



Facultad de Ingeniería

**Carrera Profesional de Ingeniería de Sistemas e
Informática**

Curva de Aprendizaje para la elaboración de calzado

Alumno:

Luis Angel García Bracamonte – u17209953

Lima – Perú

2020

ÍNDICE GENERAL

INTRODUCCIÓN.....	7
CAPITULO 1.....	8
ASPECTOS GENERALES	8
1.1. Definición del Problema.....	8
1.1.1. Descripción del Problema	8
1.2. Definición de Objetivos.....	14
1.2.1. Objetivo General	14
1.2.2. Objetivos Específicos.....	14
1.2.3. Alcance y Limitaciones	15
1.2.4. Justificación	15
1.2.5. Estado del Arte	16
CAPITULO 2.....	21
MARCO TEORÍCO	21
2.1. Fundamento Teórico.....	21
2.1.1. Producción de calzado	22
2.1.2. Curva de Aprendizaje	24
2.1.3. La Integral Definida.....	32
2.1.4. Programación Orientada a Objetos.....	33
2.1.5. IDE de Desarrollo.....	35
CAPITULO 3.....	37
DESARROLLO DE LA SOLUCIÓN	37
3.1. Diagrama de clases	37
3.2. Estructura del proyecto	38
3.3. Contenido del proyecto	40
3.4. Prototipos	50
CAPITULO 4.....	53
RESULTADOS.....	53
4.1. Prueba del Aplicativo.....	53
4.1.1. Caso	53
4.1.2. Resultados Esperados	54
4.1.3. Resultados del Aplicativo.....	55

4.2. Costos y Beneficios	56
CAPITULO 5.....	57
CONCLUSIONES.....	57
5.1. Conclusiones	57
5.2. Recomendaciones	57
TABLA DE ILUSTRACIONES	4
ÍNDICE DE TABLAS.....	6
ANEXOS	59
BIBLIOGRAFÍA.....	60

TABLA DE ILUSTRACIONES

Figura 1. Tasa de crecimiento de la producción industrial de 1985-2016.....	11
Figura 2. Proceso de elaboración de calzado	13
Figura 3. MODULO DE PRODUCCIÓN SOLUFLEX ERP	18
Figura 4. MODULO DE PRODUCCIÓN SAP BUSSINES ONE	19
Figura 5. MODULO DE PRODUCCIÓN SYSTECAI ERP	20
Figura 6 Proceso de elaboración de calzado	24
Figura 7: Curva de aprendizaje con tasa de 95%.....	26
Figura 8: Curva de aprendizaje con tasa del 75%.....	26
Figura 9: Curva de aprendizaje con tasa del 50%.....	26
Figura 10: Elementos que influyen en el aprendizaje.	27
Figura 11: Etapas del aprendizaje.	29
Figura 12: Fórmula cálculo curva de aprendizaje, método aritmético.	30
Figura 13: Curva de aprendizaje representada con método aritmético.	30
Figura 14: Fórmula calculo curva aprendizaje, método logarítmico.	31
Figura 15: Curva de aprendizaje representada con método logarítmico.	32
Figura 16: Función integral, área bajo la curva.	32
Figura 17 diagrama de clases.....	37
Figura 18 Estructura del Proyecto	38
Figura 19. Método Leer	41
Figura 20. Método Escribir.....	41
Figura 21. Método BuscarCadena.....	42
Figura 22. Clase Imagen	42
Figura 23. LecturaTiempo	43
Figura 24. Registrar Modelo	43
Figura 25. Consultar Modelo	44
Figura 26. Listar Modelo.....	44
Figura 27. Registrar Pedido.....	45
Figura 28. Consultar Pedido.....	45
Figura 29. Listar Pedido	45
Figura 30. ModeloJtable	46
Figura 31. Registrar Muestra.....	46
Figura 32. Registrar por Pedido.....	47
Figura 33. Listar Tiempo.....	47
Figura 34. Calcular tasa de aprendizaje	48
Figura 35. Porc_aprendizaje.....	49
Figura 36. Calcular HH	49
Figura 37 Login	50
Figura 38. Menú Principal	50
Figura 39. Registro de Modelos	51
Figura 40. Muestras.....	51
Figura 41. Pedidos	52
Figura42. Producción	52
Figura 43. Resultados	52
Figura 44. Cálculo de Horas Hombre.....	55

Figura 45. Resultados Obtenidos	56
--	-----------

ÍNDICE DE TABLAS

Tabla 1. Estructura de la Aplicación	39
Tabla 2. Muestra Empresa BAMBINI SHOES	53
Tabla 3 Egresos	56
Tabla 4 Ingresos	56

INTRODUCCIÓN

En la actualidad 96.7 por ciento de las empresas productoras de calzado en Perú son microempresas, y la mayoría no están capacitadas profesionalmente para tener una empresa. La capacitación y optimización es muy importante en las pequeñas y medianas empresas para medir su productividad, de lo contrario llegan a tal punto de crecimiento en el que ya no saben qué hacer.

El presente trabajo presenta un sistema que calcula la curva de aprendizaje para la elaboración de calzado, con la finalidad de incrementar la producción en las empresas de este rubro y eliminar el desperdicio. La curva de aprendizaje representa, básicamente, la descripción del grado de éxito obtenido durante el aprendizaje en el transcurso del tiempo; en este caso la producción de calzado.

CAPITULO 1

ASPECTOS GENERALES

1.1. Definición del Problema

En Perú, el 96.7 por ciento de las empresas productoras de calzado son microempresas y gran parte de estas no están capacitadas profesionalmente para tener una empresa. Esto origina que los volúmenes producidos no tengan un precio competitivo a nivel internacional; encontrándose su principal problema el área de producción. Según datos de Citeccal, uno de los principales problemas que se identifica en las áreas de producción de estas empresas es la perdida de horas hombre debido a la estimación incorrecta de los tiempos de producción. Este problema origina como consecuencia, sobre costos, personal subutilizado y pérdida de clientes.

1.1.1. Descripción del Problema

El Perú, con más de 30 millones de habitantes, consume aproximadamente 64 millones de calzado al año, con un promedio de 2.2 por habitante, de este total 32 millones son de fabricación nacional, 26 millones provienen del extranjero, en tanto se estima que no menos de 6 millones de pares corresponde al ingreso por nuestras fronteras mediante el contrabando. La importación de calzado ha ido en aumento, la mayoría proveniente de China. (CITECCAL, 2014)

Según la Sociedad Nacional de Industrias (SNI), el 96.7 por ciento de las empresas productoras de calzado en Perú son microempresas (Andina, 2011) y la mayoría no están capacitadas profesionalmente para tener una empresa. La capacitación y

optimización es muy importante en las pequeñas y medianas empresas para medir su productividad, de lo contrario llegan a tal punto de crecimiento en el que ya no saben qué hacer, porque la mayoría se desarrolló bajo un modelo familiar, en el que las decisiones solo dependían del fundador, y éste al no contar con la capacitación y/o educación adecuada, ya no podrá continuar con el crecimiento sostenible de su empresa (Gestión, 2016).

Esto se ve reflejado en uno de los rubros más importantes de la economía de nuestro país, como es el rubro de la manufactura, específicamente en el sector cuero y calzado, que es muy conocido a nivel nacional, entre ellas la ciudad de Trujillo conocida como “La Capital del Calzado”, puesto que aquí se concentra la mayor cantidad de micro y pequeñas empresas dedicadas a la fabricación de calzado. La gestión en la mayoría de este tipo de empresas no se da en forma profesional, lo que muchas veces limita el crecimiento a un mayor nivel. Por otro lado, gran parte de la mano de obra para este sector es mano de obra no calificada, es decir, recurso humano que no ha necesitado de preparación profesional o técnica, y cuya labor se basa en un conocimiento empírico. Es por ello que muchos fabricantes de calzado no han podido, o se han quedado en el intento, de llegar a un nivel más alto de “pyme”, debido a diferentes hechos, circunstancias, escenarios, que los han desfavorecido de manera interna o externa, pero que luego se ve reflejado negativamente en el resultado final.

Según José Carlos Saavedra, jefe de análisis macroeconómicos de Apoyo Consultoría; explica: “El desarrollo y la productividad van de la mano y estos resultados a nivel país sorprenden, no solo porque las empresas peruanas no

cuentan con la tecnología adecuada, sino, sobre todo, porque los trabajadores no están preparados académica o técnicamente para potenciar su productividad. Hoy en día, las empresas tienen que conducir a su personal de manera diferente, necesitan innovar, transformar y rediseñar las prácticas de capital humano. La organización del siglo XXI es global, altamente conectada y exigente” (Gestión, 2016).

Esta realidad hace que los talleres no logren alcanzar su máxima productividad y, como lo menciona Iván Olaechea (2017), la industria nacional pasa por momentos de muchos retos, debido a que la industria china y brasileña afecta en gran medida la industria nacional. Además, añade que la razón por la cual estas importaciones afectan al sector del país en gran medida, se debe a que nuestros procesos no son eficientes, generando esto que el Perú no sea considerado como un país altamente competitivo. Es por lo que surge la necesidad de implementar un sistema de planificación de recursos empresariales (ERP), por sus siglas en inglés, que permita automatizar procesos y, en consecuencia, los errores se vean reducidos, lográndose a su vez que se eleve la competitividad de la organización, tal como lo menciona Manuel Quiles (2018).

Andreas Von Wedemeyer (2016), presidente de la Sociedad Nacional de Industrias, señaló que para las micro y pequeñas empresas de calzado es muy difícil convertirse en medianas y esto les impide ser competitivas, ubicándolas en una posición negativa respecto a empresas de otros países, viéndose afectadas por las importaciones de calzado barato. Además, agrega que también factores como

infraestructura, inversión, tributación, régimen laboral han impactado de forma negativa por lo que la tasa de crecimiento de la producción anual se ha visto afectada, tal como lo muestra la siguiente figura:

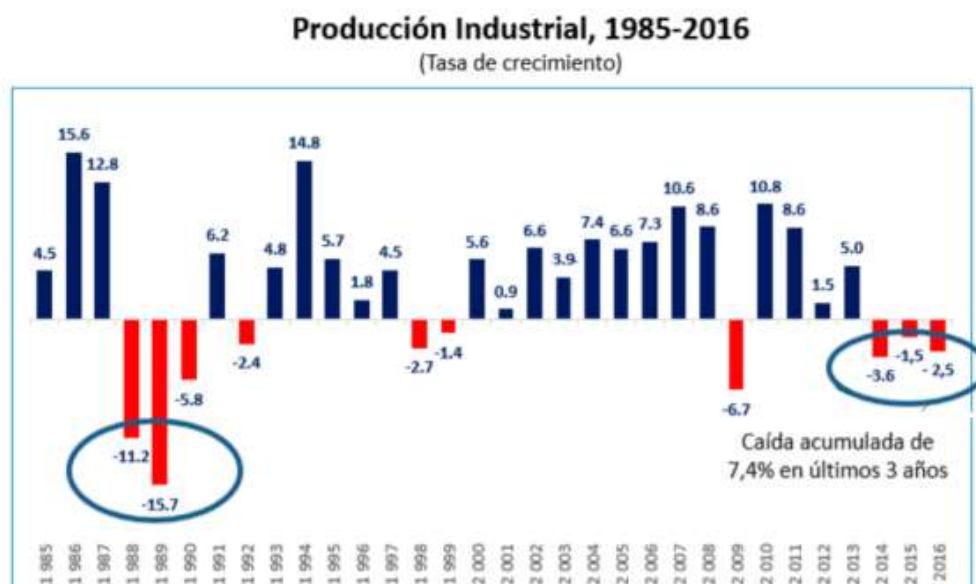


Figura 1. Tasa de crecimiento de la producción industrial de 1985-2016

Fuente: BCRP

Como fue mencionado en el párrafo anterior, para Andreas Von Wedemeyer es de suma relevancia que las micro y pequeñas empresas de calzado se vuelvan competitivas, pues como lo afirma Ricardo Espinoza (2017) la realidad actual no es favorable, ya que la mayoría de empresas no cuentan con tecnología, por lo que su desempeño operacional es presa fácil para el calzado chino, ya que la industria nacional cuenta con producciones lentas de 50 pares a la semana en lo que es calzado de cuero y esto puede cambiar de manera positiva gracias a los sistemas de planificación de materiales requeridos y el de planificación de recursos de fabricación que según Raquel García (2018) mejoran el tiempo de procesamiento

de un producto, permitiendo que el desempeño operacional aumente y, por ende, las empresas puedan volverse más competitivas.

