existente con niveles de existencias y el máximo de capacidad de stock que se denota como stock inicial, y un color seleccionable representativo para el producto.

Figura 12. Diseño de registro de producto

FERB ANZ Dist. de materiales

Buscar

CONFIGURACIÓN

- Gestor de archivos
- Calendario
- Lineamientos

AJUSTES DE USUARIO

- Perfil
- Contactos

ANÁLISIS DE DEMANDA

- Demanda

INVENTARIOS

- Productos
- Nuevo producto
- Lista de productos
- Categorías
- Compras

Productos

Home / Productos / Registrar

Registro de productos

Nombre

Proveedor

Ladrillos Lark

Código de barras

Imagen de producto [Seleccionar Archivo](#)

Campo opcional

Precio de venta

S/.

Categoria

Ladrillos

Color

#000000

Stock actual

Stock inicial

Registrar Cancelar

Fuente: <https://tesis.demendiburuo.com/products/create>

En el ítem Productos >> Listado de productos; se tiene el registro de todos los productos, con la información de: nombre, niveles de stock (este mostrado una barra de progreso indicadora), nivel de servido, estado, acciones.

Figura 13. Diseño de listado de productos.

The screenshot displays a web application interface for product management. On the left is a dark sidebar with navigation options: 'Gestor de archivos', 'Calendario', 'Lineamientos', 'AJUSTES DE USUARIO' (Perfil, Contactos), 'ANÁLISIS DE DEMANDA' (Demanda), and 'INVENTARIOS' (Productos, Nuevo producto, Lista de productos, Categorías). The main content area is titled 'Productos' and includes a search bar, a 'Mostrar 10 registros' dropdown, and a table of products. The table has columns for 'Id', 'Imagen', 'Nombre', 'Niveles de Stock', 'Nivel de servicio', 'Estado', and 'Acciones'. Three products are listed: KingKong (stock 500/900, service 93%), Pastelero (stock 380/2500, service 88%), and Pandereta (stock 250/900, service 90%). Each product row includes a stock level bar, a service level gauge, a status indicator ('Activo'), and action buttons (edit and delete). Below the table, it shows 'Mostrando 1 a 3 de 3 registros' and pagination controls for 'Anterior', '1', and 'Siguiente'.

Id	Imagen	Nombre	Niveles de Stock	Nivel de servicio	Estado	Acciones
1		KingKong	KingKong 500/900 ▲ 471	93%	Activo ✓	
2		Pastelero	Pastelero 380/2500 ▲ 205	88%	Activo ✓	
3		Pandereta	Pandereta 250/900 ▲ 117	90%	Activo ✓	

Fuente: <https://tesis.demendiburuo.com/products>

Al hacer click en los nombres se accede a la información completa del producto, donde también haciendo uso del botón modal se visualiza los campos que se desea editar, como precios nombres, foto, descripción, nombre, etc.

Figura 14. Diseño de información de productos.

The screenshot displays the 'Información del Producto' page for 'KingKong'. The page includes a sidebar with navigation options like 'Configuración', 'Gestor de archivos', 'Calendario', 'Lineamientos', 'Ajustes de usuario', 'Perfil', 'Contactos', 'Análisis de demanda', 'Inventarios', 'Productos', 'Compras', and 'Ventas'. The main content area shows the product name 'KingKong' with a price of 's/. 1.09' and a description: 'Es el ladrillo fabricado de arcilla moldeada, extruida y quemada o cocida en un horno tipo túnel de proceso continuo.' Below the description is an 'ACTIVE' button. To the right, a table lists product details:

Nombre	KingKong	Código	123
Descripción	Es el ladrillo fabricado de arcilla moldeada, extruida y quemada o cocida en un horno tipo túnel de proceso continuo.		
Proveedor	Ladrillos Lark	Estado	ACTIVE
Categoría	Ladrillos	Nivel de Stock	500
Precio de venta	S/. 1.09	Punto de recompra	471

Below the table is a 'Modificar información' button. At the bottom, there is a date range filter from '25/08/2021' to '31/08/2021' and a 'NIVEL DE INVENTARIO' chart showing 'Nivel de Stock', 'Punto de Re-orden', and 'Inventario de seguridad'.

Fuente: <https://tesis.demendiburuo.com/products/1>

Figura 15. Diseño de ventana modal de actualización de datos del producto.

The screenshot shows the 'Actualizar datos del producto' modal window. The modal contains the following fields and options:

- Nombre:** KingKong
- Proveedor:** Ladrillos Lark
- Precio de venta:** S/. 1.09
- Descripción:** Es el ladrillo fabricado de arcilla moldeada, extruida y quemada o cocida en un
- Stock actual:** 500
- Stock inicial:** 900
- Categoría:** Ladrillos
- Color:** #007BFF
- Código de barras:** 123 (Campo opcional)
- Foto:** A photo of a red brick with 'LARK' written on it.

At the bottom of the modal are 'Actualizar' and 'Cancel' buttons.

Fuente: <https://tesis.demendiburuo.com/products/1>

- **Compras:** En este ítem se encuentra el registro de nuevas compras e historial de compras.

En el registro de compras, se tiene 2 secciones: uno dónde seleccionamos el producto y automáticamente nos trae el stock actual del producto como valor no modificable, y los campos para el formulario de cantidad precio e impuesto (IGV por defecto 18%), con un botón para agregar el producto con sus datos en el detalle de compra ubicado por debajo del botón agregar.

Con forme se van agregando valores se van mostrando el detalle de compras con las funcionalidades de borrar el ítem agregado o se registrar la compra, como también los cálculos de Total, Total con impuesto y Total a pagar todo de forma dinámica.

Figura 16. Diseño de Registro de compra de productos.

The screenshot displays the 'Registro de compras' (Purchase Register) interface. On the left is a dark sidebar with navigation options: 'Compras', 'Ventas', 'Productos', 'Categorías', 'Lista de productos', 'Nuevo producto', 'Inventarios', 'Análisis de demanda', 'Contactos', 'Perfil', 'Ajustes de usuario', 'Lineamientos', 'Calendario', and 'Gestor de archivos'. The main content area is titled 'Registro de compras' and features a form for adding a purchase item. The form includes dropdown menus for 'Producto' (set to 'Pastelero') and 'Proveedor' (set to 'Ladrillos Pirámide'), and a text input for 'stock actual' (380). Below the form are input fields for 'Cantidad', 'Precio de compra' (with a 'S/.' prefix), and 'Impuesto' (18). A blue 'Agregar producto' button is located to the right of the 'Detalles de compra' table. The table has columns for 'Eliminar', 'Producto', 'Precio(PEN)', 'Cantidad', and 'SubTotal(PEN)'. It lists two items: 'KingKong' (0.80, 1000, s/800) and 'Pastelero' (1.80, 500, s/900). At the bottom, there are summary rows for 'TOTAL: PEN 1700.00', 'TOTAL IMPUESTO (18%): PEN 306.00', and 'TOTAL PAGAR: PEN 2006.00'. A 'Registrar' button is visible at the bottom right.

Fuente: <https://tesis.demendiburo.com/purchases/create>

En el apartado de historial de compras se tiene el registro de todas las compras, donde el estado de compra detalla si la mercadería se encuentra en camino, recibido, cancelado o devuelto. Tomando en cuenta cada estado se ha programado ciertas lógicas:

En camino ⇅	En camino: representa una compra recién realizada y que le mercadería aún está por llegar,
En camino ⇅	Las únicas opciones que se desprenden de ella son <i>Recibido</i> o <i>Cancelado</i>
Recibido	
Cancelado	
Recibido ⇅	Recibido: cuando se selecciona esta opción inmediatamente todo el stock de la compra en el detalle de compras es automáticamente agregado a su respectivo producto, indicando su disponibilidad.
Recibido ⇅	
De vuelto	

Figura 17. Código de la función “change_status” responsable de aumentar el stock.

```

public function change_status(Purchase $purchase,$status)
{
    if($purchase->status == 'PROGRESS' || $purchase->status == 'BACK') {
        if($status == 1 ) {
            $purchaseDetails= $purchase->purchaseDetails;
            foreach ($purchaseDetails as $purchaseDetail) {
                $stock=$purchaseDetail->quantity;

                $product=Product::where('id','=', $purchaseDetail->product_id)->get()->first();
                $product->stockfinal= $product->stockfinal + $stock;
                if($product->stockfinal <= $product->reorden)
                {
                    $product->timeorder = Carbon::now('America/Lima');
                }
                else{
                    $product->timeorder = NULL;
                }
                $product->update();
            }
        }
    }
}

```

Fuente: Visual Studio Code, carpeta app/Http/Controller/PurchaseController

El código representa a la función **change_status** que recibe los datos de la compra seleccionada al hacer el cambio de estado del producto,

luego hace una validación para ver si se hizo el cambio de estado a: “*En camino*” o “*de vuelto*”, luego se suma al stock existente del mismo.

(*nota: en Laravel la documentación indica que es preferible trabajar con palabras en inglés, debido a esto se creó una función de traducción para visualizarlo en español, en el lado del cliente ver figura 18).*

Figura 18. Código de la función “translate” para traducir el estado del producto para la vista del cliente.

```
public function translate(){
    switch ($this->status) {
        case 'VALID':
            return 'Recibido';
            break;
        case 'BACK':
            return 'De vuelto';
            break;
        case 'CANCELED':
            return 'Cancelado';
            break;
        case 'PROGRESS':
            return 'En camino';
            break;
        default:
            # code...
            break;
    }
}
```

Fuente: Visual Studio Code, carpeta app/Models/Purchase

Cancelado ⇅

Cancelado: esta opción solo se despliega al estado “*en camino*”, presenta a una compra realizada pero que se canceló antes de recibirla. Esta opción además no tiene cambios de estado lo que si fue cancelada queda de esa forma.

De vuelto ⇅

De vuelto: este estado solo se despliega cuando previamente ha sido recibido una mercadería, denota que existió una devolución, al hacer el cambio a este estado se dispara la función **change_status** de la figura 19, pero la validación resta el stock pues previamente fue sumado al ingresar al inventario.

de forma dinámica los cálculos de Total, Total impuesto y Total a pagar con la posibilidad de remover algún ítem.

Figura 20. Diseño de Registro de compra de productos.

Registro de ventas

Registro de ventas

Cliente: Nubia

Producto: Pastelero

Precio de venta: S/. 2.70

Código de barras: 456

stock actual: 380

Porcentaje de descuento: % 0

Cantidad:

Impuesto: % 18

Detalles de venta

Eliminar	Producto	Precio Venta (PEN)	Descuento	Cantidad	SubTotal (PEN)
<input type="checkbox"/>	KingKong	1.09	0	200	s/218.00
<input type="checkbox"/>	Pandereta	0.92	0	200	s/184.00
<input type="checkbox"/>	Pastelero	2.70	0	200	s/540.00
TOTAL:					PEN 942.00
TOTAL IMPUESTO (18%):					PEN 169.56
TOTAL PAGAR:					PEN 1111.56

Cancelar Registrar

Fuente: <https://tesis.demendiburuo.com/sales/create>

Al igual que compras que posee un historial compras, en ventas posee un historial que responde a las mismas lógicas descritas en compras con sus especificaciones.

Con respecto al estado ventas, en el historial se construyó las lógicas siguientes:

En camino ⇅

En camino: representa una venta recién realizada e inmediatamente se restan los valores del detalle de ventas al stock actual de los productos correspondientes.

En camino ⇅

Recibido

Cancelado

Las únicas opciones que se desprenden de ella son *Recibido* o *Cancelado*

Recibido ⇅

Recibido ⇅

De vuelta

Recibido: cuando se selecciona esta opción solo indica que el cliente recibió la mercadería sin ningún cambio de stock, adicional a eso la única opción que se desprende de este estado sería *De vuelta*.

Cancelado ⇅

Cancelado: esta opción solo se despliega a “*en camino*”, representa una venta cancelada. Lo que inmediatamente liberará el stock de la venta. De esta opción no se despliega otra.

De vuelta ⇅

De vuelta: este estado solo se despliega cuando previamente se ha enviado la mercadería, denota que existió la entrega, pero el cliente pasó a devolverla esperando su reembolso correspondiente y para sistema implica una actualización del stock disponible.

Figura 21. Diseño de historial de ventas.

The screenshot displays a web application interface for sales management. On the left is a dark sidebar with navigation options like 'Configuración', 'Gestor de a', 'Calendario', 'Lineamien', 'Ajustes de usua', 'Perfil', 'Contactos', 'Análisis de demanda', 'Inventarios', 'Productos', 'Compras', 'Ventas', 'Nueva venta', and 'Historial de ventas'. The main area shows a 'Dashboard / Ventas' header with a search bar and a user profile for 'Luis Felipe'. Below this is a table of sales records with columns for 'Vendedor', 'Total', 'Estado', and 'Acciones'. A popup window titled 'Venta #3 por Luis Felipe a Nubia' is open, showing a detailed list of products with columns for 'Producto', 'Cantidad', and 'Total'. The popup lists three items: 490 units of a product for S/. 630.24, 370 units for S/. 1178.82, and 240 units for S/. 260.54, with a total of S/. 2069.60. The main table shows sales records with dates, times, and statuses like 'Recibido', 'Cancelado', and 'De vuelta'.

