## **MAXIMIZANDO EL POTENCIAL ATLÉTICO: CIENCIA DE DATOS Y LACTATO EN EL DEPORTE DE ALTO RENDIMIENTO**

**WILLIAM ENRIQUE MARTINEZ CRISTIAN STIVEN GUERRERO A.**

## **UNIVERSIDAD DE CUNDINAMARCA**

**Facultad de Ingeniería**

## **Programa de Tecnología en Desarrollo de Software Soacha**

**Soacha (Cundinamarca)**

**abril 2024**

## **MAXIMIZANDO EL POTENCIAL ATLÉTICO: CIENCIA DE DATOS Y LACTATO EN EL DEPORTE DE ALTO RENDIMIENTO**

## 

**AUTORES**

## **WILLIAM ENRIQUE MARTINEZ CRISTIAN STIVEN GUERRERO**

**TRABAJO PARA OBTENER TITULO (TECNOLOGO EN DESARROLLO DE SOFTWARE)**

**Director: ing. Edgar Arturo Bustos Caldas**

## **UNIVERSIDAD DE CUNDINAMARCA**

**Facultad de Ingeniería**

## **Programa de Tecnología en Desarrollo de Software Soacha**

**Soacha, abril 2024**

Nota de Aceptación

Presidente jurado

Jurado

Jurado

Soacha, 03 de abril de 2024

**Dedicatoria**

A las personas que más amo en este mundo, María Lucy Moreno, Patricia Alejandra Gómez, mi hija Eilyn Camila Martínez y mi amada esposa Diana Cifuentes,

Este proyecto de grado es el resultado de innumerables horas de esfuerzo y dedicación. Cada página, cada palabra, cada idea plasmada aquí lleva consigo una parte de mi corazón y mi alma.

A María Lucy y Patricia Alejandra, gracias por ser mi apoyo incondicional, por creer en mí incluso cuando yo no lo hacía, por ser mi faro en los momentos de oscuridad. Este logro es tan mío como suyo.

A mi hija Eilyn Camila, quiero que veas en este proyecto el valor del estudio y el trabajo duro. Que sea un ejemplo para ti de que, con esfuerzo y dedicación, no hay meta que no se pueda alcanzar. Espero que este logro te inspire a seguir tus sueños, a luchar por lo que quieres y a nunca rendirte.

A mi esposa Diana, gracias por estar a mi lado en cada paso de este camino, por tu amor, tu paciencia y tu comprensión. Tu apoyo ha sido fundamental en este logro y quiero que sepas que cada éxito mío es también un éxito tuyo.

Con todo mi amor,

William Enrique Martínez

**Agradecimientos**

Agradezco profundamente a la Universidad y a todos los docentes colaboradores que han sido parte integral de mi proyecto de grado. Su apoyo incondicional, orientación experta y aliento constante han sido fundamentales en cada paso de este viaje académico.

A la Universidad, por brindarme la oportunidad y los recursos para llevar a cabo este proyecto. Su compromiso con la excelencia académica y el desarrollo personal de los estudiantes es verdaderamente inspirador.

A los docentes colaboradores, por su paciencia y dedicación. Su conocimiento y experiencia han sido una guía invaluable. Cada consejo, cada crítica constructiva, ha sido un escalón en mi crecimiento y aprendizaje.

Este proyecto de grado no solo representa el culmen de mis estudios, sino también el comienzo de nuevas aventuras académicas y profesionales. Estoy eternamente agradecido por tener la fortuna de estar rodeado de mentores tan increíbles y una institución tan prestigiosa.

Gracias por creer en mi potencial y por ayudarme a dar lo mejor de mí.

## **COMPROMISO** **DE AUTOR**

Yo **William Enrique Martínez** con cédula de identidad No.1010173961 y con cód. 793220215 estudiante del programa de Tecnología en Desarrollo deSoftware de la Universidad de Cundinamarca, declaro que:

El contenido del presente documento es un reflejo de mi trabajo personal y manifiesto que, ante cualquier notificación de plagio, copia o falta a la fuente original, soy responsable directo legal, económico y administrativo sin afectar al director del trabajo, a la Universidad y a cuantas instituciones hayan colaborado en dicho trabajo, asumiendo las consecuencias derivadas de tales prácticas.

## **Firma:**

**COMPROMISO DEL AUTOR**

Yo **Cristian Stiven Guerrero Andrade** con cédula de identidad No. 1032796841 y con cód. 793222119 estudiante del programa de Tecnología en Desarrollo deSoftware de la Universidad de Cundinamarca, declaro que:

El contenido del presente documento es un reflejo de mi trabajo personal y manifiesto que, ante cualquier notificación de plagio, copia o falta a la fuente original, soy responsable directo legal, económico y administrativo sin afectar al director del trabajo, a la Universidad y a cuantas instituciones hayan colaborado en dicho trabajo, asumiendo las consecuencias derivadas de tales prácticas.

## **Firma:**

**Agradecimientos , Dedicatoria (Cristian)**

**Agradecimientos , Dedicatoria (Cristian)**

# **Tabla de contenido**

[Glosario 17](#_bookmark0)

[Resumen 14](#_bookmark1)

[Abstract 15](#_bookmark2)

[Introducción 16](#_bookmark3)

[Planteamiento del problema 17](#_bookmark4)

[Descripción del problema 18](#_bookmark5)

[Formulación del problema 18](#_bookmark6)

[Justificación 19](#_bookmark7)

[Objetivos 20](#_bookmark8)

[General 20](#_bookmark9)

[Específicos 20](#_bookmark10)

[Alcance 21](#_bookmark11)

[Diseño Metodológico 22](#_bookmark12)

[Tipo de investigación 22](#_bookmark13)

[Metodología 22](#_bookmark14)

[Fase I 22](#_bookmark15)

[Fase II 23](#_bookmark16)

[Fase III 23](#_bookmark17)

[Fase IV 24](#_bookmark18)

[Fuentes de Información 24](#_bookmark19)

[Población 24](#_bookmark20)

[Estado del Arte 25](#_bookmark21)

[Marco Referencial 27](#_bookmark22)

[Marco histórico 27](#_bookmark23)

[Marco teórico 29](#_bookmark24)

[Ciclo de vida del Software 29](#_bookmark25)

[Análisis 30](#_bookmark27)

[Diseño 31](#_bookmark28)

[Desarrollo 31](#_bookmark29)

[Prueba 32](#_bookmark30)

[Implementación 32](#_bookmark31)

[Mantenimiento 33](#_bookmark32)

[Base de datos 33](#_bookmark33)

[Modelo entidad-relación 35](#_bookmark35)

[Modelo relacional 35](#_bookmark37)

[Lenguaje Unificado de Modelado UML 36](#_bookmark39)

[Diagrama casos de uso 37](#_bookmark41)

[Diagrama de actividades 38](#_bookmark43)

[Diagrama de clases 39](#_bookmark45)

[Marco conceptual 40](#_bookmark47)

[Metodología Kanban 40](#_bookmark48)

[Visualización del flujo de trabajo 40](#_bookmark49)

[Limitación de trabajo en curso 40](#_bookmark50)

[Gestión de flujo 41](#_bookmark51)

[Reuniones de revisión y mejora continua 41](#_bookmark52)

[Herramientas case 41](#_bookmark53)

[MVC 41](#_bookmark54)

[Modelo 42](#_bookmark56)

[Vista 42](#_bookmark57)

[Controlador 42](#_bookmark58)

[Niveles de la ingeniería en software 43](#_bookmark59)

[Herramienta 43](#_bookmark61)

[Procesos 44](#_bookmark62)

[Métodos 44](#_bookmark63)

[Reingeniería 44](#_bookmark64)|

[Marco Legal 44](#_bookmark65)

[Ley 23 de 1982 45](#_bookmark66)

[ISO 9126 45](#_bookmark67)

[ISO 9000 45](#_bookmark68)

[Artículo 339 45](#_bookmark69)

[Articulo 342 45](#_bookmark70)

[El articulo 31 de la Ley 152 de 1994 46](#_bookmark71)