En el Perú, según el Centro de Innovación Tecnológica del Cuero, Calzado e Industrias Conexas (CITECCAL, 2018), el sector de calzado está conformado fundamentalmente por micro y pequeñas empresas, por lo que se entiende que los volúmenes producidos no permiten tener un precio competitivo a nivel internacional; siendo algunas de sus principales debilidades la carencia de tecnología en su línea de producción, el desperdicio de horas/hombre en el proceso productivo, y el poco desarrollo en su área de ventas.

En este trabajo, nos centraremos en el problema del desperdicio de horas hombre en el proceso productivo. Por lo que a continuación explicaremos el proceso productivo de calzado y los principales desperdicios que existen.

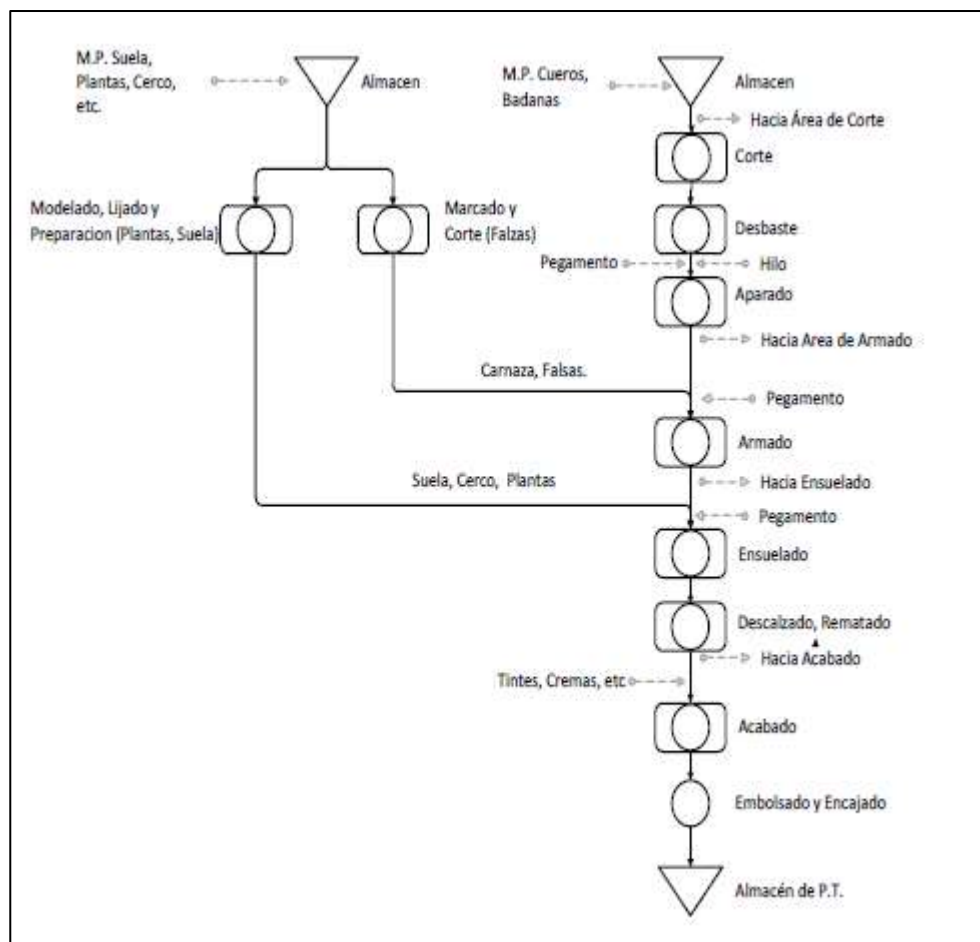


Figura 2. Proceso de elaboración de calzado

Fuente: creaciones Ruthmir

Luego de revisar el proceso productivo, se evidencia que los principales desperdicios son los siguientes:

Ítem	Causas Raíz	Efecto
1	Deficiente Programación y/o coordinación para la producción	Retrabajo
2	Desorden	Retrabajo
3	Personal No Capacitado y/o Motivado	Retrabajo
4	Procesos No estandarizados	Retrabajo
5	Mala estimación en la entrega	Trabajo adicional

Tabla 1. Desperdicios

Fuente: Elaboración Propia

Es por ello que se plantea la construcción de una aplicación que ayude a medir la curva de aprendizaje con la finalidad de incrementar la productividad de la empresa.

1.2. Definición de Objetivos

1.2.1. Objetivo General

Desarrollar una aplicación que permita calcular la curva de aprendizaje para la elaboración de calzado, tomando como insumos los tiempos de las muestras realizadas por cada modelo. Esto ayudara a eliminar desperdicio en el proceso de elaboración de calzado y a su vez mejorar los tiempos estimados de producción, lo que impacta en la optimización de horas-hombre, reduciendo el tiempo muerto y sobrecostos de producción, reprogramaciones de producción y mejorando el cumplimiento con las fechas de entrega al cliente. Esta aplicación va dirigida a los responsables de producción de las microempresas que generan calzado en Perú.

1.2.2. Objetivos Específicos

- Calcular las horas hombre requeridas por cada calzado en base a la curva de aprendizaje, que determina el tiempo que se tomaría producir un calzado en la enésima vez que es tratado el modelo. Eso permitirá optimizar el proceso productivo y tener una mejor planificación de las entregas.
- Desarrollar una aplicación en java que permita obtener el tiempo de la producción de un modelo, lo que permitirá optimizar la programación y eliminar desperdicio de h/h. Para ello, se tomarán como datos los siguientes ítems:

- Tiempo de ejecución por cada par de calzado realizado.
 - Cantidad de calzado a producir del modelo
 - Cantidad de personas asignadas a producción
- Determinar el nuevo tiempo estándar y productividad.
- Determinar la mejora de la productividad por Mano de Obra.

1.2.3. Alcance y Limitaciones

La solución solo es aplicable para empresas de producción de calzado con área de maquinado que realicen una previa toma de tiempo en la ejecución de las operaciones, ya que es un dato indispensable para determinar el comportamiento en la curva de aprendizaje.

El software a desarrollar esta limitado por lo siguiente:

- La primera versión es para entorno escritorio.
- Será desarrollada en java 8.
- El IDE usado para desarrollar la aplicación será Netbeans.
- El primer reléase de la aplicación no contará con una conexión a BBDD, por lo que se entiende que los datos se mostrarán solo en sesión.

1.2.4. Justificación

El presente trabajo busca contribuir con información esencial acerca de los tiempos de trabajo y la curva de aprendizaje que se aplican dentro de un proceso productivo y el beneficio de su implementación a la industria del calzado. Con el presente estudio se busca evidenciar las deficiencias del proceso productivo de

calzado y diseñar propuestas de mejora, que se concentren en la necesidad de elevar la producción.

Con la propuesta de mejora se logrará disminuir los niveles de desperdicio de materiales, tiempo y espacio; que ayudarán a aumentar esta productividad, así como mejorar la insatisfacción de los clientes internos. Así aplicando la inteligencia eficazmente, motivando y concientizando al personal a realizar sus tareas eficientemente desde la primera vez y así alcanzar mejores niveles de producción.

Además, le ofrece a la empresa nuevas oportunidades para propiciar y aplicar la filosofía de mejora continua, con miras de tener una creciente proyección en la comunidad industrial, puesto que, al incrementar su producción a través del mejoramiento de los procesos, mano de obra, maquinaria, materia prima, obtendrán mayores ganancias, creando una atmósfera adecuada para superar y satisfacer las exigencias de los clientes logrando mayores beneficios en el mercado competitivo.

1.2.5. Estado del Arte

El área de producción es el departamento que tiene como función principal la creación de los productos de la empresa. Transforman los insumos o recursos (energía, materia prima, mano de obra, capital, información) en productos finales. Por ello la importancia, de que los niveles de producción sean óptimos y se elimine todo tipo de desperdicio.

En el presente punto, se busca dar a conocer algunas otras opciones de software existentes en el mercado que usan las áreas de producción en las industrias de calzado en Perú.

- **SOLUFLEX ERP**

Es un Software preparado para operar en empresas del rubro de la Industria Textil y Confecciones de prendas de vestir textiles y también en la Industria del Calzado. En su módulo de producción considera las etapas de Desarrollo de Producto, Procesos de Producción y Costos de Producción

Características:

- Registro de Recetas
- Relación de Materiales (MRP)
- Ficha de Producto – Ruta de Procesos – Receta de Insumos y Servicios
- Registro de Orden de Trabajo u Orden de Producción
- Consumo de Materiales
- Registro de Tercerizaciones
- Control de Talleres de Terceros
- Seguimiento de la Orden de Trabajo (Fechas Programadas)
- Guías de Remisión
- Ubicación de Productos en Proceso
- Gastos de Fabricación – Gastos Mensuales (Luz, Agua, Gas, Mano de Obra)
- Registro de Partes de Producción por Proceso (Maquina, Operario, Horas Trabajadas, etc)
- Entrega de Productos Terminados al Almacén (parciales y totales)
- Generación de Asientos Contables Automáticos



Figura 3. MODULO DE PRODUCCIÓN SOLUFLEX ERP

Fuente: SOLUFLEX

- **SAP Business One**

Es un software ERP de gestión empresarial creado para Pymes y empresas en vías de crecimiento. Está 100% integrado, ofreciendo un conjunto de funciones como: finanzas, gestión de clientes, ventas, compras, producción, operación y logística.

La solución cuenta con una herramienta de planeación de materiales (MRP), la cual ayuda a su empresa a decidir cuándo y cuanto debe comprar o producir basado en la demanda de sus clientes, sus tiempos de compra y el nivel de su inventario.

De igual manera le facilita el control de sus Listas de Materiales y la administración de sus órdenes de producción lo que facilitará el control de los costos y tiempos de su proceso productivo permitiendo operar de una manera más efectiva.

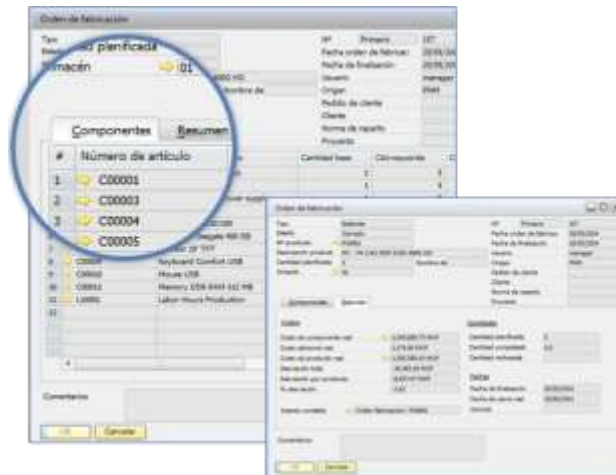


Figura 4. MODULO DE PRODUCCIÓN SAP BUSSINES ONE

Fuente: SAP

- **SYSTECA ERP**

Software de gestión para empresas de calzado, que permite controlar y planificar todas las fases de fabricación y comercialización de productos.

Características:

- Completa ficha de clientes, proveedores y artículos.
- Ficha de clientes con datos de promociones especiales.
- Control de riesgo de cliente.
- Ficha de agentes comerciales.
- Información sobre compra de materia prima.
- Modificación automática de tarifas de precios de venta.
- Sistema de exportación de facturas mediante EDI.
- Gestión de temporada de trabajo.
- Gestión de artículos por talla.
- Mantenimiento de cartas de colores.
- Gestión documentos de venta por temporada.
- Proceso automático de servir pedidos de venta.
- Creación automática de artículos de diferentes colores.
- Consultas directas a históricos del cliente/proveedor y artículos.
- Precios/Descuentos específicos por cliente y artículo.