Producto	Cantidad	Total
	490	S/. 630.24
	370	S/. 1178.82
	240	S/. 260.54
		S/. 2069.60

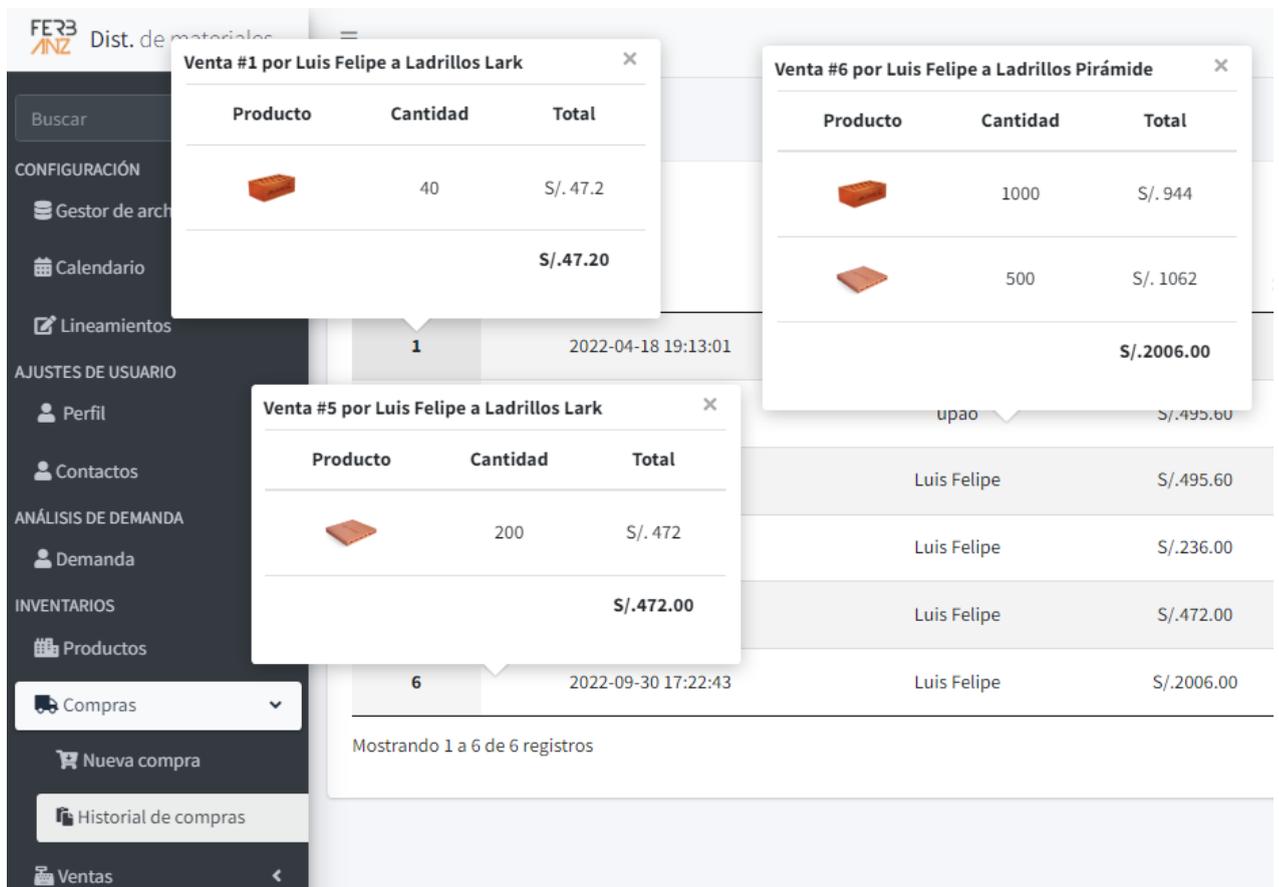
Vendedor	Total	Estado	Acciones		
Luis Felipe	S/.128.62	Recibido			
upao	S/.385.86	Recibido			
3	2022-08-23 20:03:26	Luis Felipe	S/.2069.60	Recibido	
4	2022-08-23 20:06:51	Luis Felipe	S/.3822.02	Recibido	
5	2022-08-24 15:44:01	Luis Felipe	S/.3822.02	Cancelado	
6	2022-10-03 15:35:18	Luis Felipe	S/.86.14	Cancelado	
7	2022-10-03 15:41:33	Luis Felipe	S/.185.02	De vuelta	

Fuente: <https://tesis.demendiburuo.com/sales>

- **Sistema de ventanas emergentes (POP-UP)**, se especifica 3 grupos de ventanas emergentes para facilitar la exploración de información en las áreas de Detalle de Compra, Detalle de Venta y en el apartado de Revisión Continua por producto.

Estas ventanas emergentes tienen la facultad de poder ser movidas con el mouse o cerradas si se necesitará, como si se tratase de una ventana de Windows. Se pueden abrir múltiples ventanas con fines por ejemplo de comparación de contenido.

Figura 22. Diseño de Ventana emergente para compras, múltiples ventanas abiertas



Fuente: <https://tesis.demendiburuo.com/purchases>

- **Pronóstico de Demanda e Historial**, estos dos módulos pertenecen al apartado análisis de demanda, se tomó el historial del producto estrella King Kong como base para la programación y que siempre llamé a los últimos 6 años del producto seleccionado, de una tabla llamada “demandas” esta está relacionada con los productos así que puede albergar los datos histórica de todos los productos registrados.

Figura 23. Diseño de Base de datos para la demanda de productos

Mostrando filas 0 - 5 (total de 6, La consulta tardó 0,0012 segundos)

```
SELECT * FROM `demandas`
```

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Ninguna

+ Opciones

id	product_id	year	enero	febrero	marzo	abril	mayo	junio	julio	agosto	setiembre	octubre	noviembre	diciembre	created_at
1	1	2016	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	1100	1555	1130	933	2022-09-28 11:00:00
2	1	KingKong	941	2000	NULL	NULL	1035	981	1775	1671	1728	1057	1625	1007	2022-09-28 11:00:00
3	1	2018	2540	1265	1628	1442	1550	1500	NULL	NULL	NULL	1050	NULL	1266	2022-09-28 11:00:00
4	1	2019	1316	1673	2000	1537	2108	1428	3000	2277	1805	1448	1420	1278	2022-09-28 11:00:00
5	1	2020	2599	3288	2043	1866	NULL	NULL	NULL	NULL	NULL	1940	1984	1253	2022-09-28 11:00:00
6	1	2021	3500	1945	1975	1553	1721	1547	1661	2483	1621	1538	1520	1260	2022-09-28 11:00:00

Operaciones sobre los resultados de la consulta

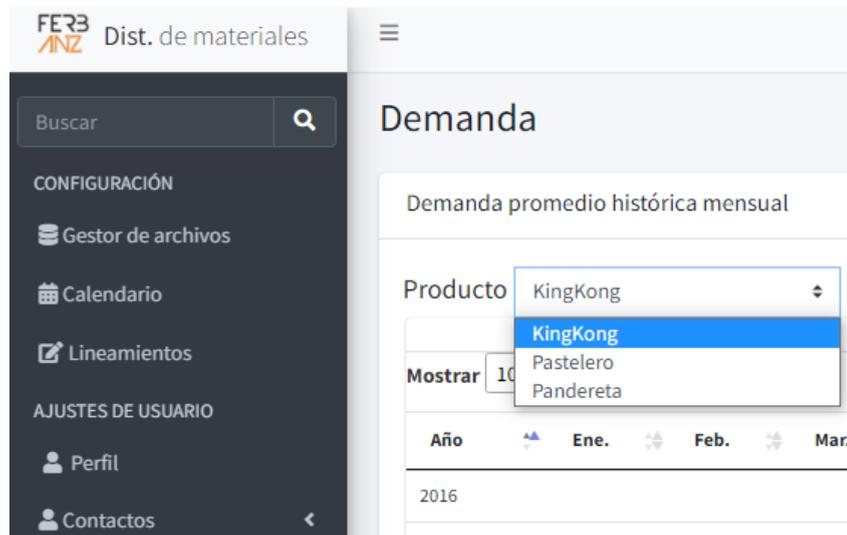
Imprimir | Copiar al portapapeles | Exportar | Mostrar gráfico | Crear vista

Fuente: PhpMyAdmin, base de datos ferbanz, tabla demandas

La interfaz de historial cuenta con dos secciones:

- **Tabla demanda mensual promedio histórica:** se muestra una tabla con los registros de las ventas según el producto en que se haya seleccionado y muestra el coeficiente de variabilidad mensual como su desviación estándar y el promedio respectivo, resaltado en negrita.

Figura 24. Selector de producto en el apartado de Demanda promedio histórica.



Fuente: <https://tesis.demendiburuo.com/demanda>

Figura 25. Tabla Demanda mensual promedio histórica Ladrillo King Kong

Año	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2016									1100	1555	1130	933
2017	941	2000			1035	981	1775	1671	1728	1057	1625	1007
2018	2540	1265	1628	1442	1550	1500				1050		1266
2019	1316	1673	2000	1537	2108	1428	3000	2277	1805	1448	1420	1278
2020	2599	3288	2043	1866						1940	1984	1253
2021	3500	1945	1975	1553	1721	1547	1661	2483	1621	1538	1520	1260
CV (%)	47.74%	37.3%	10%	11.52%	27.76%	19.06%	34.6%	19.69%	20.34%	23.61%	20.26%	13.2%
Des.Std	1040.33	758.83	191.08	184.3	445.11	259.98	742.35	422.1	318.08	338	311.12	153.96
Promedio	2179	2034	1912	1600	1604	1364	2145	2144	1564	1431	1536	1166

Fuente: <https://tesis.demendiburuo.com/demanda>

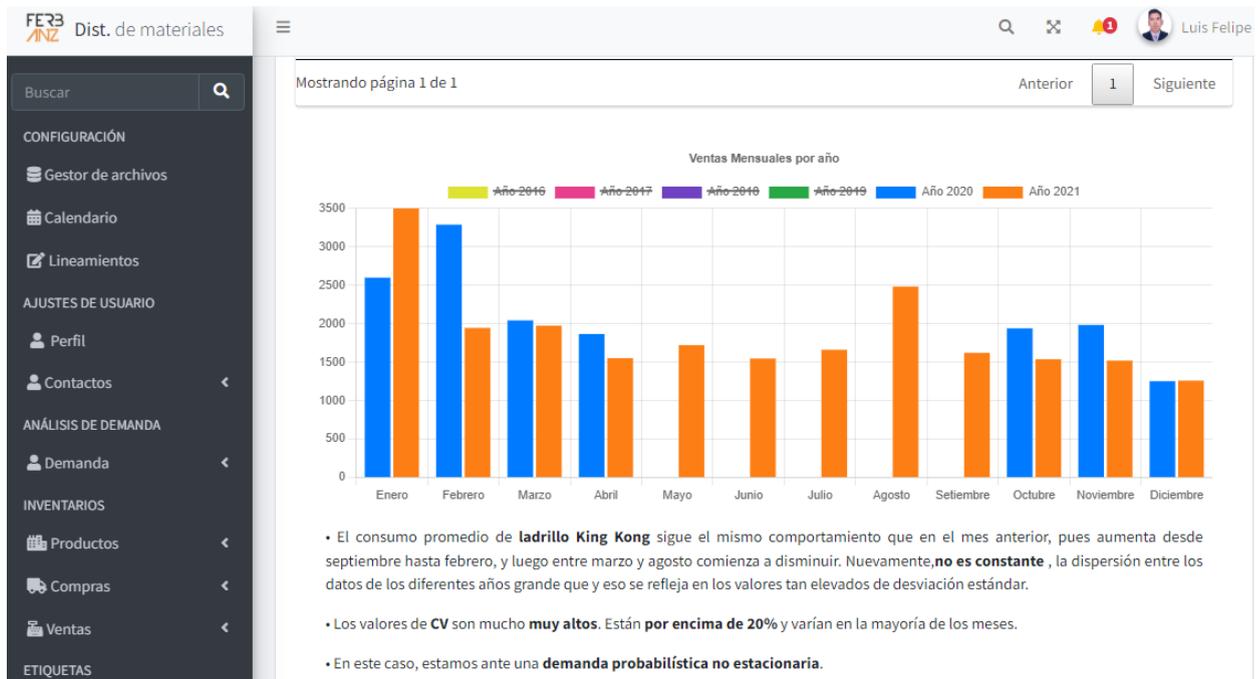
- **Gráfica de ventas mensuales por año:** corresponde a la representación gráfica de la tabla de demanda promedio mensual anual, tiene la particularidad de interactuar y visualizar los años que se desea comparar.