[La Ley 361 de 1997 46](#_bookmark72)

[La Ley 1618 de 2013 46](#_bookmark73)

[Marco tecnológico 47](#_bookmark74)

[Html 47](#_bookmark75)

[Css 47](#_bookmark76)

[JS 48](#_bookmark77)

[Xampp 48](#_bookmark78)

[PHP 49](#_bookmark79)

[Sql y MySql 49](#_bookmark80)

[Boostrap 50](#_bookmark81)

[Sublime text 50](#_bookmark82)

[Laravel 51](#_bookmark83)

[Chart.js 51](#_bookmark84)

[Marco Geográfica 52](#_bookmark85)

[Estructura temática 54](#_bookmark86)

[Fase 1 54](#_bookmark87)

[Análisis 54](#_bookmark88)

[Requerimientos funcionales 54](#_bookmark90)

[Usuario con privilegios Root 55](#_bookmark91)

[Optimización de la herramienta 55](#_bookmark92)

[Recobrar historial del graficas 55](#_bookmark93)

[Respuestas portables 56](#_bookmark94)

[Requerimientos no funcionales 56](#_bookmark96)

[Seguridad 56](#_bookmark97)

[Accesibilidad 57](#_bookmark98)

[Futuras mejoras 57](#_bookmark99)

[Base de datos 57](#_bookmark100)

[Necesidades 58](#_bookmark101)

[Fase 3 59](#_bookmark102)

[Diagramas y Mokups 59](#_bookmark103)

[Diagrama modelo de negocio 59](#_bookmark104)

[Diagrama casos de uso 60](#_bookmark106)

[Diagrama de actividades 62](#_bookmark110)

[Diagrama de entidad-relación 65](#_bookmark114)

[Diagrama relacional 66](#_bookmark116)

[Mockup index 67](#_bookmark118)

[Mockup dashboard (Root) 68](#_bookmark120)

[Mockup dashboard registro (Root) 69](#_bookmark122)

[Mockup dashboard consulta (Root) 70](#_bookmark124)

[Mockup Lista administradores (Root) 72](#_bookmark127)

[Mockup Lista usuarios (Root) 73](#_bookmark129)

[Mockup Seguimiento (Administrador) 74](#_bookmark131)

[Mockups usuario 76](#_bookmark134)

[Mockups Resultado general 76](#_bookmark135)

[Mockups Herramienta diagnostico 77](#_bookmark137)

[Fase 3 78](#_bookmark139)

[Root 82](#_bookmark144)

[Dashboard 82](#_bookmark145)

[Dashboard consulta 84](#_bookmark147)

[Lista administradores 85](#_bookmark149)

[Lista Usuarios 86](#_bookmark151)

[Seguimiento 87](#_bookmark153)

[Pdf 88](#_bookmark155)

[Herramienta Diagnostico 89](#_bookmark157)

[Fase 4 92](#_bookmark160)

[Root 92](#_bookmark161)

[Inicio de sesión root 93](#_bookmark162)

[Registro 94](#_bookmark165)

[Consulta 98](#_bookmark170)

[Listados 99](#_bookmark172)

[Administrador 101](#_bookmark175)

[Inicio de sesión administrador 101](#_bookmark176)

[Figura 55 103](#_bookmark179)

[Administrador 104](#_bookmark180)

[Inicio de sesión emprendedor 104](#_bookmark181)

[Fase 5 108](#_bookmark186)

[Estado actual del sistema 109](#_bookmark188)

[Resultados y discusiones 113](#_bookmark193)

[Conclusiones 114](#_bookmark194)

[Bibliografía 116](#_bookmark195)

**Lista de tablas**

[Tabla 1 54](#_bookmark89)

[Tabla 2 56](#_bookmark95)

# Lista de Figuras

[Figura 1 30](#_bookmark26)

[Figura 2 34](#_bookmark34)

[Figura 3 35](#_bookmark36)

[Figura 4 36](#_bookmark38)

[Figura 5 37](#_bookmark40)

[Figura 6 38](#_bookmark42)

[Figura 7 39](#_bookmark44)

[Figura 8 39](#_bookmark46)

[Figura 9 41](#_bookmark55)

[Figura 10 43](#_bookmark60)

Figura 11 53

[Figura 12 59](#_bookmark105)

[Figura 13 60](#_bookmark107)

[Figura 14 60](#_bookmark108)

[Figura 15 61](#_bookmark109)

[Figura 16 62](#_bookmark111)

[Figura 17 63](#_bookmark112)

[Figura 18 64](#_bookmark113)

[Figura 19 65](#_bookmark115)

[Figura 20 66](#_bookmark117)

[Figura 21 67](#_bookmark119)

[Figura 22 68](#_bookmark121)

[Figura 23 69](#_bookmark123)

[Figura 24 70](#_bookmark125)

[Figura 25 71](#_bookmark126)

[Figura 26 72](#_bookmark128)

[Figura 27 73](#_bookmark130)

[Figura 28 74](#_bookmark132)

[Figura 29 75](#_bookmark133)

[Figura 30 76](#_bookmark136)

[Figura 31 77](#_bookmark138)

[Figura 32 78](#_bookmark140)

[Figura 33 79](#_bookmark141)

[Figura 34 80](#_bookmark142)

[Figura 35 81](#_bookmark143)

[Figura 36 82](#_bookmark146)

[Figura 37 84](#_bookmark148)

[Figura 38 85](#_bookmark150)

[Figura 39 86](#_bookmark152)

[Figura 40 87](#_bookmark154)

[Figura 41 88](#_bookmark156)

[Figura 42 89](#_bookmark158)

[Figura 43 90](#_bookmark159)

[Figura 44 93](#_bookmark163)

[Figura 45 93](#_bookmark164)

[Figura 46 94](#_bookmark166)

[Figura 47 95](#_bookmark167)

[Figura 48 96](#_bookmark168)

[Figura 49 97](#_bookmark169)

[Figura 50 98](#_bookmark171)

[Figura 51 99](#_bookmark173)

[Figura 52 100](#_bookmark174)

[Figura 53 101](#_bookmark177)

[Figura 54 102](#_bookmark178)

[Figura 55 103](#_bookmark179)

[Figura 56 104](#_bookmark182)

[Figura 57 105](#_bookmark183)

[Figura 58 106](#_bookmark184)

[Figura 59 107](#_bookmark185)

[Figura 60 108](#_bookmark187)

[Figura 61 109](#_bookmark189)

[Figura 62 110](#_bookmark190)

[Figura 63 111](#_bookmark191)

[Figura 64 112](#_bookmark192)

# **Glosario**

**Lactato:** Un compuesto químico que se produce durante la fermentación láctica en los músculos durante el ejercicio intenso. Es un indicador comúnmente utilizado para medir la intensidad del ejercicio y la fatiga muscular.

**Automatización:** El proceso de hacer que un sistema, proceso o tarea funcione automáticamente, sin la intervención directa de humanos. Puede implicar el uso de máquinas, software o sistemas de control para realizar tareas de manera eficiente y consistente.

**Ciencia de Datos:** Un campo interdisciplinario que se enfoca en el estudio y análisis de conjuntos de datos para obtener información útil y tomar decisiones basadas en evidencia. Incluye técnicas de estadística, aprendizaje automático, minería de datos y visualización de datos.

**Registro Deportivo:** Un documento o sistema donde se registran y documentan eventos relacionados con la actividad física o el deporte. Puede incluir información como tiempos, distancias, resultados de competiciones, entrenamientos realizados, etc.

**Calibración:** El proceso de ajustar o verificar la precisión y confiabilidad de un instrumento de medición. Se realiza comparando las mediciones del instrumento con las de un estándar conocido y realizando los ajustes necesarios para garantizar que el instrumento proporcione resultados precisos.

**Interfaz:** Un punto de interacción entre dos sistemas, dispositivos o entidades, que permite la comunicación, intercambio de datos o control. Puede referirse a una interfaz de usuario, que permite a los humanos interactuar con dispositivos electrónicos o software, o a una interfaz de programación, que permite la interacción entre diferentes componentes de software.