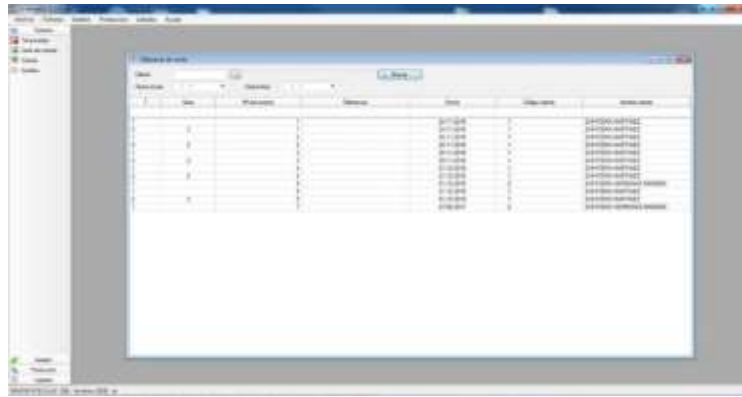


Figura 5. MODULO DE PRODUCCIÓN SYSTECAL ERP

Fuente: SYSTECAL ERP

Existe un conjunto amplio de herramientas para las áreas de producción de las empresas de calzado. Sin embargo, la mayoría de ellas dedicadas a mostrar fichas de productos, relación de materiales u ordenes de trabajo, y solo un pequeño grupo de ellas tienen informes o dashboards configurados de caja que ayudan a los productores de calzado a tomar decisiones en cuanto a optimización de la producción, teniendo como insumo datos de los trabajos anteriores.

CAPITULO 2

MARCO TEORÍCO

2.1. Fundamento Teórico

Este capítulo tiene como objetivo crear una base científica para el desarrollo del informe, el cual contiene una síntesis de los temas aplicados en el desarrollo del aplicativo.

En todo proceso de producción es importante la optimización de tiempos y costos, basado en ello se plantea la optimización del tiempo y por ende el costo. Para ello aplicaremos los principios de curva de aprendizaje en la determinación de horas hombre requeridas para una determinada producción. Este principio es aplicable en el área de producción, ya que esta recibe diferentes producciones de diferente complejidad, lo que implica que cada una de ellas tenga una curva de aprendizaje diferente.

En el desarrollo de la aplicación se hace necesario determinar las hora-hombre requerida dependiendo de la unidad producida, bajo la función de curva de aprendizaje, lo que finalmente nos lleva a obtener el total por las unidades que finalmente se producirán. Para realizar ello, utilizaremos el principio de integrales definidas, dónde se obtiene el área bajo a una curva definida por intervalos, donde los intervalos se definen en base a la información de la muestra y la cantidad a producir y el área bajo la curva bajo la función de curva de aprendizaje indicaría el número de horas hombre requeridos.

2.1.1. Producción de calzado

Podríamos afirmar que el origen del calzado comenzó con la sandalia. En el Antiguo Egipto se confeccionaban con paja trenzada o láminas de hoja de palmera, y era de uso masculino del que estaban excluidas mujeres y esclavos. Su uso era ceremonial ya que la tendencia a ir descalzo perduró durante muchos siglos en la Historia.

Respecto a cómo ha sido la evolución de los zapatos, ya en la Antigua Grecia, Homero describe a los héroes calzando lujosas sandalias. Un poco más adelante, el historiador y geógrafo griego Pausanias asegura que solo los dioses deben calzar sandalias doradas.

Mientras tanto en la antigua Persia se imponía el calzado flexible: las persikai, calzado cómodo para mujeres y ancianos. Se conocía el borceguí o zapato de caza o calzado de viaje, así como las botas lazadas llamadas endromides.

Como podemos observar, a lo largo del tiempo la elaboración de calzado ha ido evolucionando, hasta nuestra era.

A continuación, se explicará el proceso que se sigue para la elaboración del calzado en la actualidad:

- **Almacenamiento de materiales:** La elaboración del calzado se inicia con la recepción de los insumos en la fábrica. Se tienen clasificados y ordenados el tipo de material, piel sintética, tintas, lacas, suelas, adhesivos.
- **Transporte al área de proceso:** Los materiales seleccionados se transportan al área de producción.
- **Corte de piezas:** Se realiza mediante la moldura de acuerdo con la medida que se requiera para dar forma a la piel sintética, según el modelo diseñado en una actividad que pueda ser externa a la empresa.

- **Unión de piezas:** Se reúnen las piezas de un lote para su posterior elaboración. Cada zapato lleva de 7 a 12 piezas según el modelo.
- **Maquinado de corte.** Se requieren varios procesos:
 - Foliado: es la impresión en los forros de la clave, número de lote, modelo número de par, tamaño o medida para su rápida selección e identificación.
 - Grabado: impresión de la marca en la plantilla.
 - Perforado: en algunos casos se lleva a cabo de acuerdo al diseño.
 - Encasquillar: antes del montado, se pone el casquillo y contrahorte.
El casquillo es lo que le da fuerza y forma a la puntera del zapato para darle mayor consistencia.
- **Montado:** Se selecciona la horma de acuerdo a la numeración para conformar, fijar la planta a base de clavos y cemento, esto se hace manualmente y se utiliza una máquina especial para presionar y que quede bien realizado y conformado el zapato. Se montan puntas y talones. Después se realiza el proceso de asentar que consiste en hacer que el corte asiente perfectamente en la horma.
- **Acabado:** Se pegan las plantillas, se pintan los cantos de suelas y forros, se realiza el lavado del corte y forros con jabón especial; se desmancha el zapato de residuos del proceso productivo.
- **Pigmentado:** Esto se realiza con el objeto de uniformizar el color, el calzado se retoca con laca para darle brillo, lo cual se realiza con cepillos giratorios.
- **Empaque:** Se imprime el número de modelo y se guarda el producto en cajas de cartón.

- **Almacenamiento del producto terminado:** Una vez empacado se procede a clasificar en anaqueles, por estilo y número.

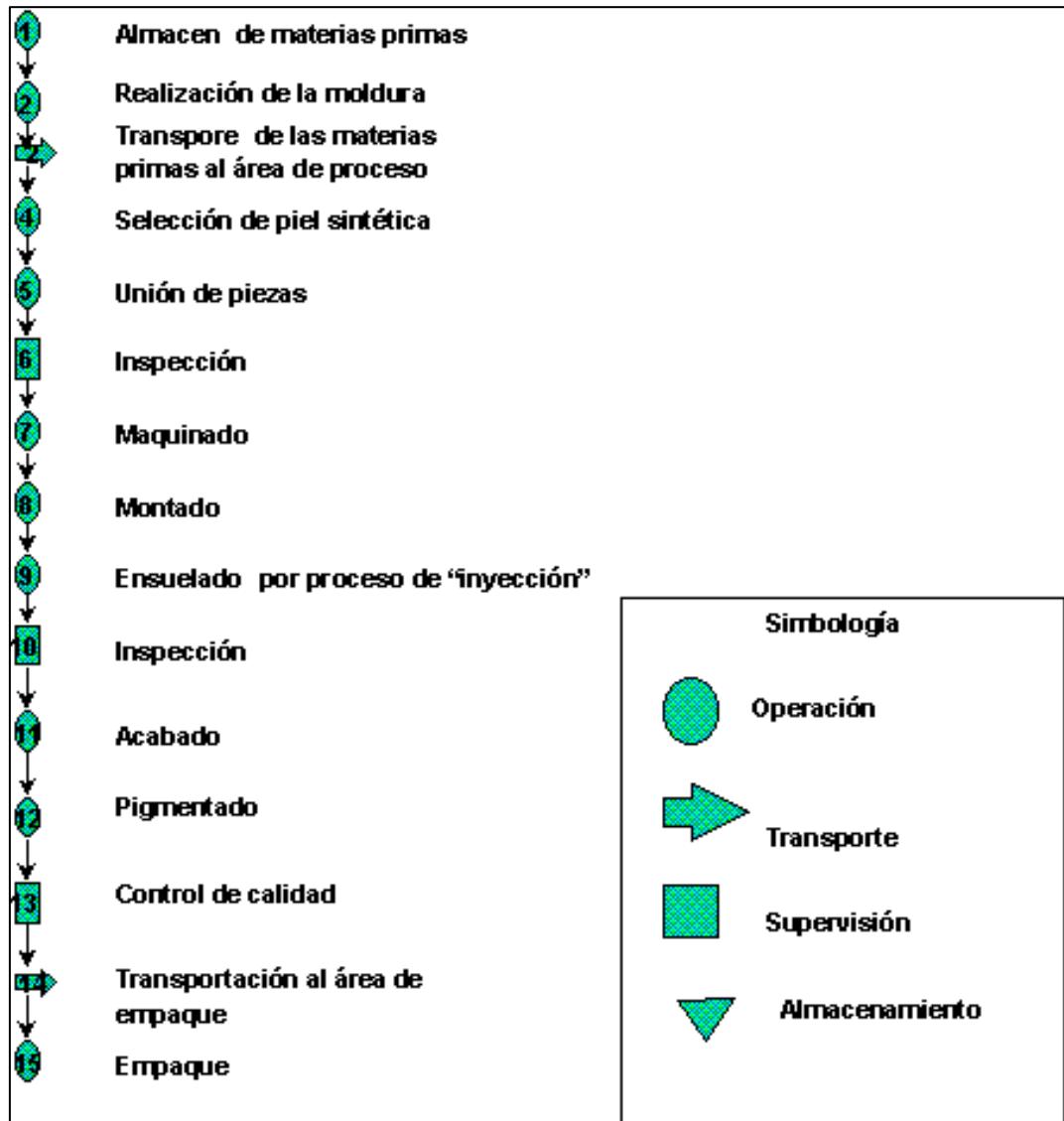


Figura 6 Proceso de elaboración de calzado
Fuente: Elaboración propia

2.1.2. Curva de Aprendizaje

Históricamente el concepto de curva de aprendizaje proviene de Hermann Ebbinghaus, quien en 1885 utilizó por vez primera ese término en su monografía "Über das Gedächtnis" (Sobre el olvido).

Todo ser humano tiene la habilidad de aprender y mejorar cada vez que repita una misma tarea, esta le permite ser más eficiente a medida que transcurre el tiempo. Este comportamiento llevó a desarrollar una teoría conocida como la “curva de aprendizaje”.

Según Wright Patterson. T.P. publicó un artículo en 1936. Afirma que el fenómeno de Curva de Aprendizaje se observó por primera vez en 1920, relacionado con los procesos de ensamble de aviones en la base de la fuerza aérea americana. Al documentar sus observaciones encontró que el ensamble de un segundo avión de cierto tipo gastaba el 80% de las horas-hombre gastadas por el ensamble del primer avión. El cuarto avión gastaba el 80% de las horas del segundo. El octavo avión gastaba el 80% del cuarto y así sucesivamente. Esto lo llevó a una conclusión (Latiff, 2005):

“las horas-hombre necesarias para completa una unidad de producción, decrecerían en un porcentaje constante cada vez que la producción se doblara. En la industria, la “curva de aprendizaje” se aplica al tiempo y costo de la producción”. (p.15).

El comportamiento de la curva de aprendizaje depende mucho de la tasa de aprendizaje y está no necesariamente se entiende como que a mayor tasa de aprendizaje mayor es la curva. En este caso es lo contrario, es decir mientras menor será la tasa de aprendizaje, mayor es la curva. Al graficar la curva de aprendizaje está se representa como una curva decreciente, donde el eje Y representa el tiempo y el eje X el número de ciclos. A continuación, una gráfica de curva aplicando una tasa de aprendizaje de 95%. En ella observamos que el salto de aprendizaje es menor.

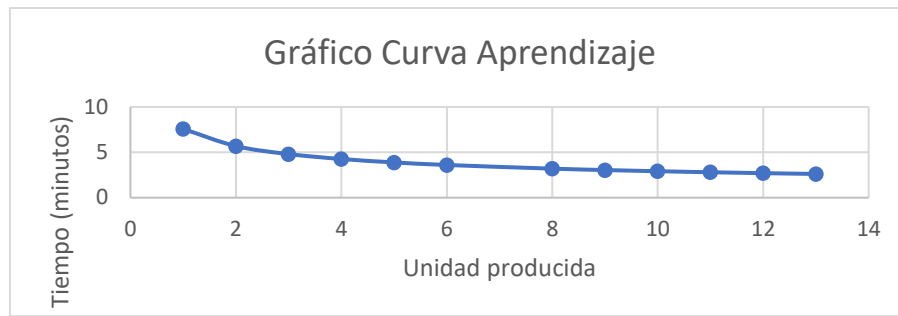


Figura 7: Curva de aprendizaje con tasa de 95%.
Fuente: Elaboración propia.