Figura 26. Gráfica de ventas Demanda mensual promedio histórica Ladrillo King Kong



Fuente: <https://tesis.demendiburuo.com/demanda/1>

Figura 27. Gráfica de ventas Demanda mensual promedio histórica Ladrillo King Kong, descartando los años previos 2016 al 2019



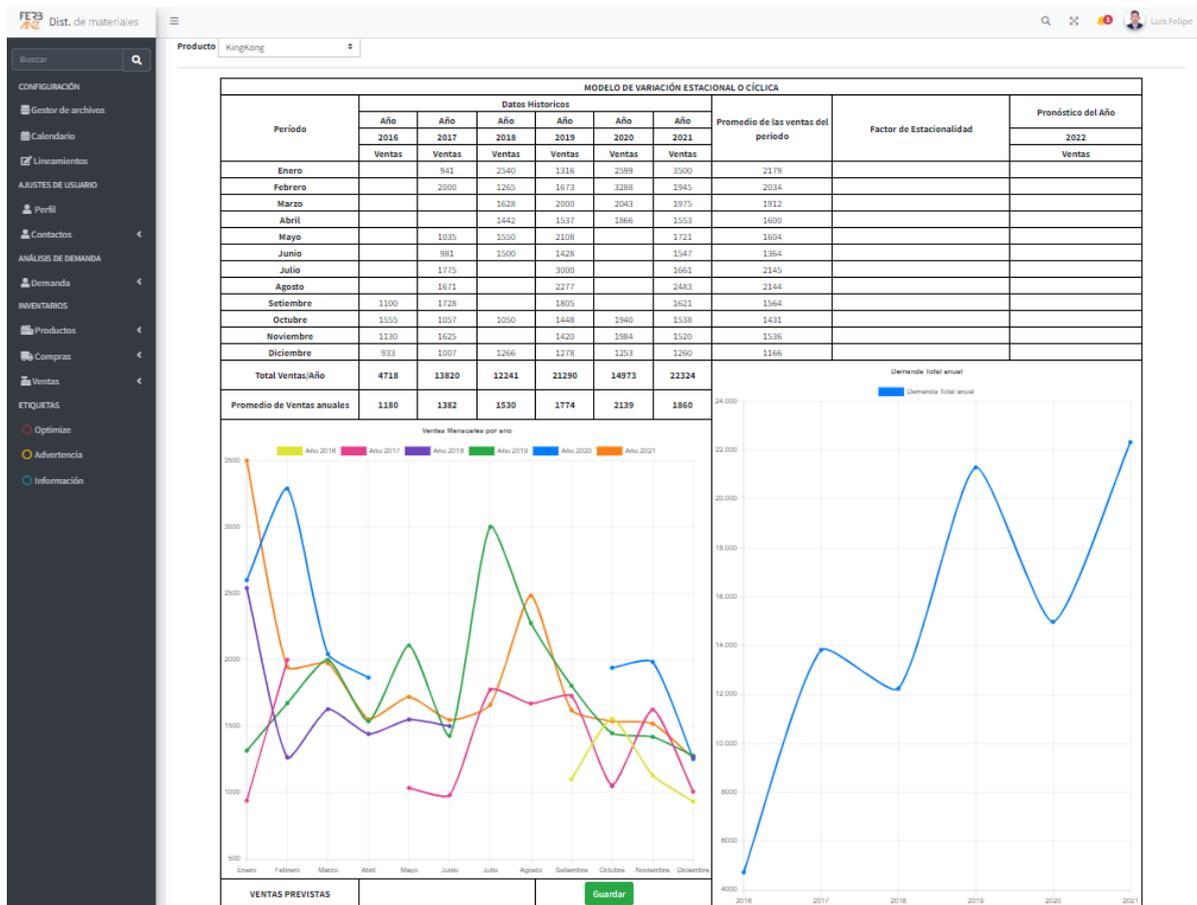
Fuente: <https://tesis.demendiburuo.com/demanda/1>

La interfaz de pronóstico cuenta con dos secciones:

- **Tabla Modelo de Variación Estacional o Cíclica:** El modelo y los cálculos se desarrollan del lado del cliente usando la librería de Javascript: JQuery para la selección de elementos para los cálculos de Factor de Estacionalidad y los respectivos pronósticos para el siguiente año por mes. Para hacer uso de ese modelo para el cliente solo se debe ingresar por teclado las ventas previstas y reactivamente se completarán los campos mencionados. Pero para ajustar más los pronósticos se apoya en el modelo de regresión lineal de las ventas totales.

Adicional a eso, también se gráfica una línea más en la gráfica de ventas mensuales por año como datos históricos en la demanda Total anual.

Figura 28. Diseño del modelo de variación estacional o cíclica para el pronóstico de demanda

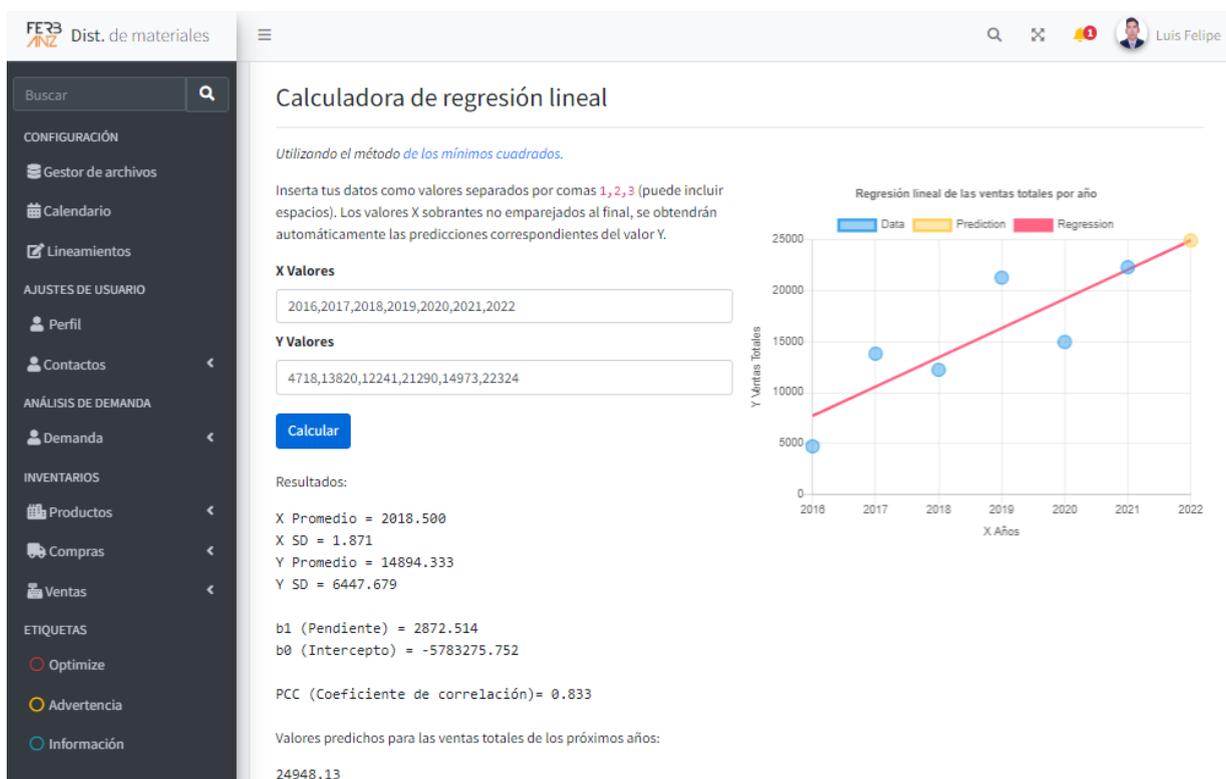


Fuente: <https://tesis.demendiburuo.com/pronostics/1>

Pronóstico de Ventas totales por regresión lineal: dado que se trabaja con estimación de promedios, en este caso se hace uso de la regresión lineal simple para la estimación el próximo año de igual forma que el apartado anterior, del lado del cliente se tiene programada toda la lógica para el cálculo y la gráfica del siguiente o siguientes puntos de la regresión.

Finalmente, al realizar el cálculo el valor del pronóstico de ventas totales para el próximo año se completa reactivamente en la tabla de modelo de variación estacional, para a su vez calcular y completar todos los pronósticos respectivos.

Figura 29. Diseño del Pronóstico de ventas totales por regresión lineal



Fuente: <https://tesis.demendiburuo.com/pronostics/1>

- **Sistema de cálculo de parámetros:** basado en el caso particular de revisión continua con demanda variable y tiempos de entrega constantes. Todos los parámetros que se necesitan para el control de

inventarios como Punto de reorden, inventario de seguridad, unidades por pedido número de ordenes al año, como los costos relacionados a ordenar, mantener y totales al año, se programaron del lado del cliente con javascript y jquery, dentro de estos cálculos se divide en 4 sectores la vista de esta página:

- **Ingreso de variables de entrada**, dado que la demanda es variable, la demanda promedio, la desviación estándar, como los costos de ordenar y mantener se muestran automáticamente y estan disponibles por teclado para el cálculo

Figura 30. Diseño del Sistema de inventario de revisión continua, entrada de parámetros

Fuente: <https://tesis.demendiburuo.com/products/1/edit>

- **Visualización del estado del inventario del producto:** en ese sector se muestra la barra de progreso que indica el nivel de existencias o stock frente al punto de reorden, adicional a eso se visualiza una alerta parpadeante que indica si el valor del stock presente es inferior al nivel de punto de reorden, de esta forma se está indicando que se necesita una recompra.

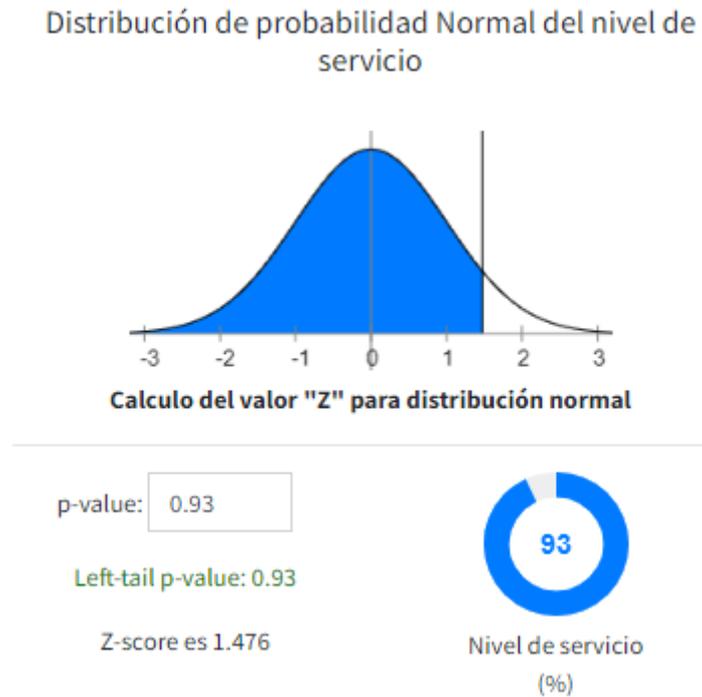
Figura 31. Barra de progreso para el nivel de existencia por producto



Fuente: <https://tesis.demendiburuo.com/products/1/edit>

- **Cálculo de “Z” para la distribución normal de nivel de servicio:** en este apartado el usuario puede variar la política del nivel de servicio del producto e inmediatamente se calcula el valor de Z correspondiente al nivel alfa, ya sea ingresado por teclado o dinámicamente girando la rueda con el mouse, que inmediatamente dispara todas las funciones javascript para los cálculos respectivos, de forma reactiva y dinámica.

Figura 32. Panel de cálculo del valor “Z” para el nivel de servicio



Fuente: <https://tesis.demendiburuo.com/products/1/edit>

- **Visualización de variables de salida:** con cada ingreso de las variables de entrada, se dispara la función Javascript “*calcular*” e inmediatamente captura los datos de las variables de entrada y procesa los cálculos según las fórmulas del modelo para luego visualizar los resultados en el apartado de cálculos.