# **Resumen**

A través de una colaboración entre el programa de Tecnología en Desarrollo de Software y el programa de Ciencias del Deporte, dos estudiantes trabajaron en el desarrollo de un software diseñado para medir los niveles de lactato en deportistas en actividad. El objetivo principal era mejorar estos niveles y, por ende, optimizar los registros históricos existentes. Este proyecto también cumpliría con el requisito de ser su trabajo de grado para obtener la graduación. Sin embargo, al considerar la actualización del software para continuar su ciclo de vida, se descubrió que no funcionaba correctamente. Con el fin de avanzar en el proyecto como una idea de negocio, fue necesario realizar una reestructuración completa del software, sometiéndolo a un proceso de ingeniería renovado para poder comercializarlo en el futuro.

El presente documento detalla las mejoras tanto visuales como funcionales del proyecto. Además, se explican las razones por las cuales el software fue reestructurado, con el objetivo de mejorar su funcionalidad y permitir que esta versión beta sea probada primero por los estudiantes de Ciencias del Deporte y posteriormente por deportistas de alto rendimiento.

**Abstract**

Through a collaboration between the Software Development Technology program and the Sports Science program, two students worked on developing software designed to measure lactate levels in active athletes. The main goal was to improve these levels and consequently optimize existing historical records. This project also fulfilled the requirement for their graduation thesis. However, upon considering updating the software to continue its lifecycle, it was discovered that it was not functioning correctly. In order to advance the project as a business idea, a complete restructuring of the software was necessary, subjecting it to a renewed engineering process to potentially market it in the future.

This document details both the visual and functional improvements of the project. Furthermore, it explains the reasons why the software was restructured, aiming to enhance its functionality and enable this beta version to be tested first by Sports Science students and subsequently by high-performance athletes.

# **Introducción**

La segunda fase del software Latasoft se enfoca en reestructurar el sistema. Esta nueva etapa tiene como objetivo desarrollar un sistema accesible y fácil de usar, que permita realizar cálculos de lactato y mejorar los registros de los estudiantes que buscan superar sus marcas personales. Utilizando las mismas herramientas que se emplearon en la versión 1.0, se procede con la actualización requerida, siguiendo las recomendaciones del anterior director del proyecto. Además, se incorporarán los nuevos requerimientos del proyecto.

En el siguiente documento, se explicará cómo se aplica la reingeniería al software, empleando diversas metodologías de investigación y programación. Esto conducirá a la primera actualización del proyecto Latasoft.

**Planteamiento del problema**

En el ámbito del deporte, el análisis de la toma de muestra de lactato es crucial para evaluar la capacidad física y determinar los niveles de esfuerzo durante el ejercicio. El lactato es un indicador importante de la producción de energía en el cuerpo y puede ayudar a optimizar el entrenamiento deportivo y la salud en general. Sin embargo, actualmente, el proceso de análisis de la toma de muestra de lactato se enfrenta a varios desafíos:

Proceso Manual y Costoso: La mayoría de los análisis de lactato se realizan manualmente, lo que requiere una gran cantidad de tiempo y recursos humanos. Esto puede resultar en costos elevados y demoras en la obtención de resultados.

Falta de Automatización: La falta de automatización en la recopilación y el análisis de datos de lactato puede llevar a errores humanos y a una falta de precisión en los resultados.

Dificultad en la Interpretación de Datos: La gran cantidad de datos generados durante la toma de muestra de lactato puede ser abrumadora y difícil de interpretar para entrenadores, médicos y deportistas, lo que limita su utilidad práctica.

Falta de Predicciones y Recomendaciones: Actualmente, se carece de sistemas que puedan utilizar datos de lactato para predecir el rendimiento deportivo o hacer recomendaciones específicas de entrenamiento.

Limitaciones en el Análisis Retrospectivo: La mayoría de los análisis de lactato se centran en resultados retrospectivos, lo que dificulta la toma de decisiones en tiempo real durante el entrenamiento o la competición.

En este contexto, surge la necesidad de desarrollar un software basado en ciencia de datos que aborde estos desafíos y permita una toma de decisiones más precisa y oportuna en el ámbito deportivo. Este software deberá automatizar la recopilación y el análisis de datos de lactato, proporcionar herramientas de visualización y predicción, y ofrecer recomendaciones personalizadas para optimizar el rendimiento físico y la salud de los deportistas y pacientes

**Descripción del problema**

El problema radica en la falta de un sistema eficiente para el análisis de lactato en el deporte. Los desafíos incluyen un proceso manual y costoso, falta de automatización, dificultad en la interpretación de datos, ausencia de predicciones y recomendaciones, y limitaciones en el análisis retrospectivo. La solución propuesta es un software basado en ciencia de datos que automatiza y optimiza este proceso.

¿Cómo poder crear una herramienta eficaz y eficiente que mida el nivel de lactato en deportistas de alto rendimiento?

¿Cómo puede la ciencia de datos optimizar el análisis de lactato para mejorar el rendimiento deportivo y la salud?

**Justificación**

Se requiere una herramienta comprobada científicamente para la toma de muestras de lactato en deportistas de alto rendimiento la que permita de manera menos invasiva posible poder establecer con eficacia el comportamiento de los niveles de lactato de los procesos de entrenamiento y pruebas de estos.

La necesidad de una herramienta científicamente validada y menos invasiva para la toma de muestras de lactato en deportistas de alto rendimiento es innegable y se fundamenta en varias consideraciones cruciales:

Precisión Científica: En el ámbito del deporte de élite, la precisión de los datos es esencial. Una herramienta validada científicamente garantiza la exactitud de las mediciones de lactato, lo que es fundamental para la toma de decisiones adecuadas en el entrenamiento y la competición.

Optimización del Rendimiento: Entender cómo los niveles de lactato varían durante el ejercicio permite a los entrenadores personalizar los programas de entrenamiento, maximizando así el rendimiento deportivo y reduciendo el riesgo de lesiones por exceso de esfuerzo.

Comodidad para los Atletas: La comodidad durante la toma de muestras es esencial para el bienestar de los deportistas. Una herramienta menos invasiva garantiza una experiencia más cómoda y menos intrusiva, lo que fomenta la cooperación y la adherencia al proceso.

Efectividad en la Planificación de Entrenamiento: Una herramienta eficaz para la toma de muestras de lactato simplifica la planificación de sesiones de entrenamiento específicas para cada atleta, lo que puede traducirse en un desarrollo físico más efectivo y un mejor rendimiento deportivo.

Seguimiento del Progreso: La monitorización continua de los niveles de lactato proporciona información valiosa sobre la evolución de un deportista con el tiempo. Esto permite ajustar el entrenamiento de manera más precisa y evaluar el progreso de manera objetiva.

La demanda de una herramienta científicamente sólida y menos invasiva para la toma de muestras de lactato en deportistas de élite no solo es justificable, sino esencial para el éxito en el deporte de alto rendimiento. Esta herramienta no solo contribuye a un rendimiento deportivo óptimo, sino que también respalda la salud y el bienestar general de los atletas, lo que respalda plenamente la necesidad de su desarrollo y aplicación en este contexto.

# **Objetivos**

# **General**

Desarrollar un sistema de análisis de lactato basado en ciencia de datos que optimice la toma de muestras en deportistas de alto rendimiento, permitiendo una evaluación precisa y menos invasiva de sus niveles de lactato durante el entrenamiento y las pruebas, contribuyendo así a la mejora del rendimiento deportivo y al bienestar de los atletas.

# **Objetivos específicos**

Diseñar una interfaz de usuario intuitiva y amigable que facilite la toma de lactato y el registro de datos para estudiantes, docentes y entrenadores, mejorando la experiencia del usuario.

Implementar algoritmos avanzados de análisis de datos que permitan una interpretación precisa de los resultados del test de lactato, proporcionando información relevante para la mejora del rendimiento deportivo y la toma de decisiones informadas.