Ahora graficamos una curva de aprendizaje con una tasa del 75% de aprendizaje.



Figura 8: Curva de aprendizaje con tasa del 75%.
Fuente: Elaboración propia.

La siguiente es una curva de aprendizaje con una tasa del 50%.

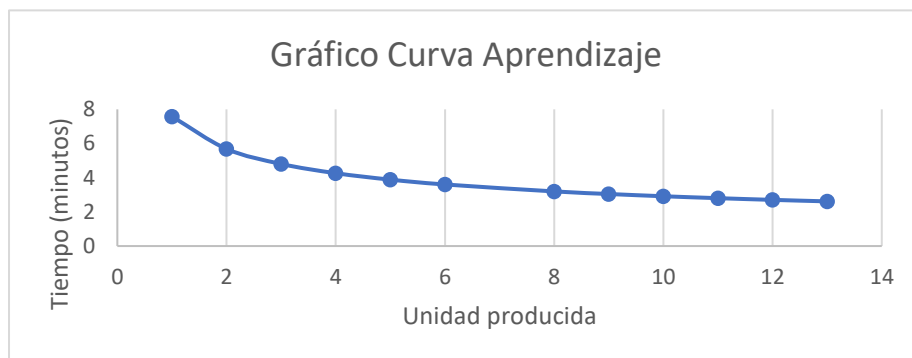


Figura 9: Curva de aprendizaje con tasa del 50%.
Fuente: Elaboración propia.

Esto resume lo expuesto anteriormente, donde se observa que, a menor tasa, mayor es el salto.

El aprendizaje se ve afectado por diversos elementos, por ello (Chango & Zambrano, 2017) nos dice:

“El aprendizaje se puede explicar como un proceso y posee elementos que lo intervienen, en general cualquier proceso de aprendizaje los motiva la necesidad y el interés intrínseco o extrínseco en la actividad, se base en valores comunes, datos e información... La practica, la disciplina y la repetición permiten alcanzar a las tareas, es decir acumular conocimiento, para potenciar el aprendizaje” (p.18).

Dentro de estos elementos podemos resaltar los expuestos en la figura 4.



Figura 10: Elementos que influyen en el aprendizaje.

Fuente: Mariela Chango e Isabel Zambrano.

Asimismo, el proceso de aprendizaje implica diferentes fases, el cual Maslow desarrollo un modelo teórico donde los divide en 4 fases: (Chango & Zambrano, 2017):

- La primera etapa del aprendizaje se denomina “incompetencia inconsciente”, es cuando la persona no sabe que no sabe, es ignorante de su ignorancia, para explicar estos pasos tomaremos por ejemplo el

desarrollo de los juegos para teléfonos inteligentes, existen tantos que ignoramos que no sabemos de su existencia.

- La segunda fase es la “incompetencia consciente”, la persona sabe que no sabe. Ud se entera de la existencia de un juego para teléfonos inteligentes, pero no lo sabe jugar. Es cuando la persona está motivada para aprender, se ilusiona y prepara toda su atención y energía en adquirir las habilidades deseadas.
- La tercera fase es la de la “competencia consciente”. La persona está adquiriendo las nuevas habilidades deseadas de manera consciente, se da cuando la persona está aprendiendo a jugar, tiene que estar pendiente de los controles, las indicaciones, los sonidos, es decir va con cautela prestando toda su atención. Además, muchas personas creen que el aprendizaje termina en esta fase y lo que se espera es que la persona termine de aprender, cuando lo cierto es que falta la cuarta etapa.
- La última etapa se denomina “competencia inconsciente”, es cuando la persona ejecuta la habilidad de forma automática. No está pendiente de detalles, actúa sin darse cuenta, puede hablar con otro jugador predecir las futuras acciones y estar pendiente de los sonidos en todo momento. Es el nivel donde la mente inconsciente se hace cargo del proceso. (p. 19,20).

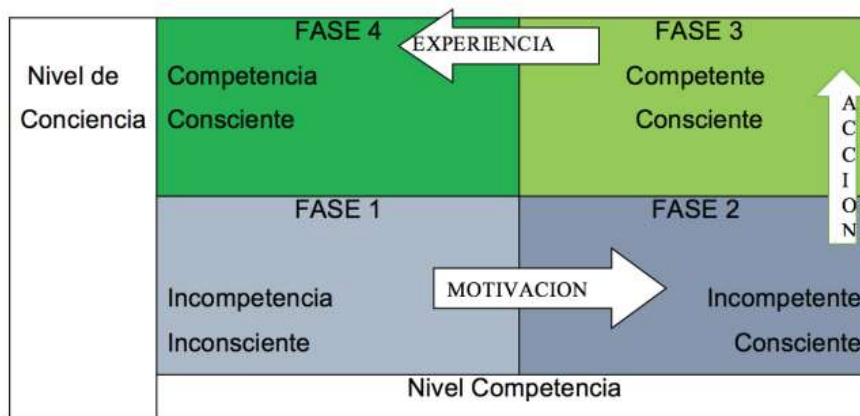


Figura 11: Etapas del aprendizaje.

Fuente: Mariela Chango e Isabel Zambrano.

Métodos de cálculo de la curva de aprendizaje

Existen diversos métodos para calcular la curva de aprendizaje, pero dentro de los métodos más conocidos y descritos por varios autores tenemos:

- Método aritmético
- Método logarítmico

Ambos métodos implican una relación matemática que permite expresar el tiempo que supone producir una determinada unidad. Esta relación es función de cuántas unidades se han producido antes y cuánto tiempo llevó producirlas. Aunque este procedimiento determina el período de tiempo que es necesario para producir una unidad dada, las consecuencias de este análisis son de mayor alcance.

Método aritmético

El análisis aritmético es el método más simple para los problemas de curvas de aprendizaje. De tal forma, cada vez que la producción se duplica, la mano de obra por unidad disminuye en un factor constante, conocido como la tasa de aprendizaje.

(Chango & Zambrano, 2017) Este método aplica la siguiente fórmula:

$$T_N = T_o * L^N$$

Siendo:

T_N = Tiempo de la unidad anterior

T_o = Unidades de factor o costo de producir la o- ésima unidad generalmente la primera.

N = Unidad de producto.

L^N = Coeficiente de la tasa de aprendizaje.

Figura 12: Fórmula cálculo curva de aprendizaje, método aritmético.

Fuente: Mariela Chango e Isabel Zambrano.

Así, si asumimos que la tasa de aprendizaje es de 80% y la primera unidad producida tomó 100 horas, las horas necesarias para producir la segunda, cuarta, octava y decimosexta unidad serían (Chango & Zambrano, 2017):

Unidad producida N	Horas para la unidad N
1	100.00
2	80.00 = 0.80*100.00
4	64.00 = 0.64*80.00
8	51.20 = 0.80*64.00
16	41.00 = 0.80*51.20

Tabla 2. Tasa de Aprendizaje

Fuente: Elaboración propia

Esto significaría un comportamiento de la curva casi lineal y constante:

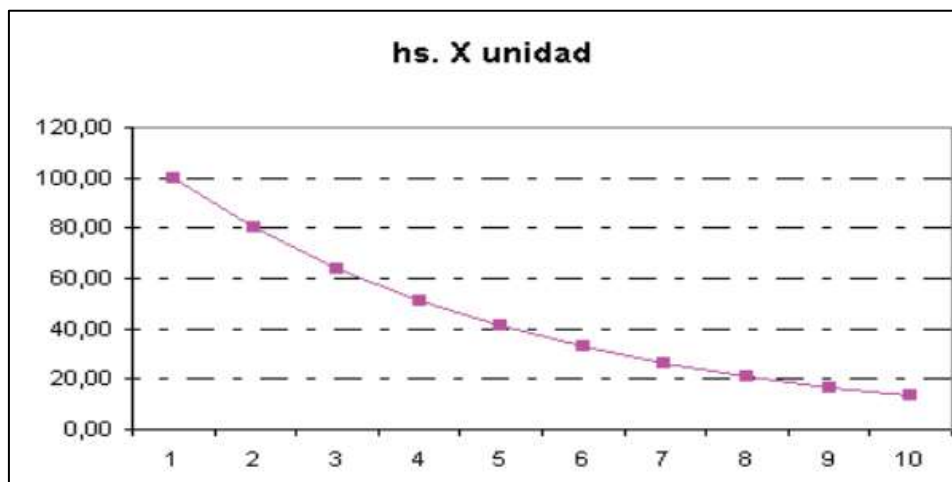


Figura 13: Curva de aprendizaje representada con método aritmético.

Fuente: Mariela Chango e Isabel Zambrano.

Método Logarítmico

Diversos autores señalan que este método logarítmico permite determina la mano de obra para cualquier unidad (Chango & Zambrano, 2017)

<p>Fórmula:</p> $T_N = T_0 \times N^b$ <p>Donde:</p> <p>x = Número de unidades.</p> <p>Y_x = Número de horas-hombre directas requeridas para producir la enésima und x.</p> <p>K = Número de horas-hombre directas requeridas para producir la primera unidad.</p> <p>n= log b/ log2, donde b= porcentaje de aprendizaje.</p>

Figura 14: Fórmula calculo curva aprendizaje, método logarítmico.

Fuente: Mariela Chango e Isabel Zambrano.

Este método permite determinar la mano de obra para cualquier unidad. Cuando se usa papel lineal para graficar, la curva de aprendizaje es una curva de potencia de la forma $y = kx^n$. En papel logarítmico (consta de un par de ejes mutuamente perpendiculares cuyas escalas son logarítmicas), la curva se representa por donde: y = tiempo de ciclo, x = número de ciclos o unidades producidas, n = exponente que representa la pendiente, k = valor del primer tiempo de ciclo.

Por definición, el porcentaje de aprendizaje es entonces igual a: Fórmula:

$$\frac{k(2x)^n}{kx^n} = 2^n$$

Tomando logaritmos en ambos lados de la ecuación: Fórmula:

$$n = \frac{\log_{10} (\text{porcentaje de aprendizaje})}{\log_{10} 2}$$

Para un 80% de aprendizaje:

$$n = \frac{\log_{10} (0,80)}{\log_{10} 2} = \frac{-0,0969}{0,301} = -0,0322$$

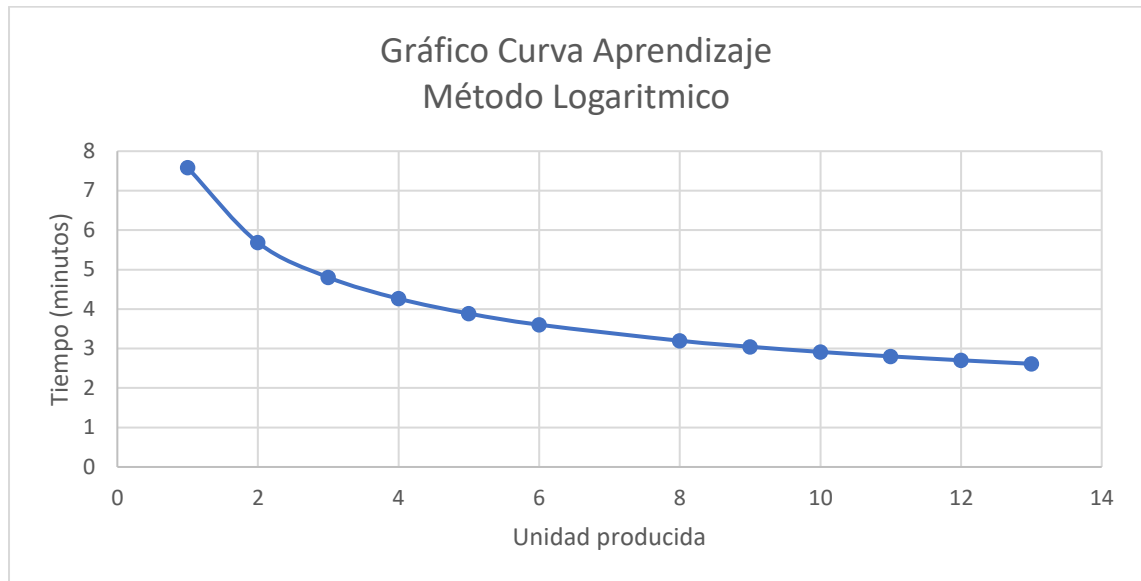


Figura 15: Curva de aprendizaje representada con método logarítmico.