Figura 33. Código fuente del script de la función “calcular”

```
contenido.blade.php X
resources > views > product > contenido.blade.php > script > calcular
1 <!-- CALCULOS DEMANDA ANUAL-->
2
3 <script>
4   function calcular(){
5     var a = parseFloat(document.getElementById("dpromedio").value) || 0;
6     b = parseFloat(document.getElementById("ddesviacion").value) || 0;
7     co = parseFloat(document.getElementById("costoordenar").value) || 0;
8     cm = parseFloat(document.getElementById("costomantener").value) || 0;
9     t = parseFloat(document.getElementById("tentrega").value) || 0;
10    z = parseFloat(document.getElementById("Z-score").value) || 0;
11    stockinicial={{$producto1->stockinicial}};
12    let tiempo1= document.getElementById("tiempo1").value;
13    const time = tiempo1;
14    const typetime= {
15      'dias' : a*365,
16      'semanas' : a*52,
17      'horas' : a*365*24
18    }
19    const timefinal=typetime[time];
20
21    document.getElementById('danual').value= timefinal;
22    document.getElementById('reorden').value= Math.round(a*t+z*b*Math.sqrt(t));
23    document.getElementById('reordencolor').innerText= Math.round(a*t+z*b*Math.sqrt(t));
24    document.getElementById('reorden2').innerText = Math.round(a*t+z*b*Math.sqrt(t));
25    document.getElementById('reorden2').style.color='{{$producto1->color}}';
26    puntodeorden=Math.round(a*t+z*b*Math.sqrt(t));
27    barra=puntodeorden/stockinicial*100-2.55;
28    document.getElementById('reorden3').style.marginLeft=barra+'%';
29    document.getElementById('iseguridad').value= Math.round(z*b*Math.sqrt(t));
30    document.getElementById('iseguridadcolor').innerText= Math.round(z*b*Math.sqrt(t));
31
32    document.getElementById('optimoporlote').value= Math.round(Math.sqrt(2*timefinal*co/cm));
33    document.getElementById('optimoporlotecolor').innerText= Math.round(Math.sqrt(2*timefinal*c
```

Fuente: Visual Studio Code, carpeta resources/views/producto/contenido.blade.php

Figura 34. Diseño de la presentación de resultados para los cálculos del modelo.



Fuente: <https://tesis.demendiburuo.com/products/1/edit>

- **Sistema de notificaciones,** se diseña un sistema de notificaciones por producto, capaz de comunicarse alrededor de 250ms con el servidor enviando micro peticiones de tipo “GET” para la obtención de información, con ello, se actualiza el menú desplegable de notificaciones, que contiene el número respectivo de productos que cumplen con la lógica establecida: para este caso, aquellos productos que se encuentren con nivel de stock igual o inferior al punto de reorden respectivo. Y con ello el usuario pueda ver en tiempo real si un producto necesita recompra, sin necesidad de recargar la página.

De igual manera los productos fuera de stock estan resaltados de un rojo muy tenue y en su estado se verá un triángulo de alerta, y el valor de los niveles de stock en la lista de productos parpadea de color rojo indicando que está por debajo del punto de recompra.

Figura 35. Diseño del sistema de notificaciones para la recompra

Mostrando 1 a 3 de 3 registros

Id	Imagen	Nombre	Niveles de Stock	Nivel de servicio	Estado	Acciones
1		KingKong	KingKong 460/4000 ▲ 471	93%	Activo ✓	
2		Pastelero	Pastelero 180/2500 ▲ 205	88%	Activo ✓	
3		Pandereta	Pandereta 250/900 ▲ 117	90%	Activo ✓	

Fuente: <https://tesis.demendiburuo.com/products>

Figura 36. Micro peticiones al servidor para actualizar las notificaciones

3 de 3 registros

Siguiente

Inspector Consola Depurador Red Editor de estilos

Filtrar las URL

Estado	Mét...	Dominio	Arc	Iniciador	Tipo	Transfe...	Tamaño
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	975 B
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	975 B
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	975 B
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	975 B
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	975 B
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	976 B
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	976 B
200	GET	ferbanz.hexaminimal.com	get	jquery.min.js:2 (xhr)	json	1.43 KB	976 B

330 solicitudes 319.72 KB / 471.49 KB transferido Finalizado: 5.49 min

Cabeceras Cookies Solicitud Respuesta Tiempos Traza de la pila Seguridad

En cola: 20.15 s Inicializado: 20.15 s Descargado: 20.40 s

Tiempo de peticiones

Bloqueado: 0 ms

Resolución DNS: 0 ms

Conectando: 0 ms

Configuración TLS: 0 ms

Enviando: 0 ms

Esperando: 254 ms

Recibiendo: 0 ms

Fuente: <https://tesis.demendiburuo.com/products>

Figura 37. Código fuente del Controlador de notificaciones.

```
NotificationsController.php X
app > Http > Controllers > NotificationsController.php > NotificationsController > getNotificationsData
1  <?php
2
3  namespace App\Http\Controllers;
4  use App\Models\Product;
5  use Illuminate\Http\Request;
6  use Carbon\Carbon;
7  use Illuminate\Support\Arr;
8  use Illuminate\Support\Facades\DB;
9
10 class NotificationsController extends Controller
11 {
12     public function getNotificationsData(Request $request)
13     {
14         // For the sake of simplicity, assume we have a variable called
15         // $notifications with the unread notifications. Each notification
16         // have the next properties:
17         // icon: An icon for the notification.
18         // text: A text for the notification.
19         // time: The time since notification was created on the server.
20         // At next, we define a hardcoded variable with the explained format,
21         // but you can assume this data comes from a database query.
22         $producto1= Product::wherecolumn('reorden','>=', 'stockfinal')->get()->all();
23
24         $notifications = (array) null;
25         foreach ($producto1 as $key) {
26             $producto1= $key->name;
27             $img= $key->image;
28             $producto2=new Carbon($key->timeorder);
29             $producto4=Carbon::now('America/Lima');
30             $productodiff=$producto2->diffAsCarbonInterval($producto4,false)->subHour(5);
31
```

Fuente: Visual Studio Code, carpeta app/Http/Controller/NotificationsController

4.5 En relación con el objetivo específico 5

Con la finalidad de validar o evaluar el Dashboard se diseña un plan de pruebas, este se inicia cada vez que se realiza un cambio o actualización en algún módulo, este consta de los siguientes apartados:

- **Módulos:** indica el nombre del módulo o proceso.
- **Rol o evaluadores:** muestra a los usuarios que evaluar el módulo.
- **Descripción:** breve reseña del escenario que consta el módulo a evaluar.
- **Resultado deseado:** proceso o procesos con los que cuenta el módulo a probar o testear.

Tabla 4. Plan de pruebas para el módulo usuarios

Módulo	Evaluadores	Descripción	Resultado deseado
Registro de Usuario	Administrador / Programador	Creación de usuario para acceder al sistema	Registro de usuario
Inicio de sesión	Administrador / Programador / Personal Administrativo	Autenticación para ingresar al sistema	Autenticación satisfactoria de usuario
Gestión de foto de usuario	Administrador / Programador / Personal Administrativo	Eliminar o reemplazar foto del usuario	Visualizar, Sustituir y/o borrar foto de perfil
Datos de perfil	Administrador / Programador / Personal Administrativo	Reemplazar datos	Visualizar, Sustituir y/o borrar datos del perfil
Gestión de contraseña	Administrador / Programador / Personal Administrativo	Reemplazar contraseña	Visualizar, Sustituir y/o borrar contraseña de usuario
Sesión de navegadores	Administrador / Programador / Personal Administrativo	Visualización y eliminación de sesiones	Visualizar y/o cerrar sesiones de navegadores

Nota: elaboración realizada por el autor.

Tabla 5. Plan de pruebas para el módulo proveedores

Módulo	Evaluadores	Descripción	Resultado deseado
Registro de Proveedores	Administrador / Programador	Creación de perfil de proveedores	Crear, perfil de proveedor
Listado de proveedores	Administrador / Programador / Personal Administrativo	Visualización de la lista de proveedores	Visualizar datos de proveedores más botones de acción
Edición de datos de proveedores	Administrador / Programador /	Visualización de datos y de proveedores con campos editables.	Visualizar y modificar información de proveedores
Eliminación de registro de proveedor	Administrador / Programador /	Eliminación datos de proveedores	Borrar datos de proveedor
Búsqueda de proveedores	Administrador / Programador / Personal Administrativo	Búsqueda y filtrado dinámico de proveedores	Filtrado de proveedores por todos los campos
Vista de perfil de proveedores	Administrador / Programador / Personal Administrativo	Visualización de datos del proveedor.	Visualizar los datos de proveedor

Nota: elaboración realizada por el autor.

Tabla 6. Plan de pruebas para el módulo clientes

Módulo	Evaluadores	Descripción	Resultado deseado
Registro de clientes	Administrador / Programador	Creación de perfil de clientes	Crear, perfil de cliente
Listado de clientes	Administrador / Programador / Personal Administrativo	Visualización de la lista de clientes	Visualizar datos de clientes más botones de acción
Edición de datos de clientes	Administrador / Programador /	Visualización de datos y de clientes con campos editables.	Visualizar y modificar información de clientes
Eliminación de registro de cliente	Administrador / Programador /	Eliminación datos de clientes	Borrar datos de cliente
Búsqueda de clientes	Administrador / Programador / Personal Administrativo	Búsqueda y filtrado dinámico de clientes	Filtrado de clientes por todos los campos
Vista de perfil de clientes	Administrador / Programador / Personal Administrativo	Visualización de datos del cliente.	Visualizar los datos de cliente

Nota: elaboración realizada por el autor.

Tabla 7. Plan de pruebas para el módulo historial de demanda

Módulo	Evaluadores	Descripción	Resultado deseado
Filtrado por productos	Administrador / Programador	Selección de producto y visualizar datos respectivos	Seleccionar y mostrar vista de datos del producto
Listado de ventas	Administrador / Programador / Personal Administrativo	Visualización de la lista de ventas histórica	Visualizar datos y resultados de cálculos
Vista de gráfica de ventas	Administrador / Programador /	Visualización de datos en gráficas de barras	Visualizar gráfica de ventas
Filtrado de gráfica por años	Administrador / Programador / Personal Administrativo	Visualización de gráficas seleccionando años	Filtrar datos usando botones de las gráficas
Búsqueda de ventas	Administrador / Programador / Personal Administrativo	Búsqueda y filtrado dinámico de proveedores	Filtrado de proveedores por todos los campos

Nota: elaboración realizada por el autor.

Tabla 8. Plan de pruebas para el módulo pronóstico de demanda

Módulo	Evaluadores	Descripción	Resultado deseado
Filtrado por productos	Administrador / Programador	Selección de producto y visualizar datos respectivos	Seleccionar y mostrar vista de datos del producto
Listado de ventas y promedio	Administrador / Programador	Visualización de la lista de ventas histórica y promedio	Visualizar los datos de ventas del producto
Vista de gráfica de ventas y Total	Administrador / Programador	Visualización de datos en gráficas de barras	Visualizar gráfica de ventas y total
Filtrado de gráfica por años	Administrador / Programador	Visualización de gráficas seleccionando años	Filtrar datos usando botones de las gráficas
Cálculos	Administrador / Programador	Cálculo de datos de factor de estacionalidad y pronósticos	Visualizar resultados de cálculos de pronóstico

Nota: elaboración realizada por el autor.

Tabla 9. Plan de pruebas para el módulo de productos

Módulo	Evaluadores	Descripción	Resultado deseado
Registro de productos	Administrador / Programador	Creación de productos	Crear, datos del producto
Listado de productos	Administrador / Programador / Personal Administrativo	Visualización de la lista de productos	Visualizar datos de productos más botones de acciones
Edición de datos de productos	Administrador / Programador /	Visualización de datos y de productos con campos editables.	Visualizar y modificar información de productos
Eliminación de registro de productos	Administrador / Programador /	Eliminación datos de productos	Borrar datos de productos
Cambio de estado de producto	Administrador / Programador /	Cambio el estado de productos de Activo Desactivado y viceversa	Visualizar el cambio de estado del producto
Búsqueda de productos	Administrador / Programador / Personal Administrativo	Búsqueda y filtrado dinámico de clientes	Filtrado de productos por todos los campos
Vista del producto	Administrador / Programador / Personal Administrativo	Visualización de datos del cliente.	Visualizar los datos de cliente
Registro de categoría productos	Administrador / Programador /	Visualización de lista de categorías del producto	Visualizar los datos de las categorías de los productos
Edición de datos de categoría de productos	Administrador / Programador /	Visualización de datos y de categorías con campos editables.	Visualizar y modificar información de categoría de productos
Eliminación de categoría	Administrador / Programador /	Eliminación de datos de categoría	Borrar datos de categoría

Nota: elaboración realizada por el autor.

Tabla 10. Plan de pruebas para el módulo de revisión continua

Módulo	Evaluadores	Descripción	Resultado deseado
Filtrado por productos	Administrador / Programador	Selección de producto y visualizar datos respectivos	Seleccionar y mostrar vista de datos del producto
Visualización de datos del producto	Administrador / Programador /	Visualización de la información del producto	Visualizar los datos del producto
Visualización de gráficas	Administrador / Programador /	Visualización de barras de progreso para el nivel de inventario y gráfica de la distribución normal para el nivel de servicio	Visualizar gráfica del nivel de inventario y de servicios.
Cálculos	Administrador / Programador /	Calcular parámetros de inventario	Visualizar resultados de cálculos del inventario
Reiniciar valores	Administrador / Programador /	Reinicio de los parámetros del producto	Visualizar los datos de los parámetros antes de los cambios
Guardar cambios	Administrador / Programador /	Guardar todos los cambios realizados en todos los campos del producto	Visualizar alerta de cambios en los parámetros y alerta de guardado satisfactorio

Nota: elaboración realizada por el autor.