Reforzar las medidas de seguridad y privacidad de datos en el software para garantizar el cumplimiento de regulaciones de protección de datos y salvaguardar la información confidencial de los estudiantes y deportistas.

# **Alcance**

* Desarrollar una herramienta de software en medición del lactato
* Validación Científica Exitosa
* Usando ciencia de datos predecir el comportamiento de un deportista con esta prueba
* Fortalecer la seguridad de los datos
* Mejora Significativa en el Rendimiento Deportivo
* Retroalimentación Positiva de los Deportistas y Entrenadores
* Impacto Duradero en el Deporte de Alto Rendimiento
* Lanzar una idea de negocio.

# **Diseño Metodológico**

# **Tipo de investigación**

Esta investigación aplicada es un enfoque y cuantitativo que busca abordar problemas concretos y prácticos en el mundo real. Se caracteriza por su orientación hacia la resolución de desafíos específicos y la aplicación de conocimientos y soluciones para mejorar situaciones existentes, el objetivo es desarrollar un software médico, científico y deportivo. para abordar las limitaciones en la toma de lactato y la gestión de datos en el entorno deportivo y académico. Este enfoque se traduce en la creación de una herramienta práctica y efectiva que beneficiará directamente a estudiantes, docentes y entrenadores al mejorar la eficiencia y precisión en la evaluación del rendimiento deportivo, lo que lo clasifica como investigación aplicada.

# **Metodología**

# **Análisis de Requisitos**

Identificación de Usuarios: En esta etapa, se identificarán los usuarios finales del software, incluyendo estudiantes, docentes y entrenadores de la Universidad de Cundinamarca - Extensión Soacha. Se recopilarán sus necesidades y requisitos específicos para la toma de lactato, el registro de datos y la retroalimentación personalizada.

Definición de Requisitos: Se establecerán los requisitos funcionales y no funcionales del software, como la capacidad de tomar datos de lactato, la seguridad de los datos, la interfaz de usuario amigable y otros aspectos clave.

# **Diseño del Software**

Diseño de la Interfaz de Usuario: Se creará una interfaz de usuario intuitiva y fácil de usar que permita a los usuarios registrar datos de lactato de manera eficiente y acceder a análisis detallados de los resultados.

Diseño de la Base de Datos: Se diseñará la estructura de la base de datos que almacenará los datos de lactato de los estudiantes y deportistas, asegurando la seguridad y privacidad de los datos.

# **Desarrollo del Software**

Desarrollo de la Aplicación Móvil: Se programará la aplicación móvil utilizando el lenguaje de programación Java y el entorno Android Studio conjunto a ciencia de datos de python. La aplicación permitirá a los usuarios ingresar datos de lactato, realizar análisis y recibir retroalimentación personalizada.

Implementación de Algoritmos de Análisis: Se desarrollarán algoritmos avanzados de análisis de datos que permitirán interpretar con precisión los resultados de la prueba de lactato y proporcionar información relevante para la mejora del rendimiento deportivo.

# **Pruebas y Validación**

Pruebas de Funcionalidad: Se llevarán a cabo pruebas exhaustivas para garantizar que todas las funciones del software funcionen correctamente, incluyendo la toma de lactato, el registro de datos y la generación de análisis.

Pruebas de Seguridad: Se evaluará la seguridad del software para garantizar la protección de los datos confidenciales de los usuarios.

# **Implementación y Despliegue**

Implementación en la Universidad: Se implementará el software en la Universidad de Cundinamarca - Extensión Soacha, y se proporcionará capacitación a los usuarios finales.

# **Evaluación y Retroalimentación:**

Recopilación de Comentarios: Se recopilarán comentarios y retroalimentación de los usuarios para identificar posibles mejoras y ajustes en el software.

# **Documentación y Entrega Final**

Documentación: Se generará documentación detallada del software, incluyendo manuales de usuario y guías de instalación.

Entrega Final: Se entregará la versión del software a la Universidad de Cundinamarca - Extensión Soacha.

# **Mantenimiento y Actualización**

Mantenimiento Continuo: Se proporcionará soporte y mantenimiento continuo para garantizar el funcionamiento óptimo del software a lo largo del tiempo.

# **Fuentes de información**

# **Tesis Doctoral**

Predicción de los umbrales de lactato y ajustes de frecuencia cardiaca en el test de leger Boucher - de Oliveira, F. R. (2004).

Diferencias fisiológicas entre ciclistas de elite y cicloturistas en un test incremental ya una misma concentración de lactato sanguíneo - Castrillón, I. S. M. (2008).

# **Artículos de Investigación**

¿Cómo interpretar los resultados del test estándar de lactato que se presentan en los atletas durante el ciclo de entrenamiento? - Berdeal, A. L. A., & Luis, A. (2014).

# **Revistas**

Identificação do lactato mínimo de corredores adolescentes em teste de pista de três estágios incrementais - Revista Brasileira de Medicina do Esporte, 17(2), 119-122

**Población**

El proyecto de software aplicado al test estándar del lactato beneficia principalmente a los siguientes grupos:

* Atletas y deportistas profesionales: quienes necesitan monitorear y optimizar su rendimiento físico.
* Entrenadores y preparadores físicos: que buscan herramientas precisas para evaluar la condición de sus atletas.
* Instituciones deportivas: como clubes y federaciones que requieren de métodos estandarizados para las pruebas de sus miembros.
* Investigadores en el campo de la medicina deportiva: interesados en estudiar la relación entre el lactato y el rendimiento atlético.
* Profesionales de la salud: que utilizan el lactato como un indicador en la recuperación y rehabilitación de pacientes.
* Este software proporciona una herramienta valiosa para la evaluación y mejora continua en el ámbito deportivo y de salud.

# **Estado del Arte**

En la actualidad, el panorama del desarrollo de aplicaciones móviles está experimentando un crecimiento significativo, con un enfoque particular en satisfacer las necesidades de los usuarios mediante soluciones tecnológicas innovadoras. Un ejemplo destacado es el proyecto Ruta-ES, respaldado por la Secretaría de Desarrollo Económico de Fusagasugá. El propósito principal de esta iniciativa es respaldar a los emprendedores locales mediante una herramienta de diagnóstico que evalúa su progreso empresarial.

Los resultados obtenidos a través de este sistema permiten a la Secretaría identificar áreas que requieren un mayor apoyo, lo que contribuye de manera considerable al desarrollo económico de la región y al fortalecimiento de la comunidad emprendedora.

Es crucial destacar que este software optimiza y simplifica los procesos, actividades y tareas para los cuales fue diseñado. Además, se observa una mejora notable en la experiencia del usuario al interactuar con estas herramientas, lo que potencia la eficiencia y la efectividad de su uso. Este enfoque se refleja en aplicaciones como Wase, una plataforma de navegación y tráfico basada en la colaboración comunitaria, que se ha consolidado como una herramienta esencial para conductores particulares.

Es importante también resaltar el papel fundamental que desempeñan las actualizaciones periódicas en la satisfacción de las cambiantes necesidades de los usuarios. Estas actualizaciones permiten a las aplicaciones adaptarse y evolucionar constantemente, garantizando su relevancia y utilidad a lo largo del tiempo en un entorno tecnológico en constante cambio.

Siguiendo esta línea de pensamiento, se ha decidido actualizar el sistema Ruta-ES, con el objetivo de corregir funcionalidades y aplicar mejoras visuales. Esta medida tiene como principal objetivo garantizar la completa satisfacción del usuario y asegurar que la aplicación siga siendo una herramienta valiosa y relevante en el futuro.

# **Marco referencial**

# **Marco Histórico**

En el marco histórico, es importante destacar que el descubrimiento del lactato o ácido láctico por parte de Carl Wilhelm Scheele en 1780 marcó un hito en la comprensión de las propiedades químicas de este compuesto. Scheele, un ayudante de farmacia sueco, realizó este descubrimiento a partir del estudio de la leche agria. Su técnica de aislamiento, aunque rudimentaria en comparación con los métodos modernos, sentó las bases para la comprensión posterior del ácido láctico.