Fuente: Elaboración propia.

2.1.3. La Integral Definida

La integral definida es un concepto utilizado para determinar el valor de áreas limitadas por curvas y rectas.

La integral definida de $f(x)$ en el intervalo $[a,b]$ es igual al área limitada entre la gráfica de $f(x)$, el eje de las abscisas, y las rectas verticales $x = a$ y $x = b$ (bajo la hipótesis de que la función f es positiva).

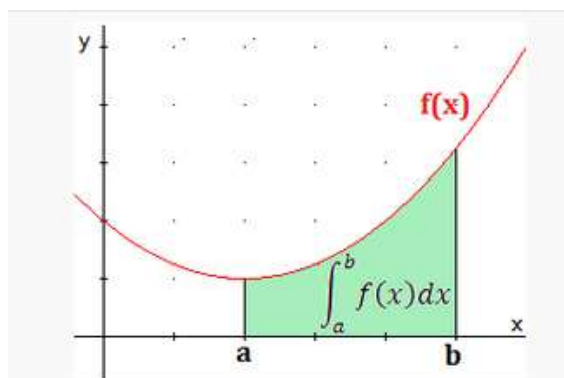


Figura 16: Función integral, área bajo la curva.

Fuente: Elaboración propia.

La integral está representada por:

$$\int_a^b f(x)dx$$

a es el límite inferior de la integración y b es el límite superior de la integración.

Si la función F es una función primitiva de f en el intervalo [a,b], por la Regla de Barrow, se tiene que:

$$\int_a^b f(s)ds = F(b) - F(a)$$

2.1.4. Programación Orientada a Objetos

En el transcurso de la historia informática, han aparecido diferentes paradigmas de programación. En primer lugar, aparece la programación secuencial, que consistía en secuencia de sentencias que se ejecutaban una tras otra. El lenguaje ensamblador o el lenguaje COBOL son lenguajes secuenciales. Entonces no existía el concepto de función, que apareció más adelante en los lenguajes procedimentales, como el BASIC o el C. La evolución no terminó aquí, y continuó hasta el paradigma más extendido en la actualidad: la programación orientada a objetos. Smalltalk, C++ o Java pertenecen a esta nueva generación. Cada nuevo paradigma ha extendido el anterior, de manera que podemos encontrar las características de un lenguaje secuencial en uno procedimental, y las de uno procedimental en uno orientado a objetos.

Java es uno de los lenguajes en el que se trabaja la programación orientada a objetos (Rodríguez, 2010)

Es en la programación orientada a objetos (POO), donde surge los conceptos de clases y objetos, para ello definamos cada uno de ellos.

Objetos

Los objetos son representaciones (simples/complejas) (reales/imaginarias) de cosas: reloj, avión, coches.

No todo puede ser considerado como un objeto, algunas cosas son simplemente características o atributos de los objetos: color, nombre.

Los objetos contienen comportamientos y características. Estos comportamientos también son conocidos como métodos y las características como atributos.
(Rodríguez, 2010)

Clases

Las clases son agrupaciones de objetos con comportamientos similares. Esta es definida como una entidad abstracta del que utilizamos para clasificar dependiendo del problema a resolver.

Ejemplo:

La clase coche puede definirse con los métodos: arrancar, avanzar, parar, y sus características definidas como color, marca, velocidad, etc.

Instanciación de Objetos

Una clase solo es un patrón de objetos, por lo que esta, realmente no existe en memoria, por ser una idea abstracta, quien ocupa un espacio en memoria es un objeto y esta se materializa cuando se instancia una clase, es decir materializamos la idea.

Un ejemplo de instanciación en Java:

Persona per;

Per = new Persona();

En la primera línea estamos declarando una variable de tipo Persona, sin embargo en la segunda línea realizamos la instanciación al indicar “new”. A partir de dicha instanciación ya estamos ocupando un lugar en memoria en el que almacenaremos las características del nuevo objeto.

Herencia

La POO establece un mecanismo fundamental que permite definir una clase en términos de otra. Esto quiere decir basarnos en una clase y hacerla más personalizada, pero aprovechando lo ya definido para la clase de quien de hereda. La herencia es siempre transitiva: una clase puede heredar características de superclases que se encuentran muchos niveles más arriba en la jerarquía de herencia (Cristina Cachero, 2011).

Asumamos la clase animal, del que podemos definir cierto comportamiento y característica, sin embargo, sobre ella podemos definir una clasificación más personalizada, como por ejemplo animal mamífero, y otro animal ovíparo. La clase mamífero inclusive puede personalizarse más, ejemplo: una nueva sub-clase perro basado en la clase mamífero.

2.1.5. IDE de Desarrollo

Hoy en día existen diferentes herramientas que facilitan el desarrollo de software. Estas herramientas son llamadas framework o IDE de desarrollo. Muchos de estos IDEs permiten desarrollar varios lenguajes de programación y así también existen IDEs que solo soportan los lenguajes por la compañía que crea los lenguajes como Microsoft.

Dentro de los IDE's más conocidos tenemos:

- Eclipse
- NetBeans
- Visual Studio
- JetBrains
- QtCreator
- CodeLite

Para este proyecto usaremos el IDE NetBeans 8.2, del cual es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo.

El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles). Entre sus características se encuentra un sistema de proyectos basado en Ant, control de versiones y refactoring.⁷

Todas las funciones del IDE son provistas por módulos. Cada módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. NetBeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente.

CAPITULO 3

DESARROLLO DE LA SOLUCIÓN

La aplicación calcula la curva de aprendizaje para la elaboración de calzado usando como input los tiempos de las muestras realizadas por cada modelo de calzado.

La aplicación ha sido desarrollada en JAVA 8 y no cuenta con base datos, pero usa ficheros para la lectura y escritura de información.

A continuación, se detallan las principales características técnicas de la aplicación:

3.1. Diagrama de clases

La Figura 17 presenta la aplicación de la propuesta de diagrama de clase UML. Como se aprecia en las Figuras 5, existe una clara analógica entre los componentes del diagrama de clases UML. Como se puede observar en el diagrama, contamos con 5 clases que están interrelacionadas entre sí.

Como se puede observar, el diagrama de clases UML ha sido elaborado tomando en cuenta los principales conceptos de programación orientada a objetos (herencia, cohesión, abstracción).

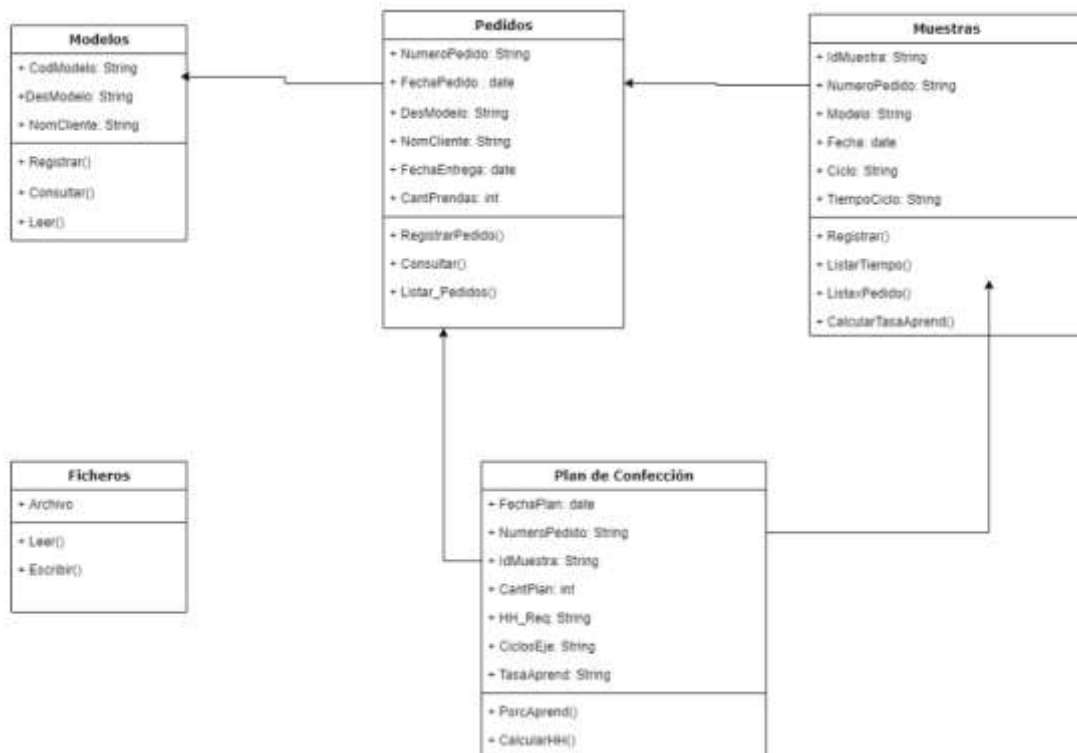


Figura 17 diagrama de clases

Fuente: Elaboración propia

3.2. Estructura del proyecto

A Continuación, se muestra la estructura del proyecto:

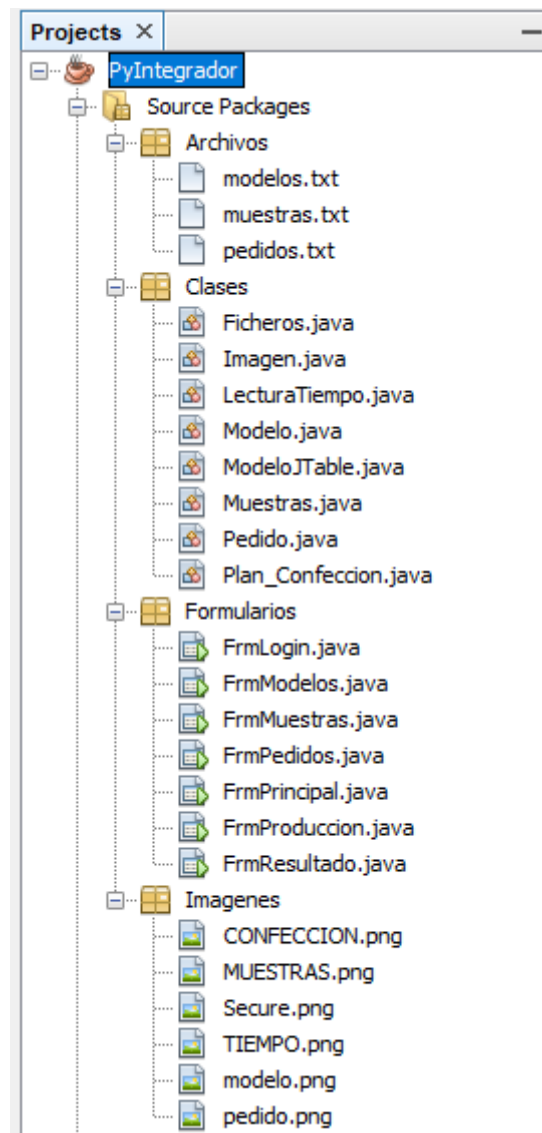


Figura 18 Estructura del Proyecto

Fuente: Elaboración propia

La aplicación está organizada en 4 paquetes principales que son las siguientes: ficheros, clases, formularios e imágenes.