Tabla 11. Plan de pruebas para el módulo compras

Módulo	Evaluadores	Descripción	Resultado deseado
Registro de compras	Administrador / Programador / Personal Administrativo	Creación de registro de compras, agregar productos y guardar	Ingresar valores de compras, visualizar en detalle de compras
Listado de compras	Administrador / Programador / Personal Administrativo	Visualización de la lista de compras	Visualizar datos de compras más botones de acción
Cambio de estado de compras	Administrador / Programador / Personal Administrativo	Modificar el estado de compras	Visualizar el cambio de estado de compras
Visualizar detalle de compras	Administrador / Programador / Personal Administrativo	Listado de detalle de compras vistas por ventana emergente.	Visualizar Pop-up con los detalles de compras
Búsqueda de compras	Administrador / Programador / Personal Administrativo	Búsqueda y filtrado dinámico de compras	Filtrado de compras por todos los campos
Vista de detalle de compras	Administrador / Programador / Personal Administrativo	Visualización de datos del detalle de compra.	Visualizar todos los datos del detalle de compras

Nota: elaboración realizada por el autor.

Tabla 12. Plan de pruebas para el módulo ventas

Módulo	Evaluadores	Descripción	Resultado deseado
Registro de ventas	Administrador / Programador / Personal Administrativo	Creación de registro de ventas, agregar productos y guardar	Ingresar valores de compras, visualizar en detalle de ventas
Listado de ventas	Administrador / Programador / Personal Administrativo	Visualización de la lista de ventas	Visualizar datos de ventas más botones de acción
Cambio de estado de ventas	Administrador / Programador / Personal Administrativo	Modificar el estado de ventas	Visualizar el cambio de estado de ventas
Visualizar detalle de ventas	Administrador / Programador / Personal Administrativo	Listado de detalle de ventas vistas por ventana emergente.	Visualizar Pop-up con los detalles de ventas
Búsqueda de ventas	Administrador / Programador / Personal Administrativo	Búsqueda y filtrado dinámico de ventas	Filtrado de ventas por todos los campos
Vista de detalle de ventas	Administrador / Programador / Personal Administrativo	Visualización de datos del detalle de ventas.	Visualizar los datos del detalle de ventas

Nota: elaboración realizada por el autor.

Tabla 13. Plan de pruebas para el módulo de notificaciones

Módulo	Evaluadores	Descripción	Resultado deseado
Icono de notificaciones	Administrador / Programador / Personal Administrativo	Visualizar icono de alerta de notificaciones	Visualizar el número de notificaciones
Listado de notificaciones	Administrador / Programador / Personal Administrativo	Visualización de la lista de notificaciones	Visualizar datos de los productos notificados
Re direccionar	Administrador / Programador / Personal Administrativo	Visualización de la vista de lista de productos	Re direccionar a vista lista de productos
Productos en alerta	Administrador / Programador / Personal Administrativo	Visualización de estilo de alerta por producto con ruptura de stock	Visualizar productos con fondo de alerta
Micro peticiones	Programador	Inspeccionar la página en el apartado redes para visualizar las peticiones constantes al servidor	Visualizar peticiones GET

Nota: elaboración realizada por el autor.

Para cada plan de pruebas se tiene pruebas unitarias con la finalidad de identificar posibles errores de funcionamiento, con ello darle un formato más preciso y óptimo para la calidad y manejo del sistema. Contando con elementos como:

- Datos del módulo evaluado
- Datos del evaluador
- Casos de uso
- Cumplimiento
- Datos de entrada
- Secuencia de pasos del módulo
- Errores
- Fallas ocasionadas
- Recomendaciones y observaciones

Tabla 14. Diseño de Pruebas Unitarias

Evaluación del dashboard									
		Fecha: <input type="text"/>							
Datos del modulo evaluado:									
Nombre:	<input type="text"/>								
Descripción	<input type="text"/>								
Datos del evaluador									
Nombre:	<input type="text"/>								
Areá:	<input type="text"/>								
Casos de uso	<input type="text"/>	Cumplimiento	<table border="1"> <tr> <td>Si</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Parcial</td> <td><input type="checkbox"/></td> </tr> <tr> <td>No</td> <td><input type="checkbox"/></td> </tr> </table>	Si	<input type="checkbox"/>	Parcial	<input type="checkbox"/>	No	<input type="checkbox"/>
Si	<input type="checkbox"/>								
Parcial	<input type="checkbox"/>								
No	<input type="checkbox"/>								
Datos de entrada	<input type="text"/>								
Secuencia de pasos del módulo	<input type="text"/>								
Resultados obtenidos									
Errores	<input type="text"/>								
Fallas ocasionadas	<input type="text"/>								
Recomendaciones u observaciones									
<input type="text"/>									

Nota: elaboración realizada por el autor.

V. DISCUSIÓN DE RESULTADOS

5.1 En relación el objetivo específico 1.

Identificar el tipo de demanda de la empresa DISTRIBUIDORA FERBANZ S.A.C.

(Blas López & Serrano Holguín, 2021) y (Alvites Socón & Villanueva Cruz, 2022) en sus trabajos de investigación si bien utiliza el coeficiente de variación para establecer el tipo de demanda, y concluyen que debido al alto valor de coeficiente de variación superior al 20%, se encuentran bajo un tipo de demanda probabilística, asumen que el mejor modelo a seguir es el de revisión periódica, salvo por el primero que si denota a la revisión continua pero no dejan en claro porque el uso de uno y no la otra.

Mi trabajo de investigación si bien trabaja también con el coeficiente de variación para determinar también el tipo de demanda que sigue el producto, no concluye con un uno de los dos modelos tan solo con ese parámetro considera que el tipo de demanda es Probabilística, y en el siguiente objetivo a definir es cuál modelo seguir.

5.2 En relación con el objetivo específico 2:

Definir el modelo de inventario más adecuado para la empresa DISTRIBUIDORA FERBANZ S.A.C.

(Gómez & Guzmán, 2016), en su trabajo de investigación aplica el sistema de revisión continua, denotando porque en un sistema de revisión periódica se alcanzan niveles de existencias muy altos; por lo que incrementa el costo y su baja rotación de inventarios hace inviable los proyectos en ingeniería Sólida Ltda, adicional a ello, si bien genera menos costos por las revisiones, el modelo de revisión periódica no es lo recomendable para la producción, partes críticas o para productos con altos costos de almacenamiento.

Este trabajo de investigación concluye que para mantener un sistema probabilístico de inventarios más controlado es necesario un panel administrativo web o Dashboard para controlar la entrada y salida de productos, así hacer frente a los costos de hacer varias revisiones y los costos por mantener un sobre dimensionamiento de inventarios por la revisión continúa. Además, aprovecha que

la empresa ya mantiene un trabajo de reaprovisionamiento con punto de recompra.

5.3 En relación con el objetivo específico 3.

Establecer la política de inventario para la empresa DISTRIBUIDORA FERBANZ S.A.C.

(Lam Noroña, 2019) en su trabajo de maestría en el apartado referente a la política de inventarios después que divide ambos modelos y especifica los pros y contras de ambos también denota que para identificar la política de inventario se debe establecer claramente todo lo referente a los cálculos para mantener los objetivos de la política de inventarios definidos como: nivel óptimo de inventario, mantener niveles adecuadas a través del seguimiento de inventario, maximizar ganancias y buen nivel de servicio.

Este trabajo de investigación, por su parte para define la política de revisión continua entre sus variantes, se basa siguiendo la naturaleza de los pedidos y entregas de los proveedores de la empresa DISTRIBUIDORA FERBANZ S.A.C, se determina que el inventario está bajo una demanda variable ya definida y un tiempo de entrega de productos constantes. Con esto, se define el abanico de cálculos que necesita el sistema para trabajar.

5.4 En relación con el objetivo específico 4.

Diseñar el código fuente del entorno web de la aplicación para la empresa DISTRIBUIDORA FERBANZ S.A.C.

(Acurana, 2017) En su trabajo de maestría, hace referencia al código fuente como medio de orden que se integra a un editor de código, con la finalidad de construir un desarrollo de aplicación en un entorno de software.

Este trabajo de investigación, para diseñar el código fuente sigue los esquemas definidos en la documentación oficial del framework Laravel y de la plantilla AdminLTE para lograr un orden y agilizar el desarrollo. De esta forma generar un código amigable con una curva de aprendizaje media.

5.5 En relación con el objetivo específico 5.

Validar el modelo de reposición de inventario bajo la plataforma web para la empresa DISTRIBUIDORA FERBANZ S.A.C.

(Robalino Gomez, 2019) En su trabajo de investigación, menciona que una vez concluido el desarrollo del sitio web, comienza con la evaluación de su funcionabilidad e interactividad bajo criterios establecidos ya sea por la empresa o por el desarrollador o equipo de desarrolladores. El resultado indica de qué manera el sitio web se encuentre listo o no para estar en marcha o si requiera mejorar ciertos aspectos.

Este trabajo de investigación, su evaluación se denota como validación pues como es un sistema en desarrollo, deja abierto varios aspectos para ser implementado más adelante con ello tener escalabilidad, afinar ciertos detalles conformé la empresa así lo requiera.

VI. CONCLUSIONES

Después de haber realizado el desarrollado del Dashboard se puede concluir en lo siguiente.

1. Con respecto al objetivo específico 1 se pudo determinar la demanda tomando al producto estrella ladrillo King Kong analizando el historial manejado por la empresa en Excel, Asimismo, se pudo observar en algunos meses durante los años 2016 al 2021 ocurre que no todos los meses sobrepasaban el 20% de la variabilidad para ser denotados como demandas probabilísticas, sin embargo, la clara segmentación de la tendencia nos impide tomar esos períodos como demandas determinísticas. Adicional a ello, se tiene también la presencia de valores faltantes lo que fortalece la premisa de hacer uso de la demanda probabilística.

2. Al definirse el modelo de revisión continua, se mantuvo el mismo ritmo de trabajo que maneja la empresa DISTRIBUIDORA FERBANZ S.A.C. de realizar recompras al llegar a un valor, que en el sistema ahora es fácilmente definible y monitoréale.
3. Estableciendo la política de inventario de revisión continua con tiempos de entrega constante, pero demanda variable, para la empresa fue determinante las entrevistas y conversaciones sobre el panorama de la gestión de inventarios, pues con ella se descarta otras posibilidades variantes del mismo modelo, pues de manera empírica se estaba trabajando ya con un modelo de punto de reorden con las características antes mencionadas.
4. El diseño del código fuente del entorno de la aplicación web, realizado por el framework Laravel, brinda una amplia gama de características que ayudaron a la agilización de la velocidad del desarrollo, como una alta seguridad, autenticación poderosa, una comunidad y soporte exclusivo al código abierto, la facilidad de crear migraciones a la base de datos sin tocar código SQL entre otras bondades.
5. La validación del modelo, permitie levantar algunos problemas que fueron surgiendo en el desarrollo del Dashboard, gracias al framework Laravel y sus descripciones de los fallos, se logra hacer el seguimiento a los errores y fallas, además de las encontradas y descritos por el personal de evaluación. Y con ello se logra hacer las modificaciones para conseguir el resultado esperado por los módulos.

VII. RECOMENDACIONES

Con el fin de mantener la sostenibilidad en el logro para la implementación del panel administrativo (Dashboard) más completo y libre de errores, se recomienda lo siguiente:

- Establecer equipos de trabajo con el personal de ventas y compras con la finalidad de lograr retroalimentación del sistema.
- Realizar una migración de la versión el lenguaje de programación de PHP v.7.4 a la versión 8 o superior, en el servidor de alojamiento así se puede contar con las nuevas mejoras, de manejo de alta densidad de datos y fluidez.
- Se recomienda en un futuro muy próximo la creación de nuevos módulos que permitan complementar más la herramienta, como el Análisis ABC, módulos para visualizar reportes más completos, de costos, contabilidad, entre otros.
- Promover reuniones con la gerencia con el fin de digitalizar toda su base de datos de Excel al sistema, como también: capacitaciones para los usuarios en el uso del sistema, para ingresar y manipular de forma correcta la entrada y salida de datos, para minimizar los errores humanos.