La descripción de la técnica de Scheele menciona la evaporación del suero agrio, la separación de la cuajada (proteínas de la leche), la saturación con leche de cal (hidróxido de calcio) y la dilución con agua. Estos pasos, aunque simples, permitieron identificar y aislar el ácido láctico de manera temprana en la historia de la química.

A partir de este descubrimiento inicial, la investigación sobre el ácido láctico se expandió, y en las décadas y siglos posteriores, se obtuvo un conocimiento más profundo sobre su estructura y sus funciones en los procesos biológicos. El estudio del lactato se ha vuelto relevante en campos como la química, la fisiología y la medicina, donde se comprende su importancia en la producción de energía en el cuerpo humano y su papel en la fermentación láctica, entre otros procesos.

# **Marco Teórico**

El software se desarrollará con el propósito de facilitar el registro de los niveles de lactato en los deportistas de la Universidad de Cundinamarca Extensión Soacha. Actualmente, no existe una herramienta que permita recopilar estos datos de manera eficiente, lo que conlleva a que los docentes o entrenadores de la universidad deban realizar este proceso manualmente. La aplicación móvil se presenta como una solución viable para mejorar la realización de las pruebas de lactato en la UDEC - Extensión Soacha, aprovechando la omnipresencia de los dispositivos móviles en la vida cotidiana, los cuales se han convertido en una extensión de las personas. Este software tendrá la capacidad de abordar una amplia gama de variables relacionadas con el ejercicio, incluyendo alta intensidad, ejercicios funcionales con variación constante de rango de movimiento, potencia, velocidad y aspectos metabólicos. Los usuarios podrán ingresar manualmente parámetros específicos para cada protocolo de prueba, como el tipo de prueba, la duración e intensidad del ejercicio, y la frecuencia cardíaca. Además, permitirá seleccionar diferentes modelos de análisis según las necesidades. Uno de los principales enfoques será medir el nivel de lactato en la sangre, un parámetro fisiológico altamente sensible a los cambios en el rendimiento deportivo. Esto facilitará una evaluación precisa del progreso o el rendimiento de los atletas, permitiendo una observación detallada de la curva de tasa de lactato. En resumen, el desarrollo de la aplicación se basa en la confianza de que resolverá los desafíos y dificultades que actualmente se presentan en la realización de las pruebas de lactato en Entrenadores, atletas y estudiantes.

# **Ciclo de vida del software**

El ciclo de vida del desarrollo de software, también conocido como SDLC o Systems Development Life Cycle, abarca las etapas esenciales para validar la creación del software y asegurar su conformidad con los requisitos de la aplicación. Además, verifica la implementación de los procesos de desarrollo, garantizando la adecuación de los métodos utilizados en el proceso.

(intelequia, 2020)

# **Figura 1**

*Ciclo de vida del software*



Nota: ciclo de vida del software. Fuente: (Gerónimo, 2023)

# **Análisis**

La fase de análisis representa un componente fundamental en el ciclo de vida del desarrollo de software. En esta etapa inicial, el equipo de desarrollo lleva a cabo una exhaustiva investigación y análisis para comprender a fondo los requisitos, la funcionalidad y el alcance del software que se va a desarrollar. Este proceso es crucial para establecer con claridad qué se construirá y por qué, ya que una comprensión incompleta o imprecisa podría conducir a un resultado final que no cumpla con las expectativas del cliente o que incluso termine prematuramente.

Como su nombre sugiere, la fase de análisis marca el inicio del proyecto, donde el equipo se reúne para planificar y establecer el enfoque que se seguirá. Se identifican todas las herramientas necesarias para la construcción del proyecto y se definen detalladamente las funcionalidades y características que los clientes requieren en el software. Esta fase no solo es la primera acción a realizar, sino que también es uno de los pasos más críticos, ya que sienta las bases para todas las etapas subsiguientes del desarrollo.

Es durante la fase de análisis donde se establece la visión y el propósito del software, se definen los objetivos del proyecto y se establecen los requisitos funcionales y no funcionales. Además, se realiza un análisis detallado de los usuarios finales y sus necesidades, lo que permite diseñar una solución que satisfaga plenamente sus expectativas y requisitos.

# **Diseño**

La fase de diseño representa un paso esencial en el ciclo de desarrollo de software, donde se da forma a la arquitectura y estructura del sistema que se construirá. Durante esta etapa, se elabora el modelo lógico del software, así como el modelo entidad-relación mediante herramientas de modelado intuitivo. Además, se crean modelos visuales, como los "Mockups", que proporcionan una representación gráfica del proyecto a nivel de interfaz de usuario.

El diseño no solo se centra en aspectos visuales, sino que también especifica la estructura subyacente a la que se conectará la aplicación. Esto incluye la definición de tipos de archivos y extensiones que se utilizarán a lo largo del proceso de desarrollo.

El proceso de diseño permite establecer una base sólida para la implementación del software, al definir claramente la estructura y funcionalidad del sistema. Esto facilita la comunicación entre los miembros del equipo de desarrollo y garantiza que todos trabajen hacia un objetivo común.

Además, el diseño cuidadoso ayuda a identificar posibles problemas o desafíos antes de que se inicie la fase de implementación, lo que permite realizar ajustes y correcciones de manera temprana en el proceso de desarrollo. Esto contribuye a minimizar costos y tiempos de desarrollo, así como a mejorar la calidad y la eficiencia del producto final.

# **Desarrollo**

La fase de desarrollo, como su nombre indica, marca el inicio de la construcción del software basándose en los lineamientos de diseño previamente establecidos. Durante esta etapa, es crucial adherirse a los requisitos funcionales y no funcionales identificados en la fase de análisis, asegurando que el producto final satisfaga las necesidades y expectativas del cliente.

Es esencial tener claridad sobre el lenguaje de programación que se utilizará, así como comprender su semántica y sintaxis. Esto garantiza un desarrollo eficiente y coherente del código. Además, se deben realizar pruebas periódicas en partes específicas del código para detectar y corregir posibles errores o fallos de manera temprana en el proceso de desarrollo.

Una vez que se ha establecido la estructura del software, es hora de iniciar la programación. La elección del lenguaje de programación adecuado para el proyecto y contar con un equipo de programadores experimentados son aspectos clave para el éxito de esta fase. Seguir las mejores prácticas de programación permitirá escalar fácilmente los proyectos y mantener un código limpio y mantenible a lo largo del tiempo.

Una vez que se ha completado una versión inicial del producto, esta se somete a una verificación exhaustiva para garantizar que todos sus componentes estén en concordancia con los requisitos previamente establecidos. Esto incluye pruebas de funcionalidad, rendimiento y seguridad, entre otras.

# **Prueba**

Una vez completada la fase de desarrollo, el producto se somete a pruebas en un entorno de prueba. El responsable de las pruebas verifica tanto el funcionamiento adecuado del software como su cumplimiento con los requisitos del usuario. Se verifica la correcta instalación de todos los componentes para evitar errores durante la presentación.

Las pruebas de funcionamiento son fundamentales para asegurar que todos los componentes estén instalados correctamente y funcionen de manera óptima. Una vez completadas las pruebas, el producto se entrega al comité de calidad.

El comité de calidad revisa minuciosamente el producto y, en caso de encontrar alguna observación, se procede a abordar las correcciones necesarias. Una vez que se han realizado todas las correcciones y se ha verificado que el producto cumple con todos los requisitos, se firma el formato de validación de requerimientos, marcando así el cierre de la fase de pruebas.

Este proceso de pruebas y validación es crucial para garantizar la calidad del producto final y su conformidad con los estándares y expectativas del usuario.

# **Implementación**

La fase de implementación representa la validación del proyecto en un entorno real, lo cual puede implicar un tiempo significativo debido a la instalación precisa del software de acuerdo con los requisitos específicos para garantizar su correcto funcionamiento. Una vez completada la instalación, se lleva a cabo un seguimiento detallado y se proporciona capacitación a los usuarios para asegurar que puedan utilizar el software de manera efectiva.