Paquete	Descripción	
Archivos	Modelos	Archivo de texto que contiene la información de los modelos registrados.
	Muestras	Archivo de texto que contiene la información de las muestras registrados.
	Pedidos	Archivo de texto que contiene el detalle de los pedidos ingresados
Clases	Ficheros	Clase que realiza la lectura y escritura de la información guardada en los archivos planos.
	Imagen	Clase que ayuda a ajustar las imágenes usadas en el proyecto
	Modelo	Esta clase contiene los métodos que permiten dar mantenimiento al registro de modelo de calzados
	ModeloJTable	Esta clase contiene un componente visual que nos permite dibujar una tabla, de forma que en cada fila/columna de la tabla podamos poner el dato que queramos.
	Muestras	Esta clase contiene los métodos que permiten almacenar y recuperar la información de las muestras registradas.
	Pedido	Esta contiene los métodos que permite dar mantenimiento al registro de Ordenes de Pedidos de Producción
	Plan_Confeccion	Esta clase contiene los métodos que permiten calcular la cantidad de hombres requeridas para un plan de confección.
Formularios	FrmLogin	Interfaz para acceder al sistema
	FrmModelos	Interfaz para registrar los modelos de calzado
		Interfaz para el registro de los ciclos de prueba y su respectivo tiempo para una determina muestra de confección.
	FrmMuestras	
	FrmPedidos	Interfaz para el registro los pedidos del cliente.
	FrmPrincipal	Interfaz que muestra las opciones que dispone el sistema.
	FrmProducción	Interfaz para el cálculo de horas hombre basado en el pedido y muestra seleccionada
	FrmResultado	Interfaz para mostrar el cálculo efectuado en FrmProduccion.

Tabla 3. Estructura de la Aplicación

Fuente: Elaboración propia

3.3. Contenido del proyecto

En este apartado se explicará el contenido de cada paquete.

- **Paquete Archivos**

Este paquete contiene los archivos planos que almacenan la información registrada en el sistema. Estos archivos tienen la información separada por palotes “|”.

En este paquete podemos encontrar los siguientes archivos:

- **Modelos:** Contiene Archivo de texto que contiene la información de los modelos registrados.
- **Muestras:** Archivo de texto que contiene la información de las muestras registradas.
- **Pedidos:** Archivo de texto que contiene el detalle de los pedidos ingresados.

- **Paquete Clases**

Este paquete contiene las clases que son usadas para realizar el registro de la data, lectura y además cálculos.

- **Clase Ficheros:** Esta clase contiene los métodos que permiten la lectura y escritura en los archivos de texto. Usa como separador de campos “|”. Además en esta clase usamos la librería java.io.*
 - **Método Leer:** Este método recibe como parámetro el nombre del archivo y a continuación procede a realizar la lectura y a devolver el resultado un arraylist de tipo string. Para abrir el archivo en modo lectura utilizamos la clase FileReader y para obtener el contenido utilizamos la clase BufferedReader, cada fila es recorrida por el método readLine de la clase BufferedReader.


```

public ArrayList<String> Leer(String p_FileName) {
    FileReader fichero;
    String line;
    ArrayList<String> lista = new ArrayList<String>();
    try {
        fichero = new FileReader(p_FileName);
        try {
            fichero = new FileReader(p_FileName);
            BufferedReader buf = new BufferedReader(fichero);
            // Leemos linea a linea el fichero
            while ((line = buf.readLine()) != null) {
                lista.add(line);
            }
        } catch (FileNotFoundException ex1) {
            JOptionPane.showMessageDialog(null, "No se encontró archivo: " + p_FileName);
        } catch (Exception e) {
            e.printStackTrace();
        } finally { // Cerramos el fichero tanto si la lectura ha sido correcta o no
            try {
                if (null != fichero) {
                    fichero.close();
                }
            } catch (Exception e2) {
                e2.printStackTrace();
            }
        }
    } catch (Exception e) {
    }
    return lista;
}

```

Figura 19. Método Leer

- **Método Escribir:** Este método permite escribir en un fichero de texto, para ello recibe 2 parámetros, el primero es el nombre del archivo y el segundo un string, con ello procede escribir en el fichero agregando filas al final. En caso el archivo no exista, procede a crearlo. Para la creación y/o apertura del archivo en modo escritura procedemos a utilizar la clase `FileWriter` y para manipular el fichero utilizamos la clase `PrintWriter` cuyo método `println` nos permite escribir la cadena recibida en el segundo parámetro.

```

public void Escribir(String p_FileName, String p_texto) {
    FileWriter fichero = null;
    PrintWriter pw = null;
    try {
        //Abro stream, crea el fichero si no existe.
        fichero = new FileWriter(p_FileName, true);
        //Escribimos en el fichero
        pw = new PrintWriter(fichero);
        pw.println(p_texto);
    } catch (FileNotFoundException ex1) {
        JOptionPane.showMessageDialog(null, "No se encontró archivo: " + p_FileName);
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        try {
            // Nos vamos a aprovechar el finally para
            // asegurarnos que se cierre el fichero.
            if (null != fichero) {
                fichero.close();
            }
        } catch (Exception e2) {
            e2.printStackTrace();
        }
    }
}

```

Figura 20. Método Escribir

- **Método BuscarCadena:** Este método permite buscar un determinado texto en un fichero, esto con el fin de determinar si un registro a almacén ya existe y así evitar duplicidad, para ello recibe como parámetro el nombre del archivo y la cadena a buscar. Como respuesta devuelve un booleano donde TRUE significa que encontró la cadena y FALSE lo contrario.

```
public boolean BuscarCadena(String p_FileName, String cadena) {
    FileReader fichero = null;
    boolean found = false;
    String line = null;
    int valor;
    try {
        //Abrir el stream, el fichero debe existir
        fichero = new FileReader(p_FileName);
        BufferedReader buf = new BufferedReader(fichero);
        //Leer el fichero
        while ((line = buf.readLine()) != null) {
            valor = line.indexOf(cadena);
            if (valor >= 0) {
                found = true;
                break;
            }
            System.out.println(line);
        }
    } catch (FileNotFoundException ex1) {
        JOptionPane.showMessageDialog(null, "No se encontró archivo: " + p_FileName);
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        try {
            if (null != fichero) {
                fichero.close();
            }
        } catch (Exception e2) {
            e2.printStackTrace();
        }
    }
}
```

Figura 21. Método BuscarCadena

- **Clase Imagen:** Esta clase contiene el código que facilita insertar imágenes en los formularios. Además, las ajusta según lo requerido. Para esta clase importamos las librerías java.awt.Dimension, java.awt.Graphics y javax.swing.ImageIcon.

```
package Clases;

import java.awt.Dimension;
import java.awt.Graphics;
import javax.swing.ImageIcon;

/**
 *
 * @author Luis Garcia
 */
public class Imagen extends javax.swing.JPanel {
    public String sRuta;
    public Imagen(int iX,int iY,int iWidth, int iHeight,String cRuta) {
        this.setSize(iWidth, iHeight);
        this.setLocation(iX, iY);
        sRuta = cRuta;
    }
    public void paint(Graphics grafico) {
        Dimension height = getSize();
        ImageIcon lng = new ImageIcon(getClass().getResource(sRuta));
        grafico.drawImage(lng.getImage(), 0, 0, height.getWidth(), height.getHeight(), null);
        setOpaque(false);
        super.paintComponent(grafico);
    }
}
```

Figura 22. Clase Imagen

- **Clase LecturaTiempo:** Esta clase se definen los métodos que se usaran para la lectura de los tiempos y se hace un llamado a los mismos.

```
public class LecturaTiempo {

    int ciclo;
    double tiempo;

    public int getCiclo() {
        return ciclo;
    }

    public void setCiclo(int ciclo) {
        this.ciclo = ciclo;
    }

    public double getTiempo() {
        return tiempo;
    }

    public void setTiempo(double tiempo) {
        this.tiempo = tiempo;
    }

    public LecturaTiempo(int ciclo, double tiempo) {
        this.ciclo = ciclo;
        this.tiempo = tiempo;
    }

    public LecturaTiempo() {
    }

}
```

Figura 23. LecturaTiempo

- **Clase Modelo:** Esta clase contiene los métodos que permiten dar mantenimiento al registro de modelo de calzado, para ello cuenta con 3 métodos principales: Registrar, Consultar y Listar.
 - **Método Registrar:** Este método toma la información registrada por el usuario y procede a instanciar la clase ficheros para almacenar en el archivo de textos.

```
//metodo que registra un modelo
public void Registrar(String pCodigo, String pDescripcion, String pCliente) {
    Ficheros fil = new Ficheros();
    Modelo dt = new Modelo();
    dt.setCodModelo(pCodigo);
    dt.setDesModelo(pDescripcion);
    dt.setNomCliente(pCliente);

    if (fil.BuscarCadena(fil.getFileName(nomarchivo), pCodigo) == true) {
        JOptionPane.showMessageDialog(null, "Modelo ya existe");
    } else {
        fil.Escribir(fil.getFileName(nomarchivo), dt.toString());
    }
}
```

Figura 24. Registrar Modelo

- **Método Consultar:** Este método permite buscar un código de modelo específico indicado en el parámetro de entrada. Para ello utiliza la clase ficheros donde procederá a verificar la existencia del registro.

```
//metodo que obtiene un modelo por el campo pCodModelo
public Modelo consultar(String pCodModelo) {
    Modelo objModelo = new Modelo();
    Modelo obj = new Modelo();
    ArrayList<Modelo> lista = new ArrayList<Modelo>();
    lista = objModelo.Listar();
    for (int i = 0; i < lista.size(); i++) {
        obj = lista.get(i);
        if (Objects.equals(pCodModelo.trim(), obj.getCodModelo().trim())) {
            objModelo = lista.get(i);
            break;
        }
    }
    return objModelo;
}
```

Figura 25. Consultar Modelo

- **Método Listar:** Este método permite obtener el listado de modelos y devolverlos en un arraylist. También utiliza la clase ficheros, pues realizará la lectura de todo el archivo de texto.

```
//metodo que lista los modelos
public ArrayList<Modelo> Listar() {
    ArrayList<Modelo> lista = new ArrayList<Modelo>();

    Ficheros fil = new Ficheros();
    ArrayList<String> lsdat = fil.Leer(fil.getFileName(nomarchivo));
    Modelo dt;
    for (int i = 0; i < lsdat.size(); i++) {
        dt = new Modelo();
        dt = convierteCadena(lsdat.get(i));
        lista.add(dt);
    }
    return lista;
}
```

Figura 26. Listar Modelo

- **Clase Pedido:** Esta clase contiene a los métodos que ayudan al mantenimiento de los pedidos. Esta clase también hereda los modelos de la Clase Modelo. En esta clase encontramos los métodos Registrar Pedido, Consultar y ListarPedido.