REFERENCIAS BIBLIOGRAFICAS

- [Alejsa, L. \(31 de Julio de 2018\). Portal de informática, internet, tecnologías y web. Obtenido de https://www.alegsa.com.ar/Dic/aplicacion_web.php](https://www.alegsa.com.ar/Dic/aplicacion_web.php)
- [Alvarez, M. A. \(28 de Julio de 2020\). Teoría de la Programación Orientada a Objetos. Obtenido de Qué es MVC: https://desarrolloweb.com/articulos/que-es-mvc.html](https://desarrolloweb.com/articulos/que-es-mvc.html)
- [Andyfg. \(24 de Marzo de 2021\). AdminLTE \(implementación\). Obtenido de https://phpscript.cubava.cu/2021/03/24/adminlte-implementacion/](https://phpscript.cubava.cu/2021/03/24/adminlte-implementacion/)
- [Buzón, M. \(26 de Enero de 2020\). Definición de software: Qué es, para que sirve y porque es tan importante. Obtenido de https://www.profesionalreview.com/2020/01/26/definicion-software/](https://www.profesionalreview.com/2020/01/26/definicion-software/)
- [Chávez, G. \(27 de Enero de 2021\). ¿Qué es Laravel Jetstream? Obtenido de https://gabrielchavez.me/que-es-laravel-jetstream/](https://gabrielchavez.me/que-es-laravel-jetstream/)
- [Garcia, M. \(20 de Mayo de 2020\). ¿QUE ES XAMPP Y COMO PUEDO USARLO? Obtenido de https://www.nettix.com.pe/blog/web-blog/que-es-xampp-y-como-puedo-usarlo](https://www.nettix.com.pe/blog/web-blog/que-es-xampp-y-como-puedo-usarlo)
- [Hernández, K. \(30 de Marzo de 2021\). Backend y Frontend, ¿Qué es y cómo funcionan en la programación? Obtenido de https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programaci%C3%B3n-de-una-aplicaci%C3%B3n-web](https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programaci%C3%B3n-de-una-aplicaci%C3%B3n-web)
- [Ingeniosa Empresa. \(7 de marzo de 2018\). Modelos probabilísticos de inventario. Obtenido de https://ingenioempresa.com/modelos-probabilisticos-inventario/](https://ingenioempresa.com/modelos-probabilisticos-inventario/)
- [Krajewski, M. &. \(2008\).](#)
- [Mrhardware. \(18 de Diciembre de 2018\). Definición de Hardware. Obtenido de https://mrhardware.es/definicion-de-hardware/](https://mrhardware.es/definicion-de-hardware/)
- [Munte, G. \(8 de Enero de 2020\). Framework: ¿qué es y cuál es su función en Internet? Obtenido de https://rockcontent.com/es/blog/framework/](https://rockcontent.com/es/blog/framework/)
- [Ortiz, D. \(20 de Octubre de 2021\). ¿Qué es un Dashboard y para qué se usa? \(2022\). Obtenido de https://www.cyberclick.es/numerical-blog/que-es-un-Dashboard](https://www.cyberclick.es/numerical-blog/que-es-un-Dashboard)
- [Palacios, D. \(02 de Julio de 2021\). Instalación de Laravel Livewire. Obtenido de https://styde.net/instalacion-de-laravel-livewire/#:~:text=Livewire%20es%20un%20framework%20para,con%20la%20instalaci%C3%B3n%20de%20Livewire.](https://styde.net/instalacion-de-laravel-livewire/#:~:text=Livewire%20es%20un%20framework%20para,con%20la%20instalaci%C3%B3n%20de%20Livewire.)
- [Rodríguez, E. \(27 de Enero de 2020\). Qué es backend y por qué es tan importante](#)

[para tu web. Obtenido de https://www.seoestudios.es/blog/que-es-backend-web/](https://www.seoestudios.es/blog/que-es-backend-web/)

[Vera, R. A. \(31 de Marzo de 2021\). Qué es Laravel: Características y ventajas / OpenWebinars. Obtenido de https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/](https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/)

[web, E. \(20 de 10 de 2020\). Control de Inventarios. Obtenido de https://controlinventarios.wordpress.com/2020/10/20/sistema-de-revision-continua/](https://controlinventarios.wordpress.com/2020/10/20/sistema-de-revision-continua/)

[Yair. \(23 de Diciembre de 2019\). Qué es Composer y cómo usarlo. Obtenido de https://styde.net/que-es-composer-y-como-usarlo/](https://styde.net/que-es-composer-y-como-usarlo/)

ANEXOS

ANEXO 1 Manual de instalación del Framework Laravel en Windows 10

Paso 1: Instalar Composer

Laravel requiere el paquete de dependencias de PHP composer, se descarga de la página web oficial y se configura siguiendo los pasos del manual de instalación. Usando la siguiente dirección (<https://getcomposer.org/doc/00-intro.md#installation-windows>)

Imagen 01. Installation Windows

Installation - Windows

Using the Installer

This is the easiest way to get Composer set up on your machine.

Download and run [Composer-Setup.exe](#). It will install the latest Composer version and set up your PATH so that you can call `composer` from any directory in your command line.

Note: Close your current terminal. Test usage with a new terminal: This is important since the PATH only gets loaded when the terminal starts.

Manual Installation

Change to a directory on your `PATH` and run the installer following [the Download page instructions](#) to download `composer.phar`.

Create a new `composer.bat` file alongside `composer.phar`:

Using cmd.exe:

```
C:\bin> echo @php "%~dp0composer.phar" %*>composer.bat
```

Using PowerShell:

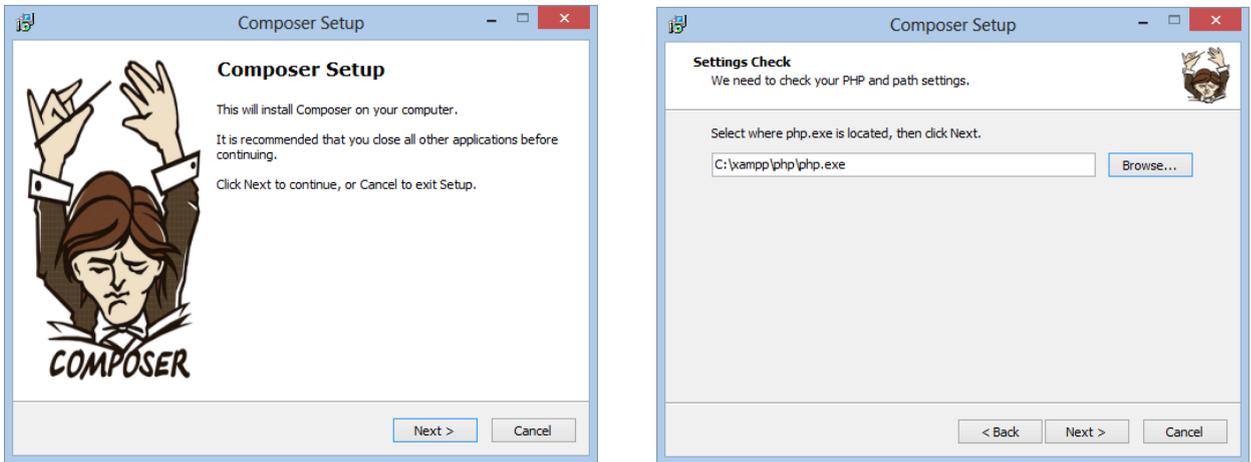
```
PS C:\bin> Set-Content composer.bat '@php "%~dp0composer.phar" %*'
```

Add the directory to your PATH environment variable if it isn't already. For information on changing your PATH variable, please see [this](#)

Fuente: Página web de Composer

Paso 2: Ejecutar el instalador

Imagen 02: Instalando Composer



Fuente: capturas de Instalación de Composer

Durante la instalación, se exige que se indique la ruta del ejecutable de PHP, en este caso se está trabajando con XAMPP el ejecutable de PHP se encuentra en la ruta C:\xampp\php\ y de damos "Next".

Imagen 03: Instalando Composer

```
cmd C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Versión 6.2.9200]
(c) 2012 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Super>php -v
PHP 5.5.15 (cli) (built: Jul 23 2014 15:05:09)
Copyright (c) 1997-2014 The PHP Group
Zend Engine v2.5.0, Copyright (c) 1998-2014 Zend Technologies

C:\Users\Super>composer -version

Composer version 1.0-dev (a309e1d89ded6919935a842faeaed8e888fbfe37) 2014-10-20 19:16:14

Usage:
  [options] command [arguments]

Options:
  --help           -h Display this help message.
  --quiet          -q Do not output any message.
  --verbose        -v|vv|vvv Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug
  --version        -V Display this application version.
  --ansi           Force ANSI output.
  --no-ansi       Disable ANSI output.
  --no-interaction -n Do not ask any interactive question.
  --profile        Display timing and memory usage information
  --working-dir   -d If specified, use the given directory as working directory
```

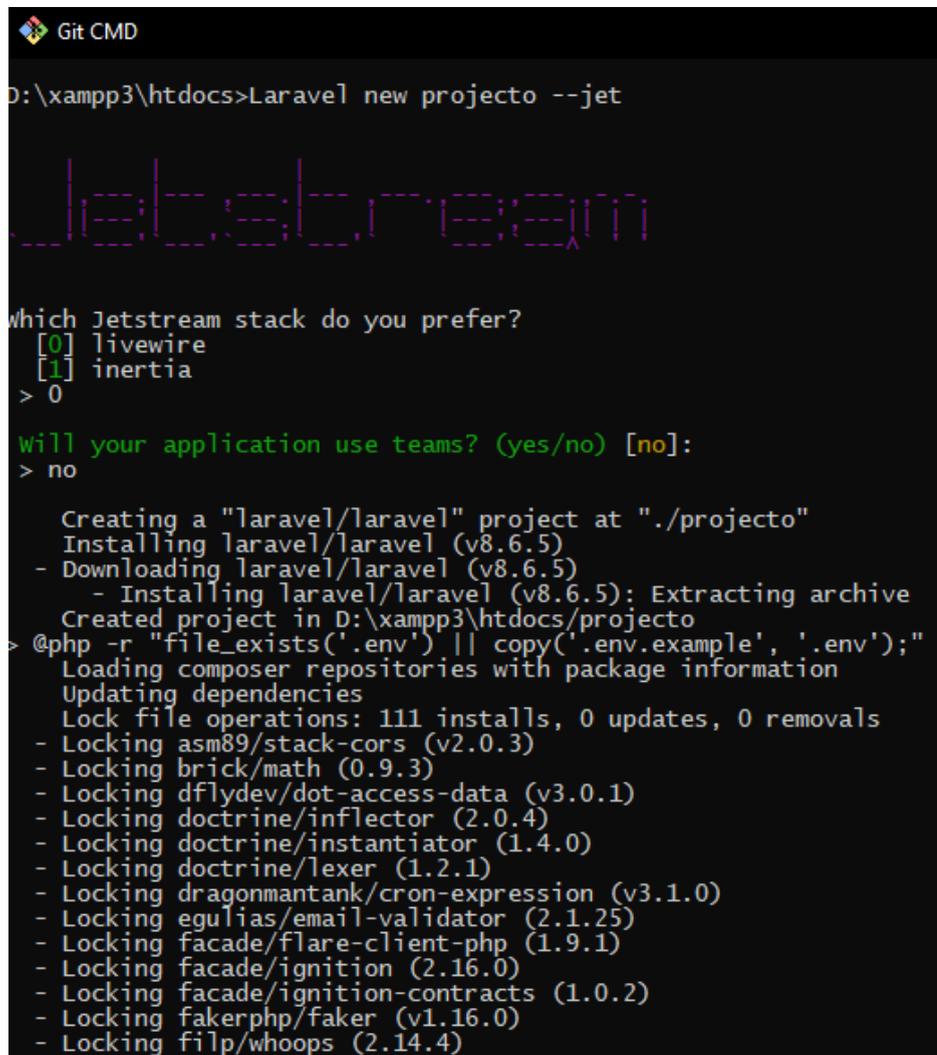
Fuente: capturas de pantalla del terminal CMD

Finalizado la instalación de Composer se comprueba su funcionamiento en Windows, con el comando: "composer --version".

Paso 3: Crear el proyecto usando Laravel

El siguiente comando se instala Laravel y Jetstream juntos:

Imagen 04: Instalando el proyecto en Laravel



```
Git CMD
D:\xampp3\htdocs>Laravel new proyecto --jet

jetstream

which Jetstream stack do you prefer?
[0] livewire
[1] inertia
> 0

will your application use teams? (yes/no) [no]:
> no

Creating a "laravel/laravel" project at "./proyecto"
Installing laravel/laravel (v8.6.5)
- Downloading laravel/laravel (v8.6.5)
  - Installing laravel/laravel (v8.6.5): Extracting archive
Created project in D:\xampp3\htdocs/proyecto
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 111 installs, 0 updates, 0 removals
- Locking asm89/stack-cors (v2.0.3)
- Locking brick/math (0.9.3)
- Locking dflydev/dot-access-data (v3.0.1)
- Locking doctrine/inflector (2.0.4)
- Locking doctrine/instantiator (1.4.0)
- Locking doctrine/lexer (1.2.1)
- Locking dragonmantank/cron-expression (v3.1.0)
- Locking egulias/email-validator (2.1.25)
- Locking facade/flare-client-php (1.9.1)
- Locking facade/ignition (2.16.0)
- Locking facade/ignition-contracts (1.0.2)
- Locking fakerphp/faker (v1.16.0)
- Locking filp/whoops (2.14.4)
```

Fuente: capturas de pantalla del terminal Git CMD.