Es esencial contar con asistencia disponible en caso de que surjan problemas o dudas, asegurando así una trazabilidad hacia la utilización efectiva del software por parte de los usuarios.

El lanzamiento del software requiere una planificación cuidadosa del entorno, teniendo en cuenta las dependencias entre los componentes individuales. Aunque ciertos componentes puedan funcionar correctamente por separado, es posible que surjan problemas de compatibilidad cuando se combinan. Por tanto, se deben utilizar combinaciones conocidas para evitar problemas de este tipo durante el lanzamiento.

# **Mantenimiento**

La fase de mantenimiento y actualización es fundamental para la viabilidad a largo plazo del software. Con el tiempo, algunas funciones pueden volverse obsoletas, dejar de funcionar correctamente o generar errores, mientras que el atractivo visual del software puede disminuir.

Es crucial realizar actualizaciones periódicas para abordar estos desafíos y garantizar que el software siga siendo relevante y funcional para los usuarios. Al pasar por alto esta etapa, el software corre el riesgo de volverse obsoleto, lo que podría conducir a su descontinuación y retirada de los servicios en línea.

El mantenimiento del software implica la corrección de errores, la optimización del rendimiento y la incorporación de nuevas características según las necesidades del usuario y los avances tecnológicos. Además, las actualizaciones periódicas también son importantes para garantizar la seguridad del software, ya que ayudan a abordar vulnerabilidades y riesgos de seguridad emergentes.

# **Método Científico**

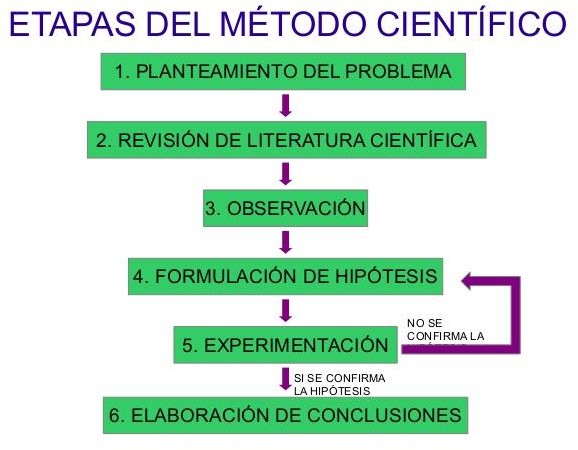
El ciclo de vida del desarrollo de software, también conocido como SDLC o Systems Development Life Cycle, abarca las etapas esenciales para validar la creación del software y asegurar su conformidad con los requisitos de la aplicación. Además, verifica la implementación de los procesos de desarrollo

"El método científico es un enfoque para adquirir nuevos conocimientos, que ha sido históricamente fundamental en el ámbito científico. Implica llevar a cabo una observación sistemática, mediciones precisas, experimentación y la formulación, análisis y posible modificación de hipótesis" (Press, 1884).

Las características principales de un método científico válido incluyen la capacidad de ser refutado, la capacidad de que los resultados puedan ser reproducidos y verificados por otros, y su validez respaldada mediante la revisión por pares. Diversas técnicas y metodologías, como la deducción, la inducción, la abducción y la predicción, se utilizan en el proceso científico para obtener y validar nuevos conocimientos."ollo, garantizando la adecuación de los métodos utilizados en el proceso.

# **Figura 2**

*Etapas del Método Científico*



Nota: Etapas del método cientifico. Fuente: (blogger, 2017)

# **Modelo entidad-relación**

Un modelo de entidad-relación (ER) es un diagrama que ilustra cómo las entidades se relacionan entre sí dentro de un sistema. Estos diagramas son ampliamente utilizados en el diseño y depuración de bases de datos relacionales en el desarrollo de software.

Los modelos ER, también conocidos como ERD (Entity-Relationship Diagrams), utilizan un conjunto específico de símbolos, como rectángulos, diamantes, óvalos y líneas de conexión, para representar las asociaciones entre entidades, relaciones y sus propiedades. Estos diagramas reflejan la estructura gramatical y utilizan entidades como sustantivos y relaciones como verbos, lo que facilita la comprensión de la estructura y las interacciones dentro del sistema.

Los rectángulos representan las entidades, que son los objetos o conceptos sobre los cuales se recopila y almacena información en la base de datos. Los diamantes representan las relaciones, que describen cómo se conectan las entidades entre sí. Los óvalos representan los atributos, que son las características o propiedades de las entidades. Y las líneas de conexión muestran las asociaciones entre entidades y relaciones.

# **Figura 3**

*Diagrama de entidad relación*

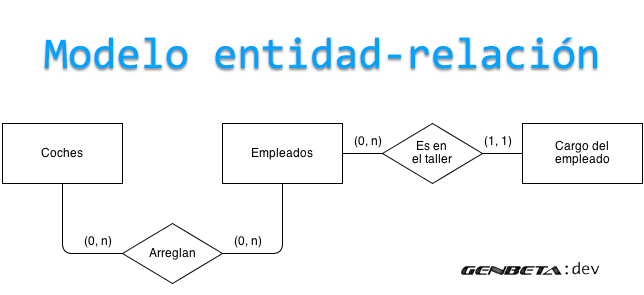
Diagrama

Descripción generada automáticamente

Nota. Diagrama de entidad relación. Fuente: (Manuel.cillero.es,2024)

# **Figura 4**

*Diagrama de entidad relación Ejemplo*



Nota. Diagrama de entidad relación. Fuente: (Genbeta,2013)

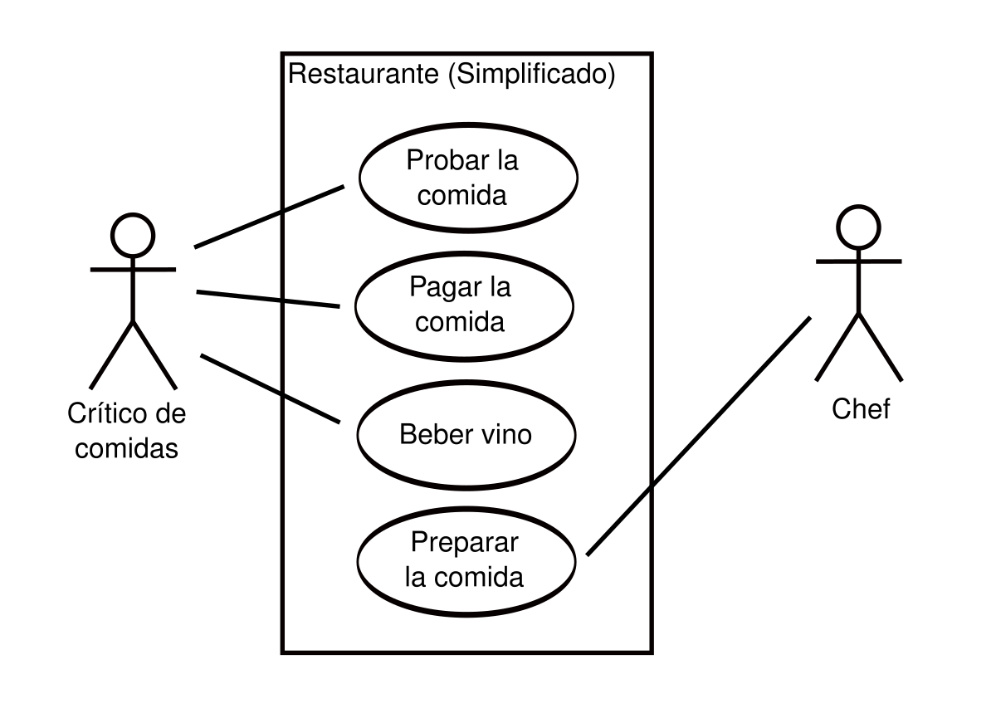
# **Diagrama casos de uso**

Un caso de uso es un artefacto que describe una secuencia de acciones que conducen a un resultado de valor observable en el contexto de sistemas y procesos comerciales. Estos casos de uso son una estructura fundamental para expresar los requisitos funcionales de un sistema.