- **Método RegistrarPedido:** Este método permite registrar la información del pedido ingresado por el usuario en un archivo de texto. Para ello instancia la clase Ficheros para utilizar el método Escribir. Antes de proceder a grabar, hace una verificación de existencia del pedido para evitar duplicidad.

```

public void RegistrarPedido(String pNumeroPedido, String pFechaPedido, String pFechaEntrega, String pDesModelo, Str
    Ficheros fil = new Ficheros();
    Pedido dt = new Pedido();
    dt.setNumeroPedido(pNumeroPedido);
    dt.setDesModelo(pDesModelo);
    dt.setNomCliente(pNomCliente);
    dt.setFechaPedido(pFechaPedido);
    dt.setFechaEntrega(pFechaEntrega);
    dt.setCantPares(pCantPares);
    if (fil.BuscarCadena(fil.getFileName(nomarchivo), pNumeroPedido) == true) {
        JOptionPane.showMessageDialog(null, "Pedido ya existe");
    } else {
        fil.Escribir(fil.getFileName(nomarchivo), dt.toString());
    }
}

```

Figura 27. Registrar Pedido

- Método Consultar: Este método permite consultar la existencia de un número de pedido y obtener la información que corresponda en caso exista.

```

public Pedido Consultar(String pNumeroPedido) {
    int compara;
    Pedido objPedido = new Pedido();
    Pedido obj = new Pedido();
    ArrayList<Pedido> lista = new ArrayList<Pedido>();
    lista = objPedido.Listar_Pedidos();
    for (int i = 0; i < lista.size(); i++) {
        obj = lista.get(i);
        compara = pNumeroPedido.indexOf(obj.getNumeroPedido());
        if (compara >= 0) {
            objPedido = lista.get(i);
            break;
        }
    }
    return objPedido;
}

```

Figura 28. Consultar Pedido

- Método Listar_Pedidos: Este método permite consultar la existencia de un número de pedido y obtener la información que corresponda en caso exista.

```

public ArrayList<Pedido> Listar_Pedidos() {
    ArrayList<Pedido> lista = new ArrayList<Pedido>();

    Ficheros fil = new Ficheros();
    ArrayList<String> lsdat = fil.Leer(fil.getFileName(nomarchivo));
    Pedido dt;
    for (int i = 0; i < lsdat.size(); i++) {
        dt = new Pedido();
        dt = convierteCadena(lsdat.get(i));
        lista.add(dt);
    }
    return lista;
}

```

Figura 29. Listar Pedido

- **Clase ModeloJTable:** Esta clase contiene un componente visual que nos permite dibujar una tabla, de forma que en cada fila/columna de la tabla podamos poner el dato que queramos.

```

import java.util.ArrayList;
import javax.swing.table.AbstractTableModel;

/**
 *
 * @author Luis.garcia
 */

public class ModeloJTable extends AbstractTableModel {
    public ArrayList<Modelo> muestras;
    public String [] encabezados = {"ID", "Title", "xxx"};
    public ModeloJTable()
    {
        int i=0;
        muestras = new ArrayList<Modelo>();
    }
    public ModeloJTable(String [] head)
    {
        for (int i = 0; i<head.length; i++)
            encabezados[i] = head[i];
        muestras = new ArrayList<Modelo>();
    }
    @Override
    public int getRowCount() {
        return muestras.size();
    }
}

```

Figura 30. ModeloJtable

- **Clase Muestra:** Esta clase contiene los métodos que permiten almacenar y recuperar la información de las muestras registradas, asimismo realiza el cálculo de la tasa de aprendizaje basado en las muestras. Sus métodos principales son: Registrar, ListarporPedido, Listartiempo y Cal_Tasa_Aprendizaje.

- **Método Registrar:** Este método permite almacenar la información registrada por el usuario. Recibe como parámetros los datos generales de la muestra y un DefaultTableModel que contiene el tiempo de cada ciclo para la muestra realizada.

```

public Boolean Registrar(String pIDMuestra, String pNumPedido, String pModelo, String pFecha, DefaultTableModel tbl) {
    Boolean estado = false;
    Ficheros fil = new Ficheros();
    if (fil.BuscarCadena(fil.getFileName(nomarchivo), pIDMuestra) == true) {
        JOptionPane.showMessageDialog(null, "Muestra ya existe");
    } else {
        for (int i = 0; i < tbl.getRowCount(); i++) {
            Muestras dt = new Muestras();
            dt.IDMuestra = pIDMuestra;
            dt.NumPedido = pNumPedido;
            dt.Modelo = pModelo;
            dt.Fecha = pFecha;
            dt.Ciclo = Integer.parseInt(tbl.getValueAt(i, 0).toString());
            dt.TiempoCiclo = Double.parseDouble(tbl.getValueAt(i, 1).toString());
            fil.Escribir(fil.getFileName(nomarchivo), dt.toString());
            estado = true;
        }
    }
    return estado;
}

```

Figura 31. Registrar Muestra

- **Método ListarporPedido:** Este método permite obtener las muestras relacionadas a un numero de pedido. La respuesta es una arraylist de tipo muestra, pero solo con la información de la cabecera de la muestra.

```
public ArrayList<Muestras> ListarporPedido(String pNumPedido)
{
    Integer pos;
    Boolean busca;
    String nroped;
    ArrayList<Muestras> lista = new ArrayList<Muestras>();

    Ficheros fil = new Ficheros();
    ArrayList<String> lsdat = fil.Leer(fil.getFileName(nomarchivo));
    Muestras dt;
    for(int i=0;i<lsdat.size();i++)
    {
        // dt = new Muestras();
        dt = convierteCadena(lsdat.get(i));
        nroped = dt.getNumPedido().trim();
        busca = Objects.equals(pNumPedido.trim(),nroped);
        if( busca == true)
        {
            if(buscar_lista(lista,dt.getIDMuestra()) == false)
                lista.add(dt);
        }
    }
    return lista;
}
```

Figura 32. Registrar por Pedido

- **Método ListarTiempo:** Este método permite obtener el listado de tiempo registrados por cada ciclo de la muestra consultada que se indica como parámetro de entrada. La respuesta es un arraylist de tipo Muestra.

```
public ArrayList<Muestras> Listartiempo(String pIDMuestra)
{
    int compare;
    ArrayList<Muestras> lista = new ArrayList<Muestras>();
    Ficheros fil = new Ficheros();
    ArrayList<String> lsdat = fil.Leer(fil.getFileName(nomarchivo));
    Muestras dt;
    for(int i=0;i<lsdat.size();i++)
    {
        // dt = new Muestras();
        dt = convierteCadena(lsdat.get(i));
        compare = pIDMuestra.indexOf(dt.getIDMuestra());
        if(compare >= 0)
        {
            lista.add(dt);
        }
    }
    return lista;
}
```

Figura 33. Listar Tiempo

- **Método Cal_Tasa_Aprendizaje:** Este método permite calcular la tasa de aprendizaje, para ello recibe como parámetro un arraylist que contiene el tiempo por cada ciclo de pruebas. Con la información procede a obtener el factor del segundo entre el primero, luego el cuarto entre el segundo, el octavo entre el cuarto y así sucesivamente. De todos los factores se procede a promediar el cual finalmente se convierte en la tasa de aprendizaje.

```
public double Cal_Tasa_Aprendizaje(ArrayList<Muestras> lis)
{
    Muestras mus = new Muestras();
    double valor1 = 0, valor2 = 0, porcen = 0;
    int next_fil = 0, reg = 0, fila_anterior = 0, fila = 0;
    Double valores[] = new Double[100];
    for(int i=0; i<lis.size(); i++)
    {
        fila = fila + 1;
        if(fila == 2) {
            mus = lis.get(fila_anterior);
            valor1 = mus.getTiempoCiclo();
            mus = lis.get(i);
            valor2 = mus.getTiempoCiclo();
            porcen = valor2 / valor1;
            next_fil = fila*2;
            valores[reg] = porcen;
            fila_anterior = i;
            reg++;
        }
        if((fila == next_fil && next_fil != 0) || (i == lis.size() - 1)) {
            mus = lis.get(fila_anterior);
            valor1 = mus.getTiempoCiclo();
            mus = lis.get(i);
            valor2 = mus.getTiempoCiclo();
            porcen = valor2 / valor1;
            next_fil = fila*2;
            valores[reg] = porcen;
            fila_anterior = i;
            reg++;
        }
    }
}
```

Figura 34. Calcular tasa de aprendizaje

- **Clase Plan_Confección:** Esta clase contiene los métodos que permiten calcular la cantidad de hombres requeridas para un plan de confección. Contiene 2 métodos principales: CalcularHH y porc_aprendizaje.
- **Método Porc_Aprendizaje:** Este método permite calcular el exponente de la integral para el cual utiliza la tasa de aprendizaje y aplica el cálculo de logaritmo.


```

public double porc_aprendizaje( double p_tasa)
{
    double porcentaje = 0;
    return porcentaje = Math.log10(p_tasa)/Math.log10(2);
}

```

Figura 35. Porc_aprendizaje

- Método CalcularHH: Este método permite calcular el total de horas hombre requerido utilizando la integral definida, para ello utiliza como datos: La cantidad inicial de calzados que se hicieron en el ciclo de muestras, la cantidad de calzados a calcular, la tasa de aprendizaje en porcentaje y el tiempo que se tomó en fabricar la primer par de calzado.

```

public double CalcularHH(Integer pInicio,Integer pFinal,double p_tasa,double p_tetandar)
{
    double porc = 0;
    double horas;
    double factor = 0;

    porc = porc_aprendizaje(p_tasa);
    factor = porc + 1;

    horas = p_tetandar* ((Math.pow((double) pFinal,factor) / factor) - (Math.pow((double) pInicio,factor) / factor));
    horas = horas/60;

    horas = redondearDecimales(horas,2);
    return horas;
}

```

Figura 36. Calcular HH

• Paquete Formularios

Este paquete contiene los formularios que han sido diseñados para la aplicación.

- FrmLogin: Interfaz que permite acceder al sistema.
- FrmModelos: Interfaz para dar mantenimiento a los modelos de calzado.
- FrmMuestras: Interfaz para el registro de los ciclos de prueba y su respectivo tiempo para una determina muestra de confección.
- FrmPedidos: Interfaz para el registro los pedidos del cliente.
- FrmPrincipal: Interfaz que muestra las opciones que dispone el sistema.
- FrmProducción: Interfaz para el cálculo de horas hombre basado en el pedido y muestra seleccionada
- FrmResultado: Interfaz para mostrar el cálculo efectuado en FrmProduccion.

3.4. Prototipos

En esta sección se incluirán las vistas o prototipos diseñados para la aplicación.

- **LOGIN**

Es la interfaz que controla el acceso individual a la aplicación mediante la identificación del usuario utilizando credenciales provistas por el usuario.



Figura 37 Login

- **MENÚ PRINCIPAL**

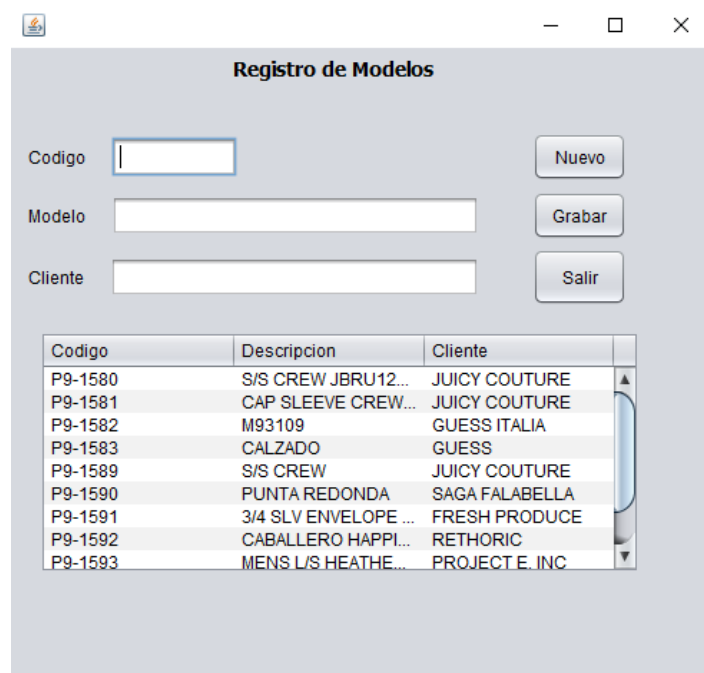
Es la interfaz que muestra las opciones del sistema. Al dar click sobre alguna de ellas, procede a redireccionarnos a dicha opción.



Figura 38. Menú Principal

- **REGISTRO DE MODELOS**

Es la interfaz que permite darle mantenimiento a los modelos de calzado existentes en la fábrica.

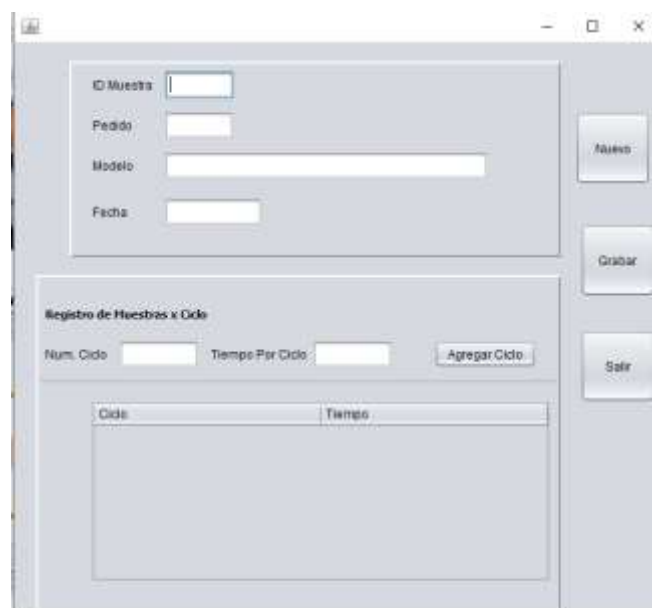


Codigo	Descripcion	Cliente
P9-1580	S/S CREW JBRU12...	JUICY COUTURE
P9-1581	CAP SLEEVE CREW...	JUICY COUTURE
P9-1582	M93109	GUESS ITALIA
P9-1583	CALZADO	GUESS
P9-1589	S/S CREW	JUICY COUTURE
P9-1590	PUNTA REDONDA	SAGA FALABELLA
P9-1591	3/4 SLV ENVELOPE ...	FRESH PRODUCE
P9-1592	CABALLERO HAPPY...	RETHORIC
P9-1593	MENS L/S HEATHE...	PROJECT E. INC

Figura 39. Registro de Modelos

- **MUESTRAS**

Es la interfaz que permite seleccionar muestras y registrar el tiempo de confección por cada ciclo.