Paso 4: configuración de la base de datos

Dentro de carpeta donde se crea el proyecto se busca el archivo ". ENV" este archivo contiene información personal de acceso al proyecto. Se coloca el

nombre de la de la base de datos del proyecto en DB_DATABASE.

Imagen 05: Definiendo el nombre de la base de datos

```
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:UhE8eRcqqAYY2Sut5LLmzgF7AqAVdCdB+Wr05rK7Rdc=
4 APP_DEBUG=true
5 APP_URL=http://panel.test
6
7 LOG_CHANNEL=stack
8 LOG_LEVEL=debug
9
10 DB_CONNECTION=mysql
11 DB_HOST=127.0.0.1
12 DB_PORT=3306
13 DB_DATABASE=panel
14 DB_USERNAME=root
15 DB_PASSWORD=
16
17 BROADCAST_DRIVER=log
18 CACHE_DRIVER=file
19 FILESYSTEM_DRIVER=local
20 QUEUE_CONNECTION=sync
21 SESSION_DRIVER=database
22 SESSION_LIFETIME=120
```

Fuente: capturas de pantalla en el Visual Studio Code

ANEXO 2. Manual de instalación de la plantilla ADMIN LTE 3.

Paso 1: Instalar dependencias con composer.

En la carpeta raíz del proyecto de Laravel, se ejecuta el comando: “composer require jeroennoten/laravel-adminlte”.

Imagen 06: Instalando dependencias del ADMIN LTE 3

```
Git CMD
programa o archivo por lotes ejecutable.

D:\xampp3\htdocs\projecto>composer require jeroennoten/laravel-adminlte
Using version ^3.7 for jeroennoten/laravel-adminlte
./composer.json has been updated
Running composer update jeroennoten/laravel-adminlte
Loading composer repositories with package information
Updating dependencies
Lock file operations: 2 installs, 0 updates, 0 removals
  - Locking almasaeed2010/adminlte (v3.1.0)
  - Locking jeroennoten/laravel-adminlte (v3.7.1)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 2 installs, 0 updates, 0 removals
  - Installing almasaeed2010/adminlte (v3.1.0): Extracting archive
  - Installing jeroennoten/laravel-adminlte (v3.7.1): Extracting archive
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
Discovered Package: fruitcake/laravel-cors
Discovered Package: jenssegers/agent
Discovered Package: jeroennoten/laravel-adminlte
Discovered Package: laravel/fortify
Discovered Package: laravel/jetstream
Discovered Package: laravel/sail
Discovered Package: laravel/sanctum
Discovered Package: laravel/tinker
Discovered Package: livewire/livewire
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
81 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi
No publishable resources for tag [laravel-assets].
Publishing complete.

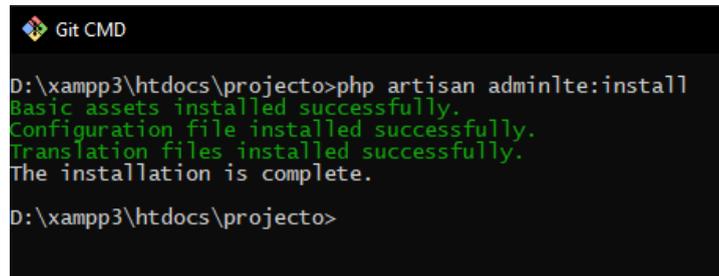
D:\xampp3\htdocs\projecto>
```

Fuente: capturas de pantalla del terminal Git CMD.

Paso 2: Instalar plantilla ADMINLTE 3.

Se instalan los recursos del paquete necesarios con el siguiente comando: “php artisan adminlte:install”. El comando instala la plantilla AdminLTE y sus dependencias (Bootstrap, jQuery, etc.) en su carpeta pública / vendor, la configuración del paquete en el archivo config / adminlte.php y las traducciones del paquete en resources / lang / vendor / adminlte / folder.

Imagen 07: Instalando dependencias del ADMIN LTE 3



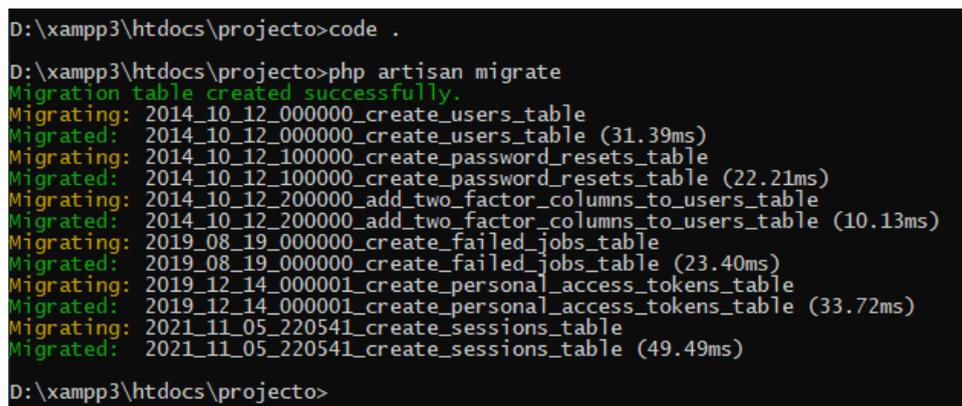
```
Git CMD
D:\xampp3\htdocs\projecto>php artisan adminlte:install
Basic assets installed successfully.
Configuration file installed successfully.
Translation files installed successfully.
The installation is complete.
D:\xampp3\htdocs\projecto>
```

Fuente: capturas de pantalla del terminal Git CMD.

Paso 3: Configurar ADMIN LTE 3

Se prepara la primera migración de datos para que la tabla usuario este se puede crear usando el comando: “php artisan migrate”.

Imagen 08: Realizando la primera migración para la creación de tabla usuario.

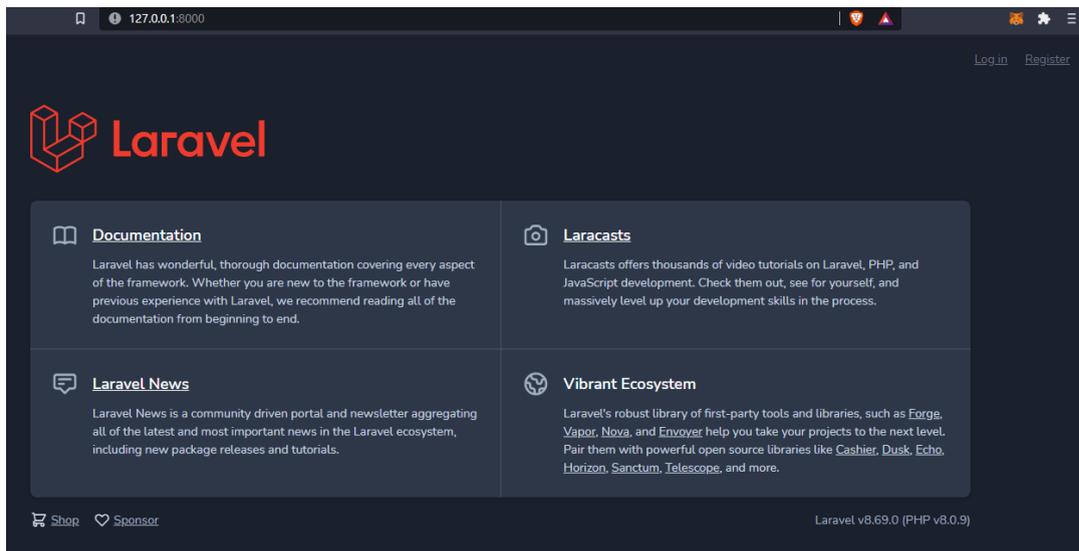


```
D:\xampp3\htdocs\projecto>code .
D:\xampp3\htdocs\projecto>php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table (31.39ms)
Migrated: 2014_10_12_000000_create_users_table (31.39ms)
Migrating: 2014_10_12_100000_create_password_resets_table (22.21ms)
Migrated: 2014_10_12_100000_create_password_resets_table (22.21ms)
Migrating: 2014_10_12_200000_add_two_factor_columns_to_users_table (10.13ms)
Migrated: 2014_10_12_200000_add_two_factor_columns_to_users_table (10.13ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table (23.40ms)
Migrated: 2019_08_19_000000_create_failed_jobs_table (23.40ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table (33.72ms)
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (33.72ms)
Migrating: 2021_11_05_220541_create_sessions_table (49.49ms)
Migrated: 2021_11_05_220541_create_sessions_table (49.49ms)
D:\xampp3\htdocs\projecto>
```

Fuente: capturas de pantalla del terminal Git CMD

Una vez se genera las tablas se ejecuta el comando “php artisan sever”, para que se comunique con el servidor y se pueda ingresar al navegador usando la ruta predeterminar: “http://127.0.0.1:8000/”

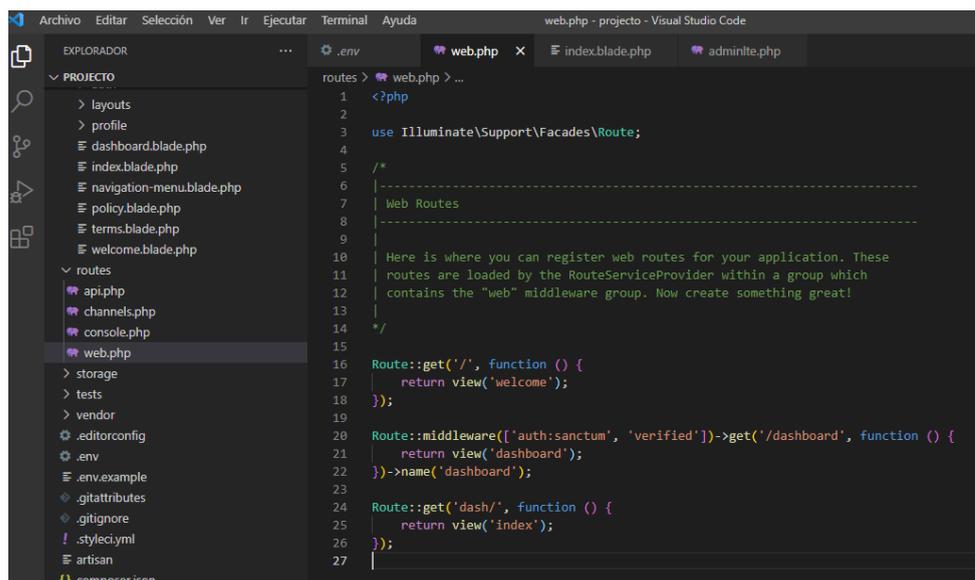
Imagen 09: Página principal del Proyecto por defecto.



Fuente: capturas de pantalla del navegador.

Ahora se define la ruta principal en el archivo de rutas web.php.

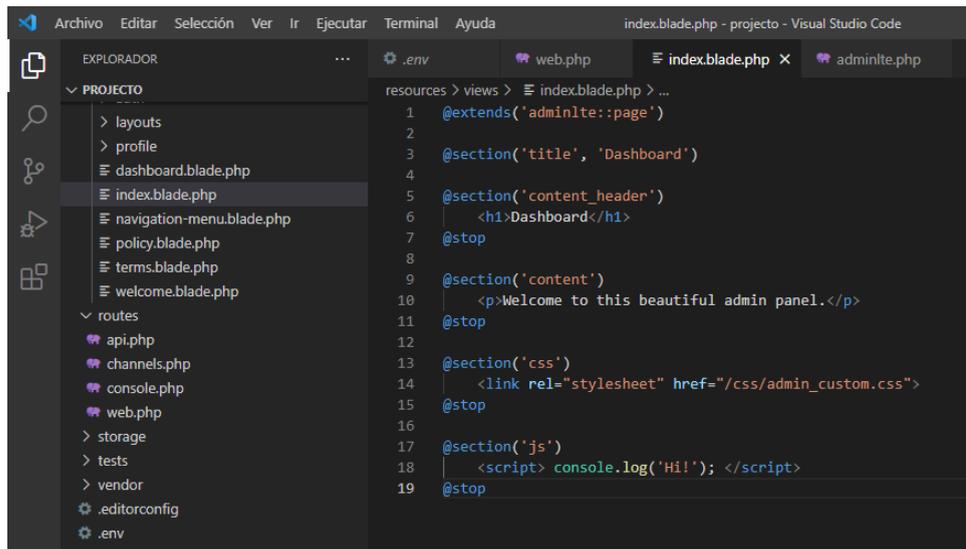
Imagen 10: Archivo de rutas web.php.



Fuente: captura de pantalla de Visual Studio Code.