Los casos de uso pueden ser representados gráficamente en diagramas, donde se muestran las interacciones entre los actores (personas, sistemas, o entidades externas) y el sistema en sí mismo. También pueden ser especificados en documentos de texto, donde se detallan las acciones y los resultados esperados en un formato narrativo.

# **Figura 5**

*Ejemplo Diagrama Casos De Uso*



Nota. Diagrama casos de uso. Fuente: (Wikipedia, 2022)

# **Modelo relacional**

El modelo relacional es una representación lógica de la información empresarial que se implementa a través de una serie de tablas, campos, restricciones y relaciones entre ellos. Este modelo refleja la semántica del negocio y es capturado y descrito por el sector de tecnología de la información en sistemas de gestión de bases de datos. La arquitectura lógica de estos sistemas, como los ERP (Enterprise Resource Planning) y CRM (Customer Relationship Management), está determinada por el modelo relacional.

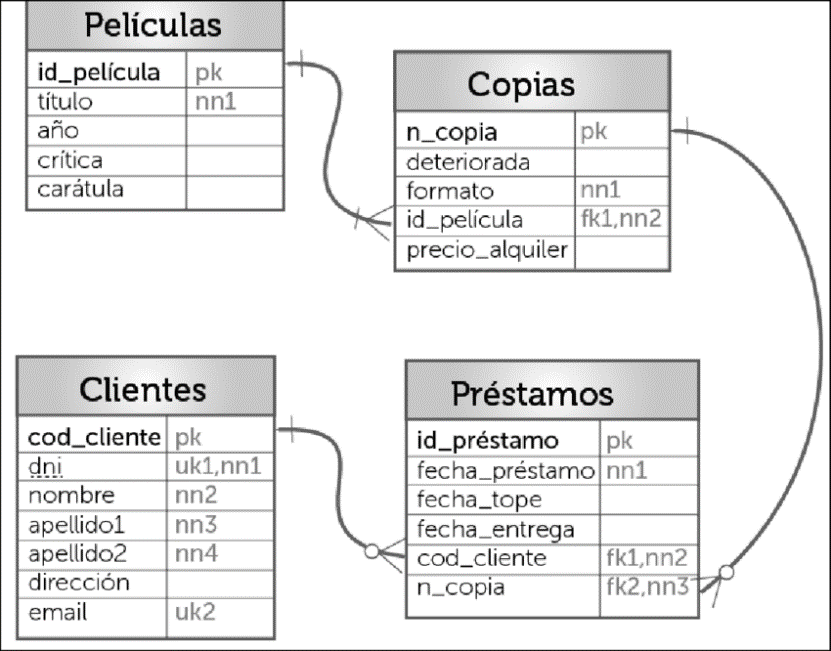
En un modelo relacional, la información se organiza en tablas, donde cada tabla representa una entidad del negocio y cada fila de la tabla representa una instancia específica de esa entidad. Los campos de la tabla representan atributos de la entidad y las relaciones entre las tablas representan las asociaciones entre las entidades.

Las restricciones, como las claves primarias y foráneas, garantizan la integridad de los datos y la coherencia entre las tablas. Esto asegura que la información almacenada en la base de datos sea precisa y consistente.

Los sistemas ERP y CRM son ejemplos de aplicaciones empresariales que utilizan modelos relacionales para almacenar y gestionar la información del negocio. Estos sistemas se basan en la arquitectura relacional para ofrecer funcionalidades como la gestión de recursos empresariales y la gestión de relaciones con los clientes de manera eficiente y efectiva.

# **Figura 6**

*Ejemplo Diagrama Modelo Relacional*



Nota. Diagrama modelo relacional. Fuente: (ResearchGate, 2022)

# **Bases de datos**

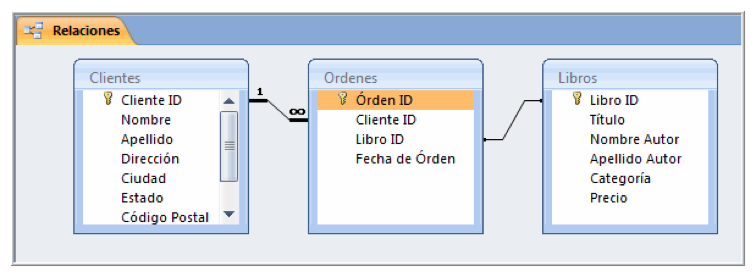
Las bases de datos son un componente fundamental en el ámbito de la informática y la gestión de la información. Este texto investigativo teórico se adentra en el estudio de las bases de datos, desde sus fundamentos conceptuales hasta su evolución histórica y sus diversas aplicaciones en la actualidad.

A través de un análisis exhaustivo, se examinan los principios teóricos que subyacen a las bases de datos, así como su importancia en el contexto de la ciencia de la computación y la gestión empresarial. Además, se exploran las diferentes tecnologías y modelos de bases de datos que han surgido a lo largo del tiempo, desde las bases de datos jerárquicas y en red hasta los sistemas de gestión de bases de datos relacionales y NoSQL.

Asimismo, se examinan las tendencias actuales en el campo de las bases de datos, como el almacenamiento en la nube, el análisis de big data y la inteligencia artificial, y se analizan las implicaciones de estas tendencias en la evolución futura de las bases de datos.

# **Figura 7**

*Ejemplo del uso de una base de datos*



Nota. Ejemplo del uso de una base de datos. Fuente: (edu.gcfglobal.org, 2007)

# **Lenguaje Unificado de Modelado UML**

El Lenguaje Unificado de Modelado (UML) es un estándar para la representación visual de objetos, estados y procesos dentro de un sistema. Sirve como un modelo de diseño que garantiza una arquitectura de información estructurada y también ayuda a los desarrolladores de software a comunicar la descripción del sistema de manera comprensible para personas no especializadas en el campo.

UML fue creado con el propósito de establecer un lenguaje de modelado visual común, semántica y sintácticamente rico para la arquitectura, diseño e implementación de sistemas. El software es inherentemente complejo tanto en estructura como en comportamiento, y UML proporciona una manera estandarizada de representar esta complejidad.

Aunque UML no es un lenguaje de programación en sí mismo, se ha vuelto indispensable en el desarrollo de software moderno. Se utiliza en una variedad de contextos, incluyendo el desarrollo de software, pero también en procesos tecnológicos en producción.

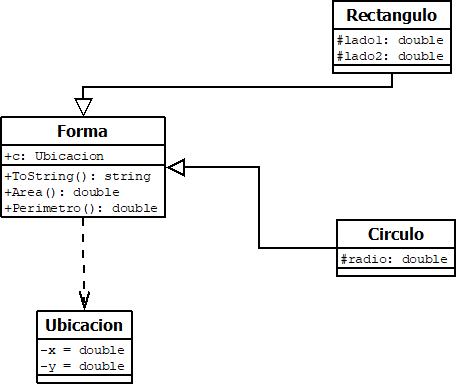
# **Diagrama de clases**

Un diagrama de clases es una herramienta de modelado estática utilizada para visualizar y representar las relaciones entre los objetos programables en un sistema de software. Además de esto, también es una excelente forma de ilustrar la estructura de capas del sistema.

Estos diagramas muestran las clases del sistema y sus relaciones, incluyendo la asociación, la composición, la herencia y la agregación entre ellas. La estructura jerárquica y las interacciones entre las clases se visualizan de manera clara y comprensible, lo que facilita la comprensión de la arquitectura del sistema y ayuda en su diseño y desarrollo.

# **Figura 8**

*Diagrama de clases*



Nota. Diagrama de clases Fuente: (exercisescsharp, 2020)

# **Marco conceptual**

En nuestro proyecto, es esencial comprender y utilizar un lenguaje técnico formal que incluye algunas expresiones clave:

Software Libre: La definición de software libre establece las condiciones necesarias para que un programa sea considerado como tal. El término 'software libre' hace referencia a programas que respetan la libertad de los usuarios y las comunidades. En resumen, esto implica que los usuarios tienen la libertad de ejecutar, copiar, distribuir, aprender, modificar y mejorar el software (Autor del Sistema Operativo GNU, 2021).