Ciclo	Tiempo
-------	--------

Figura 40. Muestras

- **PEDIDOS**

Es la interfaz permite registrar el los datos del modelo, datos del cliente, fecha de recepción del pedido, fecha de entrega del pedido y cantidad de pares solicitados.

Figura 41. Pedidos

- **PRODUCCIÓN**

Esta interfaz permite el cálculo de horas hombre basado en el pedido y muestra seleccionada.

Figura42. Producción

- **RESULTADO**

Esta interfaz muestra el resultado de los cálculos realizados en la interfaz producción (unidades a producir, total de HH, N° de personas y días requeridos).

Figura 43. Resultados

CAPITULO 4

RESULTADOS

4.1. Prueba del Aplicativo

A continuación, se detalla un caso de prueba ejecutado para el aplicativo.

4.1.1. Caso

La empresa BAMBINI SHOES ubicada en la ciudad de Trujillo, cuyo rubro de negocio es la elaboración de calzado ha recibido un pedido de su cliente JUICY COUTURE para la elaboración de 3500 pares de calzado. La empresa requiere calcular cuantas horas hombre se requerirán para la elaboración de dicho pedido, además confirmar la fecha de entrega para dicho pedido. Además, se tiene como dato la siguiente muestra tomada por el área de producción de la empresa BAMBINI SHOES.

Ciclo	Tiempo en Minutos
1	15
2	14.8
3	14.6
4	14.7
5	14.3
6	14.2
7	13.5
8	13.8
9	13.2
10	13
11	12.5
12	12.4
13	12.3
14	12.1
15	11.56
16	11.3

Tabla 4. Muestra Empresa BAMBINI SHOES

4.1.2. Resultados Esperados

A continuación, se procede a realizar los cálculos manuales para la solución de dicho caso usando los conceptos de curva de aprendizaje.

Tomando como base la table de tiempos por ciclo se procede a calcular la curva de aprendizaje de la siguiente forma:

- Procedemos a tomar los tiempos del ciclo 2 y ciclo 1 los dividimos, teniendo como resultado un porcentaje.

$$\frac{\text{Ciclo 2}}{\text{Ciclo 1}} \quad 14.8 / 15 = 0.987$$

- Luego proceder a obtener el porcentaje existente entre el ciclo 4 y ciclo 2

$$\frac{\text{Ciclo 4}}{\text{Ciclo 2}} \quad 14.8 / 15 = 0.993$$

- A continuación; procedemos a obtener el porcentaje existente entre el ciclo 8 y ciclo 4

$$\frac{\text{Ciclo 8}}{\text{Ciclo 4}} \quad 14.8 / 15 = 0.9388$$

- Finalmente obtenemos el porcentaje existente entre el ciclo 16 y ciclo 8

$$\frac{\text{Ciclo 16}}{\text{Ciclo 8}} \quad 14.8 / 15 = 0.8188$$

- Con los factores obtenidos, se obtiene el promedio

$$\text{Promedio} = 0.934$$

- Una vez que tenemos el promedio, se procede a calcular la integral con la siguiente fórmula

$$\int_a^b \text{tiempo inicial} * X^{\frac{\log_{10}(\text{Tasa aprendizaje})}{\log_{10}(2)}}$$

Donde:

a = Cantidad de pares ya confeccionada en la muestra. (15)

b = Cantidad de pares a confeccionar más la cantidad ya realizada en muestra. (3515)

tiempo inicial = 15 minutos

Tasa aprendizaje = 0.934

$$\int_{15}^{3515} 15 * X^{\log_{10}(0.934)/\log_{10}(2)}$$

$$\text{índice} = \log_{10}(0.934) / \log_{10}(2) = -0.09644385$$

$$15 \int_{15}^{3515} X^{-0.9344385} dx$$

$$15 * \frac{X^{-0.9344385+1}}{-0.9344385+1} \Big|_{15}^{3515}$$

$$\text{Exponente} = -0.9344385 + 1 = 0.0655615$$

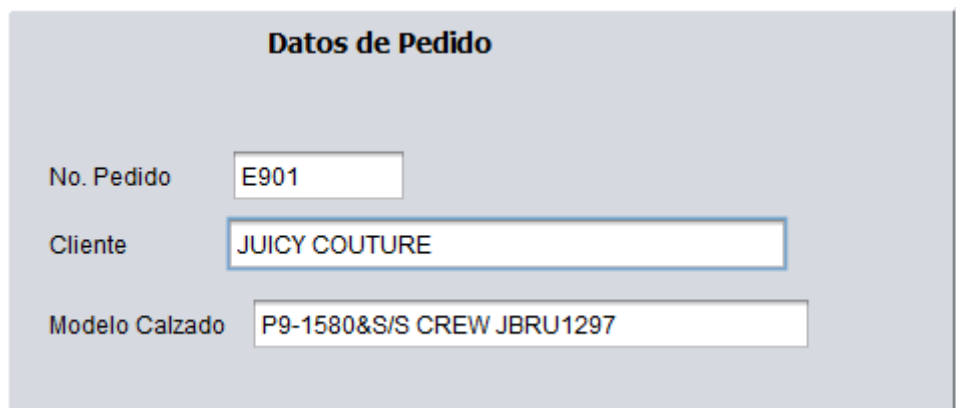
$$\text{Minutos} = 5 * \left[\frac{3515^{0.0655615}}{0.0655615} - \frac{15^{0.0655615}}{0.0655615} \right]$$

$$\text{Minutos} = 26,085.6$$

$$\text{Horas} = 434.76$$

4.1.3. Resultados del Aplicativo

En esta sección, se mostrarán los datos obtenidos con el uso de la aplicación para la solución del mismo caso.



Datos de Pedido

No. Pedido: E901

Cliente: JUICY COUTURE

Modelo Calzado: P9-1580&S/S CREW JBRU1297

Figura 26.. Datos de Pedido Juicy Couture



Datos de Muestra

de Muestra: M001 & 12.10.2018 Num. Ciclos: 16

Ciclo de muestra

Ciclo	Tiempo
1	15.0
2	14.8
3	14.6
4	14.7
5	14.3
6	14.2
7	13.5
8	13.8

Tasa Aprendizaje: 93.44

Figura 44. Cálculo de Horas Hombre

The screenshot shows a window titled "RESULTADOS" with a light gray background. Inside, there is a white rectangular box containing four rows of data. Each row has a label on the left and a text input field on the right. The labels are "Unidades a Producir", "Total Horas Hombre", "Nº de Personas", and "No de Días Requeridos". The values entered in the input fields are "3500", "434.76", "10", and "5.4" respectively. Below this box, there is a button labeled "Salir" (Exit).

Unidades a Producir	3500
Total Horas Hombre	434.76
Nº de Personas	10
No de Días Requeridos	5.4

Salir

Figura 45. Resultados Obtenidos

Como podemos observar en la **Figura 27**, la cantidad de horas hombre requeridas que calculo el sistema coinciden con el calculo que realizamos previamente de manera manual.

4.2. Costos y Beneficios

El presente análisis de costo y beneficios del proyecto busca dar a conocer si es que este es rentable, considerando que será rentable cuando la relación costo-beneficio es mayor que la unidad.

	Concepto	Monto S/.
EGRESOS	Asesoría de experto en producción de calzado.	S/ 5,000.00
	AP Java	S/ 5,000.00
	Equipo Informático	S/ 3,000.00
	Movilidades y viáticos	S/ 500.00
	TOTAL EGRESOS	S/ 13,500.00

Tabla 5 Egresos

	Concepto	Monto S/.
INGRESOS	Primer pago por instalación de la aplicación	S/ 20,000.00
	Pago por soporte y mantenimiento	S/ 2,000.00
	TOTAL EGRESOS	S/ 22,000.00

Tabla 6 Ingresos

$$B/C = 22,000 / 13,500 = 1.63$$

En base a los resultados obtenidos, podemos decir que nuestra aplicación resulta rentable.

CAPITULO 5

CONCLUSIONES

5.1. Conclusiones

Luego de haber culminado el desarrollo de la presente aplicación, se concluye lo siguiente:

- La aplicación desarrollada está enfocada en facilitar el cálculo de horas hombre requeridas para la elaboración de un pedido de manera más rápida, atractiva y sobre todo evitándonos pérdidas de tiempo al obtener esta información.
- Se obtuvieron grandes conocimientos en el manejo de las tecnologías relacionadas con un proyecto de este tipo. Se aprendieron conceptos importantes para el desarrollo de software dentro de las limitaciones típicas. También se obtuvieron conocimientos avanzados en el uso del lenguaje de programación Java.
- A medida que la aplicación fue tomando forma, se implementaron funciones adicionales al diseño original. Esto sucedió ya que se notaron algunas posibles mejoras que no representaban grandes inconvenientes desde el punto de vista de implementación, pero otorgaban considerables mejoras en la usabilidad general de la aplicación.
- El cálculo de la curva de aprendizaje que nosotros hemos empleado en la aplicación también podría ser utilizada en otros campos como en la educación, en medicina, entre otros, ya que proporciona un aporte de información muy útil.

5.2. Recomendaciones

- Se recomienda escalar la aplicación e implementar una base de datos, además de desplegarla en un servidor de aplicaciones.
- Para diseñar la siguiente fase de la aplicación, se debe tomar muy en cuenta aspectos como la navegabilidad y usabilidad de esta ya que de estos depende el éxito de la aplicación.
- Se recomienda hacer seguimiento al plan de producción para asegurar el cumplimiento del mismo. Para ello, también resultaría interesante implementar

indicadores de gestión que permitan tomar decisiones pronto para poder cumplir con la demanda.

ANEXOS

ANEXO 01: LEAN CANVAS



BIBLIOGRAFÍA

Aisemberg, Daniel (2014). Es bueno que las pymes conozcan los beneficios de un sistema ERP para que aprovechen las ventajas que las tecnologías de la información proveen. Buenos negocios. Recuperado de: <https://www.buenosnegocios.com/5-ventajas-los-sistemas-erp-n682>

Alarcón, S. (2017). Anuario del sector mundial del calzado año 2015. Revista virtual del calzado.

Chango, M., & Zambrano, I. (2017). Las curvas de aprendizaje. Sangolqui, Ecuador: Universidad de las Fuerzas Armadas ESPE.

Chero Arbañil, M. A. (2016). Implementación de un sistema de planeamiento y control de las operaciones para elevar el nivel de productividad de la línea de fabricación de mochilas en la “Industria Camel-Perú EIRL” Lima–Perú 2015, 2016.

Cristina Cachero, P. J. (2011). Herencia. Programación orientada a objetos. Universidad de Alicante.

Flores, M. (2013). Propuesta de implementación de un MRP II para una planta de confecciones textiles. Universidad Pontificia Católica del Perú.

Latiff, A. (2005). Curva de Aprendizaje. qué es y cómo se mide, 15-17.

PRODUCTORES DE CALZADO EN EL PERÚ

<http://www.andina.com.pe/Espanol/noticia-el-967-productores-calzado-peru-son-microempresas-381243.aspx>

PROBLEMÁTICA DEL CALZADO EN EL PERÚ

<http://peru21.pe/noticia/729815/peru-necesita-marca-pais-sector-calzado>

Rodríguez, A. G. (2010). Introduccion a la programacion orientada a objetos. Universitat Oberta de Catalunya.