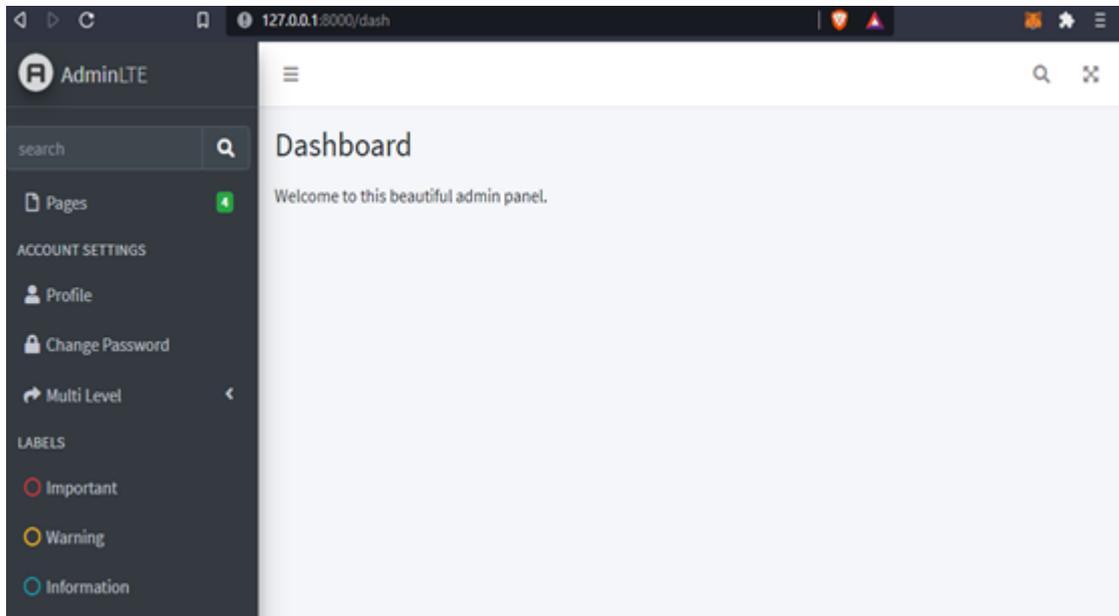
Se Prepara la primera vista del Dashboard ADMINTLTE 3, usando las sentencias base para la generación del Menú de navegación y El Sidebar.

Imagen 11: Estructura de plantilla “Blade”.



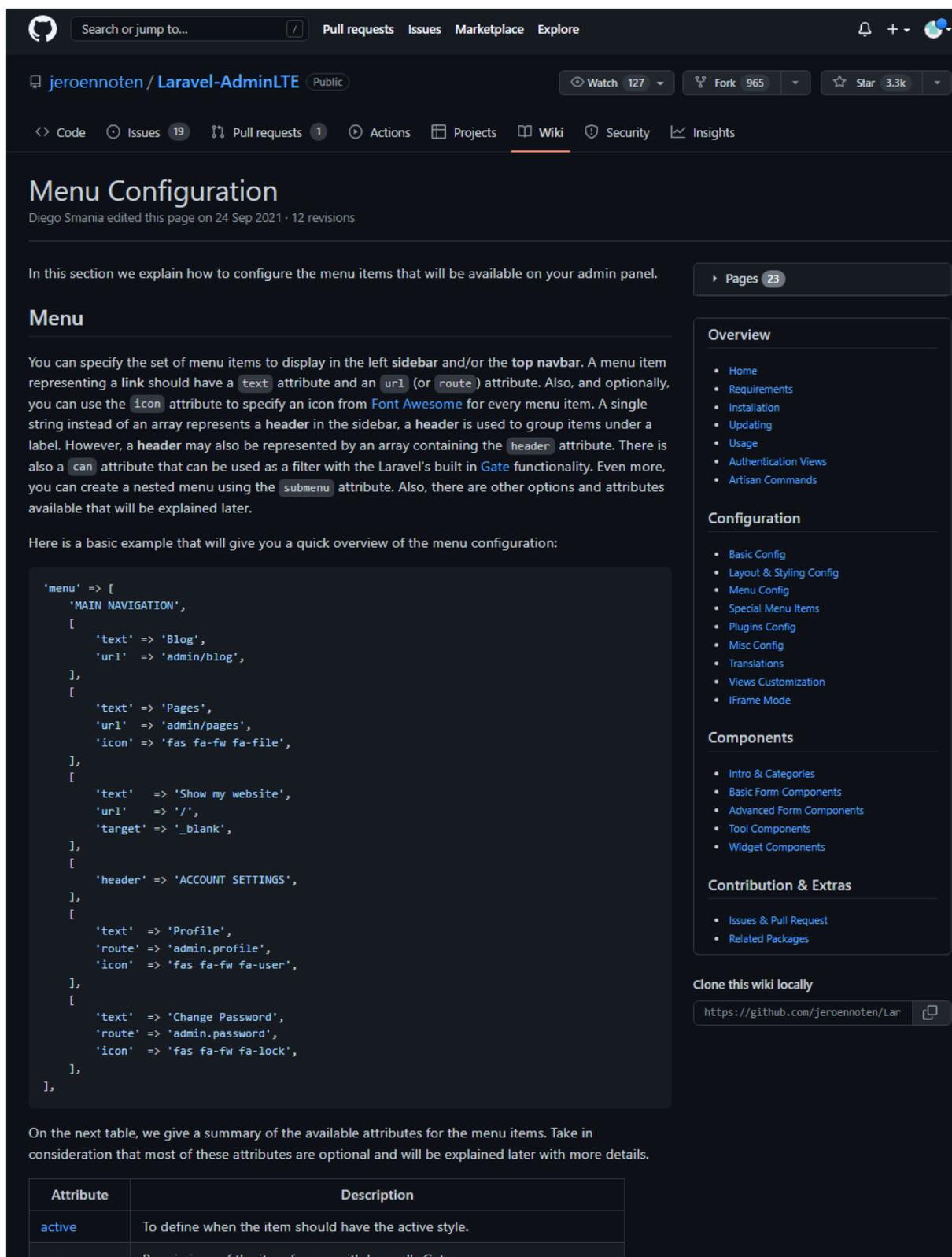
Fuente: captura de pantalla de Visual Studio Code.

Imagen 12: Vista en el navegador del Dashboard base.



Fuente: captura de pantalla del navegador

ANEXO 3. Repositorio de la plantilla integrada Laravel-AdminLTE



The screenshot shows the GitHub repository page for `jeroennoten/Laravel-AdminLTE`. The page title is "Menu Configuration" and it was edited by Diego Smania on 24 Sep 2021. The page content includes an introduction to menu configuration, a code example for the menu configuration, and a table of attributes. The right sidebar contains navigation links for Pages, Overview, Configuration, Components, and Contribution & Extras.

In this section we explain how to configure the menu items that will be available on your admin panel.

Menu

You can specify the set of menu items to display in the left sidebar and/or the top navbar. A menu item representing a link should have a `text` attribute and an `url` (or `route`) attribute. Also, and optionally, you can use the `icon` attribute to specify an icon from [Font Awesome](#) for every menu item. A single string instead of an array represents a header in the sidebar, a header is used to group items under a label. However, a header may also be represented by an array containing the `header` attribute. There is also a `can` attribute that can be used as a filter with the Laravel's built in [Gate](#) functionality. Even more, you can create a nested menu using the `submenu` attribute. Also, there are other options and attributes available that will be explained later.

Here is a basic example that will give you a quick overview of the menu configuration:

```
'menu' => [
    'MAIN NAVIGATION',
    [
        'text' => 'Blog',
        'url' => 'admin/blog',
    ],
    [
        'text' => 'Pages',
        'url' => 'admin/pages',
        'icon' => 'fas fa-fw fa-file',
    ],
    [
        'text' => 'Show my website',
        'url' => '/',
        'target' => '_blank',
    ],
    [
        'header' => 'ACCOUNT SETTINGS',
    ],
    [
        'text' => 'Profile',
        'route' => 'admin.profile',
        'icon' => 'fas fa-fw fa-user',
    ],
    [
        'text' => 'Change Password',
        'route' => 'admin.password',
        'icon' => 'fas fa-fw fa-lock',
    ],
],
```

On the next table, we give a summary of the available attributes for the menu items. Take in consideration that most of these attributes are optional and will be explained later with more details.

Attribute	Description
<code>active</code>	To define when the item should have the active style.
<code>can</code>	Permissions of the item for use with Laravel's Gate.

Pages 23

Overview

- Home
- Requirements
- Installation
- Updating
- Usage
- Authentication Views
- Artisan Commands

Configuration

- Basic Config
- Layout & Styling Config
- Menu Config
- Special Menu Items
- Plugins Config
- Misc Config
- Translations
- Views Customization
- IFrame Mode

Components

- Intro & Categories
- Basic Form Components
- Advanced Form Components
- Tool Components
- Widget Components

Contribution & Extras

- Issues & Pull Request
- Related Packages

Clone this wiki locally

<https://github.com/jeroennoten/Lar>

Fuente: <https://github.com/jeroennoten/Laravel-AdminLTE>

ANEXO 4. Ventas detalle local José Gálvez FERBANZ S.A.C 2016-2021

	A	B	C	D	E	F	G	H
1	Fecha	Producto	nCant	Pres	PUnit	Total		promedio
2	9/09/2016	KING KONG 18 HUECOS FORTES	500 UND		0.5	250		
3	19/09/2016	KING KONG 18 HUECOS FORTES	400 UND		0.54	216		
4	19/09/2016	KING KONG 18 HUECOS FORTES	3000 UND		0.52	1,560.00		
5	24/09/2016	KING KONG 18 HUECOS FORTES	500 UND		0.54	270	setiembre	1100
6	6/10/2016	KING KONG 18 HUECOS FORTES	300 UND		0.54	162		
7	6/10/2016	KING KONG 18 HUECOS FORTES	40 UND		0.5	20		
8	7/10/2016	KING KONG 18 HUECOS FORTES	0 UND		0.55	0		
9	7/10/2016	KING KONG 18 HUECOS FORTES	3000 UND		0.54	1,605.00		
10	10/10/2016	KING KONG 18 HUECOS FORTES	5000 UND		0.53	2,650.00		
11	11/10/2016	KING KONG 18 HUECOS FORTES	460 UND		0.54	248.4		
12	14/10/2016	KING KONG 18 HUECOS FORTES	1000 UND		0.55	550		
13	15/10/2016	KING KONG 18 HUECOS FORTES	1000 UND		0.55	550		
14	15/10/2016	KING KONG 18 HUECOS FORTES	6000 UND		0.51	3,060.00		
15	17/10/2016	KING KONG 18 HUECOS FORTES	5000 UND		0.53	2,650.00	Octubre	1555.26316
16	18/10/2016	KING KONG 18 HUECOS FORTES	500 UND		0.55	275		
17	22/10/2016	KING KONG 18 HUECOS FORTES	300 UND		0.55	165		
18	22/10/2016	KING KONG 18 HUECOS FORTES	40 UND		0.55	22		
19	26/10/2016	KING KONG 18 HUECOS FORTES	350 UND		0.55	192.5		
20	26/10/2016	KING KONG 18 HUECOS FORTES	1000 UND		0.54	540		
21	27/10/2016	KING KONG 18 HUECOS FORTES	2500 UND		0.54	1,350.00		
22	27/10/2016	KING KONG 18 HUECOS FORTES	1000 UND		0.25	245		
23	28/10/2016	KING KONG 18 HUECOS FORTES	60 UND		0.55	33		
24	29/10/2016	KING KONG 18 HUECOS FORTES	2000 UND		0.54	1,080.00		
25	4/11/2016	KING KONG 18 HUECOS FORTES	1000 UND		0.55	550		
26	8/11/2016	KING KONG 18 HUECOS FORTES	100 UND		0.55	55		
27	9/11/2016	KING KONG 18 HUECOS FORTES	3500 UND		0.53	1,855.00		
28	10/11/2016	KING KONG 18 HUECOS FORTES	3000 UND		0.54	1,620.00		

Fuente: archivos Excel de ventas local José Gálvez

ANEXO 5. Modelo de variación estacional o cíclica en Excel

MODELO DE VARIACIÓN ESTACIONAL O CÍCLICA					
Periodo	Datos Historicos		Promedio de las ventas del periodo	Factor de Estacionalidad	Pronostico del Año
	Año	Año			3
	1	2			Ventas
1	1957	3500	2728.5	1.42	2595
2	2478	2070	2274	1.18	2163
3	2021	1977	1999	1.04	1901
4	1702	1581	1702	0.88	1619
5	2108	1702	1905	0.99	1812
6	1428	1615	1521.5	0.79	1447
7	3000	1651	2325.5	1.21	2212
8	2277	2437	2357	1.22	2242
9	1805	1673	1739	0.9	1654
10	1694	1605	1649.5	0.86	1569
11	1702	1533	1617.5	0.84	1538
12	1266	1163	1214.5	0.63	1155
Total Ventas/Año	23438	20926			
Promedio de Ventas	1953.17	1902.36			
VENTAS PREVISTAS		22000			

Fuente: archivos Excel Ingeniosa Empresa