Base de datos: Se trata de un 'almacén' que permite la organización y el almacenamiento de grandes cantidades de información para su fácil recuperación y uso posterior. Una base de datos se define como un conjunto de datos relacionados y organizados que son recopilados y utilizados por el sistema de información de una empresa o entidad específica (Autor Maestro del Web, 2021).

Java: Es un lenguaje de programación versátil que tiene la capacidad de ejecutar una amplia variedad de proyectos y funcionar en múltiples plataformas. En nuestros recursos de aprendizaje, explorará qué es Java y cómo programar en este lenguaje a través de varios manuales. Java es un lenguaje de programación de propósito general y uno de los más populares, con una amplia gama de aplicaciones disponibles (Autor Desarrollo Web, 2021).

# **Marco Legal**

es necesario tener en cuenta las diversas normativas que le conciernen, ya sea a través de leyes, artículos o decretos que guarden relación con el mismo.

**Ley 23 de 1982:** Esta ley protege los derechos de autor de los prototipos y productos asociados al proyecto. Su cumplimiento asegura el respeto a la propiedad intelectual y la protección de la propiedad creada.

**Norma ISO 9126:** Este estándar internacional establece las características de calidad que debe tener el software, incluyendo la fiabilidad, usabilidad y eficiencia. Cumplir con esta norma garantiza un producto de calidad para los usuarios.

**Norma ISO 9000:** Esta norma complementa la ISO 9126 al establecer estándares para la gestión de calidad en el desarrollo de software, asegurando procesos eficientes y consistentes en todas las etapas del proyecto.

**Artículo 339 de la Constitución Política Nacional:** Estipula la necesidad de planes de desarrollo territorial para garantizar el uso eficiente de los recursos y el cumplimiento de las funciones asignadas por la ley.

**Artículo 342 de la Constitución Política Nacional:** Requiere la participación de la ciudadanía en la creación de planes de desarrollo, asegurando la inclusión de diversas perspectivas en la planificación.

**Ley 152 de 1994, Artículo 31 y 32:** Establece que las entidades territoriales tienen autonomía en la planificación para el desarrollo económico, social y ambiental, dentro del marco legal establecido.

**Ley 361 de 1997 (Ley de Discapacidad):** Esta ley garantiza la inclusión social de personas con discapacidad y establece medidas para eliminar la discriminación y promover su participación en la sociedad.

**Ley 1618 de 2013:** Asegura el pleno ejercicio de los derechos de las personas con discapacidad mediante la adopción de medidas de inclusión y eliminación de barreras.

# **Marco Tecnológico**

Nos referenciamos a las herramientas que se emplearán en el desarrollo del aplicativo móvil con el fin de alcanzar sus objetivos. Estas herramientas se definen y se exponen sus posibles casos de uso.

**Lenguaje Unificado de Modelado (UML):** El UML es una herramienta diseñada para capturar conceptos de manera convencional y comprensible al presentarlos a otras personas. En palabras de SCH Müller (2000), "Hoy en día, es esencial contar con un plan minuciosamente analizado. El cliente debe comprender claramente lo que un equipo de desarrolladores llevará a cabo y, al mismo tiempo, debe poder señalar cualquier cambio si sus necesidades no se han captado de manera precisa". Esto subraya la importancia del modelado UML debido a su estandarización de la comunicación, lo que facilita que el usuario comprenda el proceso sin necesidad de ser un experto en la materia y, al mismo tiempo, evita malentendidos entre el programador y el cliente en este contexto específico.

**Lenguaje de Programación Kotlin:** Kotlin es un lenguaje de programación moderno que ha ganado popularidad en la comunidad de desarrolladores debido a su elegante sintaxis y su interoperabilidad con Java. Según García (2020), Kotlin destaca por su claridad y concisión, lo que lo convierte en una herramienta ideal para desarrolladores que desean escribir código de manera eficiente y fácil de entender. La curva de aprendizaje de Kotlin es suave, lo que permite a los programadores familiarizarse rápidamente con sus características y principios fundamentales.

**Herramienta Python con Pandas para Gráficas:** Python es un lenguaje de programación ampliamente utilizado en el ámbito de la ciencia de datos y el análisis de datos debido a su simplicidad y versatilidad. Cuando se combina con la biblioteca Pandas, Python se convierte en una poderosa herramienta para manipular y analizar conjuntos de datos de manera eficiente. Según López (2023), Pandas ofrece estructuras de datos flexibles y potentes, como DataFrames, que facilitan la carga, limpieza y transformación de datos.

**Herramienta MySQL para Gestión de Bases de Datos:** MySQL es un sistema de gestión de bases de datos relacional ampliamente utilizado en aplicaciones web y empresariales debido a su fiabilidad, rendimiento y escalabilidad. Según Martínez (2022), MySQL se destaca por su facilidad de uso y su capacidad para manejar grandes volúmenes de datos de manera eficiente.

**Software:** Se define como un conjunto integral de programas, procedimientos, reglas, documentación y datos diseñados para llevar a cabo tareas específicas dentro de un sistema. Esta definición, según Sommerville (2005), abarca todos los elementos necesarios para el funcionamiento de un software. El objetivo de este proyecto es desarrollar un software que cumpla con las tareas mencionadas anteriormente, incluyendo la documentación y los requisitos funcionales y no funcionales asociados.

**IDE Android Studio:** Se trata de un entorno de desarrollo basado en IntelliJ, diseñado para proporcionar una amplia gama de funcionalidades que optimizan la compilación de aplicaciones móviles destinadas al Sistema Operativo Android. Como señala Hohensee (2014), la elección de Android Studio se basa en su estatus como entorno oficial para el desarrollo de aplicaciones Android, respaldado por Google y con licencia libre.

**Librería Scikit Learn - Machine Learning:** Scikit-learn, previamente conocida como scikits.learn, es una biblioteca de aprendizaje automático de código abierto diseñada para los lenguajes de programación Python y Java. Esta biblioteca, como se menciona en el Journal of Machine Learning Research 2011, incluye diversos algoritmos de clasificación, regresión y análisis de grupos, entre ellos máquinas de vectores de soporte, bosques aleatorios, Gradient boosting, K- means y DBSCAN. Su finalidad es trabajar de manera conjunta con las bibliotecas numéricas y científicas NumPy y SciPy,

**Base de Datos:** Se refiere a un conjunto de datos interconectados que deben estar libres de redundancia para mantener su funcionalidad. Estos datos conforman una estructura independiente en términos de uso e implementación, según lo describe Cabello García (2015). La importancia de una base de datos radica en su capacidad para agrupar y almacenar información relevante para personas y organizaciones, restringiendo el acceso a dicha información de acuerdo con los permisos otorgados.

**Herramienta Git para Control de Versiones:** Git es un sistema de control de versiones distribuido ampliamente utilizado en el desarrollo de software para gestionar cambios en el código fuente y coordinar el trabajo entre equipos de manera eficiente. Según Pérez (2023), Git se destaca por su rapidez, flexibilidad y capacidad para manejar proyectos de cualquier tamaño.

# **Marco Geográfico**

La universidad de Cundinamarca Extensión Soacha creada en el año 2000. Se encuentra

en el municipio de Soacha, conocido como la Ciudad del Varón del Sol, Soacha es uno de los municipios más poblados del Departamento. Es la más nueva de las extensiones de la Universidad de Cundinamarca, creada para fortalecer la formación de profesionales de la región. Actualmente, la institución oferta dos programas de pregrado y una especialización presencial.

¿Qué herramienta sería útil al realizar una prueba de penetración de infraestructura de red?

herramienta de escaneo de vulnerabilidades

evitando firewalls y herramientas IPS

herramienta de proxy de interceptación

herramienta de prueba de aplicaciones móviles

# **Figura 9**

*Direccion de navegador waze*

Diagrama

Descripción generada automáticamente

(Dirección sacada del navegador de Waze)

# mockup

Resultados y discusiones

concluciones

recomendaciones

bibliografia